

# AUTÓMATAS PROGRAMABLES

## PROGRAMACIÓN DE AUTÓMATAS INTRODUCCIÓN AL GRAFCET

## Programación de autómatas: Introducción al Grafcet

- Introducción a la programación del autómata
  - Definición del sistema de control
    - Descripciones literales
    - Funciones algebraicas
    - Esquemas de relés
    - Diagramas lógicos
    - Diagramas de flujo
    - Grafcet
- Definir las variables que intervienen y asignarles direcciones de memoria
- Lenguajes de programación
  - Lista de instrucciones
  - Diagramas de contactos y funciones
  - Lenguajes de alto nivel

## Programación de autómatas: Introducción al Grafcet

- Grafcet
  - Introducción
  - Símbolos normalizados
  - Reglas de evolución del Grafcet
  - Posibilidades de representación de automatismos con Grafcet
    - Estructuras base
      - Estructuras de secuencia única
      - Estructuras de secuencias paralelas
    - Estructuras lógicas
      - Divergencia OR
      - Convergencia OR
      - Divergencia en AND
      - Convergencia en AND
      - Saltos Condicionales
  - Ejemplo

## Programación de autómatas: Introducción al Grafcet

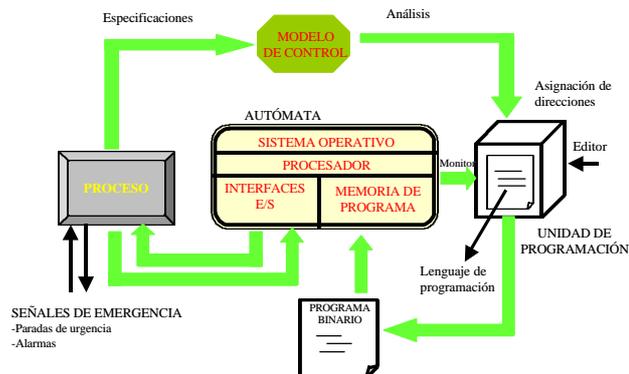
- Implementación del Grafcet sobre autómatas programables
- Niveles de Grafcet
- Representación de situaciones especiales en Grafcet
  - Secuencias exclusivas
  - Temporizadores y contadores en Grafcet

## Programación de autómatas: Introducción al Grafcet

- **Introducción a la programación del autómata** ←
- Definición del sistema de control
  - Descripciones literales
  - Funciones algebraicas
  - Esquemas de relés
  - Diagramas lógicos
  - Diagramas de flujo
  - Grafcet
- Definir las variables que intervienen y asignarles direcciones de memoria
- Lenguajes de programación
- Grafcet

## Introducción a la programación del autómata

- Sistemas y recursos envueltos en la transferencia de información que rodean al autómata

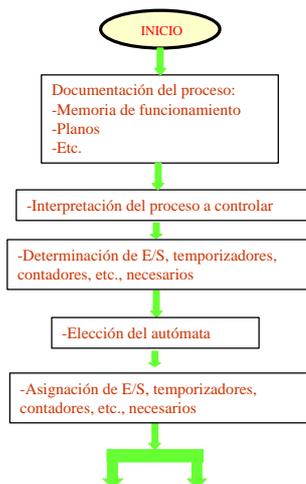


## Introducción a la programación del autómata

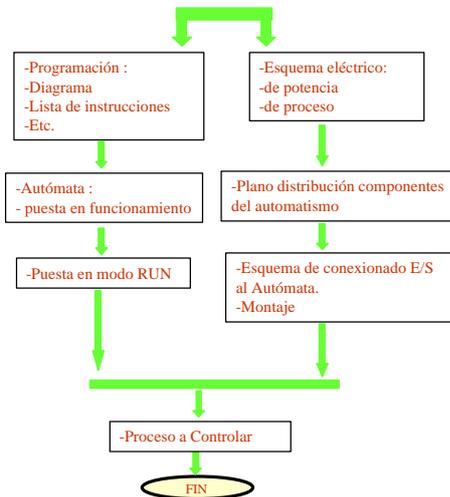
- Podríamos dividir la programación del autómata en varios pasos :
  - Definir el sistema de control ( que debe hacer, en que orden, etc.): diagrama de flujo, la descripción literal o un grafo GRAFCET.
  - Identificar las señales de entrada y salida del autómata.
  - Representar el sistema de control mediante un modelo, indicando todas las funciones que intervienen, las relaciones entre ellas, y la secuencia que deben seguir. Algebraica (instrucciones literales) o gráfica (símbolos gráficos).
  - Asignar las direcciones de entrada/salida o internas del autómata a las correspondientes del modelo.
  - Codificar la representación del modelo. Lenguaje de programación.
  - Cargar el programa en la memoria del autómata desde la unidad de programación.
  - Depurar el programa y obtener una copia de seguridad.

## Introducción a la programación del autómata

- Esquema del desarrollo de un proceso controlado por un autómata



## Introducción a la programación del autómeta



## Programación de autómatas: Introducción al Grafcet

- Introducción a la programación del autómeta
  - **Definición del sistema de control** ←
  - Descripciones literales
  - Funciones algebraicas
  - Esquemas de relés
  - Diagramas lógicos
  - Diagramas de flujo
  - Grafcet
- Definir las variables que intervienen y asignarles direcciones de memoria
- Lenguajes de programación
- Grafcet

## Definición del sistema de control

- Sistemas sin complejidad  $\Rightarrow$  lenguaje vulgar (descrip. Literal)
- Herramienta de representación basadas en símbolos(+ complejos)
- Clasificación según los símbolos utilizados:
  - *Preposicional* : descripciones literales.
  - *Algebraicas* : funciones booleanas y aritméticas.
  - *Gráfica* : esquemas de contactos, diagramas lógicos o funciones lógicas, ordinogramas, técnicas GRAFCET
- Descripciones literales
  - La descripción literal de un proceso y de su control puede hacerse enumerando literalmente las acciones a desarrollar por el mismo, expuestas secuencialmente y con indicación de las condiciones de habilitación o validación en cada caso.
  - Exhaustivamente o no define bien. Si se hace, difícil comprensión.

## Definición del sistema de control

- Se complementa con otras formas de representación
- Funciones algebraicas  $Y = (BC + \bar{B}C) \cdot A$ 
  - Las funciones algebraicas de cada una de las salidas se obtienen:
    - o bien directamente de la descripción literal del proceso a controlar
    - o bien se aplican métodos de síntesis basados en el álgebra de Boole (tablas de verdad, Karnaugh, etc.).
  - Difícil de analizar y sintetizar sistemas secuenciales. Limitado a la representación de combinaciones de variables independientes del tiempo (condiciones de alarma, operaciones aritméticas con variables analógicas, etc.)
- Esquemas de relés
  - Origen: en la representaciones electromecánicas de sistemas de mando

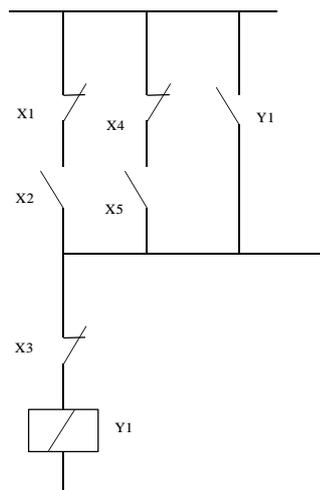
## Definición del sistema de control

- Esquemas de relés
  - representación de sistemas sencillos: señales lógicas binarias toda-nada (digitales), o bloques secuenciales predefinidos, como temporizadores y contadores.
  - Deficiencias en cuanto a la representación de funciones secuenciales complejas y sobre todo en la manipulación de señales digitales de varios bits (por ejemplo, las obtenidas de una conversión A/D).
  - Utilizado por electricistas o ingenieros con formación eléctrica

## Definición del sistema de control

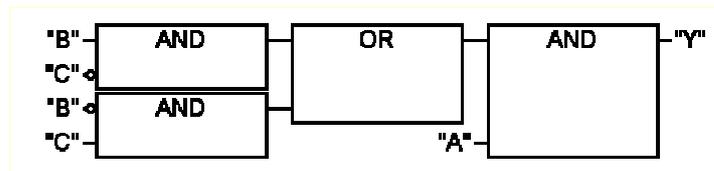
- Esquema de relés

Función Lógica :  
$$[(\bar{X}1 \cdot X2) + (\bar{X}4 \cdot X5) + Y1] \cdot \bar{X}3 = Y1$$



## Definición del sistema de control

- Diagramas lógicos
  - Este tipo de representación del sistema de control se utiliza en la documentación de las especificaciones y representación de los mismos
  - En muy pocos casos los usuarios finales de autómatas utilizan esta representación



## Definición del sistema de control

- Diagramas de flujo
  - Organigrama y flujograma, es un sistema de representación que se basa en una serie de símbolos que según un convenio establecido tienen un determinado significado.



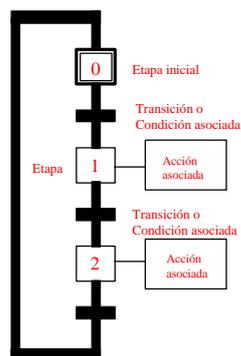
- 2 Niveles :
  - Nivel 1:
    - secuencia de acciones a realizar
    - representar el funcionamiento general del sistema
  - Nivel 2:
    - especificadas las acciones en forma de instrucción: instrucciones entendibles directamente por el autómata o en forma de funciones lógicas.

## Definición del sistema de control

- Grafcet (Graphe de Comande Etape Transition, cuya traducción literal es Gráfico de Orden Etapa Transición)
  - “es una secuencia de etapas que tienen asociadas unas determinadas acciones a realizar sobre el proceso junto con las condiciones o transiciones que provocan que se produzca el paso de una etapa a otra”
  - Normalizado: International Electrotechnical Commission IEC 848
  - Una de las mejores herramientas para representar automatismos secuenciales

## Definición del sistema de control

- Grafcet



## Programación de autómatas: Introducción al Grafcet

- Introducción a la programación del autómata
- **Definir las variables que intervienen y asignarles direcciones de memoria** 
- Lenguajes de programación
- Grafcet

## Definir las variables que intervienen y asignarles direcciones de memoria

- Definir las variables que intervienen y asignarles direcciones de memoria
  - Con los métodos anteriores:
    - descripción funcional del sistema de control
    - fijadas las entradas y salidas que tendrán nombres simbólicos
  - Ejemplo:

*“Control de una puerta corredera accionada por medio de un motor. El contactor S1 produce la apertura de la puerta, el contactor S2 controla el cierre de la puerta. El interruptor E3 de final de carrera se activa cuando la puerta esta abierta, y el interruptor E2 de fin de carrera se activa cuando la puerta esta cerrada.....”*

## Definir las variables que intervienen y asignarles direcciones de memoria

*La puerta se abre al aplicar una determinada presión sobre un sensor de paso de vehículos E1 situado enfrente de la puerta. Si el sensor E1 no se activa, la puerta se cierra—después de transcurridos 10 segundos. Si se activa E1, se cierra el contactor S2 y se mantiene cerrado hasta que el interruptor E3 de final de carrera desactive el contactor S2. Cuando se esta abriendo la puerta, o bien cuando una vez abierta haya detectado un vehículo con el sensor E1, el temporizador T1 no se activa. Si no se dan estas circunstancias y la puerta esta abierta E3 activado, se activa el temporizador T1, y transcurridos 10 segundos, la puerta se cierra mediante el contactor S1 por el temporizador T1. La acción de cerrar se produce hasta que o bien se detecta fin de carrera E2 o bien se detecta otro vehículo mediante la activación de E1 en cuyo caso se abre la puerta activando para ello el contactor S2.*

*Las lámparas LED1 y LED2 indican cuando se está cerrando o abriendo la puerta respectivamente."*

## Definir las variables que intervienen y asignarles direcciones de memoria

- Tabla de asignación de variables

Tipo	Termino	Símbolo	Descripción
Entradas	E1	VEHICULO	Sensor presencia vehículo
	E2	CERRADA	Límite puerta cerrada
	E3	ABIERTA	Límite puerta abierta
Salidas	S1	CERRAR	Contactor cerrar
	S2	ABRIR	Contactor abrir
	LED1	LEDCER	Lampara puerta cerrándose
	LED2	LEDABI	Lámpara puerta abriéndose
Temporizador	T1	TIEMPO	Temporizador 10 seg

## Definir las variables que intervienen y asignarles direcciones de memoria

- Posteriormente se asignan las direcciones físicas a estas variables

Tipo	Termino	Símbolo	Dirección
Entradas	E1	VEHICULO	I0.0
	E2	CERRADA	I0.1
	E3	ABIERTA	I0.2
Salidas	S1	CERRAR	Q0.0
	S2	ABRIR	Q0.1
	LED1	LEDCER	Q0.2
	LED2	LEDABI	Q0.3
Temporizador	T1	TIEMPO	T5

- Además de estas variables: necesitare un contador que cuente los periodos de temporización. En las tablas anteriores este contador no se ha tenido en consideración.

## Programación de autómatas: Introducción al Grafcet

- Introducción a la programación del autómata
- Definir las variables que intervienen y asignarles direcciones de memoria
- Lenguajes de programación** 
  - Lista de instrucciones
  - Diagramas de contactos y funciones
  - Lenguajes de alto nivel
- Grafcet

## Lenguajes de programación

- “Conjunto de instrucciones y/o símbolos que están disponibles para escribir el programa, se denomina lenguaje de programación”

Tipo	Lenguajes de programación
Algebraicos	Lenguajes booleanos
	Lista de instrucciones
	Lenguajes de alto nivel
Gráficos	Diagrama de contactos
	Diagrama de funciones
	Intérprete Grafcet

- Si unidad de programación acepta directamente la representación del sist. de control, se carga en la CPU. Si no hay que traducirla al lenguaje de programación pertinente.

## Lista de instrucciones

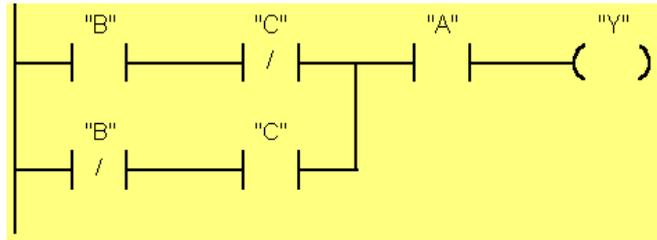
- “Es un lenguaje booleano al cual se le ha añadido una serie de instrucciones que permiten la manipulación de datos y variables digitales y gestión del programa.”

$$Y = (B\bar{C} + \bar{B}C) \cdot A$$

```
LD  "B"
AN  "C"
LDN B"
A   "C"
OLD
A   "A"
=   "Y"
```

## Diagrama de contactos. Diagrama de funciones

- Es un lenguaje gráfico procedente del lenguaje de relés que en base símbolos que representan contactos, bloques funcionales, etc. codifican la secuencia de control.



- Diagrama de funciones: deriva de la representación lógica utilizada por los técnicos en electrónica digital

## Lenguajes de alto nivel

- Lista de instrucciones, diagramas de contactos, ... Insuficientes para programar aplicaciones complejas para autómatas de gama media y alta.
- Herramientas de ingeniería:
  - S7-SCL: lenguaje de alto nivel que se asemeja al Pascal
  - M7-ProC/C++: Entorno Borland C/C++ y depurador multitarea.
- Casi todos los fabricantes de autómatas este tipo de herramientas que permiten programar en Basic, C, Pascal, Fortran, etc. mediante el uso de los intérpretes o compiladores respectivos. Estas herramientas permiten resolver tareas de cálculo científico de alta resolución, clasificaciones y tratamiento de datos, estadísticas, etc., con acceso a subrutinas específicas de lenguajes informáticos.

## Programación de autómatas: Introducción al Grafcet

- Introducción a la programación del autómata
- Definir las variables que intervienen y asignarles direcciones de memoria
- Lenguajes de programación
- Grafcet
  - **Introducción** 
  - Símbolos normalizados
  - Reglas de evolución del Grafcet
  - Posibilidades de representación de automatismos con Grafcet
  - Implementación del Grafcet sobre autómatas programables
  - Niveles de Grafcet
  - Representación de situaciones especiales en Grafcet

## Grafcet Introducción

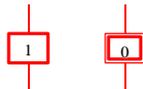
- El Grafcet es un método gráfico de modelado de sistemas de control secuenciales
- Surgió en Francia a mediados de los años 70, y fue creado por una agrupación de algunos fabricantes de autómatas, en concreto **Telemecanique** y **Aper**, junto con dos organismos oficiales, **AFCET** (Asociación Francesa para la Cibernética, Economía y Técnica) y **ADEPA** (Agencia Nacional para el Desarrollo de la Producción Automatizada).
- Fue homologado en Francia (NFC), Alemania (DIN), y con posterioridad por la Comisión Electrotecnia Internacional ( **IEC 848**, en 1998).
- Describe la evolución de un proceso que se pretende controlar, indicando las acciones que hay que realizar sobre dicho proceso y que informaciones provocan el realizar una u otra acción

## Programación de autómatas: Introducción al Grafcet

- Introducción a la programación del autómata
- Definir las variables que intervienen y asignarles direcciones de memoria
- Lenguajes de programación
- Grafcet
  - Introducción
  - **Símbolos normalizados** ←
  - Reglas de evolución del Grafcet
  - Posibilidades de representación de automatismos con Grafcet
  - Implementación del Grafcet sobre autómatas programables
  - Niveles de Grafcet
  - Representación de situaciones especiales en Grafcet

## Grafcet Símbolos normalizados

- Etapas
  - La evolución de un proceso representada mediante un gráfico Grafcet, esta formada por una sucesión de etapas que representan cada uno de sus estados, llevando cada una de ellas asociada una o varias acciones a realizar sobre el proceso.
  - Las etapas se representan con un cuadro y un número o símbolo con un subíndice numérico en su interior, en ambos casos el número indica el orden que ocupa la etapa dentro del Grafcet
  - Las etapas iniciales, aquellas en las que se posiciona el sistema al iniciarse el proceso, se representan con un cuadro doble.



## Grafcet

### Símbolos normalizados

- Acción asociada
  - Son una o varias acciones a realizar sobre el proceso, cuando la etapa de la cual dependen dichas acciones se encuentra activada.
  - Dichas acciones correspondientes a una etapa, se simbolizan mediante rectángulos conectados y situados a la derecha de dicha etapa. En el interior de estos rectángulos se indica, bien de forma literal, bien de forma simbólica, las acciones a realizar.



## Grafcet

### Símbolos normalizados

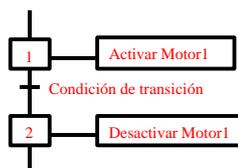
- En una primera clasificación se puede dividir las acciones en dos tipos :
  - Incondicionales : acciones que se ejecutan con solo quedar activadas las etapas correspondientes.
  - Condicionales : son las acciones que necesitan el cumplimiento de una condición además de la propia activación de la etapa correspondiente.



## Grafcet Símbolos normalizados

- Clasificar las acciones en :
  - **Internas** : acciones que se producen en el equipo de control, por ejemplo temporizaciones, contadores, cálculos matemáticos, etc.
  - **Externas** : las acciones que se producen sobre el proceso, por ejemplo abrir o cerrar una válvula, activar o desactivar una bomba, etc.
- Transición y Condición de transición
  - En el diagrama Grafcet, un proceso se compone de una serie de etapas secuenciales que se activan una tras otra unidas mediante una transición.
  - El paso de una etapa a la siguiente se realiza dependiendo de si se cumple o no la condición de transición entre ellas.

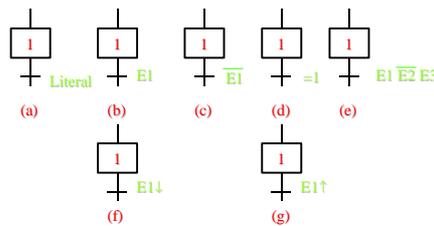
## Grafcet Símbolos normalizados



- Toda transición lleva asociada una condición de transición o función lógica booleana que se denomina **receptividad**, y que puede ser verdadera o falsa.
- Se dice que la transición está validada, cuando la etapa o etapas anteriores a la transición están activadas. El franqueamiento de la transición se producirá si, y sólo si, la transición esta validada y la receptividad es verdadera.

## Grafcet Símbolos normalizados

- Diferentes formas de anotar la receptividad:



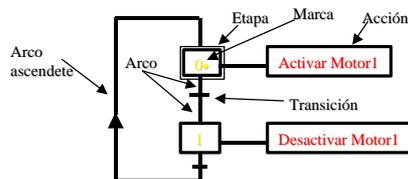
- a) Descripción literal.
- b) Condición de transición activa.
- c) Condición de transición inactiva.
- d) Incondicional, siempre se activa la etapa siguiente.
- e) Condición de transición en forma de función lógica de varias variables.
- f) Condición de transición de flanco descendente, la señal pasa de 1 a 0.
- g) Condición de transición de flanco ascendente, la señal pasa de 0 a 1.

- Arco

- Es el segmento de recta que una transición (con su condición de transición) con una etapa y viceversa, pero nunca dos elementos iguales entre sí. Los arcos pueden ser o verticales u horizontales, además los arcos verticales deben llevar una flecha indicando su sentido en el caso de ser este ascendente

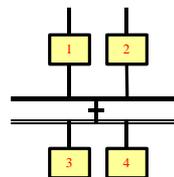
## Grafcet Símbolos normalizados

- Esquema de elementos que componen el Grafcet



La situación de etapa activada, se indica mediante la colocación de una marca en el interior del gráfico representativo de la etapa

- Líneas paralelas (conurrencia)



## Programación de autómatas: Introducción al Grafcet

- Introducción a la programación del autómata
- Definir las variables que intervienen y asignarles direcciones de memoria
- Lenguajes de programación
- Grafcet
  - Introducción
  - Símbolos normalizados
  - **Reglas de evolución del Grafcet** 
  - Posibilidades de representación de automatismos con Grafcet
  - Implementación del Grafcet sobre autómatas programables
  - Niveles de Grafcet
  - Representación de situaciones especiales en Grafcet

## Grafcet Reglas de evolución del Grafcet

- Reglas
  - La etapa inicial de un Grafcet se activan de forma incondicional. Esta situación inicial se corresponde en general con una situación de reposo.
  - Una transición esta en disposición de ser validada cuando todas las etapas inmediatamente precedentes, unidas a dicha transición, están activadas. La activación de una transición se produce cuando está validada y la condición de transición o receptividad es verdadera. Se podría definir una etapa como activable cuando la transición precedente esta validada.

## Grafcet

### Reglas de evolución del Grafcet

- Reglas
  - Franquear una transición implica la activación de todas las etapas siguientes inmediatas, y la desactivación de las inmediatas precedentes.
  - Transiciones conectadas en paralelo, se activan de forma simultánea si se cumplen las condiciones para ello.
  - Una o varias acciones se asocian a cada etapa. Estas acciones sólo están activas cuando la etapa esta activa.

## Programación de autómatas: Introducción al Grafcet

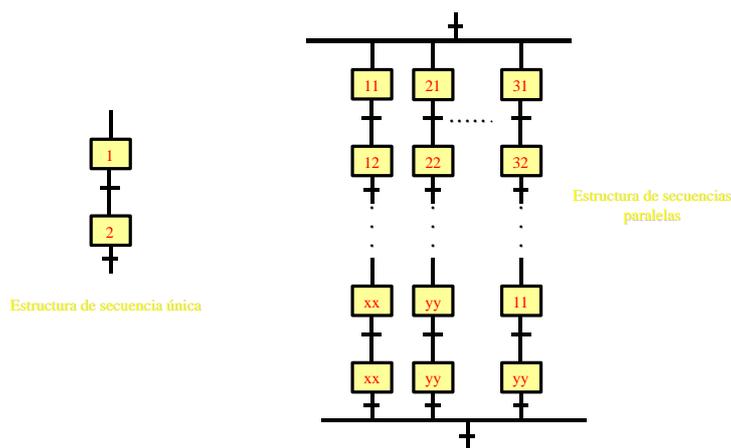
### Grafcet

- Grafcet
  - Introducción
  - Símbolos normalizados
  - Reglas de evolución del Grafcet
  - Posibilidades de representación de automatismos con Grafcet
    - Estructuras base 
    - Estructuras lógicas
    - Ejemplo
  - Implementación del Grafcet sobre autómatas programables
  - Niveles de Grafcet
  - Representación de situaciones especiales en Grafcet

## Grafcet Estructuras base

- Grafcet soporta diferentes tipos de estructura secuencial:
  - Estructura base: trata conceptos de secuencialidad y concurrencia
  - Estructura lógica: trata conceptos de concatenación de estructuras
- Estructuras base
  - Estructuras de **secuencia única**
    - Son estructuras formadas por secuencias de etapas que se van activando una tras otra, sin interacción con ninguna otra estructura
  - Estructuras de **secuencia paralela**
    - Son un conjunto de estructuras únicas activadas por una misma transición de forma simultánea. Después de la activación de las distintas secuencias su evolución se produce de forma independiente

## Grafcet Estructuras base



# Programación de autómatas: Introducción al Grafcet

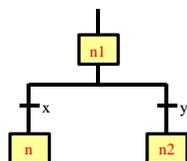
## Grafcet

- Grafcet
  - Introducción
  - Símbolos normalizados
  - Reglas de evolución del Grafcet
  - Posibilidades de representación de automatismos con Grafcet
    - Estructuras base
    - **Estructuras lógicas** ←
    - Ejemplo
  - Implementación del Grafcet sobre autómatas programables
  - Niveles de Grafcet
  - Representación de situaciones especiales en Grafcet

## Grafcet

### Estructuras lógicas

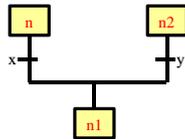
- Funciones lógicas OR, AND y saltos condicionales
- Divergencia OR
  - Se utiliza cuando lo que se trata es de modelar la posibilidad de tomar dos o más secuencias alternativas a partir de una etapa común.



La etapa **n** pasará a estar activa si estando activa la etapa **n1**, se satisface la condición de transición o receptividad **x**. De igual forma la etapa **n2** pasará a estar activa si estando activa la etapa **n1** se satisface la condición de transición o receptividad **y**.

## Grafcet Estructuras lógicas

- Convergencia OR



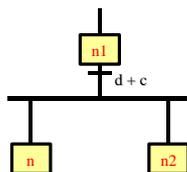
la etapa **n1** pasará a estar activa, si estando activa la etapa **n** se satisface la condición de transición o receptividad **x**; o si estando activa la etapa **n2** se satisface la condición de transición o receptividad **y**.

- Divergencia AND

- Permite la implementación de procesos concurrentes síncronos, de forma que dos o más subprocesos del sistema, representados por las secuencias paralelas, pueden activarse de forma sincronizada.

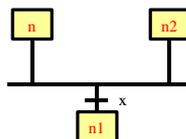
## Grafcet Estructuras lógicas

- Grafcet. Divergencia en AND



La etapa **n2** y **n** pasarán al estado activo, si estando activa la etapa **n1** se satisface la condición de transición o receptividad **d+c**

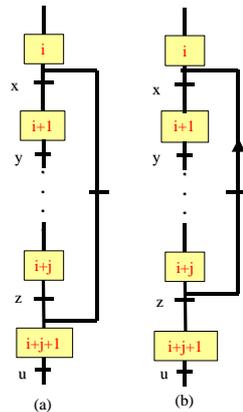
- Convergencia en AND



La etapa **n1** pasará a estar activa, si estando las etapas **n-1** y **n-2** activas se satisface la condición de transición o receptividad **x**

## Grafcet Estructuras lógicas

### ■ Saltos Condicionales



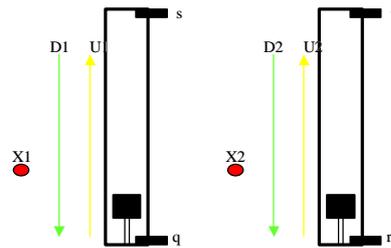
- (a) se implementa un salto condicional a la etapa  $i+j+1$  si esta activada la etapa  $i$  y se cumple la condición de transición o receptividad
- (b) se implementa un bucle que permite la repetición de la secuencia de etapas hasta que  $x$  sea igual a 1.

## Programación de autómatas: Introducción al Grafcet Grafcet

- Grafcet
  - Introducción
  - Símbolos normalizados
  - Reglas de evolución del Grafcet
  - Posibilidades de representación de automatismos con Grafcet
    - Estructuras base
    - Estructuras lógicas
    - **Ejemplo** ←
  - Implementación del Grafcet sobre autómatas programables
  - Niveles de Grafcet
  - Representación de situaciones especiales en Grafcet

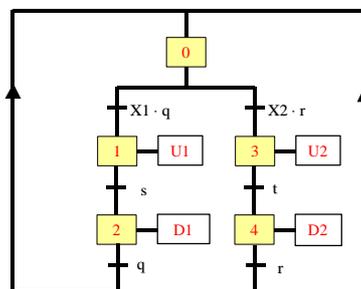
## Grafcet Ejemplo

- El siguiente ejemplo, dos montacargas se mueva cuando se pulsa un botón (x1 o x2) en sentido ascendente hasta el final de recorrido (s,t), y de inmediato se descendiende a la situación inicial (q,r).



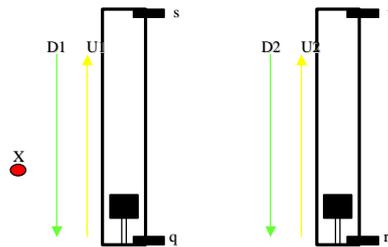
## Grafcet Ejemplo

- Primer supuesto:
  - La pulsación de x1 o x2 deberá iniciar el movimiento ascendente del montacargas m1 o m2 respectivamente. Sólo un montacargas debe estar en funcionamiento a la vez. También inicialmente se supone que el accionamiento simultáneo de los dos pulsadores no puede ocurrir.



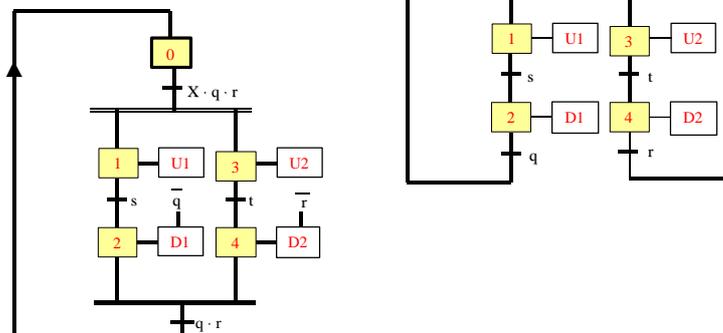
## Grafcet Ejemplo

- Variación: se requiere un sólo pulsador X para iniciar el movimiento de los dos montacargas, sincronismo en el inicio del movimiento y la única restricción que se impone es que para cada ciclo de funcionamiento ambos montacargas deben estar situados en su posición inicial (q y r). Además se deja abierta la posibilidad de que los dos montacargas posean movimientos con distintas velocidades.



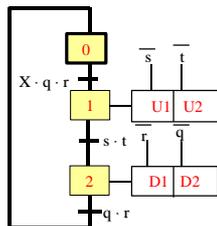
## Grafcet Ejemplo

- Ejemplo con un solo pulsador
- Con acciones condicionales

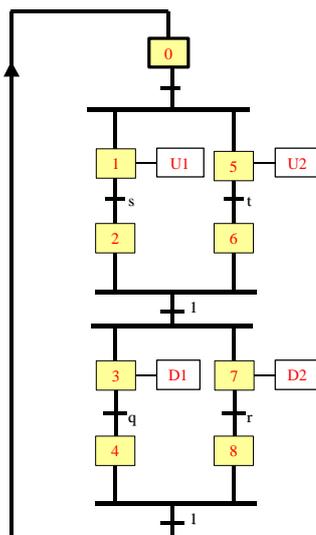


## Grafcet Ejemplo

- Si en el ejemplo anterior se pretende que exista también sincronismo en el movimiento de descenso del montacargas, para ello el sistema debe esperar a que ambos montacargas se encuentren en la posición (s, t) antes de iniciar el movimiento de descenso simultáneo hasta la situación inicial (q, r).



## Grafcet Ejemplo



## Programación de autómatas: Introducción al Grafcet

### Grafcet

- Grafcet
  - Introducción
  - Símbolos normalizados
  - Reglas de evolución del Grafcet
  - Posibilidades de representación de automatismos con Grafcet
  - **Implementación del Grafcet sobre autómatas programables** ←
  - Niveles de Grafcet
  - Representación de situaciones especiales en Grafcet

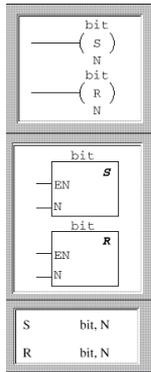
## Grafcet

### Implementación del Grafcet sobre A.P.I

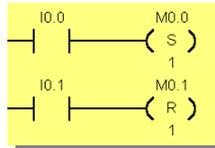
- Diagrama Grafcet que representa el proceso  $\Rightarrow$  en un algoritmo de control y su posterior programación sobre un API.
- Para ello a cada una de las etapas en las que se divide el Grafcet se le asocia una variable interna.
- La condición de transición es la encargada de activar la etapa siguiente y desactivar la anterior; para ello se utilizan las instrucciones Set y Reset que poseen todos los autómatas programables.

## Grafcet Implementación del Grafcet sobre A.P.I

- En el Siemens S7-200 las instrucciones SET y RESET son:



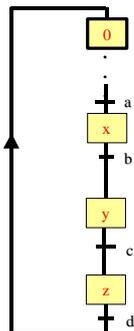
- Cuando se ejecutan las operaciones "Set" (Poner a 1 (S)) y "Reset" (Poner a 0 (R)), se activa (se pone a 1) o se desactiva (se pone a 0) el número indicado de salidas (N) a partir del valor indicado por el bit o por el parámetro OUT. El margen de E/S que se pueden poner a 0 está comprendido entre 1 y 255.
- Las instrucciones Set y Reset se utilizan para activar o desactivar las variables internas que en este autómata son las marcas internas (M0.0, M0.1, ..., etc.).



Autómatas Programables  
ISA-UMH © TDOC-99

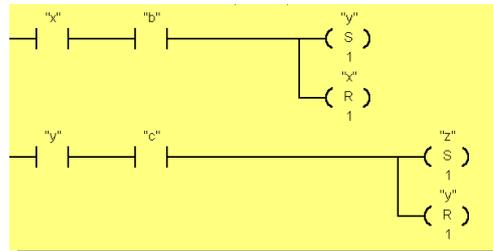
59

## Grafcet Implementación del Grafcet sobre A.P.I



Activa la etapa "y" y desactiva la etapa "x"

Activa la etapa "z" y desactiva la etapa "y"



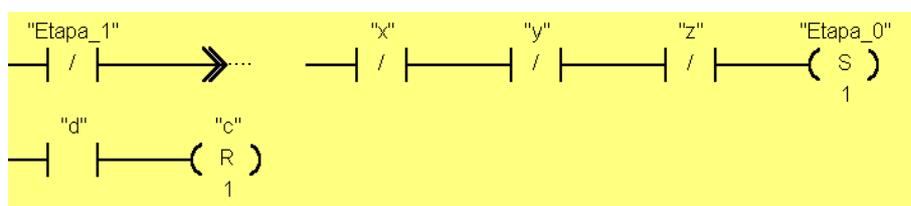
Autómatas Programables  
ISA-UMH © TDOC-99

60

## Grafcet

### Implementación del Grafcet sobre A.P.I

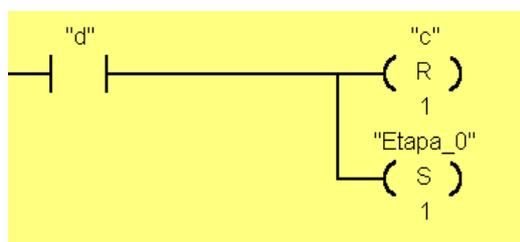
- Para activar la etapa inicial "0" existen diferentes formas :
  - Como última línea del programa de control del autómatas, se pone a cero "Reset" de la marca asignada a la última etapa, siempre que la condición de transición se cumpla. Además añadimos una línea más donde se active la marca correspondiente a la etapa inicial siempre que todas las etapas anteriores estén desactivadas.



## Grafcet

### Implementación del Grafcet sobre A.P.I

- Otra posibilidad es que cuando se cumpla la última condición de transición se desactive la última etapa activa y se active la etapa inicial.

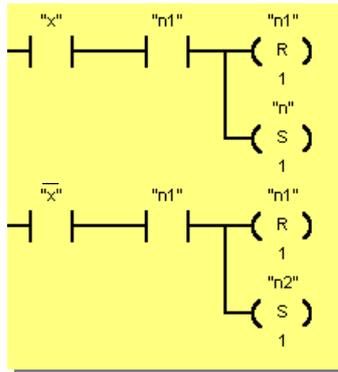
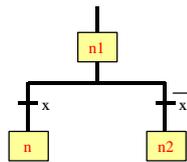


- Existe otra posibilidad de activación de la etapa inicial, que se realiza mediante el uso de variables internas del sistema. Esta última opción depende del tipo de autómatas utilizado.

## Grafcet

### Implementación del Grafcet sobre A.P.I

- Varios ejemplos de cómo codificar en lenguajes de contactos algunos casos que se pueden dar en diagramas Grafcet
  - Divergencia OR



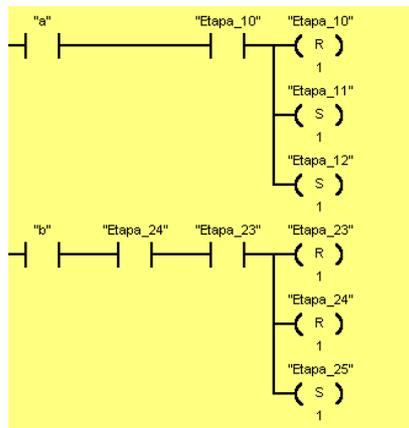
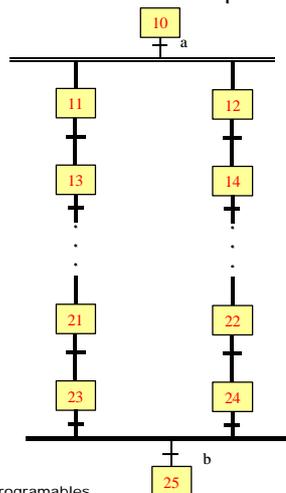
Autómatas Programables  
ISA-UMH © TDOC-99

63

## Grafcet

### Implementación del Grafcet sobre A.P.I

- caso de secuencias paralelas

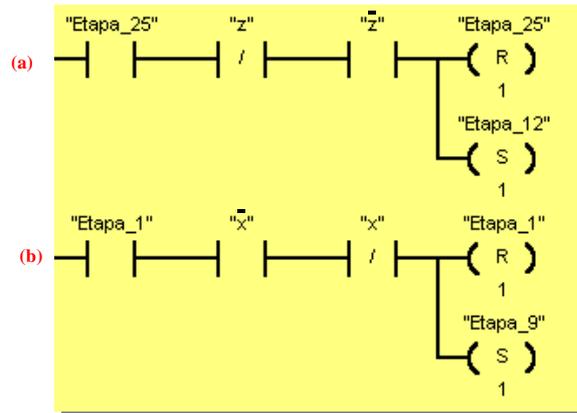
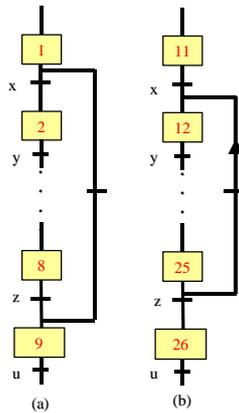


Autómatas Programables  
ISA-UMH © TDOC-99

64

## Grafcet Implementación del Grafcet sobre A.P.I

- Saltos condicionales a otras etapas



## Grafcet Implementación del Grafcet sobre A.P.I

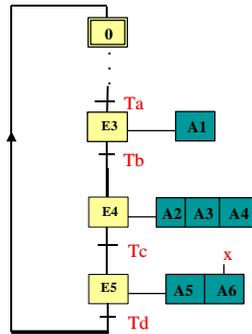
- Mediante funciones lógicas en un diagrama de escalera
  - La **ecuación de activación o desactivación asociada a la etapa** enésima :

$$E_n = E_{n-1} \cdot T_{n-1} + \overline{E_{n+1}} \cdot E_n$$

*"La etapa  $E_n$  se activará, si estando activada la etapa  $E_{n-1}$  y desactivada la  $E_{n+1}$  se satisface la transición  $T_{n-1}$ . Apartir de este instante permanecerá activada hasta que se active la  $E_{n+1}$ "*

- Aparte se tiene que afrontar las **ecuaciones de activación de las acciones asociadas a las etapas**. La expresión de estas ecuaciones dependerá del carácter condicional o incondicional de estas acciones.

## Grafcet Implementación del Grafcet sobre A.P.I



### Etapa número 3

$$E_3 = E_2 \cdot T_a + E_3 \overline{E_4}$$

$$A_1 = E_3$$

### Etapa número 4

$$E_4 = E_3 \cdot T_b + E_4 \overline{E_5}$$

$$A_2 = E_4$$

$$A_3 = E_4$$

$$A_4 = E_4$$

### Etapa número 5

$$E_5 = E_4 \cdot T_c + E_5 \overline{E_6}$$

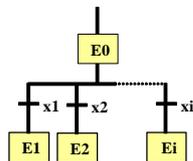
$$A_5 = E_5$$

$$A_6 = x \cdot E_5$$

## Grafcet Implementación del Grafcet sobre A.P.I

- Si Grafcet → estructuras lógicas, saltos, etc → Fun. Log. + Complejas
- Partiendo de Ecuac. General:  $E_n(t + \Delta t) = Cond_A + Cond_D \cdot E_n(t)$ 
  - Cond<sub>A</sub>: condición de activación
  - Cond<sub>D</sub>, condición de mantenimiento y desactivación

- DIVERGENCIA OR



$$Cond_D = \overline{(E_1 + E_2 + \dots + E_i)} = \prod_{n=1}^i \overline{E_n}$$

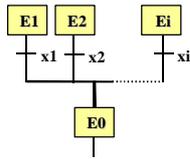
MORGAN

$$Cond_D = \prod_{n=1}^i \overline{E_n}$$

$$E_0(t + \Delta t) = Cond_A + \prod_{n=1}^i \overline{E_n} \cdot E_0(t)$$

## Grafcet Implementación del Grafcet sobre A.P.I

### ■ CONVERGENCIA OR

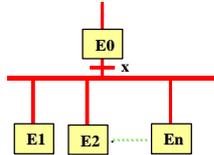


$$Cond_A = (E_1 \cdot x_1 + E_2 \cdot x_2 + \dots + E_j \cdot x_j) = \sum_{n=1}^i E_n \cdot x_n$$



$$E_0(t + \Delta t) = \sum_{n=1}^i E_n \cdot x_n + Cond_D \cdot E_0(t)$$

### ■ DIVERGENCIA AND



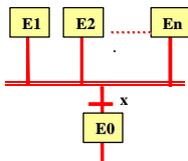
$$Cond_D = \prod_{n=1}^i E_n$$



$$E_0(t + \Delta t) = Cond_A + \prod_{n=1}^i E_n \cdot E_0(t)$$

## Grafcet Implementación del Grafcet sobre A.P.I

### ■ CONVERGENCIA AND



$$Cond_A = \left( \prod_{n=1}^i E_n \right) \cdot x$$



$$E_0(t + \Delta t) = \left( \prod_{n=1}^i E_n \right) \cdot x + Cond_D \cdot E_0(t)$$

## Grafcet

### Implementación del Grafcet sobre A.P.I

- Etapas iniciales
  - La etapa/as inicial/es deben quedar activadas cuando se inicializa el sistema.
  - las etapas iniciales no tienen ninguna etapa ni transición precedente que puedan incluirse en las condiciones de activación (CondA) de la expresión general.
  - Para solucionar este problema se introducen unas condiciones de activación iniciales CondAI, adicionales a las condiciones de activación CondA.
  - Este tipo de etapas se implementan, mediante la siguiente expresión:

$$E_0(t) = Cond_{AI} + Cond_A$$

## Grafcet

### Implementación del Grafcet sobre A.P.I

- La expresión de las condiciones de activación iniciales se puede obtener de dos formas:
  - Aprovechando el hecho de que, en la fase de inicialización del programa las variables están inicializadas a cero y por tanto, también todas las variables internas representativas de las diversas etapas existentes estarán inicializadas a cero.

$$Cond_{AI} = \prod_{n=1}^i \overline{E_n} \quad \longrightarrow \quad E_0(t) = \prod_{n=1}^i \overline{E_n} + Cond_A$$

Esta expresión puede llegar a ser laboriosa de implementar, si el Grafcet que se está realizando dispone de un gran número de etapas.

## Grafcet Implementación del Grafcet sobre A.P.I

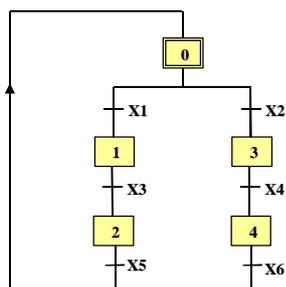
- Otra posibilidad, aparece por el hecho de que los autómatas programables disponen de una serie de marcas del sistema o marcas especiales. Existe una de ellas que está activa durante el primer ciclo de programa y desactiva en los restantes.

$$Cond_{AI} = SM$$



$$E_0(t + \Delta t) = SM + Cond_D \cdot E_0(t) + Cond_A$$

## Grafcet Implementación del Grafcet sobre A.P.I



Ejemplo de implementar las etapas iniciales :

a)  $E_0(t) = \overline{E_1} \cdot E_2 \cdot E_3 \cdot E_4 + E_2 \cdot X5 + E_4 \cdot X6$

b)  $E_0(t + \Delta t) = SM + (\overline{E_1} + E_3) \cdot E_0(t) + E_2 \cdot X5 + E_4 \cdot X6$

ó

$$E_0(t + \Delta t) = SM + (\overline{E_1} \cdot \overline{E_3}) \cdot E_0(t) + E_2 \cdot X5 + E_4 \cdot X6$$

## Programación de autómatas: Introducción al Grafcet Grafcet

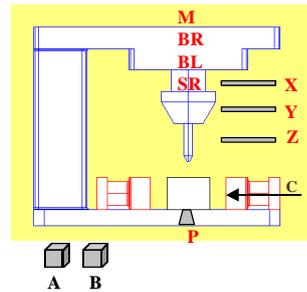
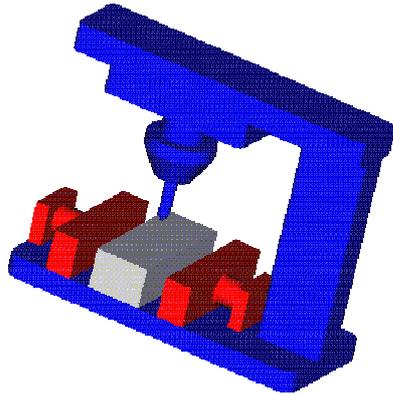
- Grafcet
  - Introducción
  - Símbolos normalizados
  - Reglas de evolución del Grafcet
  - Posibilidades de representación de automatismos con Grafcet
  - Implementación del Grafcet sobre autómatas programables
  - **Niveles de Grafcet** ←
  - Representación de situaciones especiales en Grafcet

## Grafcet Niveles de Grafcet

- A la hora de representar mediante un grafcet el sistema de control es conveniente estructurarlo dos niveles :
  - **Nivel 1** : En este nivel se representa solamente el funcionamiento lógico del sistema en una terminología próxima al lenguaje corriente, se realiza independientemente de las decisiones que a posteriori se tomen en cuanto a la tecnología a utilizar, a la nomenclatura y tipo de variables seleccionadas, etc
  - **Nivel 2** : En este nivel se tienen en cuenta las decisiones tecnológicas tomadas, la nomenclatura y tipo de variables, etc. y se hace referencia a ellas en el diagrama grafcet de este nivel

## Grafcet Niveles de Grafcet

- Ejemplo: Sistema de control de una máquina taladradora



Autómatas Programables  
ISA-UMH © TDOC-99

77

## Grafcet Niveles de Grafcet

- Funcionamiento de la taladradora:
  - existe un pulsador "B", de inicialización del sistema, con objeto de que el motor adquiera una velocidad de giro de régimen permanente, que se obtiene accionando M. El taladro posee varias velocidades en el sentido longitudinal del eje, léase bajada lenta del utensilio del taladro BL, bajada rápida BR y subida rápida SR.
  - La pieza en la que se va a realizar el taladro se detecta mediante un detector inductivo P, y se sujeta mediante dos sujeciones accionadas por C. La tarea de realizar un taladro sigue la siguiente secuencia: primero se detecta la pieza mediante el detector inductivo, posteriormente se pulsa el botón "A" de inicio de operación con lo que actúan las sujeciones de la pieza y al mismo tiempo se inicia el descenso rápido de la broca "BR".
  - Antes de empezar a realizar el taladro propiamente dicho a la pieza, el detector "Y" provoca el paso de descenso rápido de la broca a descenso lento "BL", el cual se interrumpe cuando se detecta el final de carrera "Z". Inmediatamente se produce la subida rápida de la broca hasta alcanzar la posición de reposo "X".

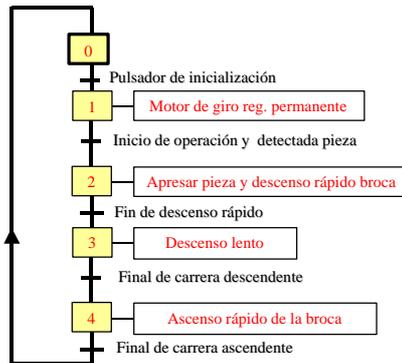
Autómatas Programables  
ISA-UMH © TDOC-99

78

# Grafcet

## Niveles de Grafcet

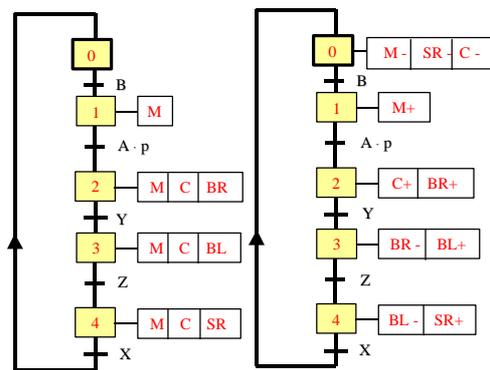
- Grafcet Nivel 1



# Grafcet

## Niveles de Grafcet

- Grafcet Nivel 2



(a) Pr. monoestable

(b) Pr. biestable

# Programación de autómatas: Introducción al Grafcet

## Grafcet

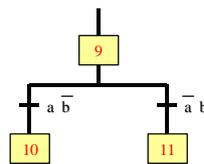
- Grafcet
  - Introducción
  - Símbolos normalizados
  - Reglas de evolución del Grafcet
  - Posibilidades de representación de automatismos con Grafcet
  - Implementación del Grafcet sobre autómatas programables
  - Niveles de Grafcet
  - **Representación de situaciones especiales en Grafcet**



## Grafcet

### Representación de situaciones especiales en Grafcet

- Secuencias exclusivas



- Temporizadores y contadores
  - Función temporización en Grafcet se implementa a través del operador de temporización

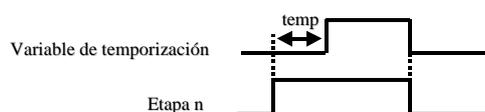
**t / n / temp**

- El valor de la temporización en seg, min u hor.
- Etapa a la cual esta referida la temporización.
- Indica la operación de temporización.

## Grafcet

### Representación de situaciones especiales en Grafcet

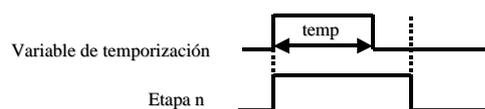
- Clasificación de las temporizaciones:
  - **Temporización con retardo:** Cada dispositivo de temporización tiene asociada una variable que en este tipo de temporización permanece en estado bajo "0", mientras que se efectúa la operación de temporización.



## Grafcet

### Representación de situaciones especiales en Grafcet

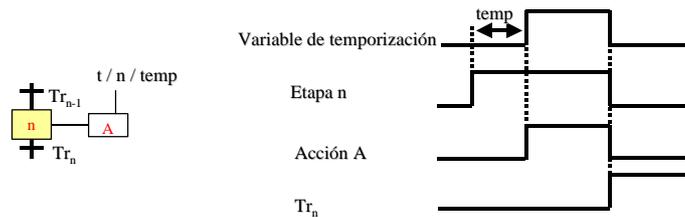
- **Temporización al arranque:** Cada dispositivo de temporización tiene asociada una variable que en este tipo de temporización permanece en estado lógico alto "1", desde el mismo comienzo de la temporización, pasando a estado lógico bajo "0" cuando ha transcurrido la temporización establecida. Su funcionamiento es el mismo que la temporización con retardo si la variable asociada al temporizador es complementada, por lo tanto este tipo de temporización se denota complementando la notación que representa la temporización con retardo.



## Grafcet

### Representación de situaciones especiales en Grafcet

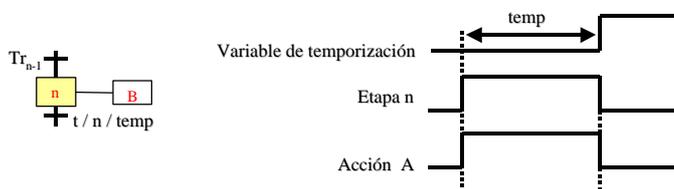
- Incorporación de las temporizaciones al Grafcet:
  - **Temporización de acciones:** Es el caso en el cual se pretende temporizar la ejecución de la acción asociada a una etapa, de forma que no se ejecute la acción hasta que transcurra un cierto instante de tiempo.



## Grafcet

### Representación de situaciones especiales en Grafcet

- **Temporización de la transición de una etapa:** Es el caso en el cual la receptividad asociada a una transición depende de que la variable de temporización sea activada. Si se trata de una temporización con retardo esta transición no será superada hasta que transcurra un cierto instante de tiempo.



## Grafcet

### Representación de situaciones especiales en Grafcet

- Contadores
  - Hay tres tipos, los que realizan la operación cuenta de forma ascendente, descendente o ambas a la vez.

