

RESPUESTAS

3.1 El programa `motorIL` controla la conexión/desconexión de un motor Dahlander. Actuando sobre el pulsador `S1` conectado a la entrada `0.00` del PLC, el usuario puede: (a) conectar el motor para que gire a velocidad baja (contactor conectado a la salida `0.00` del PLC), (b) conectar el motor para que gire a velocidad alta (contactores conectados a las salidas `0.01` y `0.02` del PLC) o (c) desconectar el motor.

```

1 PROGRAM motorIL
2 VAR
3     S1 AT %IX0.0 : BOOL;
4     KM1 AT %QX0.00 : BOOL;
5     KM2 AT %QX0.01 : BOOL;
6     KM3 AT %QX0.02 : BOOL;
7     T3s : TON;
8     x : ARRAY[1..8] OF BOOL := 1;
9     C : ARRAY[1..10] OF BOOL;
10 END_VAR
11 (*
12 * Actualización de las entradas.
13 *)
14 LD x[1]
15 AND S1
16 ST C[1]
17 LD x[2]
18 AND T3s.Q
19 ST C[2]
20 LD x[2]
21 ANDN S1
22 ST C[3]
23 LD x[3]
24 ANDN S1
25 ST C[4]
26 LD x[4]
27 AND T3s.Q
28 ST C[5]
29 LD x[4]
30 AND S1
31 ST C[6]
32 LD x[6]
33 ANDN S1
34 ST C[7]
35 LD x[5]
36 AND S1
37 ST C[8]
38 LD x[7]
39 AND S1
40 ST C[9]
41 LD x[8]
42 ANDN S1
43 ST C[10]
44 (*
45 * Actualización de los estados.
46 *)
47 LD C[10]
48 OR x[1]
49 ANDN C[1]
50 ST x[1]
51 LD C[1]
52 OR x[2]
53 ANDN C[2]
54 ANDN C[3]
55 ST x[2]
56 LD C[2]
57 OR x[3]
58 ANDN C[4]
59 ST x[3]
60 LD C[3]
61 OR x[4]
62 ANDN C[5]
63 ANDN C[6]
64 ST x[4]
65 LD C[4]
66 OR C[5]
67 OR x[5]
68 ANDN C[8]
69 ST x[5]
70 LD C[6]
71 OR x[6]
72 ANDN C[7]
73 ST x[6]
74 LD C[7]
75 OR x[7]
76 ANDN C[9]
77 ST x[7]
78 LD C[8]
79 OR C[9]
80 OR x[8]
81 ANDN C[10]
82 ST x[8]
83 (*
84 * Actualización de las salidas.

```

```

85 *)
86 LD x[3]
87 OR x[5]
88 ST KM1
89 LD x[6]
90 OR x[7]
91 ST KM2

92 ST KM3
93 LD x[2]
94 OR x[4]
95 ST T3s.IN
96 CAL T3s(PT:=t#3s)
97 END_PROGRAM

```

Este programa es una de las posibles implementaciones en PLC de la máquina secuencial binaria M que modela el comportamiento del sistema de eventos discretos S .

- (a) Dibuje el grafo asociado a M y escriba su matriz de incidencia.
- (b) Escriba las especificaciones funcionales del comportamiento del sistema S . Lo que se está pidiendo es la descripción del comportamiento manifiesto del sistema, es decir, una lista numerada de especificaciones donde se detalle cómo se conecta/desconecta el motor Dahlander en función de cómo actúe el usuario sobre el pulsador S_1

RESPUESTA:

- (a) Dibuje el grafo asociado a M y escriba su matriz de incidencia.

Podemos ayudarnos del siguiente diagrama LD que modela la lógica del programa IL anterior:

```

1 (* Actualización de las entradas.*)
2 | x[1] S1 C[1]
3 1 +---| |---| |------( )
4 |
5 (* CoDeSys no acepta este
6 segmento.*)
7 | x[2] T3s.Q C[2]
8 2 +---| |---+---| |------( )
9 | |
10 | | S1 C[3]
11 | +---|/|------( )
12 (* Segms. alternat. para CoDeSys.
13 | x[2] T3s.Q C[2]
14 +---| |---| |------( )
15 |
16 | x[2] S1 C[3]
17 +---| |---|/|------( )
18 *)
19 |
20 | x[3] S1 C[4]
21 3 +---| |---|/|------( )
22 |
23 |
24 | x[4] T3s.Q C[5]
25 4 +---| |---+---| |------( )
26 | |
27 | | S1 C[6]
28 | +---| |------( )

29 |
30 | x[5] S1 C[8]
31 5 +---| |---| |------( )
32 |
33 |
34 | x[6] S1 C[7]
35 +---| |---|/|------( )
36 |
37 |
38 | x[7] S1 C[9]
39 6 +---| |---| |------( )
40 |
41 | x[8] S1 C[10]
42 7 +---| |---|/|------( )
43 |
44 |
45 |
46 (* Actualización de los estados.*)
47 | C[10] C[1] x[1]
48 8 +---| |---+---|/|------( )
49 | x[1] |
50 +---| |---+
51 |
52 | C[1] C[2] C[3] x[2]
53 9 +---| |---+---|/|---|/|----( )
54 | x[2] |
55 +---| |---+
56 |

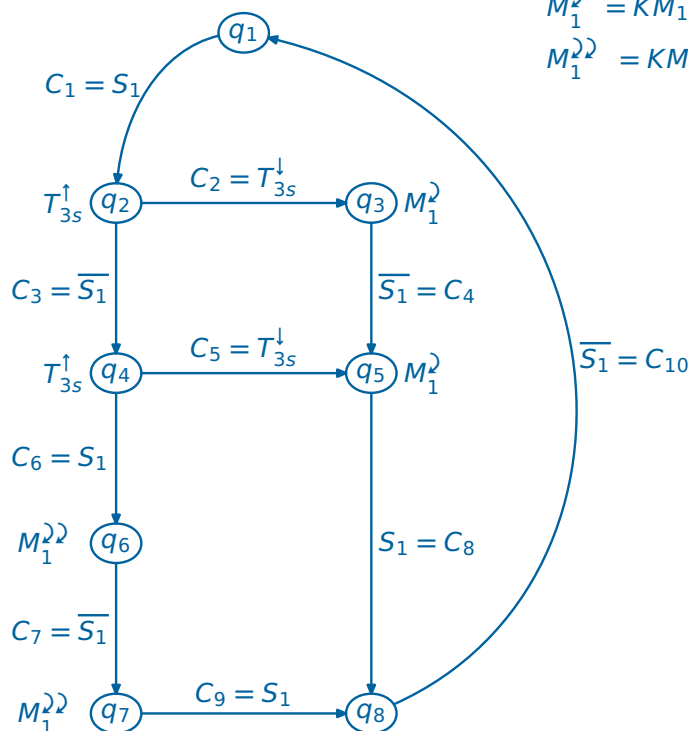
```

```

57 | C[2] C[4] x[3]
58 10 +---| |---+---|/|----- ( )
59 | x[3] |
60 +---| |---+
61 |
62 | C[3] C[5] C[6] x[4]
63 11 +---| |---+---|/|---|/|--- ( )
64 | x[4] |
65 +---| |---+
66 |
67 | C[4] C[8] x[5]
68 12 +---| |---+---|/|----- ( )
69 | C[5] |
70 +---| |---+
71 | x[5] |
72 +---| |---+
73 |
74 | C[6] C[7] x[6]
75 13 +---| |---+---|/|----- ( )
76 | x[6] |
77 +---| |---+
78 |
79 | C[7] C[9] x[7]
80 14 +---| |---+---|/|----- ( )
81 | x[7] |
82 +---| |---+
83 |
84 |
85 |
  
```

```

86 | C[2] C[4] x[3]
87 15 +---| |---+---|/|----- ( )
88 | x[3] |
89 +---| |---+
90 |
91 | C[8] C[10] x[8]
92 16 +---| |---+---|/|----- ( )
93 | C[9] |
94 +---| |---+
95 | x[8] |
96 +---| |---+
97 |
98 (* Actualización de las salidas.*)
99 | x[3] KM1
100 17 +---| |---+----- ( )
101 | x[5] |
102 +---| |---+
103 |
104 | x[6] KM2
105 18 +---| |---+-----+--- ( )
106 | x[7] | | KM3
107 +---| |---+ +--- ( )
108 |
109 | T3s
110 | +-----+
111 | x[2] | TON |
112 19 +---| |---+-----|IN Q|-
113 | x[4] | t#3s--|PT ET|-
114 +---| |---+ +-----+
  
```



$M_1^1 = KM_1$ (conexión de velocidad baja)

$M_1^2 = KM_1 \cdot KM_2$ (conexión de velocidad alta)

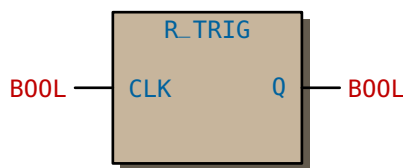
$$C = \begin{bmatrix} 0 & S_1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & T_{3s}^\downarrow & \overline{S_1} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \overline{S_1} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & T_{3s}^\downarrow & S_1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & S_1 \\ 0 & 0 & 0 & 0 & 0 & 0 & \overline{S_1} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & S_1 \\ \overline{S_1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

(b) Escriba las especificaciones funcionales del comportamiento del sistema S .

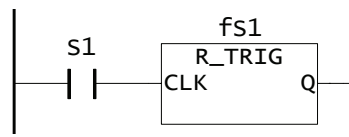
1. El estado inicial del automatismo es el de motor desconectado.
2. Si en el estado inicial se presiona el pulsador S_1 una sola vez en un intervalo de 3s, transcurrido este tiempo el motor se conectará en configuración de velocidad baja.
3. Si en el estado inicial se presiona el pulsador S_1 dos veces en un intervalo de 3s, el motor se conectará en configuración de velocidad alta la segunda vez que se presiona S_1 .
4. Si con el motor conectado en cualquiera de sus dos posibles configuraciones se presiona el pulsador S_1 , el motor se desconectará.

3.2 El programa `motorIL` del ejercicio anterior podría ser bastante más corto si empleara señales activas por flanco. En este ejercicio se pide que:

- Dibuje el grafo de una nueva máquina secuencial binaria M' que tenga en cuenta señales de entrada activas por flanco y que modele el comportamiento del sistema S . La máquina M' no modela funcionalidad adicional del sistema S , simplemente es un modelo con menos estados que la máquina M y que permite escribir un programa más sencillo.
- Escriba los elementos de la definición formal $\langle \mathbb{X}, \mathbb{U}, \mathbb{Y}, f, h, x_0 \rangle$ de M' .
- Escriba en lenguaje LD un programa de nombre `motorLD` que implemente la nueva máquina M' . En este programa se empleará el bloque de función `R_TRIG` para detectar el flanco de subida de la señal de entrada generada por el pulsador S_1 . La interfaz de este componente se muestra en la figura siguiente:



Con la variable `fs1 : R_TRIG`; podemos detectar el flanco de subida de la entrada S_1 empleando la llamada `CAL fs1(CLK:=S1)` (lenguaje IL) o el segmento (lenguaje LD)

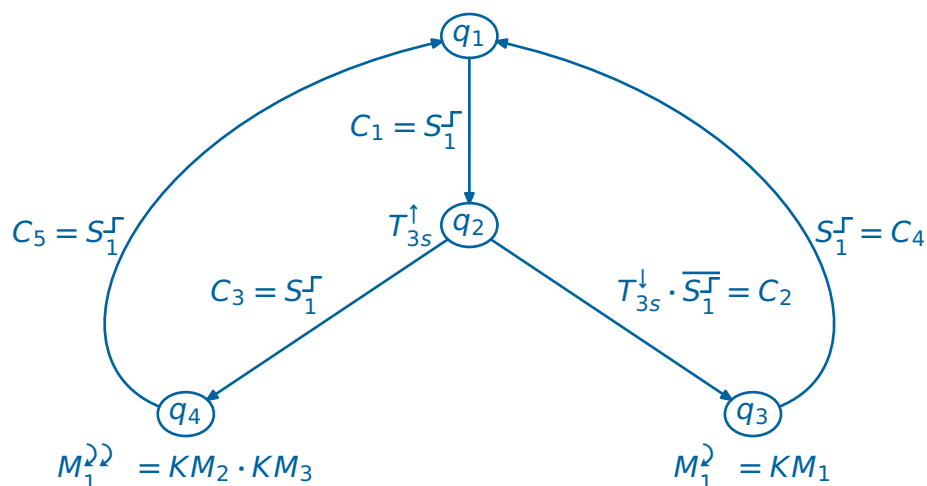


ya que en `fs1.Q` tendremos el valor `TRUE` durante el ciclo de `scan` en el que se produce el flanco de subida de S_1 . No se debe olvidar que `fs1.Q` estará activa (`TRUE`) durante un solo ciclo de `scan`.

- Introduzca las modificaciones necesarias en el programa `motorLD` para que, al conectar el motor para que gire a velocidad alta, el contactor KM_3 se conecte con un retardo de 300 ms respecto al contactor KM_2 .

RESPUESTA:

- Dibuje el grafo de una nueva máquina secuencial binaria M' ...



(b) Escriba los elementos de la definición formal $\langle \mathbb{X}, \mathbb{U}, \mathbb{Y}, f, h, x_0 \rangle$ de M' .

$$e = \begin{bmatrix} S_1^\rightarrow \\ T_{3s}^\downarrow \end{bmatrix}, \quad s = \begin{bmatrix} KM_1 \\ KM_2 \\ KM_3 \\ T_{3s}^\uparrow \end{bmatrix},$$

$$\mathbb{U} = \left\{ \overbrace{S_1^\rightarrow \cdot T_{3s}^\downarrow}^{U_1}, \overbrace{S_1^\rightarrow \cdot T_{3s}^\downarrow}^{U_2}, \overbrace{S_1^\rightarrow \cdot T_{3s}^\downarrow}^{U_3}, \overbrace{S_1^\rightarrow \cdot T_{3s}^\downarrow}^{U_4} \right\},$$

$$\mathbb{Y} = \left\{ \overbrace{KM_1 \cdot KM_2 \cdot KM_3 \cdot T_{3s}^\uparrow}^{Y_1}, \overbrace{KM_1 \cdot KM_2 \cdot KM_3 \cdot T_{3s}^\uparrow}^{Y_2}, \right.$$

$$\left. \overbrace{KM_1 \cdot \overline{KM_2} \cdot \overline{KM_3} \cdot \overline{T_{3s}^\uparrow}}^{Y_3}, \overbrace{KM_1 \cdot KM_2 \cdot KM_3 \cdot \overline{T_{3s}^\uparrow}}^{Y_4} \right\},$$

$$\mathbb{X} = \{q_1, q_2, q_3, q_4\},$$

$$x^+ = F(x, e) \Rightarrow \begin{bmatrix} x_1^+ \\ x_2^+ \\ x_3^+ \\ x_4^+ \end{bmatrix} = \begin{bmatrix} x_3 S_1^\rightarrow + x_4 S_1^\rightarrow + x_1 \overline{S_1^\rightarrow} \\ x_1 S_1^\rightarrow + x_2 S_1^\rightarrow \cdot T_{3s}^\downarrow \overline{S_1^\rightarrow} \\ x_2 T_{3s}^\downarrow S_1^\rightarrow + x_3 \overline{S_1^\rightarrow} \\ x_3 S_1^\rightarrow + x_4 \overline{S_1^\rightarrow} \end{bmatrix} = \begin{bmatrix} (x_3 + x_4) S_1^\rightarrow + x_1 \overline{S_1^\rightarrow} \\ x_1 S_1^\rightarrow + x_2 T_{3s}^\downarrow \cdot \overline{S_1^\rightarrow} \\ (x_2 T_{3s}^\downarrow + x_3) \overline{S_1^\rightarrow} \\ x_3 S_1^\rightarrow + x_4 \overline{S_1^\rightarrow} \end{bmatrix}, \quad x_0 = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix},$$

$$y = H(x) \Rightarrow \begin{bmatrix} KM_1 \\ KM_2 \\ KM_3 \\ T_{3s}^\uparrow \end{bmatrix} = \begin{bmatrix} x_3 \\ x_4 \\ x_4 \\ x_2 \end{bmatrix}.$$

(c) Escriba en lenguaje LD un programa de nombre `motorLD` que implemente la nueva máquina M' .

```

1 PROGRAM motorLD
2 VAR
3     S1 AT %IX0.00 : BOOL;
4     KM1 AT %QX0.00 : BOOL;
5     KM2 AT %QX0.01 : BOOL;
6     KM3 AT %QX0.02 : BOOL;
7     T3s : TON;
8 (* Detector del flanco de subida de S1. *)
9     fS1 : R_TRIG;
10 (* Estados. *)
11     x : ARRAY[1..4] OF BOOL := 1;
12     (* Condiciones para cambiar de estado. *)
13     C : ARRAY[1..5] OF BOOL;
14 END_VAR
15 (* Actualización de las entradas. *)
16 |           fS1
17 |           +-----+
18 |     S1   | R_TRIG | x[1]  C[1]
19 1 +---| |---|CLK   Q|---| |---( )
20 |           +-----+
21 |
22 |     x[2]           T3s.Q   fS1.Q  C[2]
23 2 +---| |---+---| |-----|/|---( )
24 |           |
25 |           | fS1.Q           C[3]
26 |           +---| |-----|---( )
27 |
28 |     x[3]  fS1.Q           C[4]
29 +---| |---| |-----|---( )
30 |
31 |
32 |     x[4]  fS1.Q           C[5]
33 3 +---| |---| |-----|---( )
34 |
35 |
36 (* Actualización de los estados. *)
37 |     C[4]           C[1]           x[1]
38 4 +---| |---+---|/|-----|---( )
39 |     C[5]           |
40 |---| |---+
41 |     x[1]           |
42 |---| |---+
43 |
44 |     C[1]           C[2]  C[3]           x[2]
45 5 +---| |---+---|/|---|/|-----|---( )
46 |     x[2]           |
47 |---| |---+
48 |

```

```

49 | C[2]          C[4]          x[3]
50 6 +---| |---+---|/|----- ( )
51 | x[3]        |
52 |---| |---+
53 |
54 | C[3]          C[5]          x[3]
55 7 +---| |---+---|/|----- ( )
56 | x[4]        |
57 |---| |---+
58 |
59 |
60 (* Actualización de las salidas.*)
61 | x[3]          KM1
62 8 +---| |----- ( )
63 |
64 | x[4]          KM2
65 9 +---| |-----+--- ( )
66 |                                     | KM3
67 |                                     +--- ( )
68 |
69 |                                     T3s
70 |                                     +-----+
71 | x[2]          | TON |
72 10 +---| |-----|IN  Q|-
73 |               t#3s--|PT  ET|
74 |               +-----+
75 |
76 END_PROGRAM

```


- (d) **Introduzca** las modificaciones necesarias en el programa `motorLD` para que, al conectar el motor para que gire a velocidad alta, el contactor `KM3` se conecte con un retardo de 300 ms respecto al contactor `KM2`.

Las modificaciones afectan a la declaración de variables, donde declararemos el temporizador `T300ms`

```

1  (*
2  * Retardo en la conexión de KM3 respecto a KM2.
3  *)
4  T300ms : TON;

```

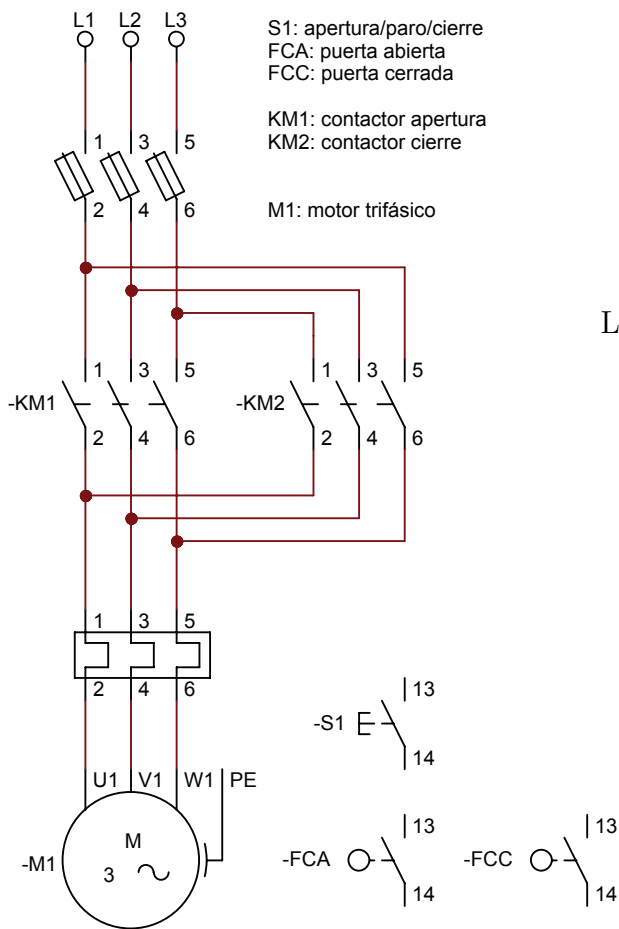
y al segmento nro. 9 de código

```

1  |
2  (* CoDeSys no acepta este segmento.*)
3  | x[4] KM2
4  9 +---| |----- ( )
5  |
6  | T300ms
7  | +-----+
8  | | TON | KM3
9  | +-----|IN Q|--- ( )
10 | t#300ms--|PT ET|
11 | +-----+
12 |
13 (* Segmentos alternativos para CoDeSys.
14 | x[4] KM2
15 9 +---| |----- ( )
16 |
17 | T300ms
18 | +-----+
19 | KM2 | TON | KM3
20 10 +---| |-----|IN Q|--- ( )
21 | t#300ms--|PT ET|
22 | +-----+
23 *)

```


3.3 Queremos implementar un automatismo programado para controlar la puerta del almacén de una planta industrial que está accionada por el motor **M1** que se muestra en la figura:



componente	entrada	salida
S1	0.00	
FCA	0.01	
FCC	0.02	
KM1		0.00
KM2		0.01

La descripción funcional del automatismo es la siguiente:

1. La puerta está inicialmente cerrada.
2. Los finales de carrera FCA y FCC indican que la puerta está abierta o cerrada respectivamente.
3. Si se presiona el pulsador **S1** estando la puerta cerrada/abierta, la puerta inicia la maniobra de apertura/cierre.
4. La maniobra de apertura/cierre finaliza cuando la puerta esté abierta/cerrada.
5. Si se presiona el pulsador **S1** antes de que finalice alguna de las maniobras anteriores, la maniobra quedará interrumpida durante un máximo de 10s. Transcurridos estos 10s se reanudará la maniobra.

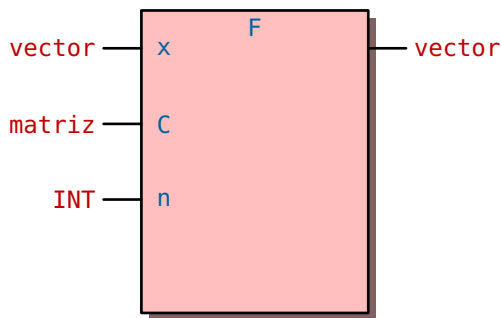
6. Si se presiona el pulsador **S1** cuando alguna de las maniobras anteriores está interrumpida, la puerta debe deshacer la maniobra previa; es decir, si la puerta estaba abriéndose/cerrándose, al deshacer la maniobra pasará a estar cerrándose/abriéndose.
7. La puerta estará abierta un máximo de 30 segundos y después iniciará la maniobra de cierre.

Se pide lo siguiente:

- (a) Dibuje el grafo de una máquina de Moore que modele el comportamiento del automatismo.
- (b) A partir del grafo anterior obtenga el vector de entrada e , la matriz de incidencia C y el vector de salida s del automatismo.
- (c) Escriba en lenguaje IL una función de nombre **F** para calcular el vector de estado del sistema con la expresión

$$x^+ = \sum_{i=1}^n \underbrace{\left(\sum_{j=1}^n x_j C_{ji} + x_i \prod_{j=1}^n \overline{C_{ij}} \right)}_{x_i^+} q_i.$$

La declaración de la función se muestra a continuación:



```

1 VAR_GLOBAL CONSTANT
2   dimX: INT := Número de estados;
3 END_VAR
4 TYPE
5   vector: ARRAY[1..dimX] OF BOOL;
6 END_TYPE
7 TYPE
8   matriz: ARRAY[1..dimX,1..dimX] OF BOOL;
9 END_TYPE

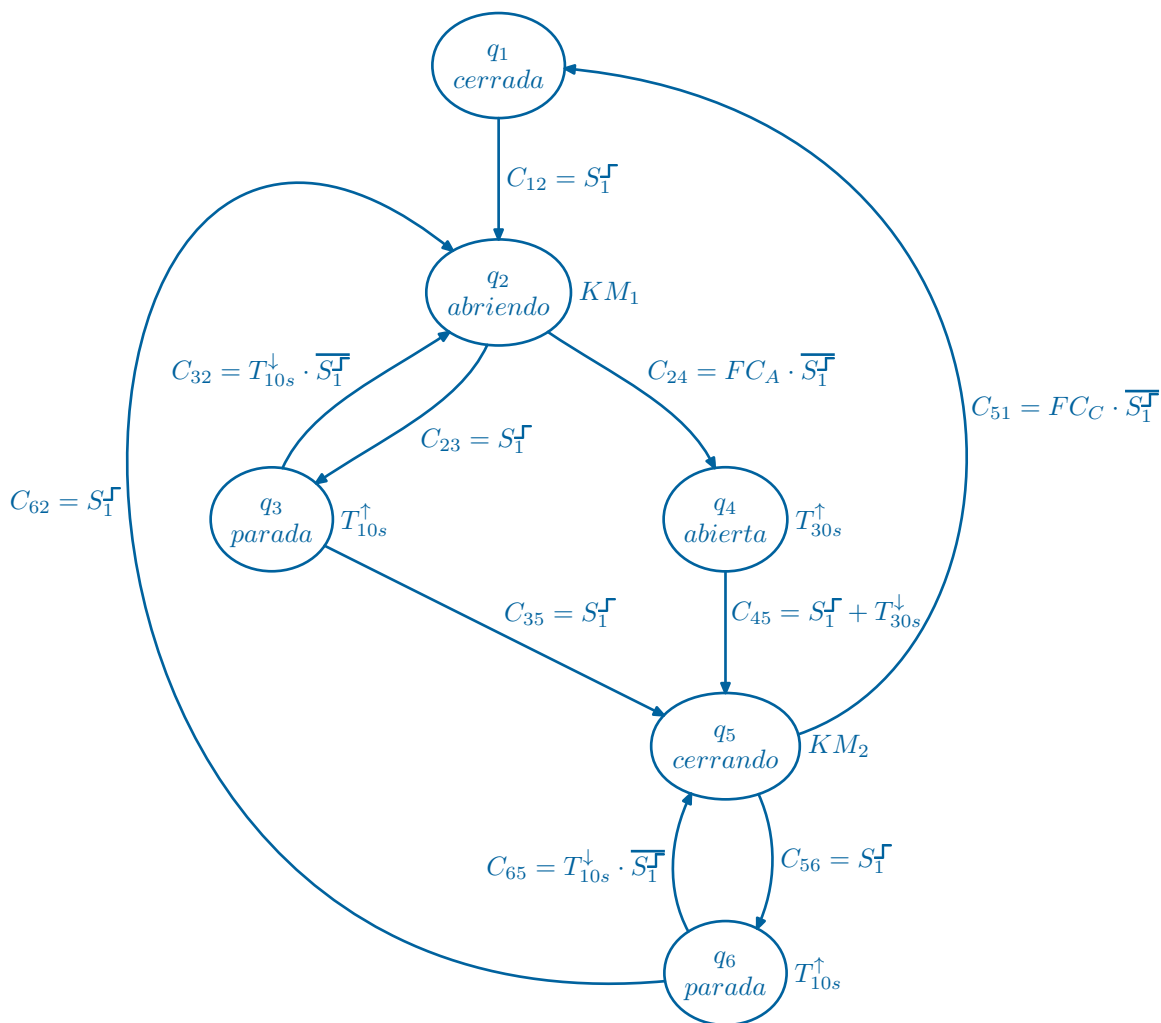
```

$n = \dim X$ es el número de estados del automatismo.

(d) Escriba, en lenguaje IL, un programa de nombre **puerta** que implemente la funcionalidad del automatismo a partir de la matriz de incidencia C , la función F y el vector de salida s .

RESPUESTA:

(a) Dibuje el grafo de una máquina de Moore que modele el comportamiento del automatismo.



(b) A partir del grafo anterior obtenga el vector de entrada e , la matriz de incidencia C y el vector de salida s del automatismo.

Elementos de la ecuación del sistema:

- El vector de entrada e ,
- el vector de estado x ,
- la matriz de transición C y,
- el vector de salida s .

A la vista del grafo de la máquina de Moore, podemos escribir directamente:

$$e = \begin{bmatrix} S_1^\uparrow \\ FC_A \\ FC_C \\ T_{10s}^\downarrow \\ T_{30s}^\downarrow \end{bmatrix}$$

$$x = \begin{bmatrix} x_1 \\ \vdots \\ x_6 \end{bmatrix}$$

$$C = \begin{bmatrix} 0 & S_1^\uparrow & 0 & 0 & 0 & 0 \\ 0 & 0 & S_1^\uparrow & FC_A \cdot S_1^\uparrow & 0 & 0 \\ 0 & T_{10s}^\downarrow \cdot S_1^\uparrow & 0 & 0 & S_1^\uparrow & 0 \\ 0 & 0 & 0 & 0 & S_1^\uparrow + T_{30s}^\downarrow & 0 \\ FC_C \cdot S_1^\uparrow & 0 & 0 & 0 & 0 & S_1^\uparrow \\ 0 & S_1^\uparrow & 0 & 0 & T_{10s}^\downarrow \cdot S_1^\uparrow & 0 \end{bmatrix}$$

$$s = \begin{bmatrix} KM_1 \\ KM_2 \\ T_{10s}^\uparrow \\ T_{30s}^\uparrow \end{bmatrix} = \begin{bmatrix} x_2 \\ x_5 \\ x_3 + x_6 \\ x_4 \end{bmatrix}$$

(c) Escriba en lenguaje IL una función de nombre F...

```

1 FUNCTION F : vector
2 (*
3 *Actualiza el estado teniendo en cuenta la matriz de incidencia.
4 *
5 *  $x^+ = F(x, C, \dim \mathbb{X}) = \sum_{i=1}^{\dim \mathbb{X}} \underbrace{\left( \sum_{j=1}^{\dim \mathbb{X}} x_j C_{ji} + x_i \prod_{j=1}^{\dim \mathbb{X}} \overline{C_{ij}} \right)}_{x_i^+} q_i$ 
6 *
7 *)
8 VAR_INPUT
9     x: vector;
10    C: matriz;
11    N: INT;
12 END_VAR
13
14 VAR
15     aux1, aux2: BOOL;
16     i, j: INT;
17 END_VAR
18 (* ----- *)
19 * INSTRUCCIONES. *
20 * ----- *)
21 LD 1
22 ST i
23 bucle_i: (* FOR i:= 1 TO N BY 1 DO *)
24     LD 0
25     ST aux1
26     LD x[i]
27     ST aux2
28     LD 1
29     ST j (* j:=1 *)
30     bucle_j: (* FOR j:= 1 TO N BY 1 DO *)
31         LD x[j]
32         AND C[j,i]
33         OR aux1
34         ST aux1 (*  $\sum_{j=1}^n x_j C_{ji}$  *)
35         LD aux2
36         ANDN C[i,j]
37         ST aux2 (*  $x_i \prod_{j=1}^n \overline{C_{ij}}$  *)
38         (* Control del bucle 2. *)
39         LD j
40         ADD 1
41         ST j (* j:=j+1 *)
42         GT N
43         JMPCN bucle_j (* Salta si CR = 0, i.e.,  $j \leq \dim \mathbb{X}$ . *)
44     LD aux1
45     OR aux2

```

```
46  ST  F[i]  (*  $x_i^+ = \sum_{j=1}^n x_j C_{ji} + x_i \prod_{j=1}^n \overline{C_{ij}}$  *)
47  (* Control del bucle 1. *)
48  LD  i
49  ADD 1
50  ST  i (* i:=i+1 *)
51  GT  N
52  JMPCN bucle_i (* Salta si  $CR = 0$ , i.e.,  $i \leq \dim \mathbb{X}$ . *)
53  RET
54  (* ----- *)
55  * FIN de la función. *
56  * ----- *)
57  END_FUNCTION
```

(d) Escriba, en lenguaje IL, un programa de nombre puerta...

```

1 PROGRAM puerta
2 VAR
3 (* Detector del flanco de subida de 'S1'. *)
4   fsS1: R_TRIG;
5 (* Temporizadores. *)
6   T10s, T30s : TON;
7 (* Vector de estado. 'q1' estado inicial. *)
8   x : vector := 1; (* Equivalente a '1(1), 5(0)', ie.,
9     x1 = 1, x2...dimX = 0. *)
10  C : matriz;
11 END_VAR
12 (* ----- *)
13 * INSTRUCCIONES. *
14 * ----- *)
15 (* ----- *)
16 * Actualización de las ENTRADAS. *
17 * ----- *)
18 (* Detección de flancos de subida. *)
19 (* Detección de flancos de subida. *)
20 CAL fSS1(CLK:=S1)
21 (* Construcción de la matriz de transiciones. *)
22 (* Esta matriz es realmente una 'matriz DISPERSA' (sparse) ya que de los 36»
23   » elementos que tiene solo utilizamos 10. El factor de utilización es »
24   » 10/36 → 27,78%. *)
25 LD fSS1.Q
26   ST C[1,2]
27   ST C[2,3]
28   ST C[3,5]
29   ST C[5,6]
30   ST C[6,2]
31   OR T30s.Q
32   ST C[4,5] (* C45 = S1↑ + T30s↓ *)
33 LD FCA
34   ANDN fSS1.Q
35   ST C[2,4]
36 LD T10s.Q
37   ANDN fSS1.Q
38   ST C[3,2]
39   ST C[6,5]
40 LD FCC
41   ANDN fSS1.Q
42   ST C[5,1]
43 (* ----- *)
44 * Actualización del ESTADO. *
45 * ----- *)
46 LD x
47 F C, dimX (* CR := F(x,C,dimX) *)

```



```
46 ST x (*  $x^+ = F(x, C, \dim \mathbb{X})$  *)
47 (* ----- *)
48 * Actualización de las SALIDAS. *
49 * ----- *)
50 LD x[2]
51 ST KM1
52 LD x[5]
53 ST KM2
54 (* Actualización de los temporizadores. Los temporizadores se pueden »
   » actualizar junto con las entradas o junto con las salidas. *)
55 LD x[3]
56 OR x[6]
57 ST T10s.IN
58 CAL T10s(PT:=t#10s)
59 LD x[4]
60 ST T30s.IN
61 CAL T30s(PT:=t#30s)
62 (* ----- *)
63 * FIN del programa. *
64 * ----- *)
65 END_PROGRAM
```
