

# Autòmats programables

Lluís Martínez Novoa

**Automatismes industrials**





# Índex

<b>Introducció</b> .....	5
<b>Resultats d'aprenentatge</b> .....	6
<b>1. Maquinari i funcionament intern dels autòmats programables</b> .....	7
1.1. Instal·lació electromecànica .....	7
1.1.1. Tipus de senyals elèctrics .....	9
1.1.2. Circuit d'automatisme amb autòmat programable .....	10
1.2. Programa de l'autòmat programable .....	11
1.2.1. Codificació interna dels programes dels autòmats programables .....	13
1.2.2. Memòries .....	15
1.3. Arquitectura i funcionament intern dels autòmats programables .....	16
1.3.1. Arquitectura dels autòmats programables .....	17
1.3.2. Funcionament intern dels autòmats programables .....	19
1.3.3. Interpretació del programa dels autòmats programables .....	21
1.3.4. Estructura i components dels autòmats programables ...	23
1.4. Perifèria dels autòmats programables .....	24
1.4.1. Entrades digitals .....	25
1.4.2. Sortides digitals .....	27
1.4.3. Senyals analògics .....	29
1.4.4. Entrades analògiques .....	31
1.4.5. Sortides analògiques .....	32
1.5. Funcionalitat dels autòmats programables .....	32
1.5.1. Llenguatges de programació .....	33
1.5.2. Característiques generals dels autòmats programables .....	34
1.5.3. Evolució dels autòmats programables .....	35
<b>2. Instruccions bàsiques de programació   dels autòmats programables 1</b> .....	37
2.1. L'autòmat programable S7-200 de Siemens .....	37
2.1.1. Maquinari .....	37
2.1.2. Programari .....	38
2.1.3. Exemple de programació amb MicroWin i comprovació del programa amb el simulador S7-200 .....	39
2.2. Programació d'entrades i sortides .....	40
2.2.1. Activació d'una sortida mitjançant una entrada .....	41
2.2.2. Activació de diverses sortides amb diverses entrades .....	42
2.2.3. Programació de contactes en sèrie i en paral·lel .....	42

2.2.4. Programació de contactes en connexió mixta sense parèntesis .....	43
2.2.5. Programació de contactes en connexió mixta amb parèntesis .....	44
2.2.6. Programació de contactes oberts i tancats de sortides .....	45
2.3. Programació de biestables .....	46
2.3.1. Circuit amb polsadors d'aturada i marxa .....	47
2.3.2. Programació de sortides amb instruccions de <i>set-reset</i> ...	48
2.3.3. Programació d'una mateixa sortida diverses vegades ..	49
2.3.4. Funcionament del registre de sortides .....	49
2.4. Flancs i modificador <i>not</i> .....	50
2.4.1. Programació de flancs positiu i negatiu .....	50
2.4.2. Programació del modificador <i>not</i> .....	52
2.5. <i>Merkers</i> .....	53
2.5.1. Programació de <i>merkers</i> .....	53
<b>3. Instruccions bàsiques de programació dels autòmats programables 2</b> .....	55
3.1. Temporitzadors .....	55
3.1.1. Programació del temporitzador a la connexió .....	56
3.1.2. Posada en marxa d'un contactor mitjançant polsadors i temporitzadors a la connexió .....	58
3.1.3. Posada en marxa progressiva de tres contactors mitjançant dos temporitzadors a la connexió .....	58
3.1.4. Programació del temporitzador a la desconexió .....	59
3.1.5. Programació del temporitzador a la desconexió accionat mitjançant flanc positiu .....	60
3.1.6. Programació del temporitzador acumulatiu .....	61
3.1.7. Control del temps de funcionament d'un contactor mitjançant un temporitzador acumulatiu .....	63
3.2. Comptadors .....	63
3.2.1. Comptador progressiu .....	64
3.2.2. Programació d'un comptador progressiu amb selecció d'inicialització manual o automàtica .....	65
3.2.3. Comptador regressiu .....	66
3.2.4. Comptador progressiu-regressiu .....	67
3.2.5. Control de fabricació de peces en un procés de producció .....	68
3.3. Multivibradors .....	69
3.3.1. Generador d'impuls .....	70
3.3.2. Funcions especials .....	71
3.4. Programació estructurada .....	73
3.4.1. Instruccions de salt .....	73
3.4.2. Subrutines .....	75

## Introducció

Els autòmats programables van aparèixer l'any 1968 com els substituïts dels armaris de relés i altres components de maniobra en els processos d'automatismes, i avui són els equips de control que s'utilitzen en la majoria d'instal·lacions automatitzades, des de la instal·lació més gran, com pot ser una planta petroquímica, a la més petita, com el control de llums, calefacció i altres equips d'una casa.

La majoria de tècnics en automatismes, quan es refereixen als autòmats programables, ho fan amb les sigles *PLC*, inicials del terme anglès *programmable logic controller* ('controlador lògic programable').

La unitat "Autòmats programables" està dissenyada per estudiar els principis de funcionament dels autòmats programables, i també les principals instruccions de programació i la seva aplicació en els circuits de comandament i regulació.

Aquesta unitat està temporitzada dintre del mòdul, de manera que abans s'han tractat tots els aspectes relatius als sistemes de comandament i regulació de manera convencional, és a dir, mitjançant sistemes cablats, i que serviran de base per entendre millor les aplicacions dels autòmats programables.

En l'apartat "Maquinari i funcionament intern dels autòmats programables" es descriuen els components que configuren els autòmats programables, se n'estudia el funcionament i s'analitza la manera com la unitat central de processament (CPU) de l'autòmat executa el programa.

En els apartats "Instruccions bàsiques de programació dels autòmats programables 1 i 2", s'estudien exhaustivament les diferents instruccions que es poden programar en l'autòmat perquè la instal·lació es comporti segons les condicions definides en els esquemes d'automatització.

Per treballar els continguts d'aquesta unitat és convenient fer les activitats i els exercicis d'autoavaluació, llegir els annexos i consultar els recursos multimèdia.

## Resultats d'aprenentatge

En acabar la unitat, heu de ser capaços de fer el següent:

1. Identificar els components i dispositius utilitzats en automatismes cablats i programables, a partir d'esquemes i documentació tècnica.
2. Diferenciar entre la part de comandament i la part de potència en automatismes cablats i programables, a partir d'esquemes i documentació tècnica.
3. Interpretar el programa de control d'un automatisme programable, a partir de les relacions de cada part o subrutina del programa amb les especificacions funcionals de l'equip.
4. Relacionar modificacions en elements del circuit o amb parts del programa de control amb variacions dels paràmetres característics de l'equip.
5. Elaborar el quadern de càrregues d'un automatisme a partir de les especificacions funcionals sol·licitades.
6. Seleccionar la solució més adient, mitjançant un sistema programable, per a cada aplicació que s'ha d'automatitzar, a partir de les seves característiques.
7. Codificar programes de control en algun llenguatge (contactes, llista d'instruccions i altres), a partir de les especificacions del projecte i de diagrames de seqüència.

## 1. Maquinari i funcionament intern dels autòmats programables

Abans d'estudiar els autòmats programables, heu de tenir clar què són i per a què s'utilitzen. Per aclarir aquests conceptes, fixeu-vos en les diferències entre un automatisme convencional, basat en el cablatge de tots els seus components, i un automatisme programat, basat en el programa que s'ha introduït a l'autòmat programable.

En l'**automatisme convencional o cablat** les condicions de funcionament de la instal·lació estan definides per les connexions mitjançant cables entre tots els elements de maniobra que componen la instal·lació, que connecten en sèrie o paral·lel els contactes de pulsadors, tèrmics, relés, temporitzadors, etc., que acabaran connectant les bobines dels relés, temporitzadors, contactors, etc., que en activar-se faran que la instal·lació faci els moviments desitjats.

En l'**automatisme programat** tots els elements de control estan connectats mitjançant cables, directament a l'autòmat programable, i no estan interconnectats entre ells. És mitjançant les instruccions introduïdes al programa de l'autòmat que es defineixen com han d'estar connectats, en sèrie o paral·lel, i quan han d'obrir o tancar els contactes dels elements de comandament i, en definitiva, quan s'han de connectar les bobines dels contactors que activaran els actuadors de la instal·lació i faran que la instal·lació faci els moviments desitjats.

### Exemples d'instal·lacions electromecàniques

D'instal·lacions electromecàniques n'hi ha moltes i de tot tipus, petites i molt senzilles com la porta automàtica d'un pàrquing, o grans i complexes com una cadena de producció de cotxes.

### 1.1. Instal·lació electromecànica

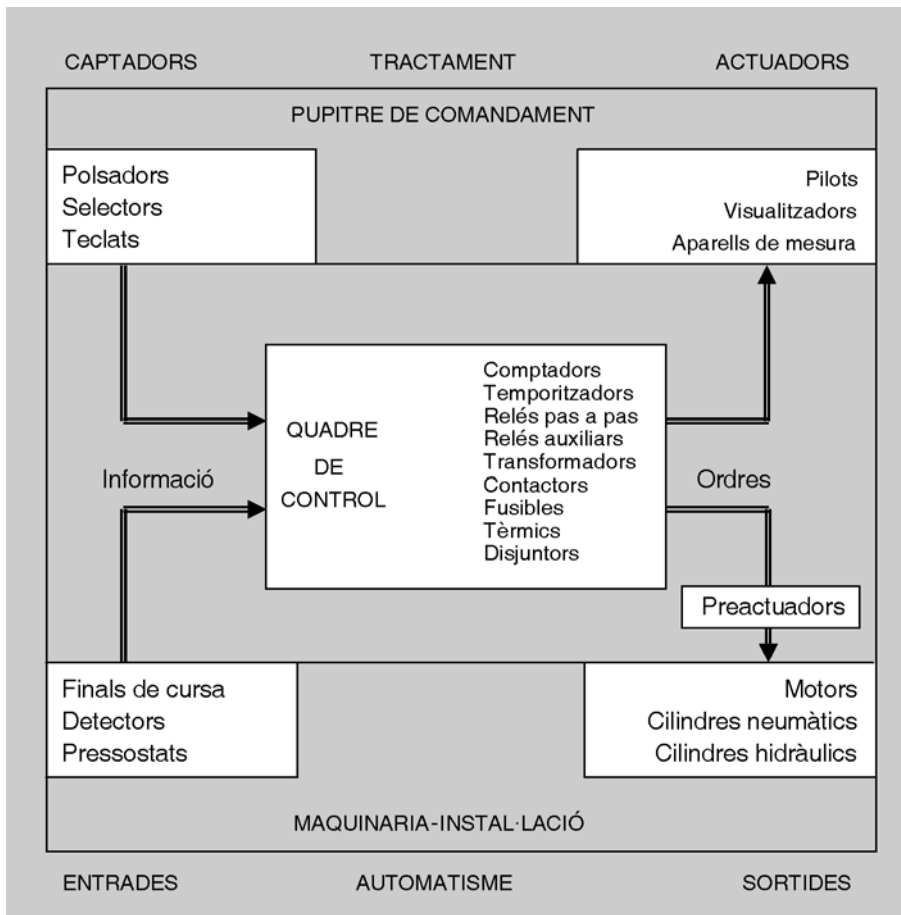
Entenem per **instal·lació electromecànica** qualsevol instal·lació que s'alimenta amb energia elèctrica i mitjançant components mecànics duu a terme una tasca determinada i produeix un producte final; es componen, com es veu en la figura 1, de tres parts:

- **Pupitre de comandament.** El lloc des del qual els operaris poden intervenir en el funcionament de la instal·lació enviant informació al quadre de control mitjançant els captadors, com pulsadors, selectors, etc., i on es pot visualitzar com funciona la instal·lació mitjançant les ordres que reben els actuadors, com pilots, visualitzadors, etc., des del quadre de control.
- **Maquinaria o instal·lació.** El conjunt d'elements que configuren l'estructura i components que fan la tasca per a la qual ha estat dissenyada la instal·lació; rep les ordres del quadre de control per activar els actuadors, com motors, cilindres, etc., i envia informació de l'estat de la instal·lació mitjançant els captadors, com finals de cursa, detectors, etc.

- **Quadre de control.** El lloc de la instal·lació on hi ha definit el tipus de funcionament o seqüència de treball que ha de fer la instal·lació; en funció de la informació que li envien els captadors dóna ordres de connexió als actuadors.

Conté tots els elements de maniobra com comptadors, relés, temporitzadors etc., que poden ser substituïts per un autòmat programable, i també els elements de potència i protecció, com transformadors, contactors, fusibles, tèrmics, etc.

Figura 1. Components d'una instal·lació electromecànica



#### Preactuadors

Per adaptar el senyal elèctric del PLC, que és de petita potència, en els actuadors que acostumen a consumir més potència s'utilitzen els preactuadors, com els contactors.

Quan una instal·lació electromecànica tingui un autòmat programable en el quadre de control per definir la seqüència de treball, tots els **captadors**, és a dir, tots els elements que envien informació al quadre de control, es consideren entrades, i tots els **actuadors**, és a dir, tots els elements que reben ordres del quadre de control, sortides del PLC, independentment que estiguin situats al pupitre de comandament o a la instal·lació.

La informació que envien els captadors i les ordres que reben els actuadors es fan mitjançant senyals elèctrics.



En la secció "Recursos de contingut" del web d'aquest mòdul podeu veure un vídeo amb exemples d'instal·lacions automatitzades.

### 1.1.1. Tipus de senyals elèctrics

El **senyal elèctric**, referit als circuits d'automatismes, és la quantitat d'energia elèctrica que circula pels components de comandament d'una instal·lació per enviar informació i rebre ordres del quadre de control.

Els senyals elèctrics poden ser de tensió o d'intensitat i de corrent continu o altern.

- Els **senyals de tensió** més freqüents són 10 V o 24 V en corrent continu i 220 V en corrent altern.
- Els **senyals d'intensitat** més freqüents són 0-20 mA (mil·liamperes) o 4-20 mA en corrent continu.

#### Exemple del tipus de senyal

El senyal que envia l'interruptor d'una habitació quan l'accionem és de 220 V de corrent altern, i el senyal que envia el pulsador d'un ascensor és de 24 V de corrent continu.

En els circuits d'automatismes hi ha dos tipus de senyals elèctrics, els digitals i els analògics.

- **Senyal digital.** Normalment són senyals de tensió, tant de corrent continu com d'altern; només poden tenir dos estats, que es representen utilitzant el sistema de numeració binari, que està format per zeros i uns.
  - Estat 0: relacionat amb: “no dóna senyal”, “no funciona”, “està desconnectat”, “circuit obert”.
  - Estat 1: relacionat amb: “sí que dóna senyal”, “sí que funciona”, “està connectat”, “circuit tancat”.
- **Senyal analògic.** Acostumen ser tant de tensió com d'intensitat de corrent continu; són senyals elèctrics que, entre un valor mínim i un màxim, poden tenir qualsevol valor, i per representar-los s'utilitza el sistema de numeració decimal. És a dir, el captador enviarà 1,2 V o 6,4 V, etc., el PLC convertirà aquest senyal analògic en digital per poder processar-lo, visualitzant-lo convertit en una magnitud física que puguem entendre. En la taula 1 tenim un exemple de captadors (entrades) i actuadors (sortides), que envien o reben informació mitjançant senyals elèctrics digitals i analògics.

#### Tractament del senyal analògic

Quan ens pesem en una bàscula electrònica, el senyal que es genera és una tensió entre 0 i 10 V, que la bàscula converteix en un valor binari per processar-lo i visualitzar-lo en la pantalla convertit en un valor proporcional expressat en kg.

Taula 1. Exemple d'entrades i sortides digitals i analògiques

	Digital	Analògica
<b>Entrades</b>	Sensor ultrasònic de presència. Quan detecta en qualsevol punt de la seva zona d'accés, dóna 1.	Cèl·lula fotoelèctrica. En funció de la distància on detecti dintre de la seva zona d'acció, dóna un valor entre 0 i 10 V.
<b>Sortides</b>	Contactador per a l'accionament d'un motor. El motor funciona, girant a la velocitat nominal, o no funciona i està parat.	Variador de velocitat d'un motor. En funció de la tensió de consigna entre 0 i 10 V el motor gira a més o menys velocitat.

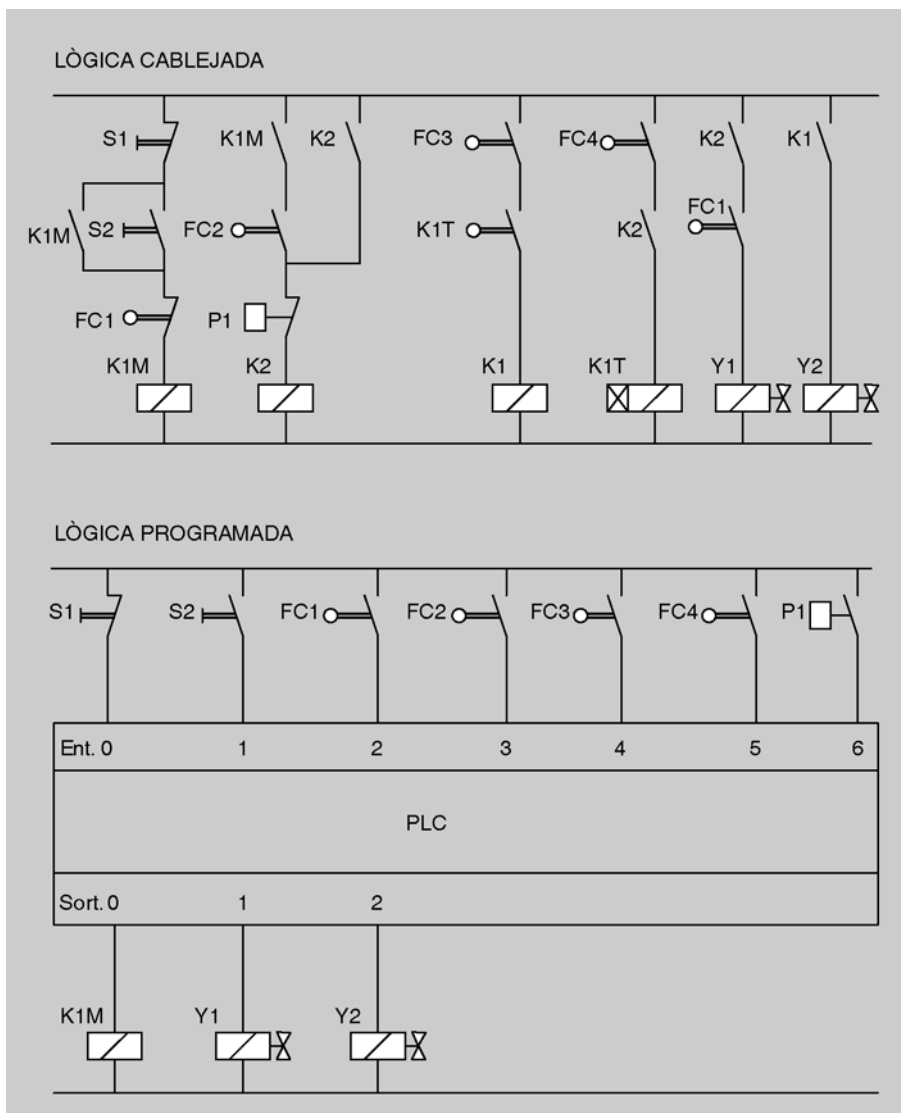
### 1.1.2. Circuit d'automatisme amb autòmat programable

L'autòmat programable simplifica i abarateix molt els circuits d'automatisme, i per això actualment estan molt introduïts, s'utilitzen molt i hi ha models de diferents configuracions per adaptar-se a les característiques de qualsevol instal·lació, des del més simple fins al més complex. En la figura 2 es mostren les diferències fonamentals entre un circuit fet amb lògica cablejada, el sistema convencional, és a dir, sense PLC, i amb lògica programada amb un PLC.

#### Aplicacions dels autòmats

Hi ha PLC per a aplicacions molt petites i molt econòmiques, pràcticament pel preu d'un parell de relés i un temporitzador, però també podeu trobar un PLC amb més de cent relés i deu temporitzadors interns, entre altres funcions.

Figura 2. Circuit d'automatisme amb lògica cablejada i programada



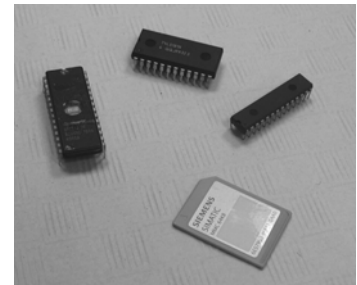
Els dos esquemes de la figura 2 representen un mateix circuit d'automatisme fet amb les dues tecnologies, cablada i programada. Si us fixeu bé en els esquemes podreu observar les diferències següents:

- El cablatge, és a dir, la interconnexió entre els elements que configuren la instal·lació, és molt més complexa en la lògica cablada, ja que hi ha connexions en sèrie, en paral·lel i mixtes, mentre que en la lògica

programada la connexió és molt senzilla i consisteix en un cable que uneix tots els captadors i un altre cable que uneix tots els actuadors, i després hi ha uns cables que uneixen cada captador i actuator amb el PLC.

- Amb la lògica cablada hi ha molts més contactes, en total quinze, i amb la programada només set. Això es deu al fet que:
  - El captador FC1 (final de cursa) fa dues funcions i amb lògica cablada necessita dos contactes, però amb la programada amb un contacte n'hi ha prou.
  - Amb lògica cablada són necessaris els contactes auxiliars de contactors, com K1M, de relés, com K1 i K2, i de temporitzadors, com KIT, i amb lògica programada no cal connectar-los, ja que podem programar els que té el PLC internament.
- Amb la lògica cablada hi ha més bobines corresponents a actuadors, en total sis, i amb la programada només tres. Això es deu al fet que les bobines dels temporitzadors i relés auxiliars no cal connectar-les, ja que podem programar els que té internament el PLC.
- El circuit amb lògica programada utilitza un autòmat programable, que necessita un programa, i amb la cablada no cal.

La diferència fonamental entre el circuit amb lògica cablada i lògica programada és que en el primer circuit, si el connectem a la xarxa elèctrica i accionem el captadors, la instal·lació funciona, i per contra, el circuit amb lògica programada no ho fa, i això és degut al fet que el circuit amb autòmat programable necessita que aquest tingui un programa emmagatzemat a la memòria del PLC.

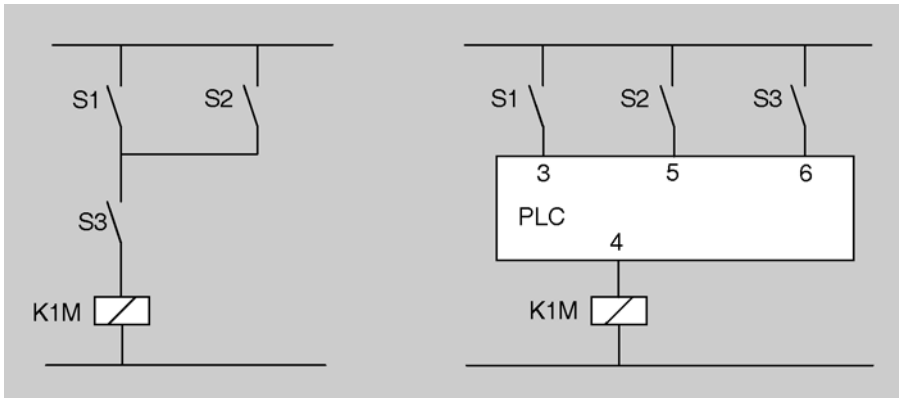


Memòries de PLC

## 1.2. Programa de l'autòmat programable

El programa de l'autòmat programable d'una instal·lació està format per una sèrie ordenada d'instruccions que corresponen a l'esquema elèctric que faríeu si realitzéssiu la instal·lació amb lògica cablejada; cada instrucció ha de contenir tota la informació necessària de cada element de l'esquema elèctric. En la figura 3 teniu un exemple d'un circuit molt elemental en què es veu com es connecten els elements d'un esquema amb lògica cablada, i com es connecten a l'autòmat programable per fer el mateix circuit amb lògica programada, que ens servirà d'exemple per raonar com hem de fer el programa del PLC, que consistirà a indicar com estan interconnectats aquests elements.

Figura 3. Connexió d'entrades i sortides al PLC



En l'esquema de l'esquerra de la figura 3, que correspon al circuit amb lògica cablada, podeu comprovar el funcionament de la instal·lació, i veureu que perquè funcioni l'actuador K1M han d'estar activats el captador S3 i un dels captadors S1 o S2.

A la dreta de la figura 3 teniu la interconnexió dels captadors i l'actuador amb el PLC, i podeu observar que S1 està connectat al born 3 del PLC, S2 al born 5 i S3 al born 6. Això implica que a partir d'aquest moment els captadors els considerem com a entrades del PLC i ja no seran S1, S2 i S3, sinó les entrades E3, E5 i E6, respectivament, i pel mateix motiu el captador K1M serà la sortida S4.

Com que l'esquema té quatre elements necessitem quatre instruccions per fer el programa del PLC; cada instrucció del programa ha de contenir tota la informació de cada element i això ho aconseguim mitjançant tres dades:

**1) Operació.** Ha d'indicar què fa aquest element dintre del circuit o, dit d'una manera més entenedora, com està connectat. En principi descriurem les operacions més elementals per entendre aquest exemple, que són:

- Inicial: si està connectat al principi de l'esquema, en aquest exemple correspondria a S1.
- Final: si està connectat al final de l'esquema, en aquest exemple correspondria a K1M.
- Paral·lel: si està connectat en paral·lel, en aquest exemple correspondria a S2.
- Sèrie: si està connectat en sèrie, en aquest exemple correspondria a S3.

**2) Operand.** Ha d'indicar quin tipus d'element és. De moment només descriurem els dos que necessitem per entendre aquest exemple, que són:

- Entrades: elements que envien senyals elèctrics al PLC.
- Sortides: elements que reben senyals elèctrics del PLC.

El conjunt d'operacions, operands i paràmetres d'un PLC es coneix com a mapa de memòria del PLC.

### 3) Paràmetre: ha d'indicar quina entrada o sortida és.

Cada instrucció del programa del PLC correspon a un element de l'esquema. Normalment l'ordre de programació es porta a terme amb els elements situats d'esquerra a dreta i de dalt a baix.

Totes les instruccions dels autòmats programables estan formades per tres dades, que són: operació, operand i paràmetre.

En la taula 2 s'indica com seria el programa corresponent a l'esquema de la figura 3.

Taula 2. Exemple de realització programa de PLC

Informació de l'element	Adreça	Operació	Operand	Paràmetre
S1	1	Inici	Entrada	3
S2	2	Paral·lel	Entrada	5
S3	3	Sèrie	Entrada	6
K1M	4	Final	Sortida	4

Aquest programa està format per quatre instruccions; s'han descrit les diferents operacions, operands i paràmetres amb unes denominacions molt generals i fàcils d'entendre, però heu de tenir en compte que cada fabricant d'autòmats utilitza uns codis diferents i que no corresponen a aquesta denominació general. Així doncs, la primera cosa que heu de fer per fer un programa per a un PLC és conèixer els codis del fabricant d'aquest autòmat, que es coneixen com a **àrees de memòria i funcions**. En la taula 3 s'indica com seria el programa de la taula 2 adaptat als codis de l'autòmat S7-200 de Siemens.

Taula 3. Adaptació del programa de la figura 3 adaptat a l'autòmat S7-200 de Siemens

Adreça	Programa general	Programa S7-200 de Siemens
1	Inici entrada 3	LD I 0.3
2	Paral·lel entrada 5	O I 0.5
3	Sèrie entrada 6	A I 0.6
4	Final sortida 4	= Q 0.0

#### 1.2.1. Codificació interna dels programes dels autòmats programables

Quan nosaltres introduïm el programa corresponent a un esquema en un autòmat programable, ho fem escrivint directament amb la programadora els codis corresponents a l'operació, operand i paràmetre d'aquest autòmat (LD, O, I, Q, etc.), però, lògicament, dintre del PLC no es guarda



Per veure les característiques del maquinari i programari de les diferents CPU de l'autòmat S7-200 de Siemens, aneu a la secció "Annexos" del web d'aquest mòdul.

#### Codi binari

És el codi utilitzat pels autòmats programables per realitzar totes les operacions internes. Cada dígit del sistema de numeració binari que pot tenir el valor zero o u, l'anomenem *bit*.

aquesta informació tal com l'escrivim, sinó que cada codi corresponent a l'operació, operand i paràmetre es converteix en una seqüència de zeros i uns, és a dir, es guarda codificat en binari.

L'operació que fa l'autòmat programable de guardar el programa codificat en binari s'anomena **compilació**.

Cada instrucció del programa necessita 16 bits per guardar la informació corresponent a cada element, i cada grup de 16 bits s'anomena **paraula**, encara que s'utilitza molt el terme anglès **word**. El programa codificat en binari en *words* es guarda en la memòria del PLC. En la taula 4 teniu un exemple de com es codifiquen en binari les instruccions de programació d'un autòmat programable.

Taula 4. Exemple d'una possible codificació en binari de les instruccions d'un PLC

Instrucció	Codi operació bits 1 a 3	Codi operand bits 4 a 6	Codi paràmetre bits 7 a 16
Inici	000		
Sèrie	010		
Paral·lel	101		
Final	111		
Entrada		000	
Sortida		111	
Paràmetre 0			0000000000
Paràmetre 1			0000000001
Paràmetre 2			0000000010
Paràmetre 3			0000000011
Paràmetre 4			0000000100
Paràmetre 5			0000000101
Paràmetre 6			0000000110

En la taula 5 teniu com es guardaria el programa de la figura 3 segons la codificació que hem posat com a exemple en la taula 4.

Taula 5. Exemple de com queda codificat un programa a la memòria del PLC

Programa general	Programa codi S7-200 Siemens	Programa codificat en binari
Inici entrada 3	LD I 0.3	000 000 0000000011
Paral·lel entrada 5	O I 0.5	101 000 0000000101
Sèrie entrada 6	A I 0.6	010 000 0000000110
Final sortida 4	= Q 0.0	111 111 0000000100

## 1.2.2. Memòries

Els programes que fa l'usuari perquè la instal·lació faci una tasca concreta i els que introdueix el fabricant perquè l'autòmat faci les operacions internes necessàries es guarden en les **memòries del PLC**, que són un circuits integrats que emmagatzemen tota la informació codificada en binari.

El programa s'introdueix mitjançant una programadora, que actualment consisteix en un programari que s'instal·la en qualsevol ordinador personal; així doncs, mentre s'està confeccionant el programa s'utilitza la memòria de l'ordinador i quan es vol provar el funcionament es transfereix el programa a la memòria del PLC.

Les memòries es poden dividir en dos grups segons la funció que fan: les memòries d'usuari i les de sistema.

- **Memòria d'usuari.** És la memòria on es guarda el programa fet pel tècnic en automatismes i que fa que la instal·lació funcioni executant la seqüència de treball dissenyada.
- **Memòries de sistema.** Són les memòries que programa el fabricant per emmagatzemar els programes i la informació necessàries per al funcionament intern correcte del PLC, entre d'altres:
  - Programa per convertir les instruccions introduïdes per l'usuari en un llenguatge que el PLC pugui processar, és a dir, per compilar el programa.
  - Dades de procés i de control.
  - Variables internes.
  - Dades de configuració del PLC.
 En funció de les tecnologies emprades i de seva funcionalitat, es pot donar la classificació següent de tipus de memòries:

- **RAM** (*random access memory*) o memòria d'accés aleatori

És una memòria volàtil, i per això si falla l'alimentació elèctrica es perden les dades emmagatzemades; es pot posar una pila o una bateria per mantenir les dades en cas que es talli l'alimentació elèctrica i es poden modificar les dades en qualsevol moment.

- **ROM** (*read only memory*) o memòria únicament de lectura

No perd la informació en cas de tall d'alimentació elèctrica; és una memòria amb un programa fix, muntada pel fabricant amb components electrònics, no està programada, l'usuari no hi té accés i no es pot modificar. Sol tenir les instruccions de funcions necessàries per al funcionament correcte de l'autòmat, com rutines d'inicialització, test, autodiagnòstic...



Programadores de PLC

- **PROM** (*programmable read only memory*) o memòria únicament de lectura programable

És com una memòria ROM, però que el fabricant ha programat, a diferència de la ROM, que està muntada físicament amb components electrònics, i una vegada programada no es pot modificar.

- **EPROM** (*erasable PROM*) o memòria únicament de lectura programable

És possible esborrar-la totalment si se sotmet a l'exposició de rajos ultraviolats entre 10 i 20 min la finestreta que té a la part superior del circuit integrat.

- **EEPROM** (*erasable electrically PROM, programmable read only memory erasable electrically*) o memòria únicament de lectura programable i esborrable elèctricament

És com una memòria PROM, però es pot esborrar tot o part del programa mitjançant la programadora, sense necessitat d'exposar-la als rajos ultraviolats, i no perd la informació si falla el corrent.

Els tipus de memòria d'usuari més utilitzats són la RAM, EPROM i EEPROM.

El programa que fabrica l'usuari normalment es fa en una memòria RAM, per la facilitat de fer modificacions, i una vegada que el programa és definitiu, es pot deixar en aquesta memòria amb el suport d'una bateria, o copiar-lo a una memòria EPROM si no es vol que es modifiqui, o a una memòria EEPROM si es preveu que durant el funcionament normal de la instal·lació s'hagi de modificar alguna dada.

Els programes propis del sistema utilitzen tots els tipus de memòria: RAM i EEPROM per a les dades que es modifiquen durant el processament del programa, i les ROM, PROM i EPROM per a les dades fixes.

### 1.3. Arquitectura i funcionament intern dels autòmats programables

L'autòmat programable és un equip electrònic programable que en funció de l'estat de les entrades i de l'algorisme de control que té introduït mitjançant el programa, dona ordres de connexió o desconnexió a les sortides; a escala interna es pot dir que tots els autòmats programables estan constituïts per tres grans blocs:

- **CPU** (*central processing unit*) o unitat central de processament.

És l'encarregada d'organitzar el tràfic de dades, fer les operacions internes i controlar les entrades i sortides.

El funcionament intern dels PLC està basat en un sistema amb microprocessador.

- **Memòries.** De diferents tipus de suport físic.

Contenen el programa fet per l'usuari, les dades que es produeixen mentre s'executa el programa i els programes introduïts pel fabricant per al funcionament intern del PLC.

- **Entrades i sortides**

Mitjançant els corresponents mòduls d'interfície per adaptar els senyals, s'encarreguen de comunicar la CPU amb l'exterior, reben senyals de les entrades i n'envien a les sortides.

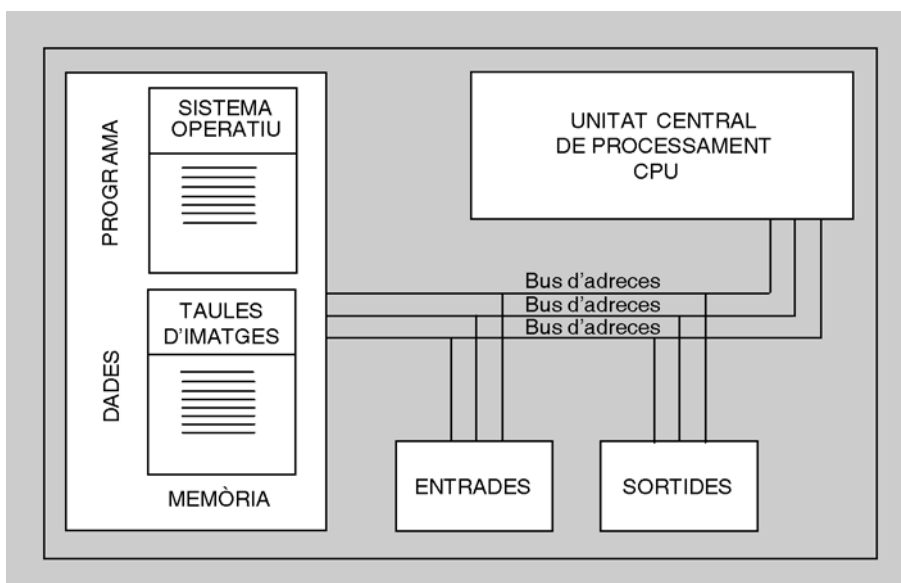
### 1.3.1. Arquitectura dels autòmats programables

Els tres grans blocs de què consten els PLC (CPU, memòries i entrades/sortides) es troben units mitjançant unes línies elèctriques per comunicar-se dades entre ells, anomenades **busos de comunicació**, que es divideixen en tres grups:

- **Bus d'adreces.** És per on el processador envia l'adreça de l'element al qual vol enviar informació o que vol que li envii informació.
- **Bus de dades** És per on tots els elements envien les dades. És bidireccional, i la CPU pot posar dades perquè una adreça la llegeixi o pot llegir una dada que li envii una adreça.
- **Bus de control.** És on el processador indica quina operació està fent; les més freqüents són llegir i escriure. Controla els blocs de memòria i d'entrades i sortides.

En la figura 4 teniu un exemple de com es comuniquen els diferents blocs que configuren els PLC.

Figura 4. Blocs constituents dels autòmats programables

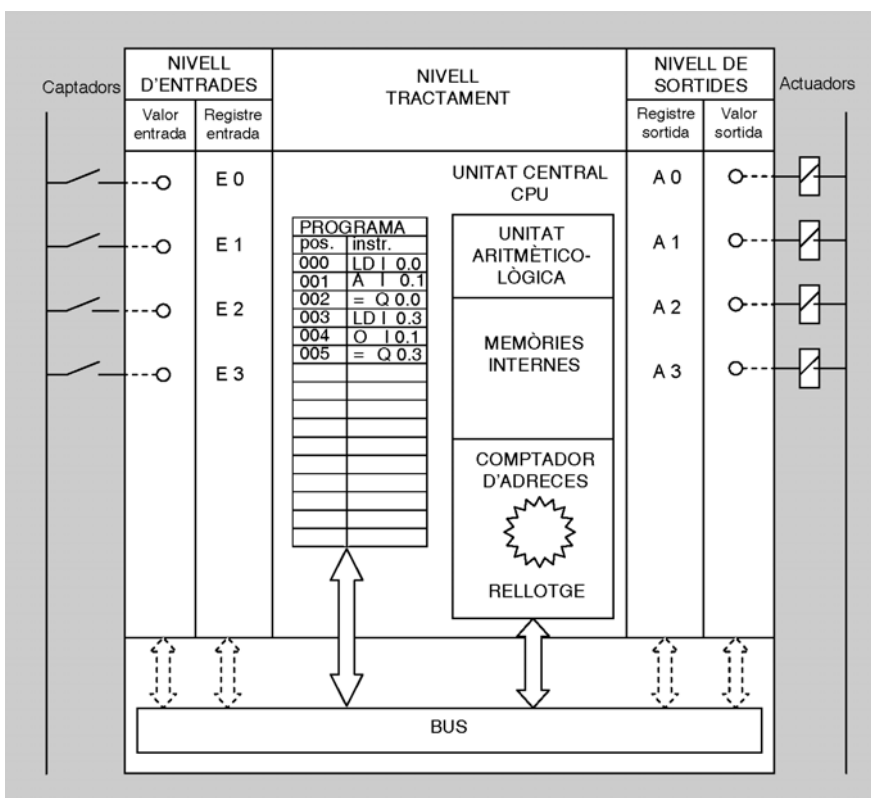


La **CPU** dels autòmats programables s'encarrega d'organitzar totes les tasques de control i comunicació entre els diferents blocs, per executar el programa fet perquè la instal·lació funcioni.

Els components que intervenen en el procés d'execució del programa (representats en l'esquema de la figura 5) són els següents:

- Valor d'entrada: es posa a 1 o 0 en funció de l'estat del captador.
- Registre d'entrada: emmagatzema el valor de l'entrada corresponent.
- Programa: instruccions corresponents a l'esquema elèctric que defineix el funcionament de la instal·lació.
- ALU (unitat aritmètica): és l'encarregada de fer les operacions aritmètiques, bàsicament sumes o productes, en funció del tipus de connexió dels contactes del circuit, paral·lel o sèrie.
- Memòries internes: emmagatzemen els resultats de les operacions aritmètiques fetes per l'ALU, les dades de configuració del PLC, i altres dades de control.
- Comptador d'adreces o rellotge: encarregat de dir en cada moment quina instrucció del programa s'ha d'executar. La CPU executa totes les instruccions del programa, d'una en una i en l'ordre que li marca el rellotge.
- Registre de sortides: emmagatzema el valor corresponent a la sortida en funció del resultat de l'execució del programa.
- Valor de sortida: es posa a 1 o 0 quan la CPU carrega el valor del registre de sortides a la sortida corresponent.

Figura 5. Components que intervenen en l'execució del programa



### 1.3.2. Funcionament intern dels autòmats programables

Els PLC poden tenir bàsicament dos estats: *run* i *stop*.

- **Run.** El PLC està actiu, la CPU llegeix el programa i dona ordres de connexió o desconnexió als actuadors.
- **Stop.** El PLC està connectat al corrent però la CPU no llegeix el programa i, per tant, no es dona cap ordre de connectar les sortides.

Quan el PLC està en *run*, la CPU, unitat central de processament de l'autòmat, executa el programa cíclicament, és a dir, llegeix contínuament el programa, instrucció rere instrucció.

Hi ha tres conceptes que heu de tenir clar quan treballeu amb autòmats programables, que són: *scan*, *temps de scan* i *watchdog*.

*Watchdog* significa 'gos de vigilància', i fa referència a la tasca que duu a terme de controlar el temps de cicle.

#### 1) *Scan*

Cada cop que executa el programa una vegada, es diu que ha executat un cicle o un *scan*.

L'execució d'un *scan* consisteix en el següent:

- a) La CPU fa un autodiagnòstic intern i dels perifèrics.
- b) Mira el valor que tenen totes les entrades i guarda els valors en el registre d'entrades corresponent. Cada entrada té un bit que emmagatzema el seu valor en el registre d'entrades.
- c) El rellotge marca la primera adreça per llegir-ne la instrucció.
- d) La unitat aritmeticològica fa l'operació corresponent a la instrucció.
  - Aquesta operació pot ser:
    - LD: inici, llegeix el valor de l'operand corresponent.
    - A: sèrie, fa l'operació aritmètica del producte.
    - O: paral·lel, fa l'operació aritmètica de la suma.
    - =: final, assigna el valor a l'operand corresponent.
  - En aquest procés, quan ha de llegir el valor d'una entrada ho fa del registre de l'entrada, no de l'entrada directament, i si ha d'assignar un valor a una sortida, ho fa sobre el registre de la sortida.
- e) Guarda en la memòria interna el resultat de l'operació aritmètica si aquesta és LD, O, A, o el guarda en el registre de sortides si l'operació aritmètica és =.

- f) El rellotge marca l'adreça següent.
- g) Es repeteixen el passos 3 i 4 fins que executa la darrera instrucció.
- h) Mira els valors emmagatzemats en el registre de sortides i els assigna a les sortides corresponents.
- i) Es torna a repetir el procés des del pas 1 de manera cíclica.

*Scan* és l'execució del programa una vegada i consisteix en el següent:

- Fer un autodiagnòstic.
- Llegir l'estat de totes les entrades i carregar els valors al registre d'entrades.
- Executar tot el programa llegint les dades de les entrades del registre d'entrades i escrivint els resultats de les combinacions en el registre de sortides.
- Copiar els valors del registre de sortides a les sortides.

## 2) Temps de *scan* o de cicle

És el temps que triga la CPU a executar un cicle; aquest temps és molt important, ja que les ordres de connexió o desconexió en els actuadors connectats a les sortides es fan al final del cicle, encara que la instrucció corresponent estigui al principi. Per tant, si el temps és massa llarg, podria succeir que es donés l'ordre d'aturar un motor perquè ha saltat una alarma i aquesta trigués massa a executar-se.

### Temps de cicle

Mentre s'executa el cicle, les ordres de connexió i desconexió es fan en el registre de sortides, no en la sortida, que es fa en finalitzar el cicle.

El temps de cicle depèn del següent:

- Tipus de CPU: tots els fabricants d'autòmats disposen de diferents CPU per a cada model de PLC, i així s'adapten a les necessitats dels usuaris. Una de les característiques més importants que diferencien les diferents CPU és la velocitat de processament.
- Nombre i tipus d'instruccions del programa: com més instruccions tingui el programa, més temps trigarà la CPU a llegir-lo, de la mateixa manera que hi ha instruccions que són més complexes que altres i lògicament es tarda més a executar-les.
- Nombre i tipus de perifèrics: al principi de cada cicle es fa un autodiagnòstic dels perifèrics; per tant, com més n'hi hagi i més complexos siguin més temps trigarà la CPU.

És més complex processar una entrada analògica que una de digital.

### 3) *Watchdog*

El *watchdog* controla la durada del temps de cicle, i quan aquest supera els valors establerts, dóna ordre d'aturar l'autòmat i passa de *run* a *stop*; per tant, la CPU deixa d'executar el programa i es desconnecten totes les sortides.

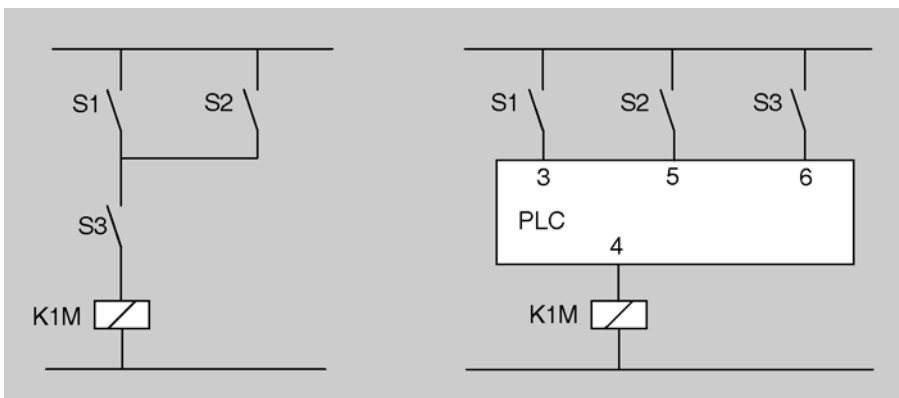
El **temps del *watchdog*** pot ser fix o programable, i depèn del tipus del CPU.

Les CPU petites, dirigides a instal·lacions senzilles i sense requeriments de seguretat, estan configurades de fabrica amb un temps de *watchdog* que el fabricant considera acceptable, i que pot arribar fins a 500 ms; en d'altres CPU, les més potents, el temps de cicle es pot definir en funció de la complexitat de la instal·lació i, com són molt més ràpides, es poden fixar temps entre 10 i 200 ms.

#### 1.3.3. Interpretació del programa dels autòmats programables

Un **programa d'un PLC** és la traducció d'un esquema elèctric en un conjunt d'instruccions, però el PLC el que fa és interpretar el programa i no l'esquema elèctric. En la figura 6 tenim un esquema elèctric que correspon a un circuit que hem muntat amb un PLC; s'hi pot veure en quines entrades i sortides estan connectats els captadors i els actuadors.

Figura 6. Interpretació del programa. Connexió elèctrica



En les taules 6 i 7 tenim el programa corresponent a l'esquema amb les operacions que fa la CPU en cada línia de programa en funció de l'estat de les entrades.

Taula 6. Interpretació del programa. Lectura de la CPU amb el captador S3 accionat

Adreça	Programa	Valor d'entrades o sortides	Unitat aritmètica	Memòria interna
001	LD I 0.3	0	Carrega el valor 0	0

Adreça	Programa	Valor d'entrades o sortides	Unitat aritmeticològica	Memòria interna
002	O I 0.5	0	$0 + 0 = 0$	0
003	A I 0.6	1	$0 \times 1 = 0$	0
004	= Q 0.0		Assigna el valor 0	0

Interpretació de la taula 6:

- Com que només està accionat el captador S3, fixeuvos que el valor corresponent a la seva entrada, la I0.6, està a 1 i la resta d'entrades estan a 0.
- En la instrucció 001 hi ha programada l'operació LD; per tant, la CPU carrega el valor de l'entrada programada en aquesta instrucció, l'entrada I0.3, que val 0, en la memòria interna.
- En la instrucció 002 està programada la instrucció O, que correspon a una connexió en paral·lel, però la CPU del PLC, que no interpreta cap esquema elèctric, fa l'operació de sumar i suma el valor guardat a la memòria interna, que és 0, al valor de l'entrada programada en aquesta instrucció, I0.5, que també és 0, i el resultat de l'operació, que és 0, el torna a guardar en la memòria interna.
- En la instrucció 003 està programada la instrucció A, que correspon a una connexió en sèrie, però la CPU del PLC fa l'operació de multiplicar i multiplica el valor guardat en la memòria interna, que és 0, amb el valor de l'entrada programada en aquesta instrucció, I0.5, que és 1, i el resultat de l'operació, que és 0, el torna a guardar en la memòria interna.
- En la instrucció 004 està programada l'operació =, que correspon a l'ordre d'assignar el valor guardat en la memòria interna, que és 0, al registre de la sortida programada en aquesta instrucció, que és la Q0.0.

Ara podeu analitzar el funcionament del circuit quan estan accionats els captadors S2 i S3 (taula 7), que corresponen a les entrades I0.5 i I0.6, seguint el mateix raonament que hem fet amb la taula 6.

Taula 7. Interpretació del programa. Lectura de la CPU amb els captadors S2 i S3 accionats

Adreça	Programa	Valor d'entrades o sortides	Unitat aritmeticològica	Memòria interna
001	LD I 0.3	0	Carrega el valor 0	0
002	O I 0.5	1	$0 + 1 = 1$	1
003	A I 0.6	1	$1 \times 1 = 1$	1
004	= Q 0.0		Assigna el valor 1	1

La CPU del PLC, sense interpretar cap esquema elèctric, però llegint el programa i fent les operacions aritmètiques en funció de les operacions programades, arriba a la mateixa conclusió que raonant l'esquema, i dóna ordre de desconnectar la sortida Q0.0 en l'exemple de la taula 6 i de connectar-la en el de la taula 7.

### 1.3.4. Estructura i components dels autòmats programables

Hi ha molts tipus d'autòmats programables, i la classificació que se'n pot fer és molt diversa, en funció dels criteris que es vulguin aplicar, però la més usual és segons l'estructura física que té i segons com estan ubicats tots els components del PLC; atenent a aquest criteri els podem dividir en dos grans grups: compactes i modulars.

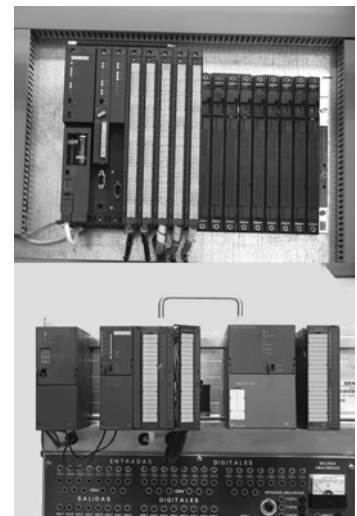
- **Estructura modular**

Cada component del PLC es troba en un mòdul diferent, anomenats normalment *targetes*, que es munten damunt d'un bastidor, que pot ser un carril DIN o un perfil específic de cada fabricant, i que s'uneixen elèctricament mitjançant els busos de comunicació, que poden estar incorporats al bastidor mateix, o mitjançant uns cables amb connectors que uneixen tots les targetes.

- **Estructura compacta**

En un únic element, hi ha tots els components del PLC: CPU, bateries, memòries, mòdul d'entrada i de sortida, etc.

Molts dels PLC compactes de gama baixa i amb poques entrades i sortides tenen incorporada una petita consola de programació. Es coneixen com a **microautòmats**, i actualment la majoria d'aquests autòmats tenen la possibilitat d'ampliar les funcions mitjançant mòduls perifèrics, com ho fan els modulars.



PLC modulars



PLC compactes

Independentment de l'estructura constructiva que tinguin els autòmats, tots estan configurats per una sèrie de components per fer les diferents funcions; els més comuns són:

- **Unitat central de processament o CPU.** La CPU és l'encarregada d'organitzar el trànsit de dades, executar el programa i controlar les entrades i sortides.
- **Mòdul d'alimentació.** És l'encarregat de subministrar totes les tensions auxiliars a la resta de components.

- **Mòdul de bateries.** Conté la pila que alimenta els relés amb memòria, el temporitzador horari o rellotge i la memòria RAM perquè no perdi la informació quan falla l'alimentació elèctrica.
- **Mòdul de memòries.** Conté el xip amb la memòria del programa.
- **Mòdul de temporitzadors.** La majoria de temporitzadors són programables, és a dir, el temps es pot variar mitjançant les instruccions de programació, però també n'hi ha d'ajustables, que tenen l'avantatge que el temps es pot modificar externament, ajustant un petit potenciòmetre, sense haver d'entrar en el programa.
- **Mòduls d'entrada i sortida digitals.** Són els mòduls on es connecten les entrades i sortides digitals que hi ha a la instal·lació, i tenen uns LED que ens indiquen en cada moment quins elements estan activats; en funció de les necessitats de l'usuari hi ha mòduls:
  - Per a diferents tensions: 24 V CC, 24 V CA, 127 V CA, etc.
  - Que suporten diferents intensitats: 20 mA, 2,5 A, etc.

També n'hi ha amb el següent:

- Separació galvànica, amb la qual cosa es protegeix el circuit interior del mòdul.
- Sense separació galvànica.

- **Mòduls d'entrada i sortida analògics.** N'hi ha de molts tipus:
  - Amb senyals d'intensitat o tensió.
  - Amb diferents resolucions, com 8, 10 o 12 bits.
  - Amb diferents codificacions: binari, BCD.
- **Mòduls especials.** Dissenyats per fer tasques complexes, s'utilitzen en PLC modulars i de gran potència; els més freqüents són: controladors PID, control d'eixos i comunicacions.

Els temporitzadors ajustables són adequats quan el personal que ha d'ajustar els temps no és tècnic en PLC.

#### Rangs de les entrades i sortides analògiques

La majoria d'entrades i sortides analògiques treballen amb els senyals elèctrics de 0-10 V CC (volts de corrent continu) i 4-20 mA (mil·liamperes).

## 1.4. Perifèria dels autòmats programables

La característica més important dels autòmats programables és que són uns equips totalment oberts, ampliables i dissenyats per admetre la connexió de molts tipus diferents de components; els més comuns en tots els PLC són les **entrades** i **sortides**, tant digitals com analògiques.

La majoria d'autòmats tenen incorporades a la CPU (unitat central de processament) una sèrie d'entrades i sortides que es coneixen com a *perifèria integrada*.

Per connectar o desconectar les targetes al PLC, el PLC ha d'estar sense alimentar.

### 1.4.1. Entrades digitals

Les entrades digitals subministren informació al PLC de l'estat de la instal·lació mitjançant les targetes d'entrada.

#### Tipus d'entrades

Les entrades poden ser amb contacte elèctric i sense contacte elèctric.

- Amb contacte elèctric: pulsadors, interruptors, finals de cursa, contactes auxiliars de contactors, contactes auxiliars de relés de protecció, *encoders*, etc.
- Sense contacte elèctric: detectors inductius, capacitius, fotoelèctrics, ultrasònics, etc.

Poden estar preparades per funcionar amb corrent continu o altern. N'hi ha que donen el senyal elèctric amb lògica positiva o negativa, i aquesta polaritat ha de coincidir amb la del tipus de targeta en què es connectin.

#### Tipus de detectors

Els detectors són entrades sense contacte elèctric, és a dir, no tenen un contacte físic que obre o tanca per transmetre el senyal, sinó que envien el senyal mitjançant un component electrònic. Hi ha detectors de dos o de tres fils.

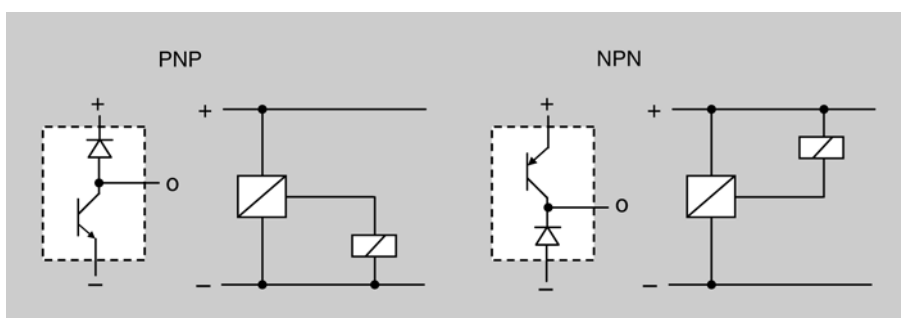
- Detectors de dos fils: s'ha de tenir en compte el corrent residual, que és el corrent que necessita el detector en estat obert per funcionar i que pot arribar a activar l'entrada del PLC. N'hi ha de dos tipus: no polaritzats i polaritzats.
  - No polaritzats: no importa la polaritat; n'hi ha per treballar amb corrent continu i amb corrent altern i alguns funcionen indistintament en totes dues tensions.
  - Polaritzats: només poden treballar amb corrent continu i se n'ha de controlar la polaritat. No tenen corrents residuals, necessiten alimentació externa i poden ser amb **lògica positiva (PNP)** o amb **lògica negativa (NPN)**. En la figura 7 podeu veure la polaritat a la sortida dels detectors PNP i NPN.
- Detectors de tres fils: tots els detectors de tres fils són polaritzats.



Tipus de detectors

Si intercanviem els cables de connexió en un detector polaritzat, es pot avariar.

Figura 7. Esquema detectors NPN i PNP



#### Connexió de detectors de tres fils

Si el detector és del tipus PNP, vol dir que dona sortida + a la bobina i, per tant, el comú de la bobina s'ha de connectar al -. Si és del tipus NPN, és a l'inrevés.

## Condicionament del senyal de les entrades

Les targetes d'entrada s'encarreguen d'adaptar i filtrar el senyal procedent de les entrades, normalment a 24 V, a les característiques del funcionament intern del PLC, que solen treballar a 5 V.

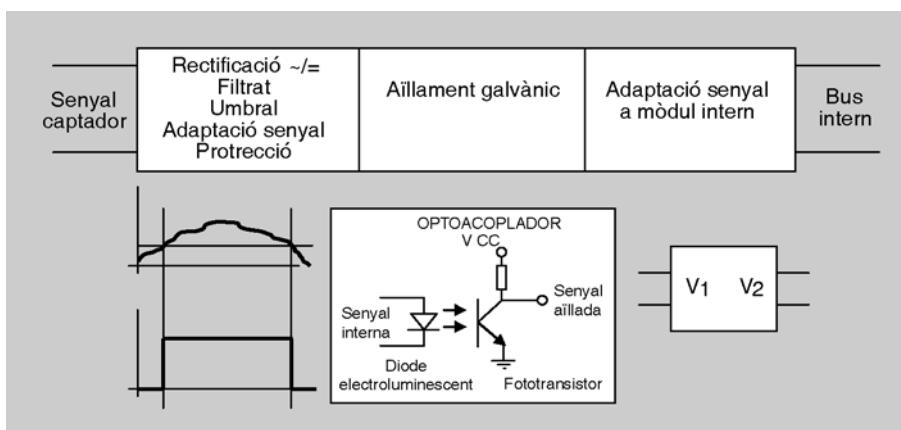
Les targetes d'entrada poden ser de dos tipus, amb el senyal d'entrada aïllat galvànicament o sense aïllar.

L'aïllament galvànic del senyal pot ser:

- a) Inductiu: per a entrades amb senyals de corrent altern; es fa mitjançant un transformador.
- b) Òptic: per a entrades amb senyals de corrent continu; es fa mitjançant un *optoacoblador*.

En la figura 8 podeu veure el procés d'adaptació del senyal.

Figura 8. Condicionament del senyal d'entrada



### Optoacoblador

El senyal que arriba a l'optoacoblador produeix un raig de llum en el díode electroluminescent que capta el fototransistor i el converteix un altre cop en un senyal elèctric proporcional al raig de llum.

Fixeu-vos que el senyal del captador que arriba a la targeta d'entrades és tractat en tres fases per adaptar-la al bus intern del PLC:

- 1) Si el senyal és de corrent altern:
  - a) Es converteix a corrent continu.
  - b) El senyal passa per un filtre per evitar que hi entrin paràsits.
  - c) Es defineixen els valors mínims i màxims del senyal, en què es considera l'entrada activada o no, s'eliminen les petites fluctuacions i es converteix el senyal en un valor digital, que serà 1 o 0.
  - d) Es disposen els elements de protecció per evitar sobretensions.

2) S'aïlla galvànicament el senyal que arriba de l'exterior del senyal interior, és a dir, no hi ha contacte elèctric entre tots dos senyals; en aquest exemple es fa mitjançant un optoacobrador.

3) S'adapta la tensió generada per l'optoacobrador als valors en què treballa internament la targeta d'entrades.

Els mòduls o targetes d'entrada s'encarreguen del següent:

- Amplificar el senyal que arriba del captador.
- Adaptar els nivells de tensió del captador als del PLC.
- Filtrar el senyal per evitar comportaments no volguts.
- Codificar el senyal perquè pugui ser processat pel PLC.

### 1.4.2. Sortides digitals

Les sortides digitals reben ordres del PLC i en posar-se en funcionament fan que la instal·lació faci les tasques per a la qual ha estat concebuda.

Els **actuadors** o elements de sortida més freqüents són:

- Pilots i pantalles per subministrar informació.
- Motors i cilindres pneumàtics i hidràulics per realitzar moviments.

Els motors no es poden connectar directament a les targetes de sortida, ja que acostumen a ser trifàsics i de molta potència, i els cilindres realitzen els moviments mitjançant les electrovàlvules, que també tenen una potència superior a la que subministren les targetes; així doncs, aquests elements es posen en funcionament des del PLC a través dels **preactuadors**, que adapten el senyal elèctric que subministra la targeta al senyal que necessiten els actuadors per funcionar correctament.

#### Tipus de targetes de sortida

- Amb sortida de tensió alterna o contínua.
- Amb lògica positiva o negativa.
- Lliures de potencial o a una tensió determinada.
- Aïllades galvànicament o sense aïllar.
- Tipus d'aïllament:
  - Aïllament inductiu o magnètic mitjançant relé.
  - Aïllament òptic mitjançant optotriac o optoacobrador.

#### Targetes de sortida a corrent continu

El senyal de sortida pot ser per transistor o per relé.

#### Preactuadors

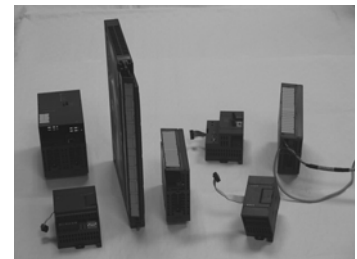
Els preactuadors més freqüents són els contactors; tenen la bobina amb un consum molt baix i es pot connectar directament a la targeta de sortida, i mitjançant els seus contactes de potència es pot activar qualsevol actuator.

- Per transistor (estat sòlid):
  - Entra en conducció quan la sortida s'activa.
  - És adequat per accionar elements de petita potència.
  - Pot ser amb negatiu comú o positiu comú.
- Per relé (lliure de potencial):
  - Són sortides lliures de potencial, la sortida es connecta a un contacte del relé que no està alimentat elèctricament.
  - Cada sortida pot accionar una sortida a tensió diferent.
  - Té l'inconvenient de la velocitat de resposta.

### Targetes de sortida de corrent altern

El senyal de sortida pot ser per triac o per relé.

- Per triac (estat sòlid):
  - Entra en conducció quan la sortida s'activa.
  - Quan la sortida es desactiva, el triac continua conduint fins al primer pas per zero del corrent, i per aquest motiu no serveix en corrent continu.
- Per relé (lliure de potencial):
  - Contacte lliure de potencial.
  - Cada sortida pot accionar un accionador a tensió diferent.
  - Té l'inconvenient de la velocitat de resposta.



Targetes PLC

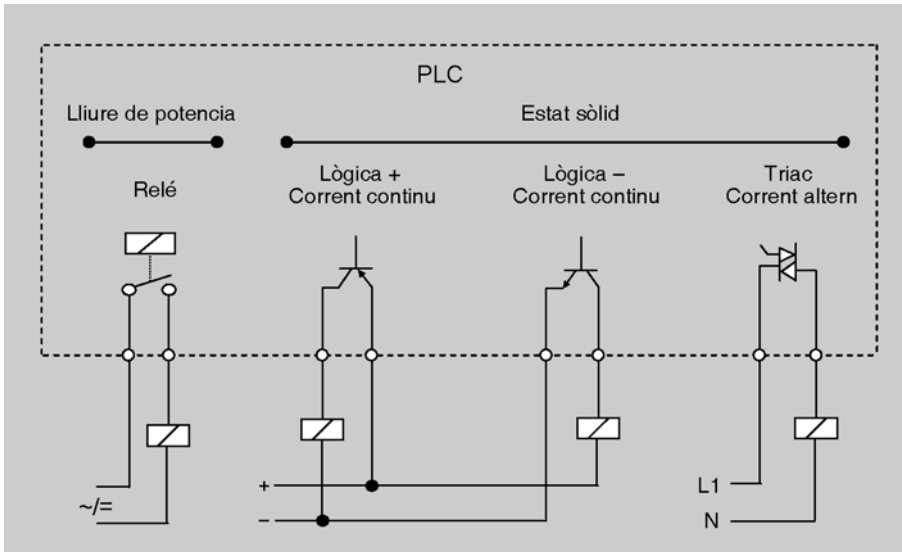
Els mòduls o targetes de sortida s'encarreguen de:

- Descodificar la informació procedent del registre de sortides (rebre els valors 0 i 1).
- Amplificar i codificar el senyal de sortida que s'envia a l'actuator.
- Adaptar el nivells de tensió del senyal al tipus d'actuadors.

Les targetes de sortida que envien el senyal a l'exterior mitjançant un relé s'anomenen **lliures de potencial** i les que ho fan mitjançant un component electrònic com díodes o triacs, s'anomenen **d'estat sòlid**.

En la figura 9 podeu veure com es connecten els actuadors a la targeta de sortides en funció del tipus de senyal que envia el mòdul de sortida.

Figura 9. Connexió de les sortides



### 1.4.3. Senyals analògics

Els **senyals analògics** s'utilitzen per convertir magnituds físiques, com temperatures, cabals, pressions, etc., en senyals elèctrics, mitjançant una relació entre els valors mínims i màxims de totes dues magnituds.

La temperatura d'una sala amb un calefactor augmenta de manera progressiva, sense fer salts.

Les variacions produïdes en la magnitud física produeixen una variació del senyal elèctric directament proporcional a aquest senyal elèctric analògic, que obté infinits valors entre els seus valors mínim i el màxim, i necessita convertir-se en un senyal digital per poder ser tractat per l'autòmat. Aquest senyal digital també serà proporcional al senyal elèctric analògic, però no obtindrà valors infinits, sinó que entre el valor mínim i màxim canviarà d'unitat en unitat i en funció del nombre de bits del seu convertidor tindrà més o menys resolució.

#### Característiques dels senyals analògics

Els senyals analògics poden ser de:

- Tensió:
  - Els valor més freqüents són 0 a 10 V i -10 a +10 V.
  - Poden tenir caigudes de tensió en funció de la distància.
- Intensitat:
  - Els valors més freqüents són 0 a 20 mA i 4 a 20 mA; aquest últim és el més emprat perquè detecta si el fil es trenca.
  - Els senyals d'intensitat no els afecta la distància.

En un senyal analògic de 0 a 20 mA, no hi diferència entre el valor mínim de la magnitud física, que donaria 0 mA, i la trencadora de fil, que també donaria 0 mA.

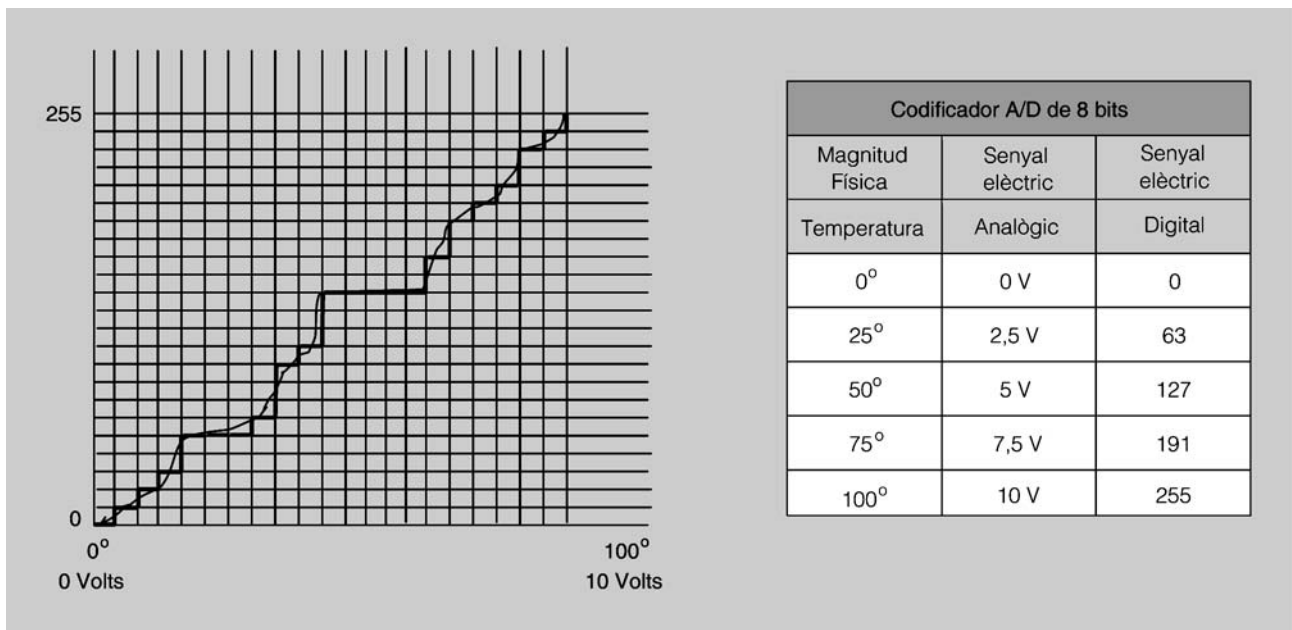
Els senyals d'entrada requereixen un convertidor A/D (analògic/digital) per convertir el senyal analògic en digital, i els de sortida un convertidor D/A (digital/analògic); aquests convertidors poden ser de 8, 10 o 12 bits.

La resolució del convertidor varia en funció del nombre de bits; com més bits millor resolució, i per tant més precisió.

La **resolució d'un convertidor A/D o D/A** és la capacitat que té per discernir entre dos valors molts similars de la magnitud d'entrada.

En la figura 10 podeu observar la relació entre el valor analògic i el digital d'un senyal, i com el senyal analògic varia de manera progressiva sense esglaons i el digital fa salts d'un en un, entre 0 i 255 unitats, en un exemple amb un codificador A/D de 8 bits.

Figura 10. Gràfic d'un senyal analògic



### Càlcul de la resolució d'un convertidor

Amb un convertidor de 8 bits es poden fer 256 combinacions binàries diferents, des de 0000 0000, fins a 1111 1111, corresponents als valors en decimal de 0 a 255, i si el senyal analògic que li entra és 0-10 V, vol dir que la sortida digital incrementarà positivament o negativament una unitat el seu valor cada:  $10/256 = 0,03906 \sim 39 \text{ mV}$ .

En la taula 8 podeu comprovar la diferència de resolució entre els convertidors de 8, 10 i 16 bits.

Taula 8. Càlcul de la resolució dels codificadors A/D o D/A

Nre. de bits	Valor de tensió	Valor decimal	Càlcul de la resolució	Valor de la resolució
8	0-10 V	0-255	$10 / 256 = 0,0390$	39 mV
10	0-10 V	0-1.023	$10 / 1.024 = 0,0097$	9,8 mV
12	0-10 V	0-4.095	$10 / 4.096 = 0,0024$	2,4 mV

Observeu com passant d'un convertidor de 8 bits a un de 10 augmentem la resolució 4 vegades, i amb un de 12 bits, 16 vegades.

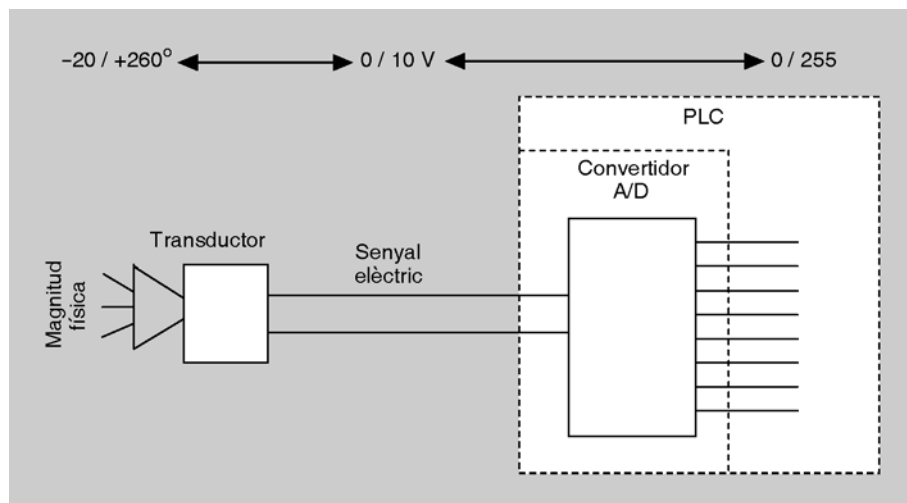
#### 1.4.4. Entrades analògiques

Les entrades analògiques s'utilitzen per fer mesures de totes les magnituds físiques que es produeixen en un procés de fabricació, com nivells, pressions, cabals, nivell de CO<sub>2</sub>, etc.

El captador per a senyals analògics consta bàsicament de dues parts:

- **Transductor**, que s'encarrega de convertir la magnitud física a mesurar en una magnitud elèctrica.
- **Convertidor A/D**, analògic/digital, que converteix el senyal elèctric analògic en un senyal digital (figura 11), quan aquest senyal s'ha d'enviar a un PLC, un autòmat programable, s'utilitzen captadors sense convertidor A/D, ja que les entrades analògiques dels PLC tenen incorporats els convertidors.

Figura 11. Exemple del procés de transformació de l'entrada analògica



#### Procés de transformació del senyal analògic

El valor de la temperatura es troba entre  $-20$  i  $+260^{\circ}$ , es converteix en un valor de tensió entre 0 i 10 V i mitjançant el convertidor A/D es torna a convertir en un valor binari entre 0 i 255.

Per a la connexió de les entrades analògiques, convé utilitzar fil trenat i apantallat per evitar perturbacions i obtenir valors incorrectes, i s'ha de connectar la pantalla al terra al costat del PLC, i el negatiu al costat del captador.

Les entrades analògiques requereixen un convertidor A/D (analògic/digital) per adaptar el senyal del captador al PLC. Aquest convertidor el pot tenir el captador mateix, o si es treballa amb PLC el té incorporat el mòdul d'entrades analògiques mateix del PLC.

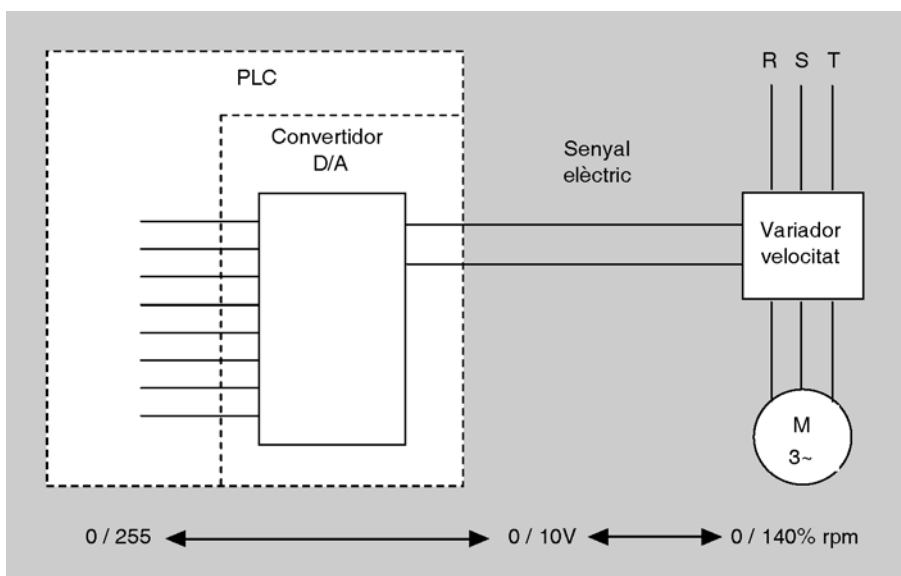
### 1.4.5. Sortides analògiques

Les sortides analògiques permeten accionar vàlvules proporcionals, variadors de velocitat i tot tipus d'actuadors en què convingui que el funcionament no sigui tot o res, sinó que es pugui ajustar.

Les sortides analògiques requereixen un convertidor D/A (digital/analògic) per adaptar el senyal del PLC. Aquest convertidor el pot tenir l'actuador mateix, o si es treballa amb PLC el té incorporat el mòdul de sortides analògiques mateix del PLC.

En la figura 12 podeu comprovar el procés de transformació del valor binari que processa el PLC en un valor analògic de tensió entre 0 i 10 V, que produeix en el variador de velocitat uns canvis de freqüència que fa que el motor giri entre el 0% i el 140% de la seva velocitat nominal.

Figura 12. Exemple del procés de transformació de la sortida analògica



Per a la connexió de les sortides analògiques, convé utilitzar fil trenat i apantallat, i s'ha de connectar la pantalla al terra al costat del PLC, i el negatiu al costat de l'actuador, atès que el negatiu de les diferents sortides del mateix PLC sol ser comú.

## 1.5. Funcionalitat dels autòmats programables

Tots els autòmats tenen una sèrie de característiques comunes i que és important conèixer, com l'estat de funcionament, que pot ser:

- *Stop*: està connectat a l'alimentació però la CPU (unitat central de processament) no executa el programa i, per tant, la instal·lació no funciona.

### Sortida analògica

Una vàlvula proporcional permet obrir més o menys la comporta d'un dipòsit en funció de la quantitat de producte que volem que caigui.



Variador de velocitat

### Programadora d'autòmats programables

Fa uns quants anys cada PLC tenia la seva consola de programació, i actualment tots els fabricants de PLC utilitzen ordinadors portàtils amb els programaris corresponents, com programadores.

- *Run*: la CPU executa el programa i la instal·lació funciona.

La programadora pot treballar en dues situacions respecte al PLC:

- *Off line*: la programadora treballa sense estar connectada al PLC.
- *On line*: la programadora està connectada al PLC i pot intercanviar informació.

Quan el PLC treballa en línia s'ha de tenir molta cura amb les operacions que es fan, ja que pot variar el funcionament de la instal·lació, i per això només ho ha de fer personal especialitzat.

Quan s'elabora un programa es poden fer moltes funcions que ajuden a l'edició d'aquest; les més comunes són:

- Esborrar un element o conjunt d'elements.
- Inserir un element o conjunt d'elements.
- Modificar el tipus, adreça o dades associades a l'element.
- Esborrar, copiar o afegir blocs o subrutines.
- Cercar o substituir elements.
- Crear taules d'ocupació d'elements i referències creuades.

Mentre la instal·lació està funcionant és molt interessant poder monitorar i intervenir en l'execució del programa; tots els PLC tenen funcions com:

- Visualització de l'estat dels elements.
- Visualització de l'estat del programa.
- Forçar entrades, sortides, registres, temporitzadors, etc.

### 1.5.1. Llenguatges de programació

Els autòmats han adaptat els llenguatges de programació als usuaris d'aquests equips i són molt intuïtius; els més usuals són:

- Esquema de contactes: es programa amb un esquema elèctric.
- Diagrama de funcions: es programa mitjançant un circuit de portes lògiques.
- Llista d'instruccions: utilitza unes instruccions relacionades amb l'àlgebra de Boole.
- GRAFCET, *graphe de commande étape-transition*: es descriu el funcionament de la instal·lació sense necessitat de raonar el circuit elèctric o electrònic.
- Llenguatges d'alt nivell: per fer programes molt complexos es poden confeccionar en BASIC, Pascal, C, etc.



Per veure un exemple de com es pot realitzar un programa en els diferents llenguatges de programació que podeu utilitzar amb els autòmats programables, aneu a la secció "Annexos" del web d'aquest mòdul.

El programari del PLC converteix automàticament els programes fets en un llenguatge de programació concret, en els altres llenguatges disponibles en el programari.

### 1.5.2. Característiques generals dels autòmats programables

Es pot definir un PLC com un equip que té les característiques següents:

- És una caixa “negra” configurada per mòduls estàndards, de configuració fàcil per a l'usuari o programador.
- És de dimensions reduïdes i amb gran capacitat de compactació.
- És fàcil de muntar i instal·lar.
- És fàcil de programar amb programaris molt intuïtius.
- Disposa de programaris que generen molta documentació del projecte.
- Disposa de programaris que subministren molta informació en línia.
- Disposa de la possibilitat d'emmagatzemar molts programes.
- És de reutilització fàcil per part d'altres aplicacions.

Un **autòmat programable** és un equip electrònic amb un microprocessador, que rep uns senyals de les entrades i que, en funció de l'algorisme de control que s'hagi programat, dona unes ordres de connexió a les sortides.

La utilització dels PLC en el camp de l'automatització ha representat una sèrie d'avantatges molt importants, com ara els següents:

- Menys temps dedicat a la realització del projecte, ja que:
  - No és necessari dibuixar esquemes de manera detallada, perquè els genera el programari del PLC mateix.
  - Disposa de molta memòria i no és necessari simplificar les equacions.
  - S'utilitzen menys materials i components, ja que el PLC té incorporades moltes funcions.
- Possibilitat d'introduir modificacions sense canviar el cablatge i sense haver d'afegir components.
- Menys cost de mà d'obra en la instal·lació.
- Economia en el manteniment, ja que en el PLC mateix es poden programar funcions per a la detecció d'errors i tasques de manteniment.
- S'utilitzen pocs elements amb accionament electromecànics.
- Possibilitat de governar diverses màquines amb un sol PLC. En cas de necessitar-ne diversos es poden comunicar fàcilment.
- Com són modulars és fàcil d'adaptar la seva constitució i funcionalitat.
- Hi ha la possibilitat d'introduir modificacions en línia, sense haver d'aturar la producció.
- Simplificació de les posades en marxa.

Amb tota la gamma d'autòmats que hi ha al mercat, és molt important tenir en compte una sèrie de dades a l'hora de seleccionar el més adequat per a cada projecte; entre altres, poden ser:

- Dades de catàleg: marca, model, fabricant, data de comercialització.
- Mapa de memòria: nombre màxim d'entrades, sortides, temporitzadors, etc.
- Dades de la CPU (unitat central de processament): temps de *scan* per a 1 kB de memòria, operacions que fa, etc.
- Ampliacions: tipus i quantitat de mòduls que es poden acoblar.
- Comunicacions: tipus de xarxes de comunicació que es poden configurar.
- Programació: llenguatges de programació disponibles, possibilitat de programació en línia, etc.
- Criteris econòmics: preu de l'equip, formació, assessorament, manteniment, servei tècnic, etc.

### 1.5.3. Evolució dels autòmats programables

Els autòmats programables van aparèixer l'any 1968 per encàrrec de l'empresa automobilística General Motors, ja que en les seves cadenes de producció es trobaven que hi havia:

- Excessives avaries en components electromecànics com els relés, dels quals n'hi havia milers en els quadres de control.
- Gran varietat de targetes electròniques: cada targeta estava dissenyada per a una funció concreta.
- Dificultat per fer modificacions en els processos de producció quan es volia introduir alguna millora.

El PLC apareix l'any 1968 com el substitut del següent:

- **Armaris de relés.** Internament disposen de multitud de relés, temporitzadors i altres components de comandament.
- **Targetes electròniques.** El PLC està basat en un microprocessador i es poden programar totes les funcions electròniques que calguin.
- **Rigidesa en els processos automatitzats.** Els PLC són programables, i per això és fàcil fer modificacions en els processos de producció.

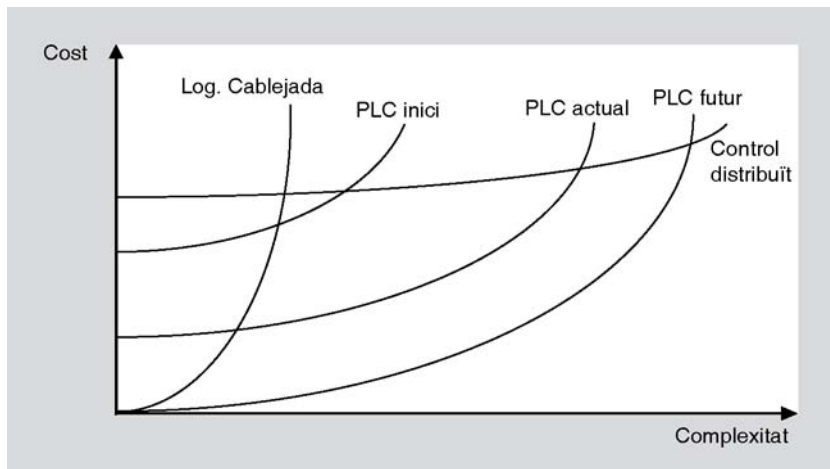
Actualment disposen de funcions de càlcul complexes, com operacions matemàtiques, llaços de control, control numèric, etc., disposen de mò-

duls de comunicació per a diferents xarxes, com PROFIBUS, ethernet, etc. Poden comunicar-se amb sistemes SCADA (sistemes d'adquisició i visualització de dades en processos de control) per visualitzar i controlar el procés des d'un ordinador, poden treballar en sistema redundat en el qual dos PLC controlen un procés i comparen en cada *scan* els registres d'entrades i sortides i, si detecten discrepàncies, porten el procés a una posició estable i segura i s'aturen els dos, o poden tenir un sistema que determini quin dels dos equips falla.

En un futur no molt llunyà serà el substitut dels ordinadors de processos i dels equips de control distribuït en aplicacions mitjanes.

En la figura 13 podeu observar l'evolució dels PLC des de la seva aparició: fixeuvos que en un principi resultava car per a petites aplicacions i no servia per a aplicacions molt complexes, però actualment la relació cost-complexitat és molt competitiva, i comprèn quasi tot el terreny de la lògica cablejada i moltes aplicacions dels controls distribuïts, i la tendència en un futur molt pròxim és agafar tot el camp de la lògica cablejada i la majoria d'aplicacions dels controls distribuïts.

Figura 13. Evolució dels PLC



## 2. Instruccions bàsiques de programació dels autòmats programables 1

L'autòmat programable és un equip electrònic amb un microprocessador que rep uns senyals de les entrades i que, en funció de l'algoritme de control que s'hagi programat, dona unes ordres de connexió a les sortides. Cada fabricant d'autòmats programables utilitza un programari de programació propi; per tant, heu de conèixer l'autòmat i el programari de l'autòmat S7-200 de Siemens, amb el qual fareu tots els exemples de programació.

Amb els autòmats podeu fer programes per a instal·lacions molt senzilles, i en aquest cas parlariem d'**automatismes**, o molt complexes, i en aquest cas parlariem d'**automatització i control industrial**. La diferència està en el tipus d'instruccions que utilitzem: normalment es diu que en automatismes s'utilitzen molt les instruccions de bit i en control industrial les instruccions de paraula o *word*.

### 2.1. L'autòmat programable S7-200 de Siemens

L'autòmat programable S7-200 de Siemens és un PLC compacte, ja que incorpora en una mateixa carcassa la CPU (unitat central de processament), una font d'alimentació integrada, les memòries i els mòduls d'entrades i sortides, però a més és ampliable, per la qual cosa es poden connectar al bloc de la CPU altres mòduls d'entrades i sortides digitals i analògiques, mòduls de comunicació, etc. Totes les característiques d'aquest autòmat, tant de maquinari com de programari, són molt semblants a la majoria de PLC d'altres fabricants.

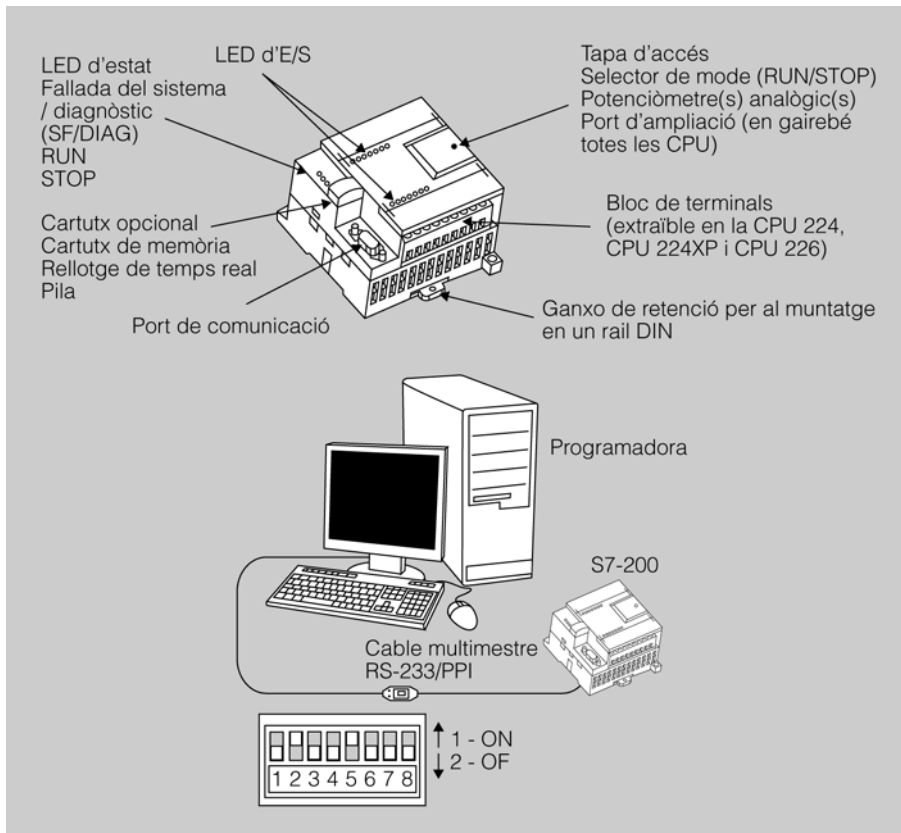
#### 2.1.1. Maquinari

La CPU 224 de l'autòmat S7-200 té unes dimensions molt petites, 120 × 80 × 62 mil·límetres, però unes característiques molt potents, com:

- Memòria de programa de 12 MB.
- 14 entrades i 10 sortides digitals.
- Ampliable fins a 7 mòduls.
- 2 potenciómetres analògics.
- Rellotge en temps real.
- Port de comunicació.
- 32 bytes de registres interns.
- 256 temporitzadors.
- 256 comptadors.

En la figura 14 podeu veure les característiques del maquinari de l'autòmat S7-200 i com es connecta a la programadora.

Figura 14. Maquinari autòmat programable



### 2.1.2. Programari

El programari que es fa servir per programar el PLC S7-200 és el **MicroWin**, que us permet escriure els programes en tres llenguatges de programació:

- KOP, o esquema de contactes.
- FUP, o diagrama de funcions.
- AWL, o llista d'instruccions.

Per poder escriure un programa heu de conèixer el mapa de memòria del PLC que esteu treballant; en la taula 9 teniu un resum del mapa de memòria del PLC S7-200 amb els elements mínims necessaris per començar a treballar amb el MicroWin.

Els autòmats de Siemens treballen amb els paràmetres en format de byte, és a dir, en agrupacions de 8 bits, i per això tots els bits van del 0 al 7 i no existeixen els bits 8 i 9; per tant, el bit següent a 0.7 és 1.0, i el següent a 3.7 és 4.0.



En l'espai de l'"Aula", trobareu les carpetes amb els arxius necessaris per instal·lar els programaris MicroWin i el simulador S7-200.

Taula 9. Mapa memòria Autòmat S7-200

Operand	Codi	Paràmetres
Entrada	I	I 0.0 a I 1.5
Sortida	Q	Q 0.0 a Q 1.1
Registre (marca)	M	M 0.0 a M 31.7
Temporitzador	T	T 0 a T 255
Comptador	C	C 0 a C 255

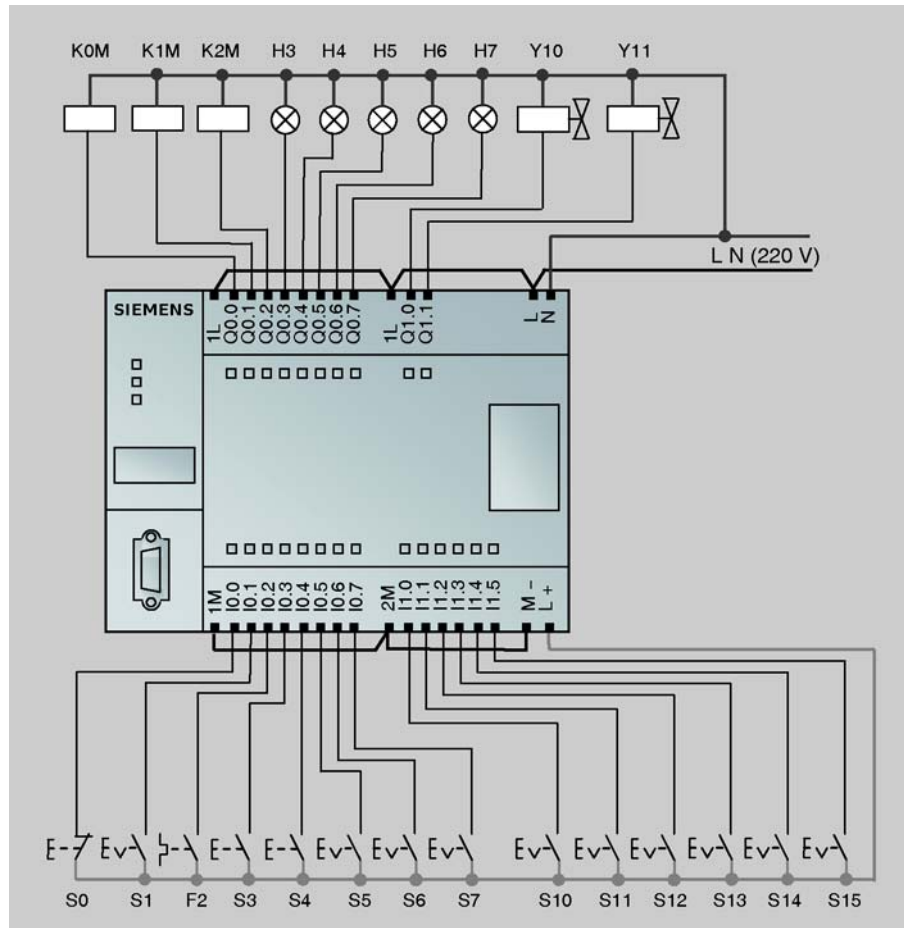
### 2.1.3. Exemple de programació amb MicroWin i comprovació del programa amb el simulador S7-200

Per poder introduir un esquema d'automatismes en l'autòmat programable, és a dir, per escriure un programa per a un PLC, heu de conèixer en primer lloc en quins borns de connexió del PLC estan connectades les entrades i les sortides de l'esquema; en aquest exemple estan representades en la figura 15, i podeu veure, per exemple, que l'interruptor S1 està connectat al born corresponent a l'entrada I0.1 i el pilot H3 al born de la sortida Q0.3.

#### Paràmetres dels operands

Per simplificar els exemples, encara que en la realitat en la indústria no coincideix pràcticament mai, fem coincidir el número de l'element amb el de l'entrada o sortida: S1 connectat a l'entrada I0.1 i K3M a la sortida Q0.3.

Figura 15. Connexions d'entrades i sortides al PLC



#### Connexió d'entrades i sortides al PLC

Fixeu-vos que en les sortides Q0.0 a Q0.2 hi ha connectades bobines de contactor, en la Q0.3 a la Q0.7 hi ha bombetes i en Q1.0 i Q1.1 electrovàlvules.

En l'entrada I0.2 hi ha connectat el tèrmic F2, en la I0.0 i I0.3 interruptors normalment tancats i en la resta interruptors normalment oberts.

També heu de saber que l'algoritme de control, és a dir, les condicions de funcionament de la instal·lació, està representat en aquest exemple en l'esquema de lògica cablada de la figura 17.

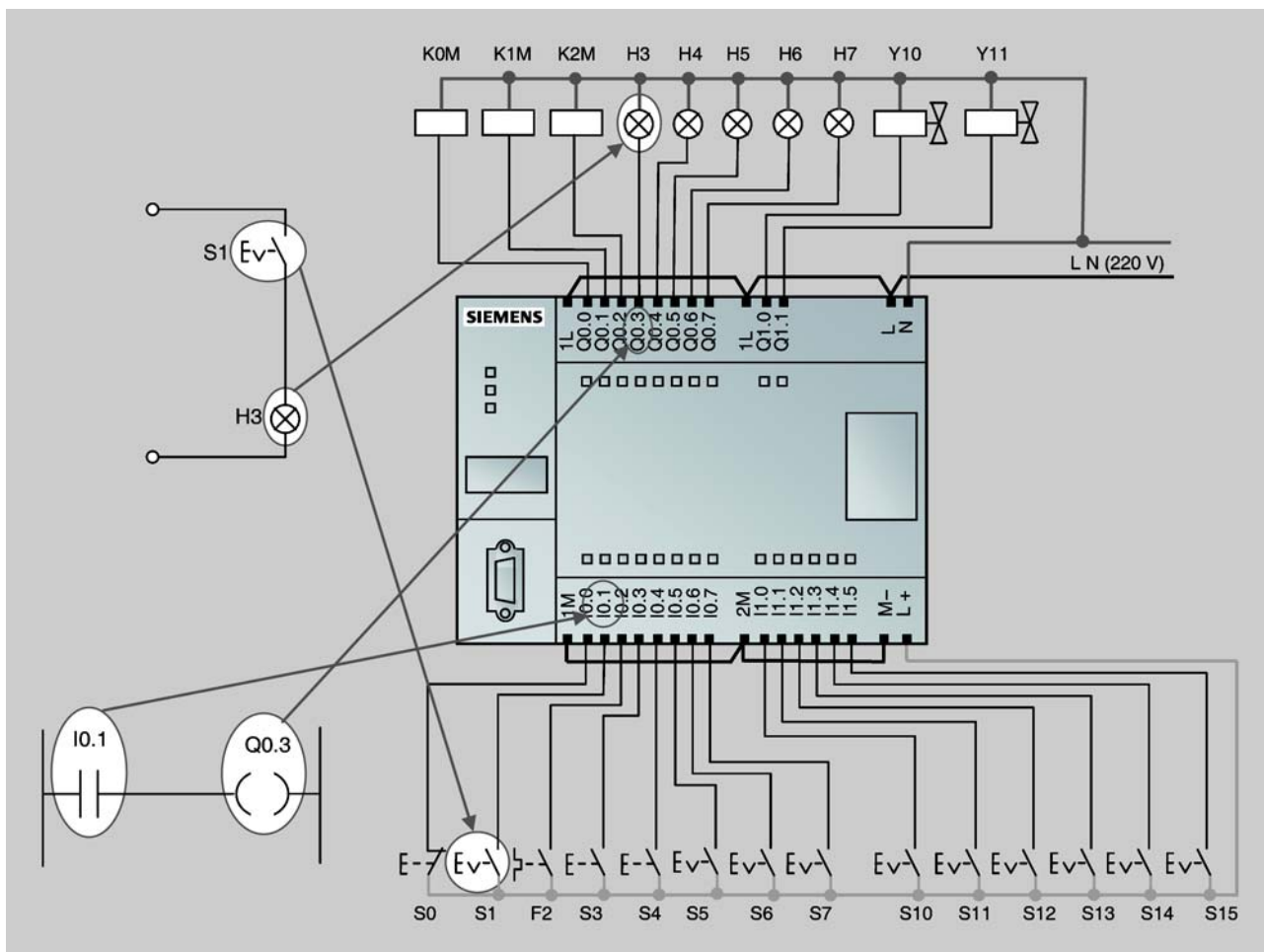
Una vegada que teniu això clar, només es tracta de convertir l'esquema simplificat segons la norma europea, amb esquema de lògica cablada, a la representació americana, amb representació KOP i esquema de contactes, i amb els paràmetres d'entrades i sortides corresponents.

En la figura 16 es mostren els passos per fer un programa per al PLC a partir de l'esquema en lògica cablejada.

**!!**

Ara feu l'activitat "Instal·lació i comprovació del funcionament dels programaris MicroWin i el simulador de PLC S7\_200 SIMU en l'ordinador personal de l'alumne", que trobareu en la secció "Activitats" del web d'aquest mòdul.

Figura 16. Conversió de l'esquema d'automatisme en programa per al PLC



Mentre no treballem connectats amb el PLC, comproveu el funcionament dels programes corresponents a les activitats i exercicis fets amb MicroWin amb el simulador S7-200.

## 2.2. Programació d'entrades i sortides

Les instruccions més senzilles que es poden programar en els autòmats programables són les corresponents a entrades i sortides, però també són

**!!**

En la secció "Recursos de contingut" del web del mòdul trobareu el programa d'aprenentatge per a la comprovació dels programes de l'autòmat S7-200 de Siemens amb els programaris MicroWin i S7\_200 SIMU.

les que més s'utilitzen, i per això és important entendre com es programen i com les tracta la unitat central del PLC, ja que us servirà per raonar amb fonament programes més complexos.

### 2.2.1. Activació d'una sortida mitjançant una entrada

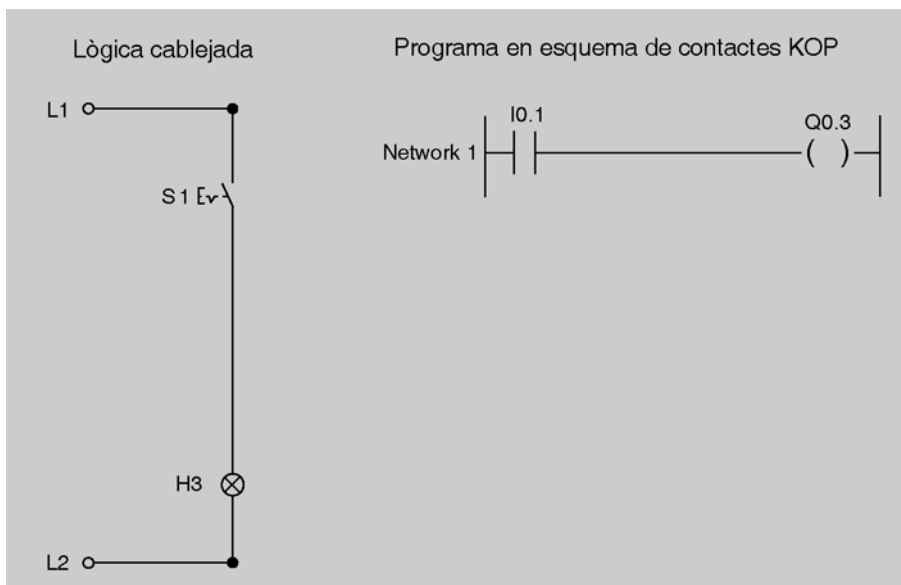
En la figura 17 podeu veure l'esquema de lògica cablejada i el programa en esquema de contactes corresponent a l'enunciat.

Si analitzeu l'esquema de lògica cablejada de la figura 17 veureu que, en tancar l'interruptor S1, arriba la línia L1 al born superior de la bombeta H3 mitjançant el fil que uneix els dos elements, i com a la part inferior de la bombeta arriba la línia L2, aquesta s'il·lumina en arribar-li les dues línies d'alimentació.

Amb el PLC no hi ha connexió elèctrica entre l'interruptor i la bombeta; és el programa que fa que en funció de l'estat de l'interruptor doni ordre de connectar o desconnectar la bombeta, i ho fa de la manera següent:

- L'autòmat llegirà el valor que tingui l'entrada I0.1 i ho assignarà a la sortida Q0.3; per tant, si l'interruptor està obert, no deixarà passar corrent i no arribarà tensió al born corresponent a l'entrada I0.1, i per això el PLC prendrà aquesta informació com un "zero" i carregarà aquest zero a la sortida Q0.3, és a dir, no subministrarà tensió al born corresponent i no funcionarà la bombeta H3.
- Quan es tanqui l'interruptor, arribarà tensió al born de l'entrada I0.1, i per això el PLC prendrà aquesta informació com un "u", que assignarà a la sortida Q0.3 i llavors s'il·luminarà la bombeta H3.

Figura 17. Activació d'una sortida mitjançant una entrada



Ara feu l'activitat "Activació d'una sortida mitjançant una entrada", que trobareu en la secció "Activitats" del web d'aquest mòdul.

## 2.2.2. Activació de diverses sortides amb diverses entrades

En la figura 18 podeu veure l'esquema de lògica cablejada i el programa en esquema de contactes corresponent a l'enunciat.

Si analitzeu l'esquema veureu que les condicions perquè funcioni, per exemple la bombeta H3, són diferents de les que necessita qualsevol altra sortida, és a dir, que siguin circuits independents, encara que estiguin representats en el mateix esquema. Cadascun d'aquests circuits independents normalment s'anomenen *branques*; per tant, en aquest esquema teniu cinc branques.

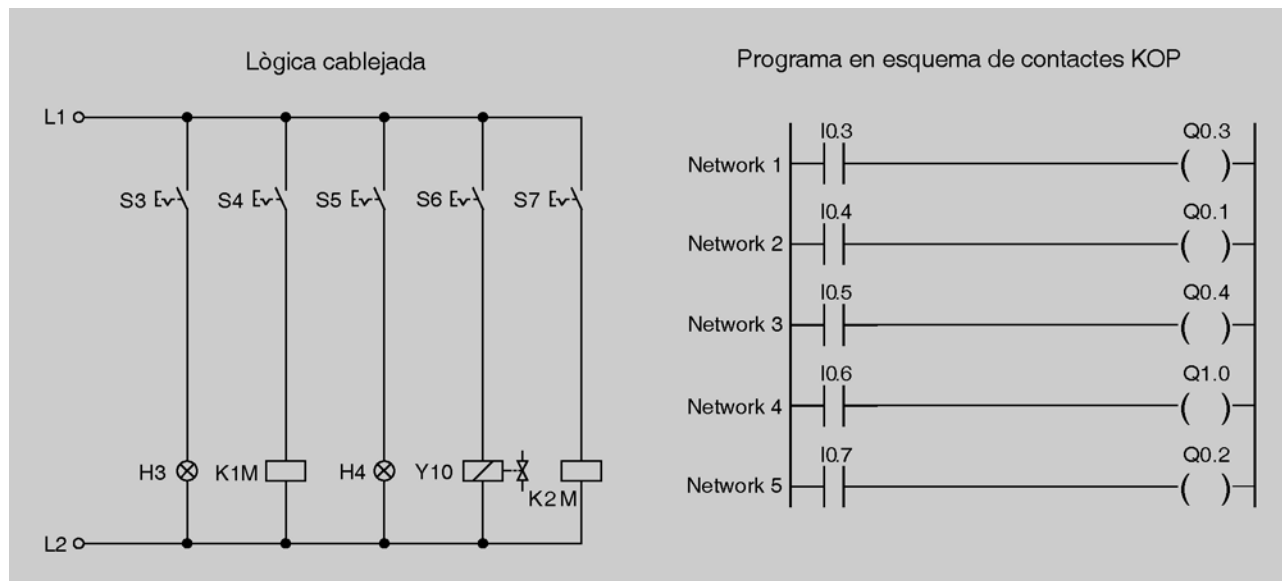
En el programari de Siemens s'ha de programar cada branca en un *network*, en un segment diferent; si no ho feu així, veureu que a l'hora de transferir el programa al PLC, quan aquest es compili perquè el PLC el pugui executar, us donarà un error de compilació i no es podrà transferir.

Hi ha autòmats que permeten programar totes les branques juntes, és a dir, que no divideixen el programa en segments.



En la secció "Recursos de contingut" del web del mòdul trobareu una animació en què podeu veure com es pot editar un programa.

Figura 18. Connexió de diverses entrades amb diverses sortides



## 2.2.3. Programació de contactes en sèrie i en paral·lel

En la figura 19 podeu veure l'esquema de lògica cablada i el programa en esquema de contactes corresponent a l'enunciat, i podeu comprovar la diferència entre la lògica cablada i la programada. En la cablada es veu que l'interruptor S3 té dues funcions diferents i, per tant, necessita dos contactes, és a dir, serà un interruptor de doble càmera, fet que encareix el producte. En la lògica programada l'interruptor S3 només té



Ara feu l'activitat "Connexió de diverses entrades amb diverses sortides", que trobareu en la secció "Activitats" del web d'aquest mòdul.

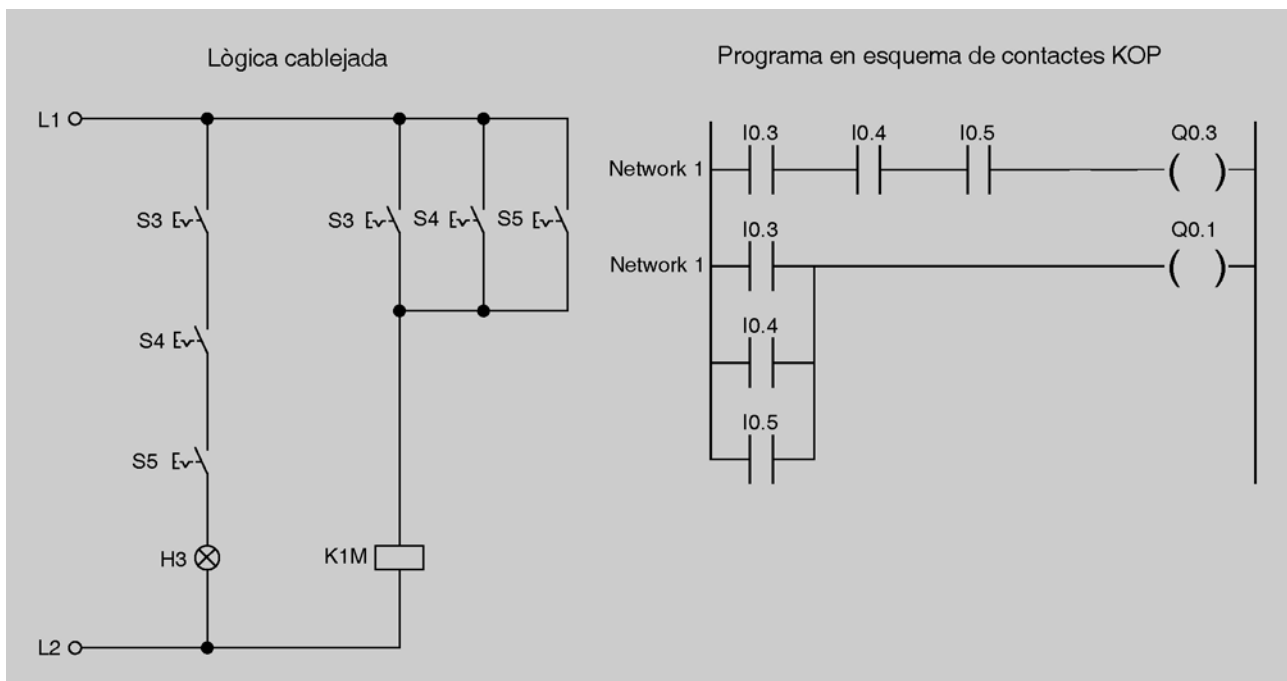
un contacte, que està connectat a l'entrada I0.3, encara que després es programi dues vegades.

En un programa es poden programar, repetir, els contactes de les entrades tantes vegades com sigui necessari, encara que només hi hagi una entrada física. El PLC, en llegir el programa, consulta l'estat de l'entrada cada vegada que la trobi programada i multiplica o suma el valor d'aquesta amb el resultat anterior emmagatzemat en la memòria interna en funció de l'operació en sèrie o paral·lel que tingui l'entrada.



En la secció "Recursos de contingut" del web del mòdul trobareu una animació en què podeu veure com es poden programar contactes connectats en sèrie o en paral·lel.

Figura 19. Programació de contactes en sèrie i en paral·lel



#### 2.2.4. Programació de contactes en connexió mixta sense parèntesis

En la figura 20 podeu veure l'esquema de lògica cablada i el programa en esquema de contactes corresponent a l'enunciat.

Un **programa sense parèntesis** és aquell en què, cada vegada que programem un contacte en sèrie o en paral·lel, està en sèrie o en paral·lel amb tots els contactes programats abans en la mateixa branca.

Vegeu la figura 20 i comproveu com l'entrada I1.4 està en sèrie amb I1.0, i la I1.1 està en paral·lel amb les dues anteriors, i la I1.5 en sèrie amb tot l'anterior, i així successivament. Això vol dir que, encara que l'esquema és llarg, el programa és senzill, ja que l'expressió algebraica corresponent no conté parèntesis:  $I1.0 \times I1.4 + I1.1 \times I1.5 + I1.2 \times I1.7 + I1.3 = Q0.3$

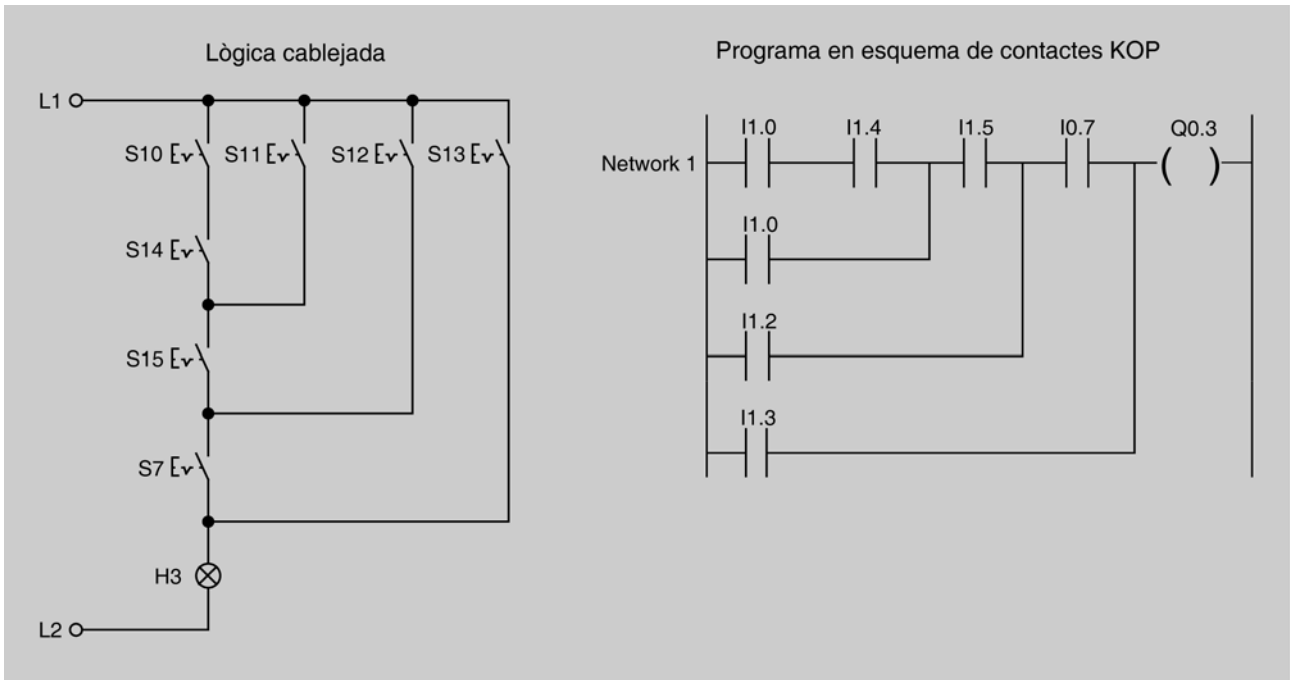


Ara feu l'activitat "Programació de contactes en sèrie i en paral·lel", que trobareu en la secció "Activitats" del web d'aquest mòdul.



En la secció "Recursos de contingut" del web del mòdul trobareu una animació en què podeu veure com es poden programar contactes en connexió mixta (connectats en sèrie i en paral·lel entre ells).

Figura 20. Programació de contactes en connexió mixta sense parèntesis



### 2.2.5. Programació de contactes en connexió mixta amb parèntesis

En la figura 21 podeu veure l'esquema de lògica cablejada i el programa en esquema de contactes corresponent a l'enunciat.

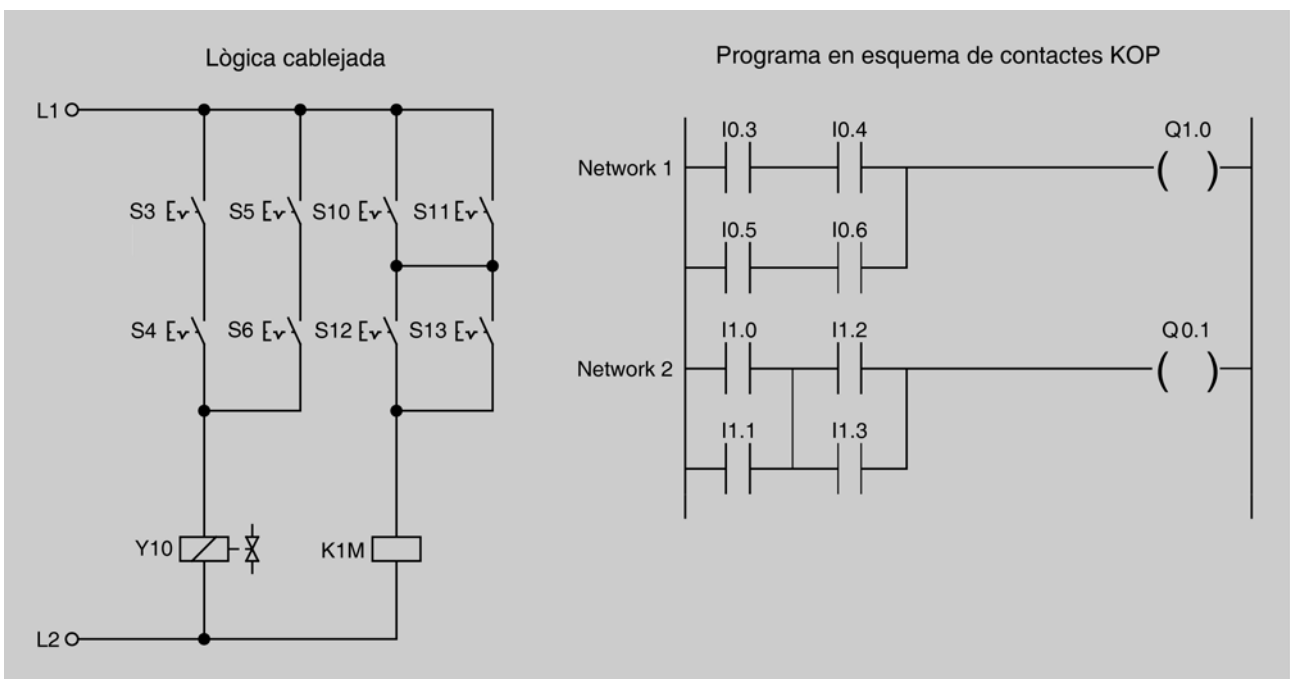
**!!**

Ara feu l'activitat "Programació de contactes en connexió mixta (sense parèntesis)", que trobareu en la secció "Activitats" del web d'aquest mòdul.

Funcionament de la branca corresponent a l'electrovàlvula Y10:

- L'interruptor S3 està en sèrie amb l'S4.
- L'interruptor S5 està en sèrie amb l'S6.
- El conjunt de S3 i S4 està en paral·lel amb el conjunt format per S5 i S6.

Figura 21. Programació de contactes en connexió mixta amb parèntesis



Això correspon a l'expressió algebraica  $(S3 \times S4) + (S5 \times S6) = Y10$ , que, com podeu veure, conté parèntesis.

De la mateixa manera a la segona branca li correspon l'expressió algebraica  $(S10 + S11) \times (S12 + S13) = K1M$ .



Ara feu l'activitat "Programació de contactes en connexió mixta (amb parèntesis)", que trobareu en la secció "Activitats" del web d'aquest mòdul.

### 2.2.6. Programació de contactes oberts i tancats de sortides

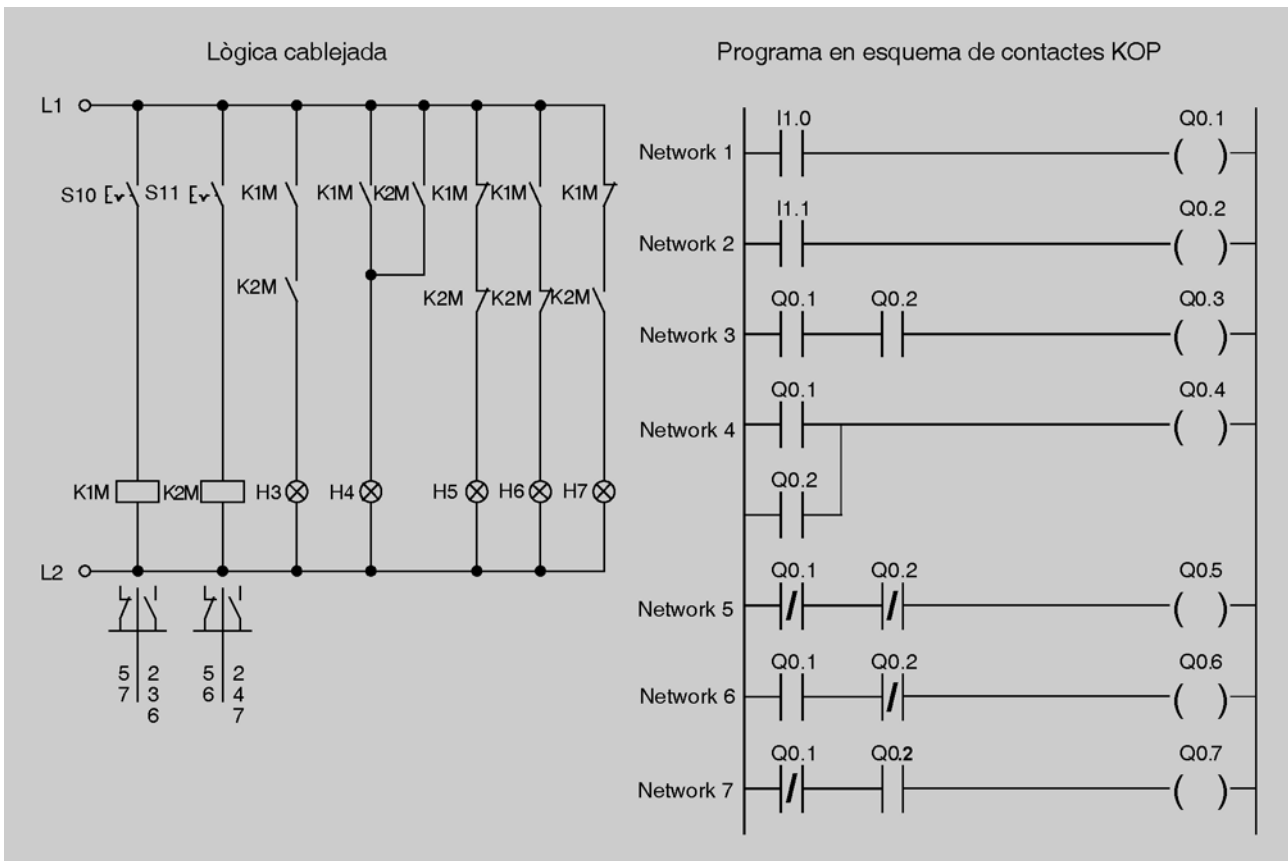
En la figura 22 podeu veure l'esquema de lògica cablada i el programa en esquema de contactes corresponent a l'enunciat.

Fixeu-vos que en aquest exemple hi ha contactes que no corresponen a entrades, és a dir, contactes que no podeu accionar manualment com feu amb les entrades, sinó que es tanquen i obren automàticament quan funciona la bobina corresponent. Aquests contactes, que corresponen a la bobina del preactuador identificats amb el símbol  $KxM$ , es programen en el PLC amb l'operand de la sortida corresponent, i són uns contactes interns, o millor dit, uns registres interns que tenen valor 1 quan s'activa la sortida. Per tant, els contactes oberts valen 0 quan la sortida està a 0, i 1 quan aquesta sortida s'activa. Per contra, els contactes negats valen 1 quan la sortida està a 0, i 0 quan aquesta sortida s'activa. Com que són contactes interns, es poden programar tants contactes de sortides com se'n necessitin; l'únic límit és la capacitat de la memòria del PLC.



Ara feu l'activitat "Programació de contactes oberts i tancats de sortides", que trobareu en la secció "Activitats" del web d'aquest mòdul.

Figura 22. Programació de contactes oberts i tancats de sortides



### 2.3. Programació de biestables

Els **biestables** són elements de memòria que amb un impuls d'entrada conserven indefinidament qualsevol dels dos estats de sortida 1 o 0. S'anomenen *bàscules*, *flip-flop*, *forrellat*, *multivibrador biestable*, i n'hi ha de dos tipus:

- **Asíncrons.** No necessiten cap senyal de rellotge per activar la sortida.
- **Síncrons.** Els senyals d'entrada han de coincidir amb un senyal de rellotge perquè les sortides siguin connectades. Aquest senyal de rellotge pot ser:
  - Amb flanc de pujada: val 1 quan el senyal del rellotge passa de 0 a 1.
  - Amb flanc de baixada: val 1 quan el senyal del rellotge passa d'1 a 0.
  - Que valgui 1: val 1 mentre el senyal del rellotge està a 1.

S'anomena **senyal de rellotge** l'entrada que ha de coincidir en el temps amb els senyals que activen els biestables; no són, per tant, senyals proporcionats per cap rellotge.

Hi ha diferents tipus de biestables depenent del valor que pren la sortida en funció de l'estat de les entrades.

- **Biestable R-S.** Amb un impuls a l'entrada S i ordre de connectar *set*, connecta la sortida, i amb un altre impuls a l'entrada R i ordre de desconnectar *reset*, la desconnecta. En el cas d'enviar senyal a la vegada a les entrades S i R, la sortida es posa a 0 sigui quin sigui el seu valor anterior. Aquests biestables són els que es poden programar en la majoria d'autòmats programables, però amb la diferència que en el cas d'enviar senyal a la vegada a les instruccions *set* i *reset*, no posa a 0 la sortida, sinó que predomina la que estigui programada en últim lloc.

Les instruccions de *set/reset* s'utilitzen molt en circuits d'automatismes i realitzen la mateixa funció que un circuit d'aturada i marxa, amb l'avantatge que el circuit *set/reset* es programa en dues branques independents i el d'aturada i marxa ho fa en una de sola, per la qual cosa no es poden separar les condicions d'aturada de les de marxa. La instrucció *set* correspon a l'ordre de marxa, és a dir, amb un impuls a la bobina *set* és suficient perquè aquesta es connecti, i resta connectada fins que arribi un altre impuls a la bobina del *reset*, que correspon a l'ordre d'aturada, i es fa, com podeu veure, la mateixa funció que els polsadors de marxa i aturada.

- **Biestable J-K.** Amb un impuls a l'entrada J i ordre de connectar, connecta la sortida, i amb un altre impuls a l'entrada K i ordre de descon-

nectar, la desconnecta. En el cas d'enviar senyal a la vegada, en les entrades J i K la sortida inverteix el valor que tenia anteriorment.

- **Biastable T.** És un biastable J-K en el qual s'han unit les dues entrades; així doncs, sempre tenim el mateix valor en les entrades J i K i cada vegada que arriba senyal a les entrades, canvia el valor de la sortida.
- **Biastable D.** És un biastable síncron, i cada vegada que arriba un sincronisme, és a dir, cada vegada que arriba senyal a l'entrada del rellotge, posa a la sortida el mateix valor que hi ha a l'entrada D.

### 2.3.1. Circuit amb polsadors d'aturada i marxa

En la figura 23 podeu veure l'esquema de lògica cablada i el programa en esquema de contactes corresponent a l'enunciat.

Podeu observar que els polsadors d'aturada S3 de l'esquema de lògica cablada són tancats, i en el programa en esquema de contactes KOP estan programats oberts; això és degut al fet que en l'entrada I0.3 del PLC hi ha connectat un polsador normalment tancat i, per tant, en repòs deixa passar corrent i l'entrada I0.3 té el valor 1.

Les entrades, que tenen el contacte normalment tancat, es programen obertes quan en l'esquema de lògica cablejada estan tancats, i negades quan en l'esquema de lògica cablejada estan oberts.

Si mireu l'esquema de lògica cablada i n'analitzeu el funcionament, veureu el següent:

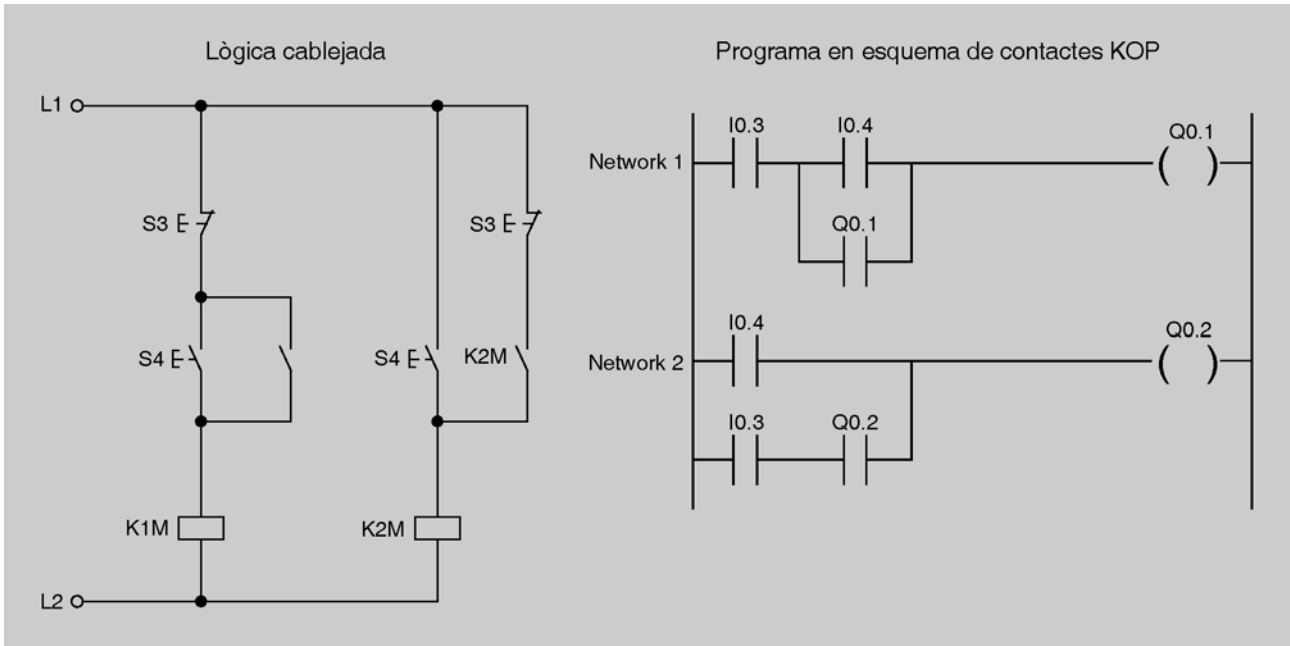
- En accionar el polsador de marxa S4, es posen en funcionament les bobines K1M i K2M i en deixar el polsador continuen funcionant, i s'autoalimenten, doncs, pels contactes oberts de les bobines.
- Quan les bobines estan funcionant i accioneu el polsador d'aturada S3, deixen de funcionar.

Aparentment el funcionament de les dues bobines és igual, però si accioneu a la vegada els polsadors de marxa i aturada, podeu comprovar com només funcionarà la bobina K2M, i això vol dir que, en el circuit de K1M predomina l'ordre d'aturada i en el de K2M predomina l'ordre de marxa.



Ara feu l'activitat "Circuit amb polsadors d'aturada i marxa", que trobareu en la secció "Activitats" del web d'aquest mòdul.

Figura 23. Circuit amb pulsadors d'aturada i marxa



### 2.3.2. Programació de sortides amb instruccions de *set-reset*

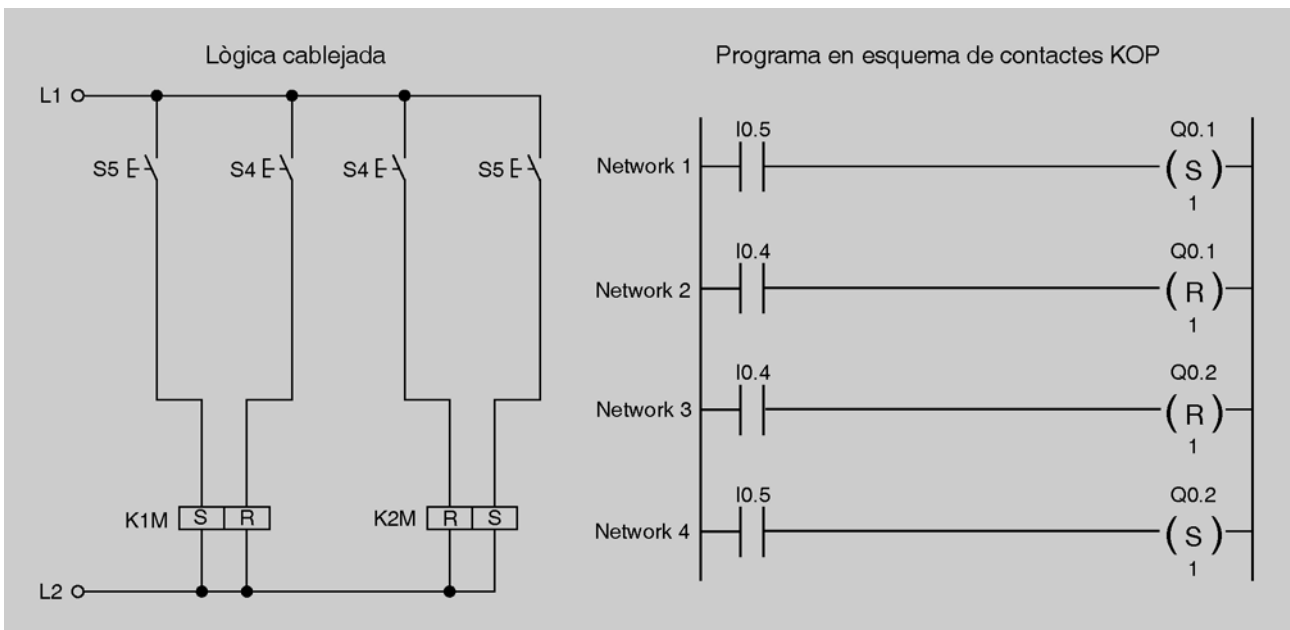
En la figura 24 podeu veure l'esquema de lògica cablada i el programa en esquema de contactes corresponent a l'enunciat.

Una sortida accionada mitjançant les instruccions *set-reset* es connecta quan li arriba tensió, és a dir, un 1, a la bobina *set*, i continua connectada encara que es posi a 0. Per desconnectar-la hem de donar tensió, és a dir, un 1, a la bobina *reset*, i heu de tenir en compte que si arriba a la vegada un 1 a les bobines *set* i *reset*, predomina la que està programada en últim lloc.

**!!**

Ara feu l'activitat "Programació de sortides amb instruccions de *set-reset*", que trobareu en la secció "Activitats" del web d'aquest mòdul.

Figura 24. Programació de sortides amb instruccions de *set-reset*



### 2.3.3. Programació d'una mateixa sortida diverses vegades

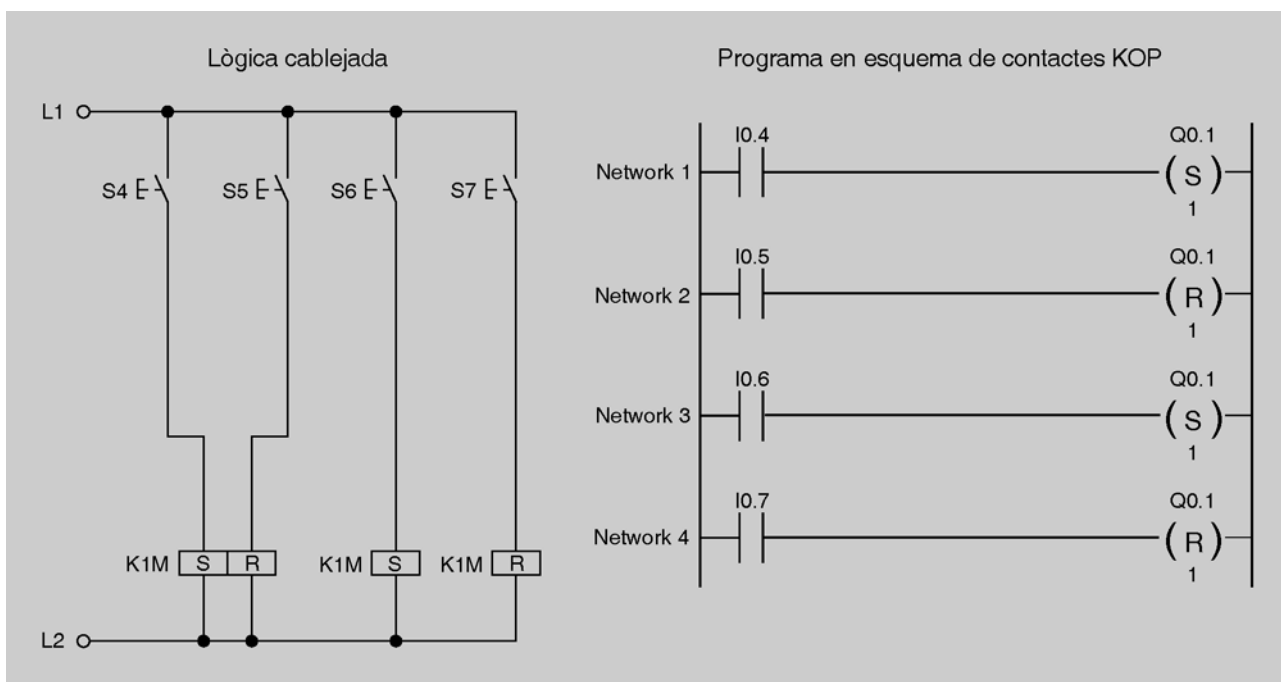
En la figura 25 podeu veure l'esquema de lògica cablejada i el programa en esquema de contactes corresponent a l'enunciat.

Si programeu una sortida diverses vegades, heu de tenir en compte que físicament només existeix una sortida i, per tant, totes les diferents ordres de connexió i de desconexió actuen sobre aquesta sortida real; si programeu una mateixa sortida diverses vegades mitjançant les instruccions *set-reset*, aquesta es connectarà cada vegada que arribi un 1 a qualsevol de les bobines *set*, i es desconnectarà cada vegada que arribi un 1 a qualsevol de les bobines *reset*. Si arriba un 1 a la vegada a una bobina *set* i a una *reset*, predominarà la programada en últim lloc.



Ara feu l'activitat "Programació d'una mateixa sortida diverses vegades", que trobareu en la secció "Activitats" del web d'aquest mòdul.

Figura 25. Programació d'una mateixa sortida diverses vegades



### 2.3.4. Funcionament del registre de sortides

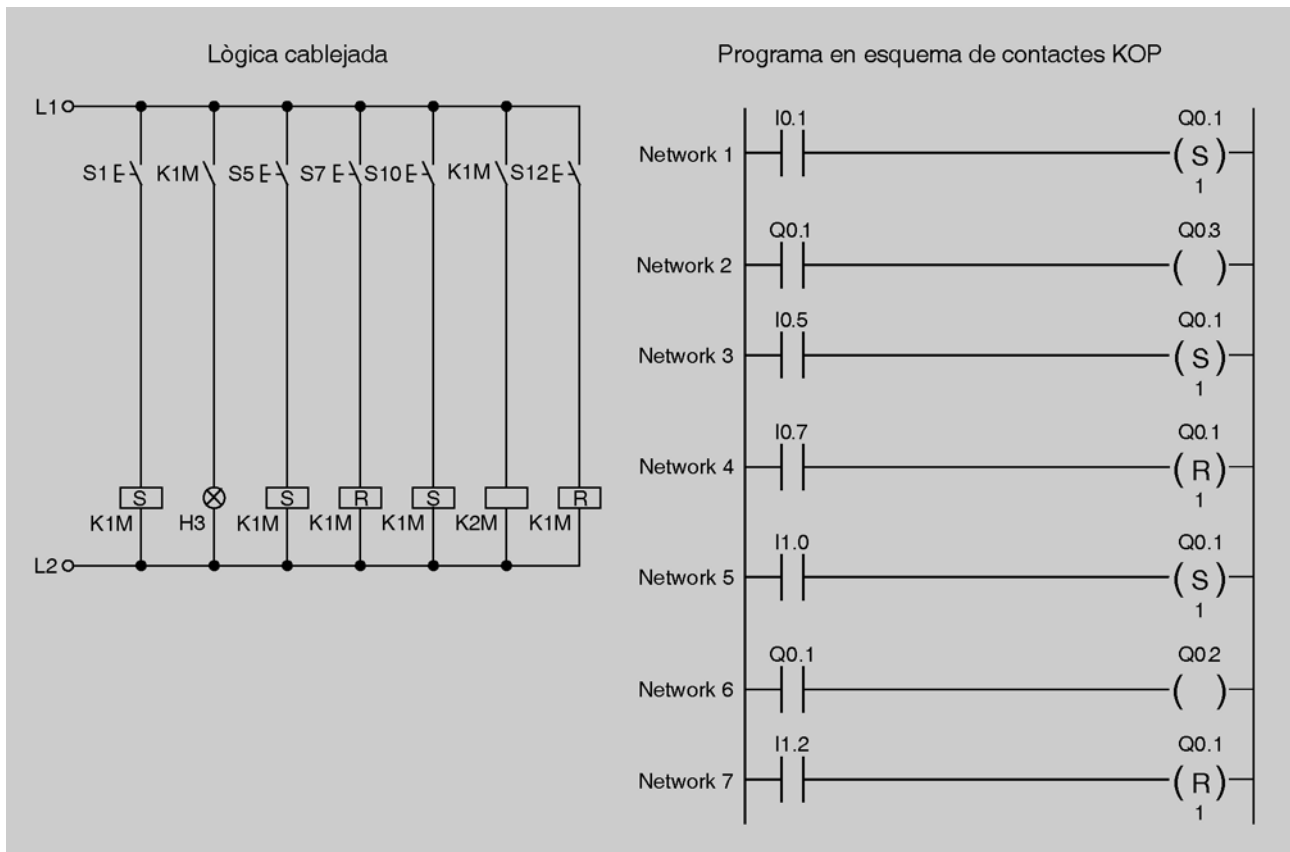
En la figura 26 podeu veure l'esquema de lògica cablejada i el programa en esquema de contactes corresponent a l'enunciat.

Quan programeu diverses vegades una mateixa sortida, el registre d'aquesta sortida pren el valor que correspon a la instrucció programada i, per tant, pot tenir diferents valors en diferents llocs del programa, encara que el valor que transfereix a l'actuador connectat a la sortida serà el que tingui el registre al final del programa, que correspondrà a la darrera instrucció programada.



Ara feu l'activitat "Funcionament del registre de sortides", que trobareu en la secció "Activitats" del web d'aquest mòdul.

Figura 26. Funcionament del registre de sortides



## 2.4. Flancs i modificador *not*

Els autòmats programables fonamenten la seva programació en la conversió de l'esquema de contactes corresponent al circuit d'automatismes en una llista d'instruccions, però poden utilitzar moltes instruccions basades en l'electrònica digital que no corresponen a cap circuit elèctric, i això fa que augmenti la potència dels programes que es poden fer amb els autòmats programables. Entre d'altres instruccions, les més bàsiques són els **flancs positiu i negatiu**, i el **modificador *not***.

### 2.4.1. Programació de flancs positiu i negatiu

Un **flanc** és un registre intern de l'autòmat que converteix un senyal permanent en un pols que dura un *scan*. En la figura 27 podeu veure l'esquema de lògica cablejada i el programa en esquema de contactes corresponent a l'enunciat.

Els flancs positius també es coneixen com a *ascendents* i els negatius com a *descendents*.

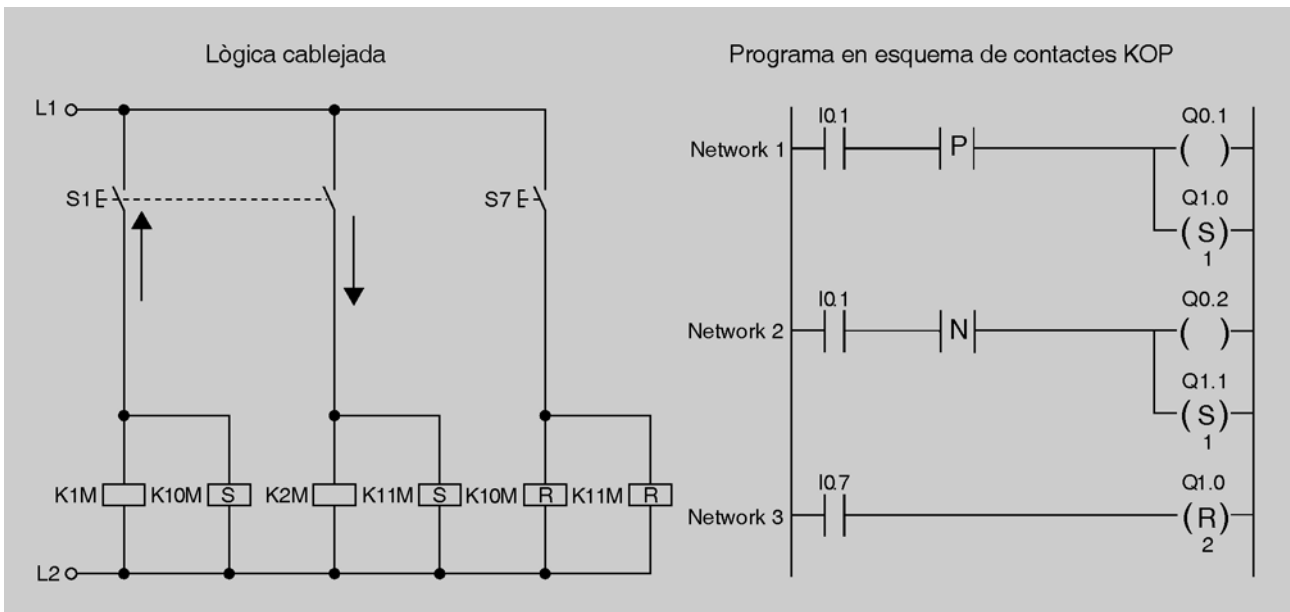
El funcionament és el següent: en accionar l'entrada I0.1 es produeix el flanc positiu i arriba senyal a les sortides Q0.1 i Q1.0; la primera només funcionarà durant un cicle, que és el temps que dura el flanc, i la segona es manté activada, ja que està programada amb la instrucció *set*. De la mateixa manera, en deixar l'entrada I0.1 es produeix el flanc negatiu i arriba

senyal a les sortides Q0.2 i Q1.1; la primera només funcionarà durant un cicle, que és el temps que dura el flanc, i la segona es manté activada, ja que està programada amb la instrucció *set*. Amb l'entrada I0.7 es fa el *reset* de les sortides Q1.0 i Q1.1.

Els **flancs positius i negatiu** donen senyal només durant un cicle en activar o desactivar l'entrada corresponent.

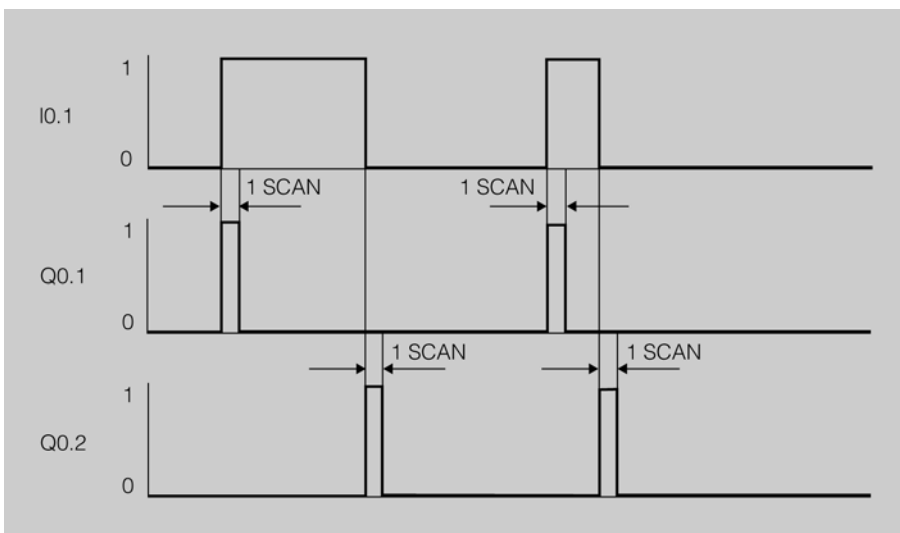
En la figura 28 es mostra el diagrama de funcionament de les entrades amb flancs, i podeu veure com, en el moment que l'entrada passa de l'estat 0 a 1, que és quan s'executa el flanc positiu, es connecta la sortida Q0.1, però ho fa només durant un cicle, de la mateixa manera en el moment que l'entrada passa de 1 a 0, que és quan s'executa el flanc negatiu, es connecta la sortida Q0.2 també només durant un cicle.

Figura 27. Programació de flancs positius i negatius



Observeu que fem el *reset* de les sortides Q1.0 i Q1.1 amb només una instrucció posant 2 a la instrucció *reset*.

Figura 28. Funcionament dels flancs



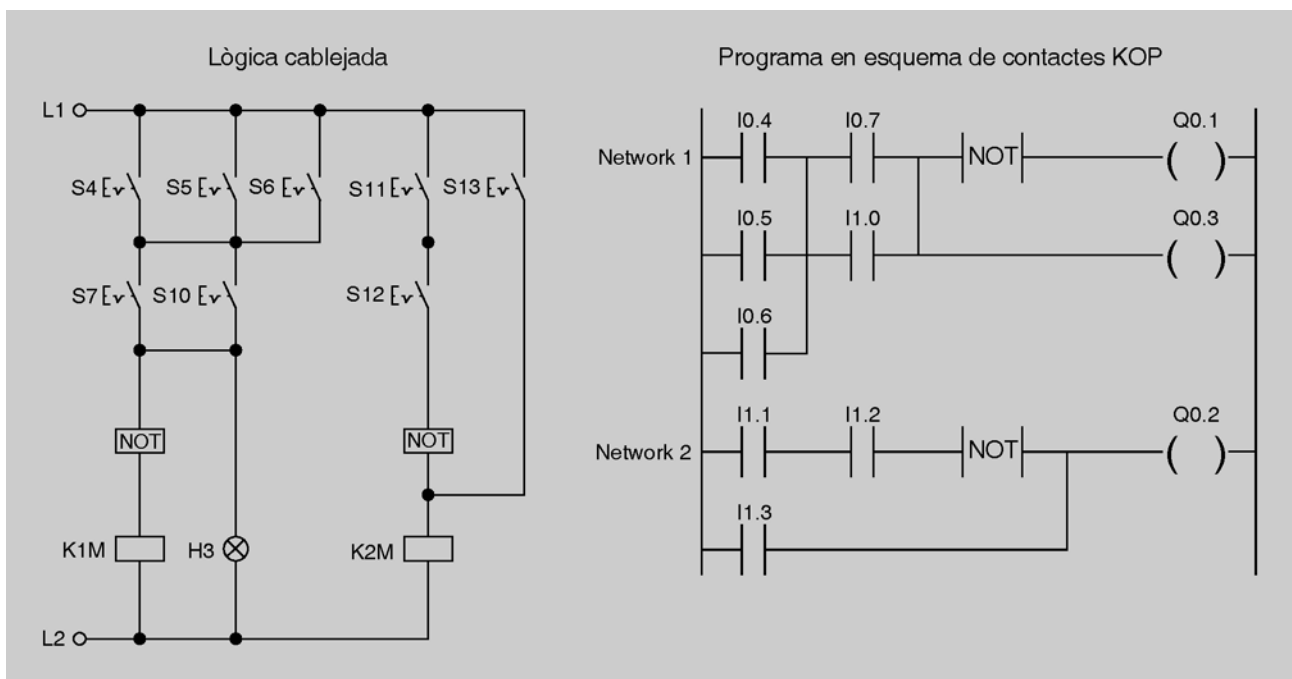
**!!**

Ara feu l'activitat "Funcionament de flancs positius i negatius", que trobareu en la secció "Activitats" del web d'aquest mòdul.

### 2.4.2. Programació del modificador *not*

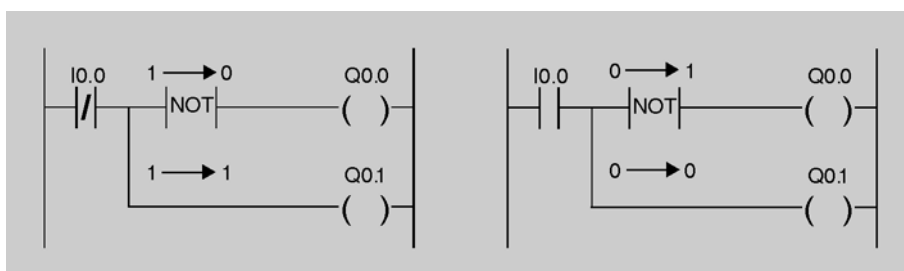
El **modificador *not*** és un registre intern de l'autòmat que inverteix el valor que li arriba; per tant, si el resultat de la combinació de contactes programats abans del modificador és 1, en la instrucció següent del *not* hi haurà 0, és a dir, si en l'entrada tenim 1 a la sortida tindrem 0, i a l'inrevés, si a l'entrada tenim 0 en la sortida tindríem 1. En la figura 29 podeu veure l'esquema de lògica cablejada i el programa en esquema de contactes corresponent a l'enunciat. Hi podeu veure que en el *network 1*, el modificador *not* no afecta el funcionament de la sortida Q0.3, i que en el *network 2* el modificador *not* no inverteix el valor de l'entrada I1.3.

Figura 29. Programació del modificador *not*



En la figura 30 es mostra l'esquema de funcionament del modificador *not*; hi podeu veure com, quan l'entrada I0.0 val 1, arriba un 0 a la sortida Q0.0 connectada mitjançant el modificador *not*, i en canvi, a la sortida Q0.1, que no té el modificador *not*, li arriba el mateix valor de l'entrada, és a dir, un 1. En l'esquema de la dreta podeu veure que quan l'entrada I0.0 val 0, la sortida que funciona és la Q0.0, ja que el modificador inverteix el valor que li arriba a l'entrada.

Figura 30. Funcionament de modificador *not*



**!!**

Ara feu l'activitat "Funcionament de modificador *not*", que trobareu en la secció "Activitats" del web d'aquest mòdul.

## 2.5. Merkers

Un *merker* és un registre intern de l'autòmat programable que es posa a 1 o 0 en funció de l'expressió algebraica corresponent a la interconnexió dels contactes que l'activen. Es representen amb la lletra M i estan agrupats en bytes, és a dir, en grups de vuit bits. Així doncs, els paràmetres dels *merkers* són iguals que els de les entrades i sortides.

M12.4 representa el *merker* del byte 12 i bit 4.

El primer dígit del paràmetre d'un *merker* correspon al byte i el segon al bit.

El funcionament dels *merkers* és similar al de les sortides; l'única diferència és que la sortida dona un senyal extern i el *merker* no, però es representa igual, és a dir, per activar-lo es programa una bobina, igual que es fa amb les sortides, i es poden programar tants contactes oberts i tancats d'aquesta bobina com calguin.

Els relés que s'utilitzen en els circuits d'automatismes equivalen, quan treballem amb els autòmats programables, als registres interns, coneguts com a *merkers* o *marques*.

### 2.5.1. Programació de *merkers*

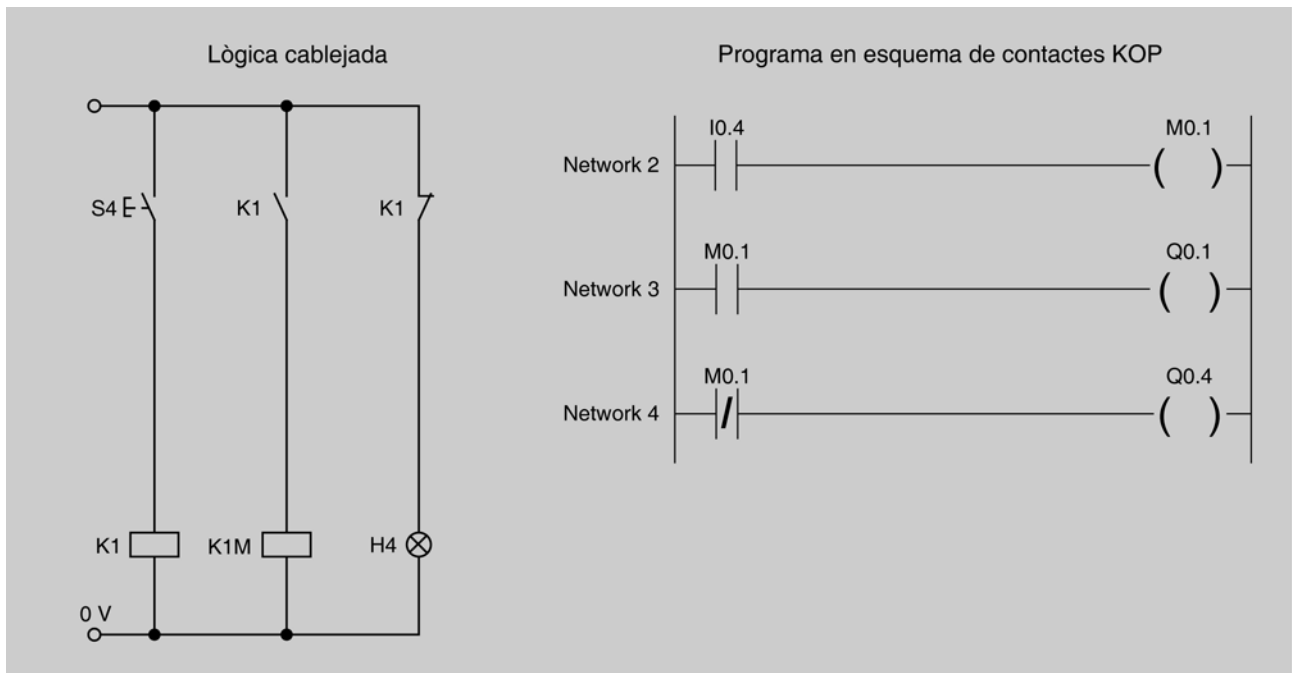
En la figura 31 podeu veure l'esquema de lògica cablejada i el programa en esquema de contactes corresponent a l'enunciat.

El funcionament és el següent: amb la instal·lació en repòs el contacte tancat del relé K0 deixa passar corrent i, per tant, la bombeta H4 funciona. Quan accioneu l'interruptor S1 es connectarà el contactor K1M i el relé K0, i en aquest moment els contactes del relé canviaran d'estat, el contacte obert es tancarà i connectarà la bombeta H3, i el contacte tancat s'obrirà i desconnectarà la bombeta H4.

El contactor K1M i el relé K0 estan connectats en paral·lel, i funcionen a la vegada; ara bé, si mireu el PLC veureu que el contactor està connectat a la sortida Q0.1 i, per tant, hi ha un LED que s'encén quan funciona la sortida, però el relé no està connectat a cap born i, per tant, externament no tenim proves del seu funcionament, ja que és un registre intern, encara que podeu veure com, mitjançant contactes d'aquest registre, podeu accionar elements externs, en aquest exemple les bombetes H3 i H4.



Ara feu l'activitat "Programació de *merkers*", que trobareu en la secció "Activitats" del web d'aquest mòdul.

Figura 31. Programació de *markers*

### 3. Instruccions bàsiques de programació dels autòmats programables 2

Els autòmats programables han evolucionat molt des que van aparèixer: en un principi només s'utilitzaven com a substituïts dels elements electro-mecànics com relés, contactors, temporitzadors, etc., però en l'actualitat tenen funcions molt avançades que permeten realitzar el control d'instal·lacions automàtiques molt complexes, que abans requerien equips molt cars com els controls distribuïts. Els autòmats programables actuals incorporen funcions que permeten realitzar el programa de manera estructurada, dividint-lo en parts, programant cada part en una subrutina diferent i mitjançant instruccions de *Salt executar* en cada moment la subrutina necessària per al funcionament correcte de la instal·lació o per realitzar funcions avançades com control de la producció, control de qualitat o tasques de manteniment.

Els autòmats programables porten incorporats molts temporitzadors, comptadors, multivibradors, generadors d'impulsos, *markers* especials i moltes més funcions que simplifiquen i abarateixen el muntatge, posada en marxa i manteniment de les instal·lacions automatitzades.

#### Relés, temporitzadors i comptadors

Un autòmat programable de gamma petita, com pot ser l'S7-200 de Siemens, pot tenir incorporats un mínim de 250 temporitzadors i 250 comptadors.

#### 3.1. Temporitzadors

Els **temporitzadors** són elements semblants als relés, formats per una bobina i uns contactes que, quan els arriba senyal a la bobina, compten temps, com si fossin rellotges amb alarma, és a dir, que podeu prefixar un temps i quan aquest ha transcorregut els contactes del temporitzador canvien d'estat.

El comptatge del temps d'un temporitzador es pot fer de manera ascendent o descendent, i cada fabricant d'autòmats ho fa de manera diferent. L'autòmat S7-200 de Siemens ho fa ascendent, és a dir, si volem que conti 300 ms, ho farà de 0 a 300; en canvi els autòmats que fan el comptatge descendent ho farien de 300 a 0. Hi ha autòmats que quan arriben al valor prefixat, deixen de comptar: és el cas més freqüent. Però l'autòmat S7-200 de Siemens, quan arriba al valor prefixat, canvia l'estat dels contactes però continua comptant.

Hi ha molts tipus de temporitzadors, però bàsicament tots es basen en dos, que són:

- **Temporitzadors a la connexió.** Fan l'operació de comptar quan arriba senyal a la bobina.
- **Temporitzador a la desconexió.** Fan l'operació de comptar quan es treu el senyal de la bobina.

Els temporitzadors es poden programar en la majoria d'autòmats de dues maneres:

- Mitjançant operands: representant els símbols de la bobina amb el temps i els corresponents contactes oberts i tancats.
- Mitjançant funcions: representant una caixa amb les entrades corresponents al temps i l'ordre de marxa a la dreta, i la sortida que s'ha d'activar al final de l'operació de comptatge a l'esquerra. Aquestes entrades i sortides cada autòmat les identifica amb unes sigles diferents, però totes són molt semblants; així doncs, si coneixeu com es programen els temporitzadors de l'autòmat S7-200 de Siemens, us serà molt fàcil programar els temporitzadors d'altres autòmats.

Hi ha autòmats en què en tots els seus temporitzadors podem indicar, entre un mínim i un màxim, qualsevol temps i en diferents unitats, com per exemple: 100 ms, 3 s, 2 min i 20 s, etc. Però l'autòmat S7-200 de Siemens no ho fa així: aquest autòmat agrupa diferents temporitzadors i cada grup té una base de temps fixa. En la taula 10 s'indica la base de temps que tenen assignats els diferents temporitzadors a la connexió i desconnexió.

Els temporitzadors representats en la taula 10 es poden programar a la connexió o a la desconnexió, però quan un temporitzador s'ha programat d'una manera, ja no es pot tornar a programar de cap altra.

Taula 10. Base de temps dels temporitzadors a la connexió i desconnexió

Base de temps	Número temporitzador
1 ms	T32 i T96
10 ms	T33 a T36 i T97 a T100
100 ms	T37 a T63 i T101 a T255

Observant la taula podeu comprovar que podeu obtenir el mateix temps combinant valors prefixats i diferents bases. Així doncs, si volem un temps d'1 s podeu programar, per exemple:

- T32 amb valor 1.000:  $1.000 \times 1 \text{ ms} = 1.000 \text{ ms} = 1 \text{ s}$
- T33 amb valor 100:  $100 \times 10 \text{ ms} = 1.000 \text{ ms} = 1 \text{ s}$
- T37 amb valor 10:  $10 \times 100 \text{ ms} = 1.000 \text{ ms} = 1 \text{ s}$

El temps programat en els tres casos serà igual, però heu de tenir en compte que el temps real no ho és, ja que els temporitzadors poden tenir un error equivalent a la base de temps; així doncs, el T32 pot arribar a tenir un error d'1 ms, i el T37 de 100 ms.

### 3.1.1. Programació del temporitzador a la connexió

Els **temporitzadors a la connexió** es representen en l'autòmat programable S7-200 de Siemens amb les sigles **TON**. En la figura 32 podeu veure

l'esquema de lògica cablada i el programa en esquema de contactes corresponent a l'enunciat, i també el diagrama de temps, en què podeu veure que el temporitzador comença a comptar quan l'entrada es posa a 1, i que quan ha transcorregut el temps prefixat es connecta la sortida; també podeu veure com quan l'entrada es posa a 0, la sortida es desconnecta, i que si l'entrada està connectada menys temps del que s'ha programat la sortida no arriba a connectar-se.

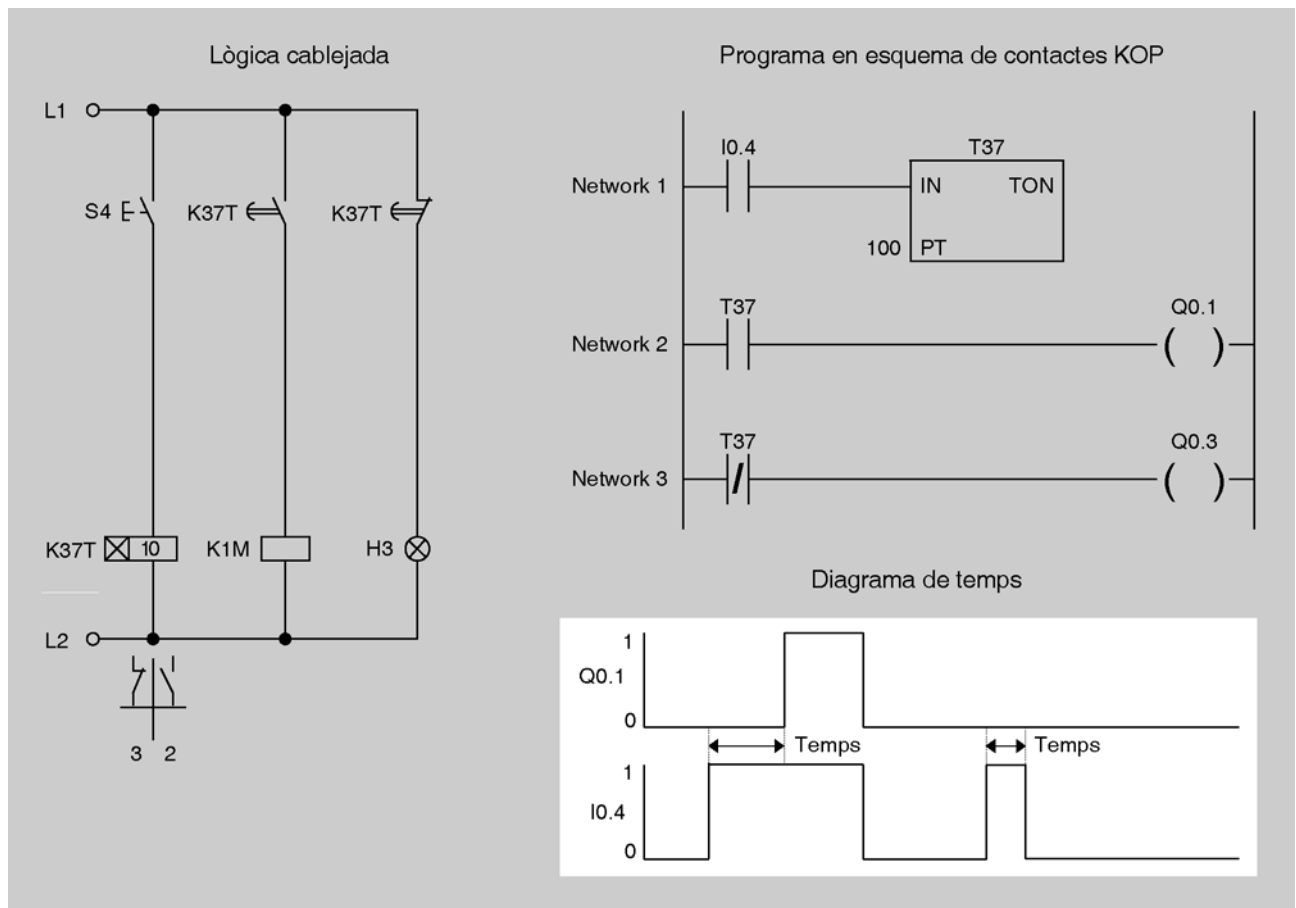
El funcionament és el següent: en repòs, està accionada la sortida Q0.3, ja que el contacte negat de T37 li dona senyal; quan accioneu l'entrada I0.4 arriba senyal a l'entrada IN, i llavors el temporitzador comença a comptar i quan arriba a 100, que és el valor programat en l'entrada PT, els contactes del temporitzador canvien d'estat, es connecta la sortida Q0.1 i es desconnecta la Q0.3, i el temporitzador continua contant fins a arribar al valor màxim, que és 32.767. Quan obrim l'entrada I0.4, és a dir, quan traiem el senyal IN, els contactes del temporitzador tornen a l'estat inicial i, per tant, la sortida Q0.1 es desconnecta i es torna a activar la Q0.3.

La base de temps de T37 és 100 ms; com que en l'entrada PT hi ha el valor 100, comptarà 100 vegades 100 ms, que és igual a 10.000 ms o 10 s.

Si l'entrada està accionada menys temps del que està programat, els contactes del temporitzador no canvien d'estat, i si torneu a activar l'entrada el temporitzador començarà a comptar de nou, i per tant el temps no s'acumula d'una ordre de connexió a una altra.

**!!**  
Ara feu l'activitat "Programació del temporitzador a la connexió", que trobareu en la secció "Activitats" del web d'aquest mòdul.

Figura 32. Programació del temporitzador a la connexió



### 3.1.2. Posada en marxa d'un contactor mitjançant pulsadors i temporitzadors a la connexió

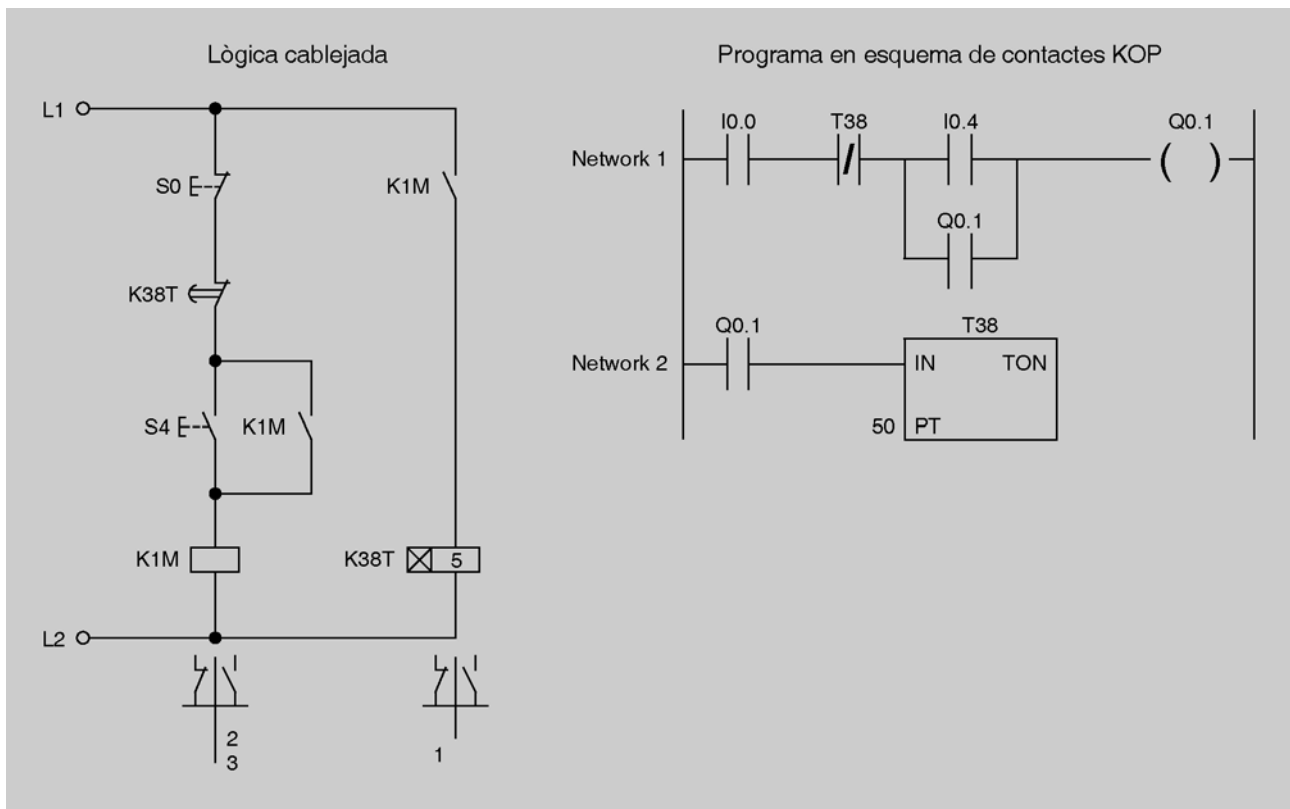
En la figura 33 podeu veure l'esquema de lògica cablada i el programa en esquema de contactes corresponent a l'enunciat.

El funcionament és el següent: la sortida Q0.1 es posa en marxa amb el pulsador I0.4 i s'atura de manera manual mitjançant el pulsador d'aturada I0.0 o automàticament en estar 5 s en funcionament mitjançant el contacte negat de T38.



Ara feu l'activitat "Posada en marxa d'un contactor mitjançant pulsadors i temporitzadors a la connexió", que trobareu en la secció "Activitats" del web d'aquest mòdul.

Figura 33. Posada en marxa d'un contactor mitjançant pulsadors i temporitzador a la connexió



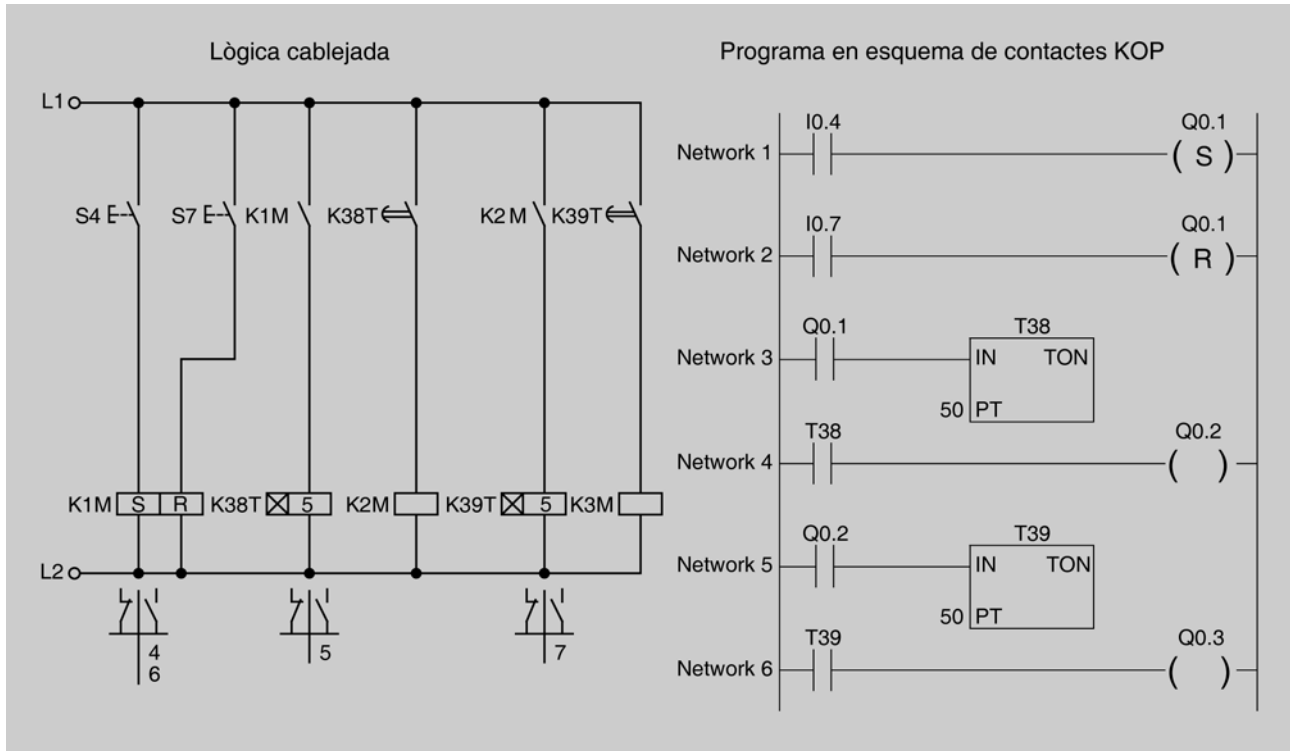
### 3.1.3. Posada en marxa progressiva de tres contactors mitjançant dos temporitzadors a la connexió

En la figura 34 podeu veure l'esquema de lògica cablejada i el programa en esquema de contactes corresponent a l'enunciat.

El funcionament és el següent: teniu tres sortides, Q0.1, Q0.2 i Q0.3; la primera es posa en marxa mitjançant instruccions *set-reset* accionant els pulsadors I0.4 per posar-la en marxa i I0.7 per aturar-la. Quan funciona Q0.1 a través del seu contacte obert programat en el *network* 3, posa en marxa el temporitzador T38, que en passar 5 s accionarà la sortida Q0.2 mitjançant el seu contacte obert programat en el *network* 4. Al mateix temps

tancarà el contacte de Q0.2 programat en el *network* 5, que posarà en marxa el temporitzador T39, que en passar 5 s i mitjançant el seu contacte obert programat en el *network* 6 accionarà la sortida Q0.3. En el moment que s'accióni el pulsador I0.7 que dona ordre de *reset* a la sortida Q0.1, s'aturaran totes les sortides que estiguin en marxa.

Figura 34. Posada en marxa progressiva de tres contactors mitjançant temporitzadors a la connexió



Ara feu l'activitat "Posada en marxa progressiva de tres contactors mitjançant temporitzadors a la connexió", que trobareu en la secció "Activitats" del web d'aquest mòdul.

### 3.1.4. Programació del temporitzador a la desconexió

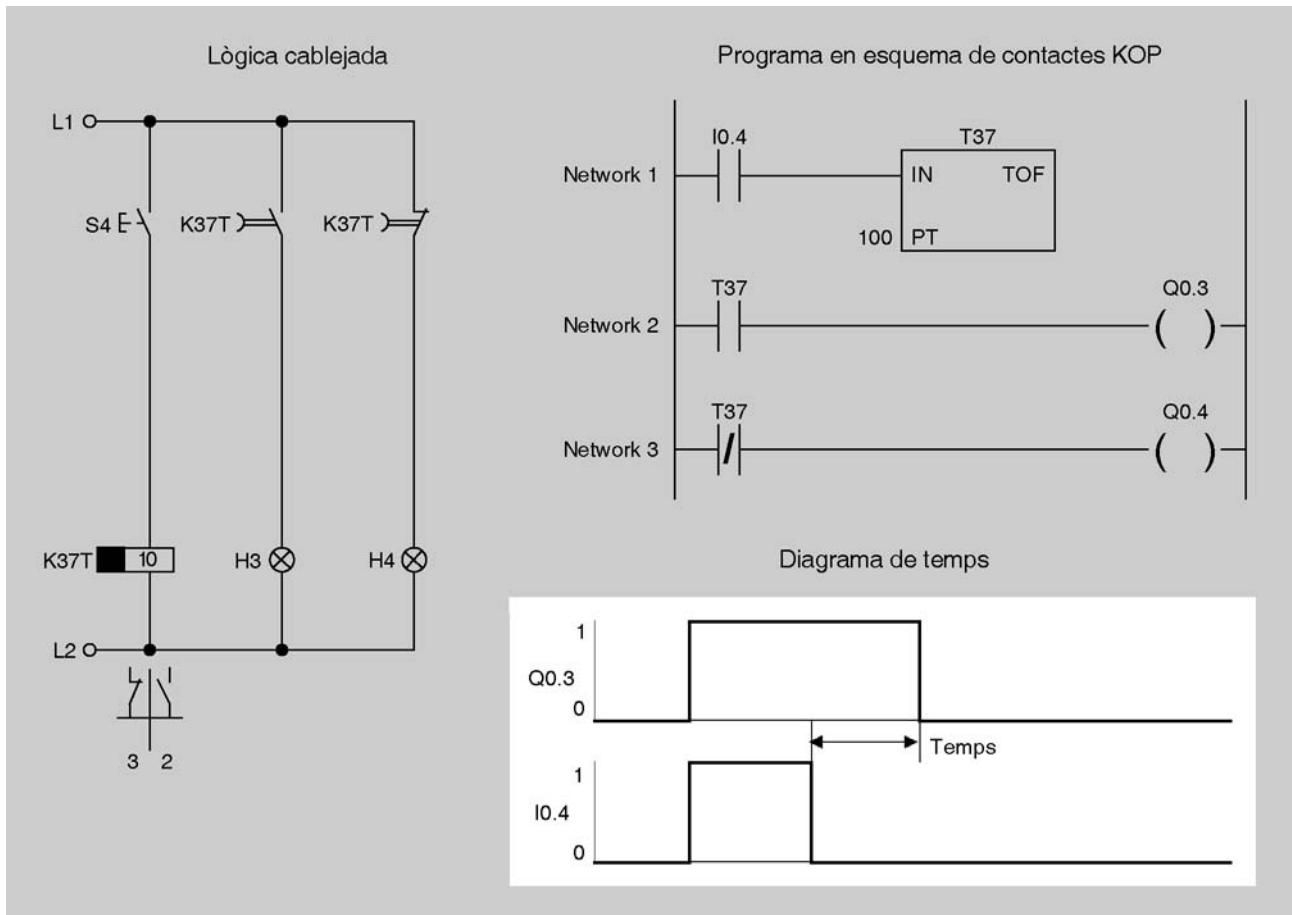
Els temporitzadors a la desconexió es representen en l'autòmat programable S7-200 de Siemens amb les sigles *TOF*. En la figura 35 podeu veure l'esquema de lògica cablada i el programa en esquema de contactes corresponent a l'enunciat, i també el diagrama de temps, en què podeu veure que la sortida es connecta al mateix moment que es connecta l'entrada, és a dir, quan l'entrada es posa a 1; quan aquesta es posa a 0, la sortida continua funcionant, i és en aquest moment quan el temporitzador comença a comptar i una vegada que ha transcorregut el temps prefixat es desconnecta la sortida.

El funcionament és el següent: en repòs, està accionada la sortida Q0.4, ja que el contacte negat de T37 li dona senyal; quan accioneu l'entrada I0.4 li arriba senyal a l'entrada IN, i en aquest moment canvien d'estat els contactes del temporitzador, i accionen la sortida Q0.3 i desconnecten la Q0.4, però el temporitzador no fa l'operació de comptar. Quan desconnecteu l'entrada I0.4 el temporitzador comença a comptar i quan arriba al valor prefixat, en aquest cas 10 s, els contactes tornen a la posició inicial, activen la sortida Q0.4 i desactiven la Q0.3.



Ara feu l'activitat "Programació del temporitzador a la desconexió", que trobareu en la secció "Activitats" del web d'aquest mòdul.

Figura 35. Programació del temporitzador a la desconexió



### 3.1.5. Programació del temporitzador a la desconexió accionat mitjançant flanc positiu

En la figura 36 podeu veure l'esquema de lògica cablada i el programa en esquema de contactes corresponent a l'enunciat.

El funcionament és el següent: cada vegada que accioneu l'entrada I0.4 i mitjançant la instrucció del flanc positiu, arriba senyal durant un cicle a l'entrada IN del temporitzador T38; per tant, com que li arriba senyal, canvia l'estat dels seus contactes, es connecta la sortida Q0.5 i es desconnecta la Q0.7, però com que al cicle següent ja no arriba senyal, el temporitzador comença a comptar i 5 s després ( $50 \times 100 \text{ ms} = 5.000 \text{ ms}$ ) els contactes tornen a la posició inicial, connecten de nou la sortida Q0.7 i desconnecten la Q0.5.

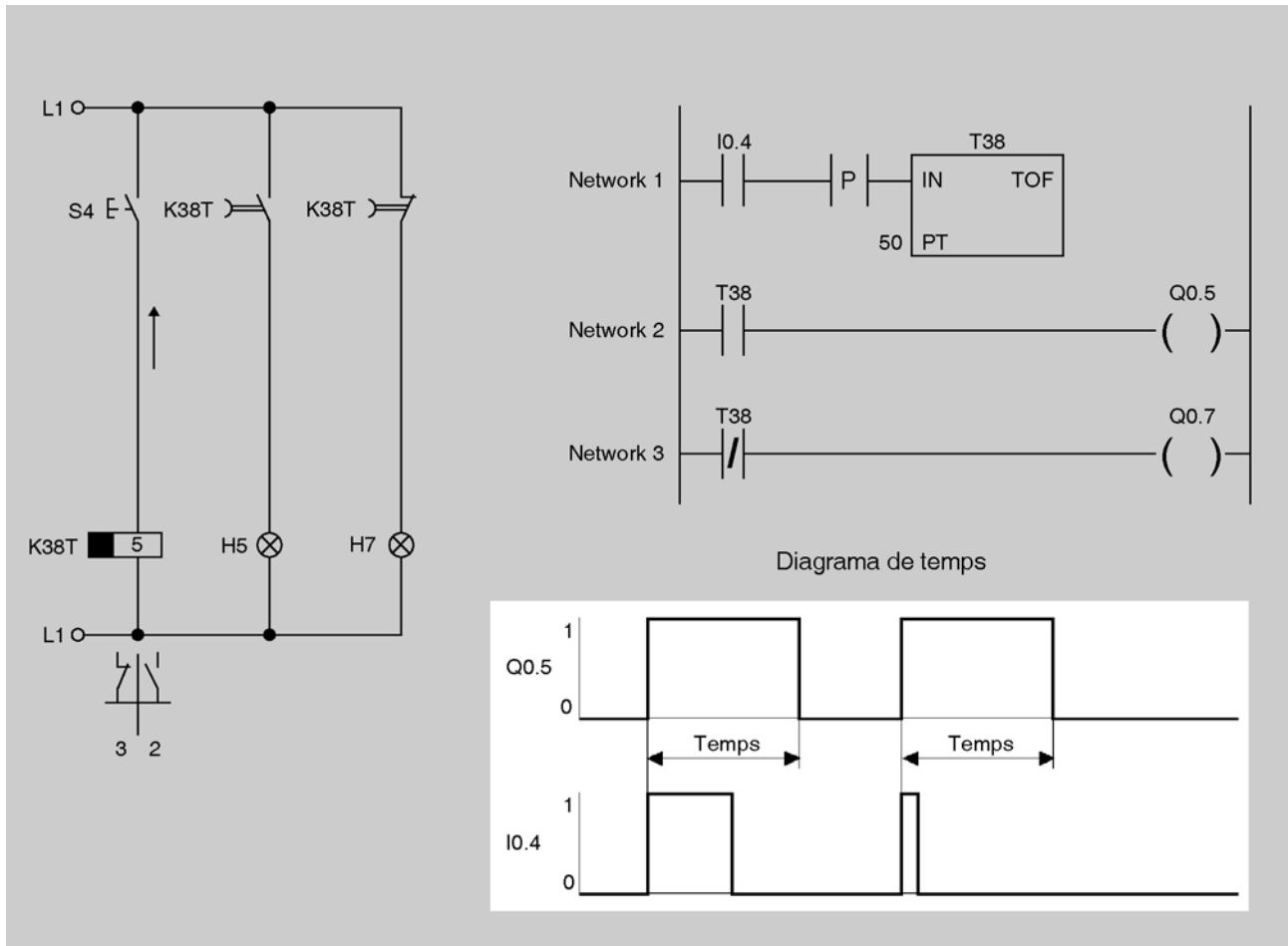
Si mentre el temporitzador està comptant torneu a accionar l'entrada I0.4, el temporitzador es posa a zero, però els seus contactes continuen en el mateix estat, per tant continua connectada la sortida Q0.3 i, un cicle després d'haver tornat a accionar l'entrada I0.4, el temporitzador inicia el comptatge de nou.

Els flancs positius converteixen el senyal elèctric permanent que li arriba en un senyal instantani, que dura un cicle d'execució del PLC.



Ara feu l'activitat "Programació del temporitzador a la desconexió accionat mitjançant flanc positiu", que trobareu en la secció "Activitats" del web d'aquest mòdul.

Figura 36. Programació del temporitzador a la desconexió accionat mitjançant flanc positiu



### 3.1.6. Programació del temporitzador acumulatiu

En la taula 11 s'indica la base de temps que tenen assignats els temporitzadors acumulatius amb numeracions diferents de les dels temporitzadors a la connexió i desconexió.

Taula 11

Base de temps	Núm. de temporitzador
1 ms	T0 i T64
10 ms	T1 a T4 i T65 a T68
100 ms	T5 a T31 i T69 a T95

Els temporitzadors acumulatius es representen en l'autòmat programable S7-200 de Siemens amb les sigles *TONR*. En la figura 37 podeu veure l'esquema de lògica cablada i el programa en esquema de contactes corresponent a l'enunciat, i també el diagrama de temps, en què podeu veure que el temporitzador incrementa el seu valor mentre mantinguem el valor 1 en l'entrada I0.4, i que quan ha transcorregut el temps prefixat es connecta la sortida. També podeu veure que, quan l'entrada I0.4 es posa a 0, la

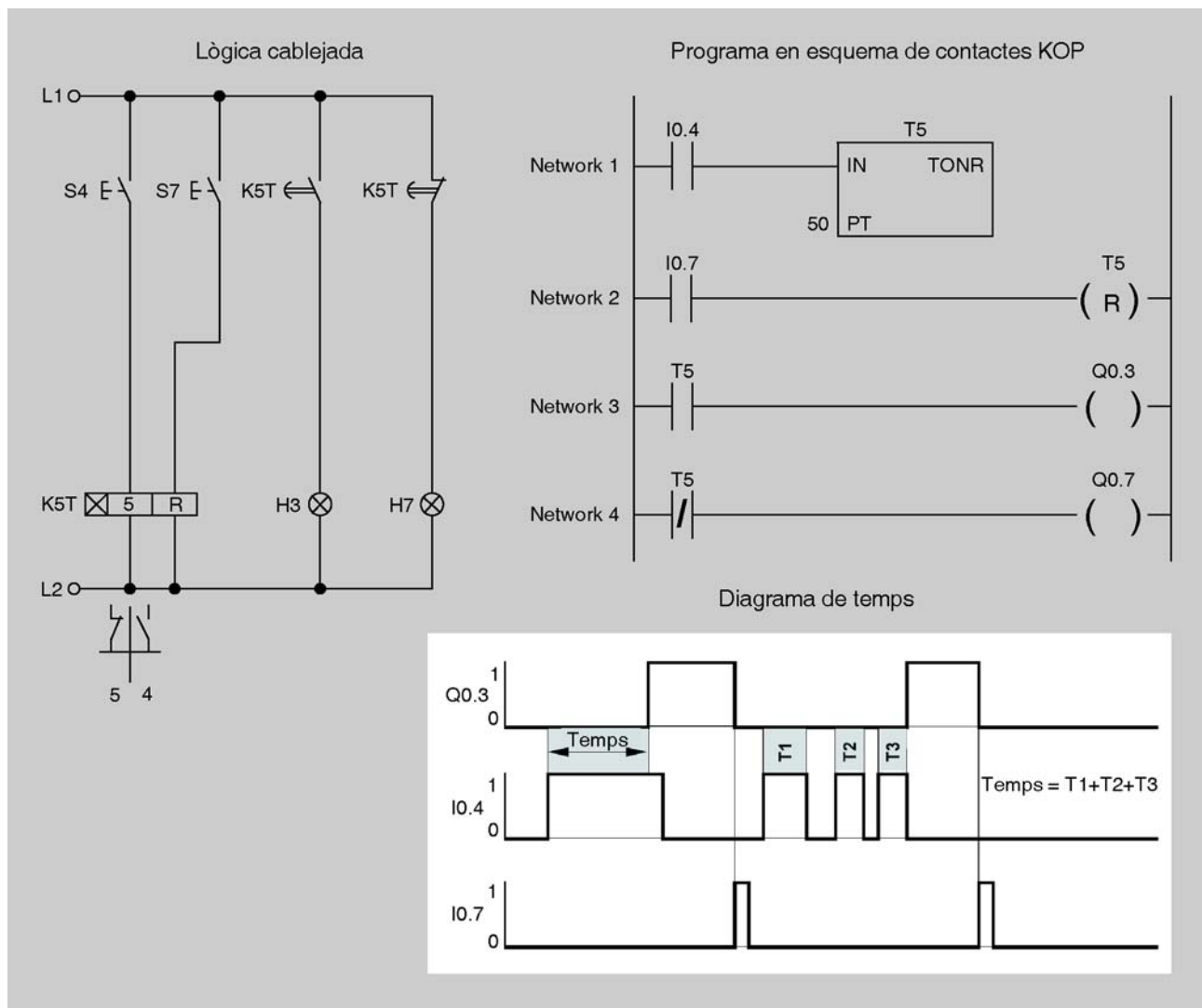
sortida no es desconnecta, i que si l'entrada està connectada menys temps del que hi ha programat, el temporitzador guarda el valor acumulat i cada vegada que es connecta l'entrada el temporitzador incrementa el seu valor a partir d'aquest valor, no des de zero. Per desconnectar la sortida s'ha de fer mitjançant una instrucció de *reset*.

El funcionament és el següent: en repòs, està accionada la sortida Q0.7, ja que el contacte negat de T5 li dona senyal; quan tanqueu l'entrada I0.4 i mentre arriba el valor 1 a l'entrada IN del temporitzador, va incrementant el seu valor. Si obriu l'entrada I0.4, per tant, posem un 0 a l'entrada IN, i el temporitzador manté el valor; si tornem a tancar l'entrada I0.4 el temporitzador incrementa el valor partint del valor anterior, no de zero. Quan el valor del temporitzador arriba al valor preseleccionat, en aquest cas 5 s, els contactes canvien d'estat, es connecta la sortida Q0.3 i es desconnecta la Q0.7. A partir d'aquest moment, si es manté el senyal en l'entrada IN del temporitzador, aquest continua comptant fins al valor màxim de 32.767, però l'estat dels seus contactes no varia. Si accioneu l'entrada I0.7, es fa el *reset* del temporitzador i els contactes tornen a la posició inicial, i tornen a connectar la sortida Q0.7 i a desconnectar la Q0.3

**!!**

Ara feu l'activitat "Programació del temporitzador acumulatiu", que trobareu en la secció "Activitats" del web d'aquest mòdul.

Figura 37. Programació del temporitzador acumulatiu



### 3.1.7. Control del temps de funcionament d'un contactor mitjançant un temporitzador acumulatiu

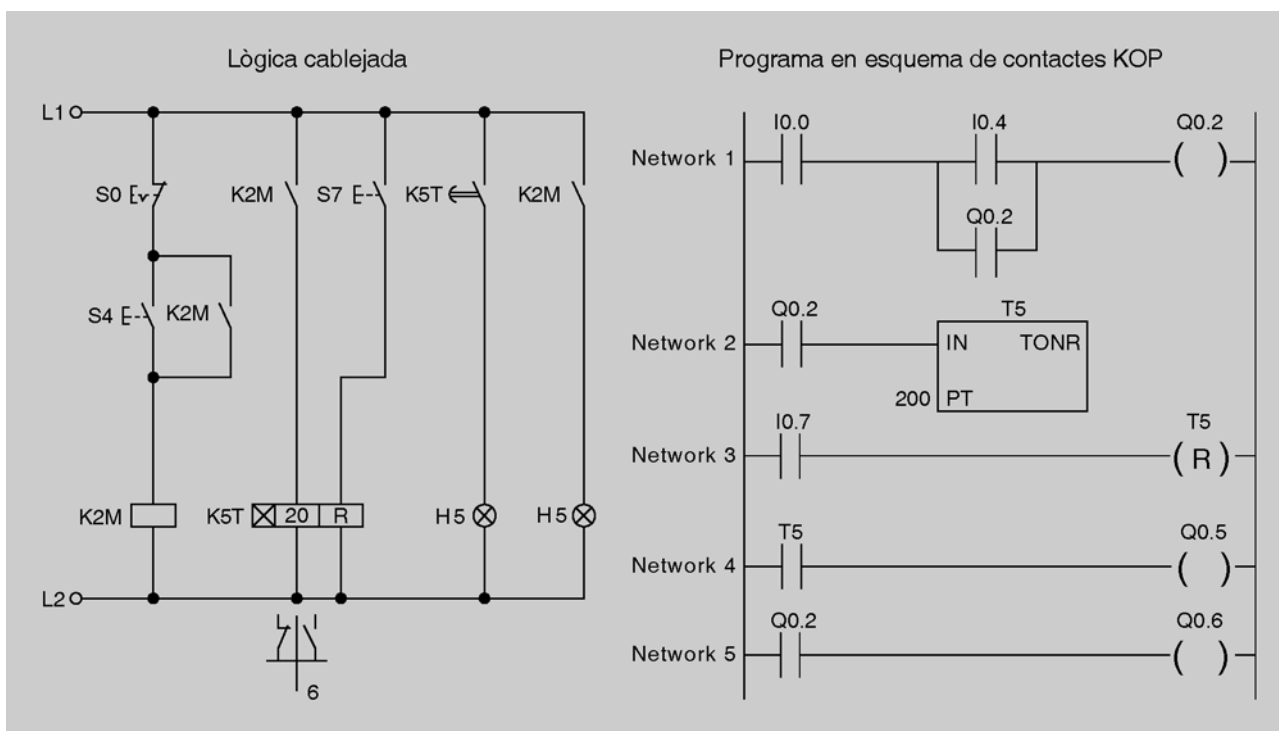
En la figura 38 podeu veure l'esquema de lògica cablejada i el programa en esquema de contactes corresponent a l'enunciat.

El funcionament és el següent: el contactor K2M connectat a la sortida Q0.2 de l'autòmat programable es posa en marxa cada vegada que accioneu el polsador de marxa S4 connectat a l'entrada I0.4 i s'atura cada vegada que accioneu el polsador d'aturada S0 connectat a l'entrada I0.0. Mentre funciona la sortida Q0.2, el seu contacte programat en el *network* 2 està tancat i dona senyal a l'entrada IN del temporitzador T5; així doncs, el temporitzador incrementa el seu valor sempre que funciona la sortida i, per tant, controla el temps de funcionament del contactor K2M. Aquesta és una funció molt interessant en tasques de manteniment, ja que mitjançant el contacte obert programat en el *network* 5 connectarà la bombeta de senyalització H5 i indicarà que ha transcorregut el temps fixat per fer l'operació de manteniment pertinent. La bombeta de senyalització H6 indica que el contactor està funcionant, i amb el polsador S7 connectat a l'entrada I0.7 es pot posar el temporitzador a zero una vegada feta la tasca de manteniment.



Ara feu l'activitat "Control del temps de funcionament d'un contactor mitjançant un temporitzador acumulatiu", que trobareu en la secció "Activitats" del web d'aquest mòdul.

Figura 38. Control del temps de funcionament d'un contactor mitjançant un temporitzador acumulatiu



## 3.2. Comptadors

Els **comptadors** són elements que compten els impulsos elèctrics que els arriben. Aquests impulsos poden ser senyals elèctrics, que per exemple,

els envia una fotocèl·lula que detecta el pas de persones per una zona determinada, o senyals d'un detector de proximitat, que indiquen que una ampolla ha estat dipositada en una caixa.

Bàsicament hi ha dos tipus de comptadors: sense preselecció i amb preselecció.

- Comptadors sense preselecció: es dediquen a comptar el nombre de senyals que els arriben, i guarden el valor en un registre intern.
- Comptadors amb preselecció: a més de comptar els impulsos i guardar el valor en un registre, permeten preseleccionar un valor concret i quan el comptador arriba a aquest valor, envien un senyal mitjançant els contactes oberts i tancats de què disposen.

Els comptadors amb preselecció poden fer el comptatge de manera ascendent o descendent.

El **comptador amb preselecció ascendent** s'inicialitza posant el comptador a zero; per fer el comptatge de 0 al valor preseleccionat, normalment es duu a terme aquesta operació mitjançant la instrucció *reset*. En canvi, el **descendent** s'inicialitza posant el comptador al valor preseleccionat, i fa el comptatge des d'aquest valor fins a zero; normalment es fa aquesta operació mitjançant la instrucció *set o load*.

La majoria de comptadors, tant si són amb preselecció o sense, poden incrementar impulsos positivament o negativament, ja que encara que la funció bàsica és comptar, ha de tenir l'opció de descomptar en el cas que es faci una operació errònia.

L'autòmat S7-200 de Siemens disposa de 256 comptadors, que s'identifiquen amb la lletra C; per tant, es poden programar des de C0 fins a C255, i tots es poden utilitzar amb o sense preselecció, poden ser progressius, regressius o progressius-regressius, i el nombre màxim d'impulsos que poden comptar és 32.767.

### 3.2.1. Comptador progressiu

El **comptador progressiu** de l'autòmat S7-200 de Siemens s'identifica amb les sigles CTU, compta impulsos endavant i, per tant, incrementa el seu valor quan li arriba un 1 a l'entrada CU; s'inicialitza, és a dir, es posa a zero, quan li arriba un 1 a l'entrada R (*reset*) i els seus contactes canvien d'estat quan el comptador ha comptat un nombre d'impulsos igual o major que el valor preseleccionat.

#### Exemple de comptadors

La pantalla dels partits de basquet disposa d'un comptador sense preselecció, que és el comptador de cistelles realitzades, i d'un comptador amb preselecció, que és el comptador de temps de cada part del partit i que quan arriba al final del temps, fa sonar una sirena.

El comptador de cistelles d'un partit de bàsquet permet descomptar una cistella que l'àrbitre havia donat com a triple i que només era de dos punts.

Si voleu que el comptador funcioni com a comptador amb preselecció, heu d'indicar el nombre d'impulsos de l'entrada PV. Si es posa 0, es comporta com un comptador sense preselecció.

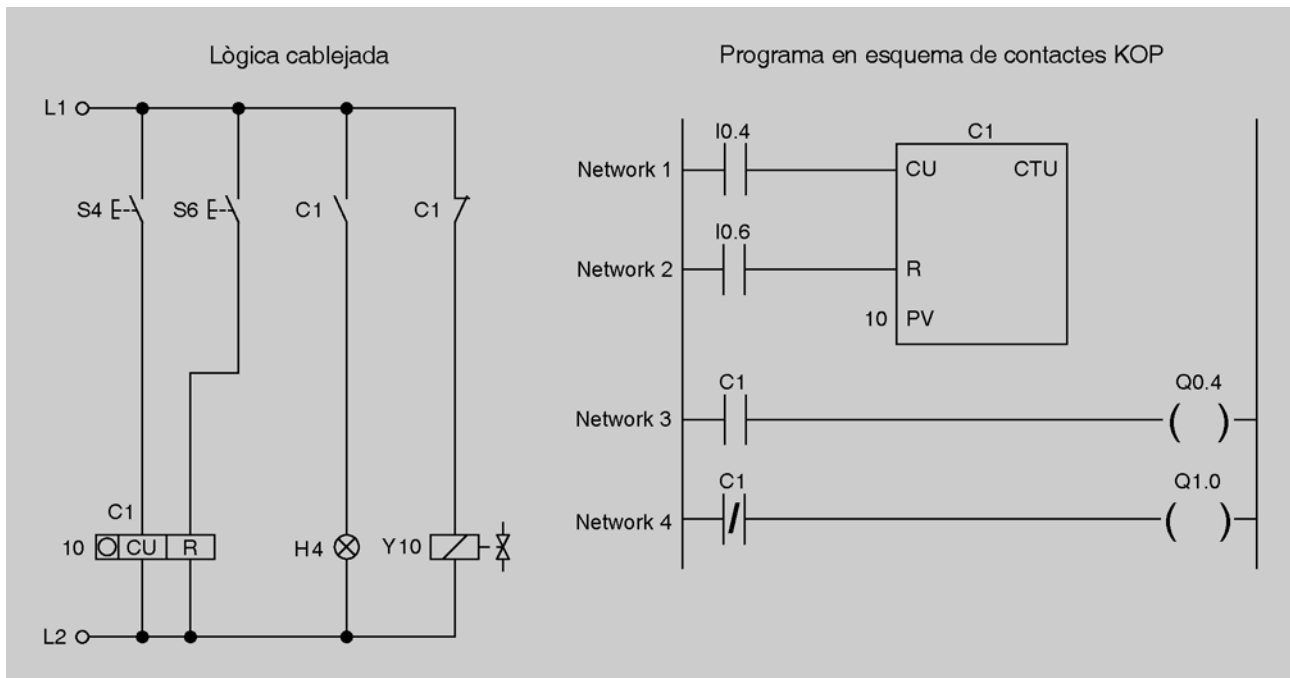
En la figura 39 podeu veure l'esquema de lògica cablejada i el programa en esquema de contactes corresponent a l'enunciat.

El funcionament és el següent: el comptador s'inicialitza fent un *reset* mitjançant l'entrada I0.6; en aquest moment funciona la sortida Q0.1, i cada vegada que s'acciona l'entrada I0.4 el comptador incrementa una unitat el seu valor i quan aquest és igual o superior a 10 (que és el valor preseleccionat), canvia l'estat dels contactes, acciona la sortida Q0.4 i desconnecta la Q1.0.



Ara feu l'activitat "Comptador progressiu", que trobareu en la secció "Activitats" del web d'aquest mòdul.

Figura 39. Comptador progressiu



### 3.2.2. Programació d'un comptador progressiu amb selecció d'inicialització manual o automàtica

En la figura 40 podeu veure l'esquema de lògica cablada i el programa en esquema de contactes corresponent a l'enunciat.

El funcionament és el següent:

- El comptador C3 compta els impulsos (incrementa el seu valor) mitjançant l'entrada I0.4.

#### Inicialització automàtica o comptador cíclic

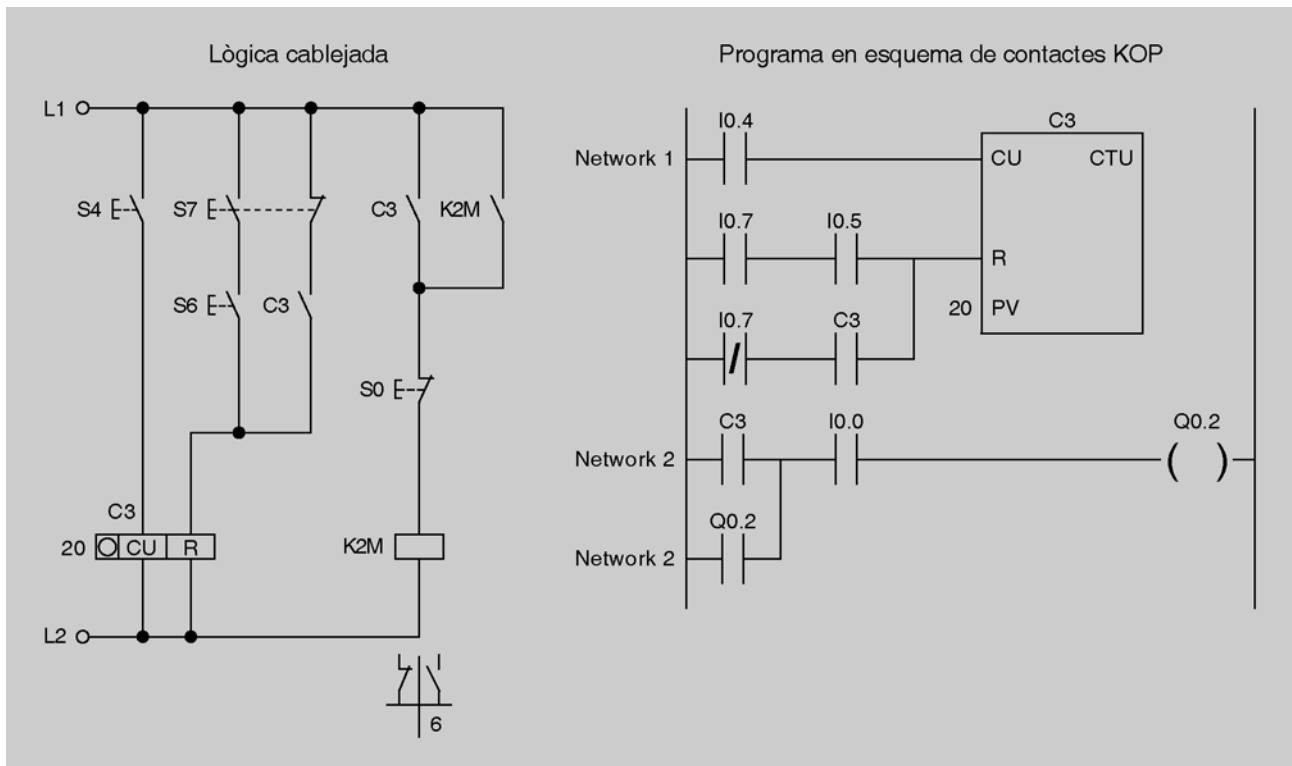
Un comptador cíclic és aquell que quan compta un nombre d'impulsos igual que el de la preselecció es torna a posar a zero i inicialitza el procés de comptatge, mitjançant un contacte del comptador mateix. En aquest cas els seus contactes canvien d'estat només durant un cicle o *scan*.

- Amb l'entrada I0.7 seleccioneu el tipus d'inicialització:
  - En repòs, comptador cíclic. Es posa a zero mitjançant el contacte del comptador C3 quan aquest arriba al valor preseleccionat, en aquest cas 20.
  - Accionada, inicialització manual. Es posa a zero quan accioneu l'entrada I0.5.
- Quan el comptador hagi comptat vint impulsos tanca el contacte C3 programat en el *network* 2 i dóna ordre de marxa a la sortida Q0.2, que es realimenta pel contacte que la sortida mateixa té connectat en paral·lel amb el contacte de C3, que només està connectat durant un cicle.
- L'entrada I0.0 actua com a polsador d'aturada de la sortida Q0.2



Ara feu l'activitat "Programació d'un comptador progressiu amb selecció d'inicialització manual o automàtica", que trobareu en la secció "Activitats" del web d'aquest mòdul.

Figura 40. Programació d'un comptador progressiu amb selecció d'inicialització manual o automàtica



### 3.2.3. Comptador regressiu

El **comptador regressiu** de l'autòmat S7-200 de Siemens s'identifica per les sigles CTD, compta impulsos endarrere i, per tant, incrementa negativament el seu valor quan li arriba un 1 a l'entrada CD; s'inicialitza, és a dir, es posa en el valor preseleccionat, quan li arriba un 1 a l'entrada LD (*load*): en aquest moment els seus contactes estan en repòs i canvien d'estat quan el comptador, després de comptar impulsos incrementant negativament el seu valor, arriba a zero.

El comptador regressiu s'atura quan arriba a zero.

En la figura 41 podeu veure l'esquema de lògica cablejada i el programa en esquema de contactes corresponent a l'enunciat.

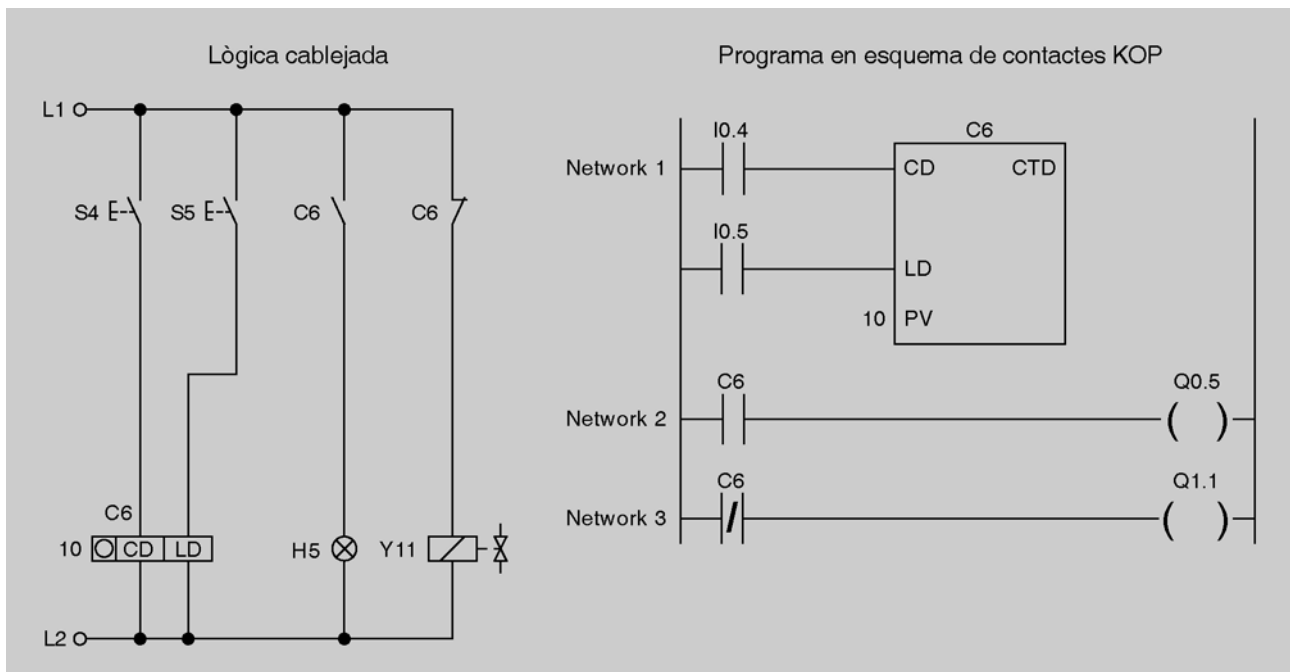
El funcionament és el següent:

- El comptador s'inicialitza, és a dir, es posa en el valor 10, quan accioneu l'entrada I0.5.
- En aquest moment està connectada la sortida Q1.1.
- Cada vegada que accioneu l'entrada I0.4 el comptador incrementa negativament una unitat el seu valor.
- Quan arriba a zero canvia l'estat dels contactes, es connecta la sortida Q0.5 i es desconnecta la Q1.1.



Ara feu l'activitat "Comptador regressiu", que trobareu en la secció "Activitats" del web d'aquest mòdul.

Figura 41. Comptador regressiu



### 3.2.4. Comptador progressiu-regressiu

El **comptador progressiu-regressiu** es coneix també com a **comptador reversible** i en l'autòmat S7-200 de Siemens s'identifica amb les sigles *CTUD*; compta impulsos endavant quan li arriba senyal a l'entrada CU i endarrere quan li arriba senyal a l'entrada CD. S'inicialitza, posant el comptador a zero, quan li arriba senyal a l'entrada R, i els seus contactes canvien d'estat quan el valor del comptador és igual o superior al valor de preselecció, que pot estar entre 32.767 i -32.768.

En la figura 42 podeu veure l'esquema de lògica cablejada i el programa en esquema de contactes corresponent a l'enunciat.

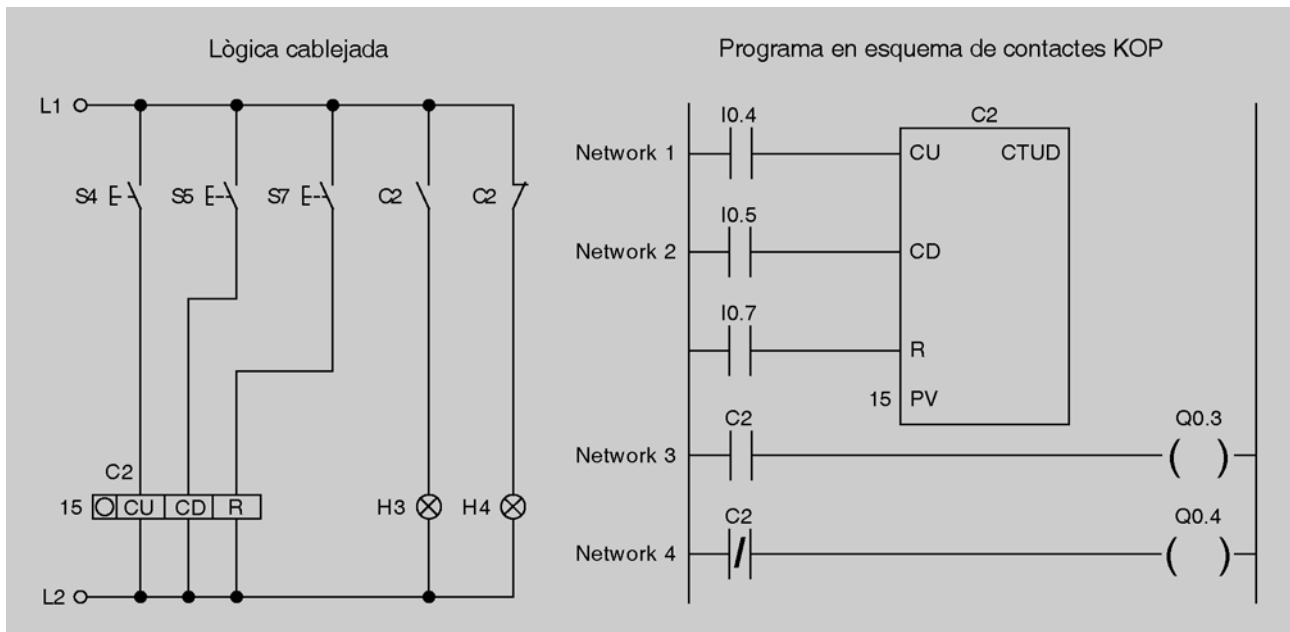
El funcionament és el següent:

- El comptador s'inicialitza, és a dir, es posa a zero quan accioneu l'entrada I0.7.
- En aquest moment està connectada la sortida Q0.4.
- Cada vegada que accioneu l'entrada I0.4 el comptador incrementa positivament una unitat el seu valor.
- Cada vegada que accioneu l'entrada I0.5 el comptador incrementa negativament una unitat el seu valor.
- Quan el valor del comptador sigui igual o superior a 15 canvia l'estat dels contactes, es connecta la sortida Q0.3 i es desconnecta la Q0.4.



Ara feu l'activitat "Comptador progressiu-regressiu", que trobareu en la secció "Activitats" del web d'aquest mòdul.

Figura 42. Comptador progressiu-regressiu



### 3.2.5. Control de fabricació de peces en un procés de producció

En la figura 43 podeu veure l'esquema de lògica cablejada i el programa en esquema de contactes corresponent a un circuit en què s'utilitzen els tres tipus de comptadors, CTU, CTD i CTUD, com a comptadors sense preselecció.

El circuit de la figura 44 formaria part d'un circuit més complex d'un procés de fabricació i s'encarregaria de controlar les peces que produeix aquest procés. El sensor S1 donaria senyal cada vegada que surt una peça acabada i el sensor S4 donaria senyal cada vegada que la peça és rebutjada pel sistema de control de qualitat.

El funcionament és el següent:

- El comptador C1 compta el total de peces fabricades, incrementa el seu valor mitjançant l'entrada I0.1, el sensor de peces totals, i es posa a zero amb el polsador de *reset* I0.7.

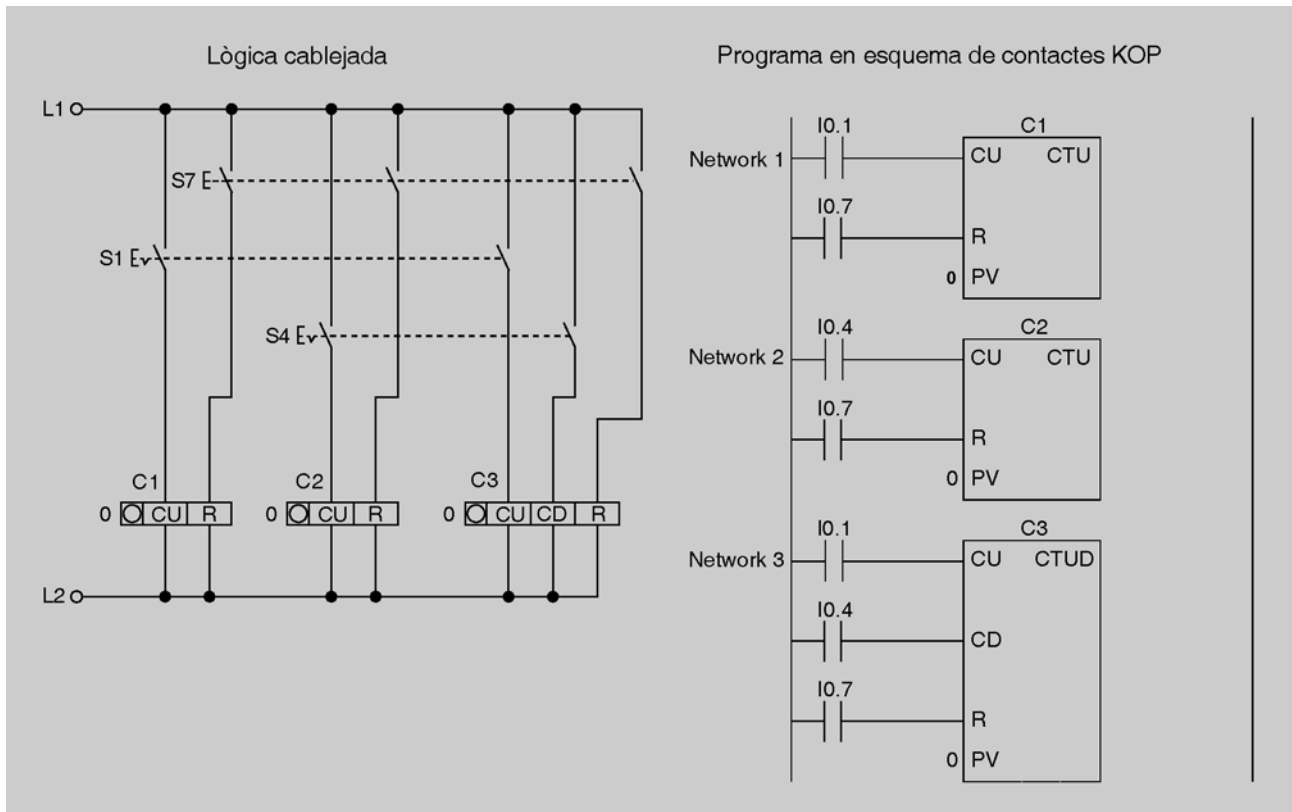
Com l'entrada PV de tots els comptadors té el valor 0, són comptadors sense preselecció.

- El comptador C2 compta les peces dolentes, incrementa el seu valor mitjançant l'entrada I0.4, el sensor de peces dolentes, i es posa a zero amb el polsador de *reset* I0.7.
- El comptador C3 compta les peces bones, incrementa el seu valor mitjançant l'entrada I0.1, el sensor de peces totals, i si la peça és dolenta la descompta mitjançant l'entrada I0.4, el sensor de peces dolentes, i es posa a zero amb el polsador de *reset* I0.7.



Ara feu l'activitat "Control de fabricació de peces en un procés de fabricació", que trobareu en la secció "Activitats" del web d'aquest mòdul.

Figura 43. Control de fabricació de peces en un procés de producció



### 3.3. Multivibradors

El **multivibrador** és un generador de pols que, quan li arriba un senyal elèctric a l'entrada, produeix un senyal elèctric i polsa la sortida.

S'entén per *pols* els canvis d'estat d'un senyal elèctric que pren valors de 0 i 1.


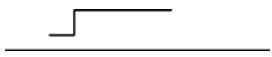

Hi ha tres tipus de multivibradors:

- **Astable** Genera un flux continu de pols, i s'anomena **multivibrador autònom** (*free running*); normalment en els programaris dels autòmats programables es coneixen com a *generadors d'impulsos*.
- **Bistable** Un senyal a l'entrada genera un canvi d'estat a la sortida que sempre està en un dels dos estats, 0 o 1; s'anomena **flip-flop** i normalment en els programaris dels autòmats programables es coneixen com a **instruccions set-reset**.

- **Monoestable** Un senyal a l'entrada genera un pols a la sortida de durada fixa; s'anomena **multivibrador d'un tret** (*one shot*) i normalment en els programaris dels autòmats programables es coneixen com a **funcions especials**. Els PLC tenen diverses d'aquestes funcions especials, en les quals el pols de sortida pot ser:
  - De durada fixa o ajustable.
  - De senyal elèctric positiu o negatiu.

En la taula 12 es mostra el tipus de senyal elèctric que donen els diferents tipus de multivibradors.

Taula 12. Pols generats per multivibradors

Tipus	Senyal de sortida	Aplicació del PLC
Astable		Generador d'impulsos
Biestable		Set-reset
Monoestable		Funcions especials

### 3.3.1. Generador d'impulsos

El **generador d'impulsos** és un multivibrador astable que genera un senyal elèctric d'ona quadrada en el qual les freqüències dels valors 0 i 1 poden ser iguals o diferents. Tots els autòmats disposen de diferents generadors d'impulsos, i en els cas concret de l'autòmat S7-200 de Siemens, en té tres i tots generen un senyal en el qual els temps de valor 0 i 1 són iguals i envien els senyals mitjançant un registre intern identificat amb les sigles *SM* (*special merker* o marques especials) i són:

- SM0.4: està 30 s en *on* (valor 1) i 30 s en *off*.
- SM0.5: està 0,5 s en *on* i 0,5 s en *off*.
- SM0.6: està 1 *scan* en *on* i 1 *scan* en *off*.

En la figura 44 podeu veure l'esquema de lògica cablejada i el programa en esquema de contactes corresponent a un circuit en què s'utilitzen els tres tipus de generadors d'impulsos.

El funcionament del programa és el següent:

- Els contactes de les marques especials generen els polsos corresponents de manera automàtica quan el PLC està en mode *run*.

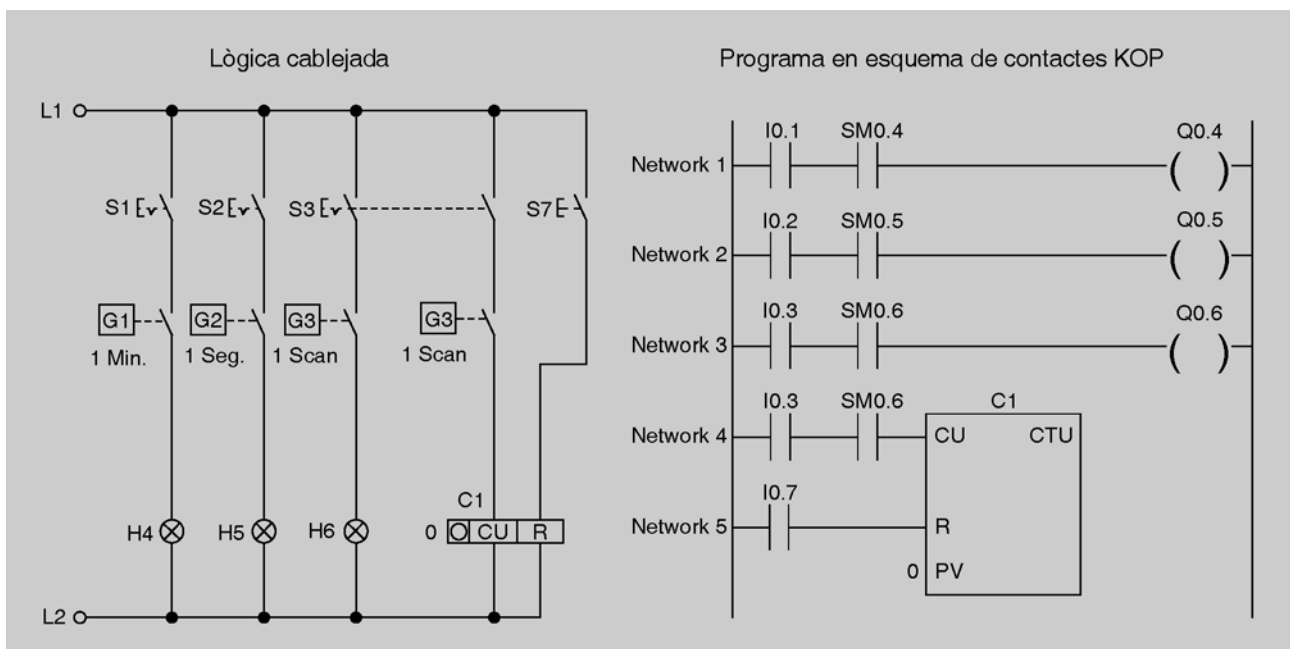
Els termes *on/off* són els més emprats en els manuals tècnics dels autòmats programables per definir els valors 1 i 0.

- Quan connecteu l'entrada I0.1 funcionarà de manera intermitent la sortida Q0.4 a través de SM0.4 i, per tant, veureu que genera un pols cada minut, en el qual està 30 s activada i 30 s desactivada.
- Amb I0.2 funcionarà de manera intermitent la sortida Q0.5 amb un pols que dura 1 s.
- Amb I0.3 funcionarà de manera intermitent la sortida Q0.6 i estarà 1 *scan* activada i un altre desactivada, i el comptador C1 incrementarà el seu valor cada dos *scan*.
- Amb l'entrada I0.7 es posa a zero el comptador.



Ara feu l'activitat "Generadors d'impulsos", que trobareu en la secció "Activitats" del web d'aquest mòdul.

Figura 44. Generadors d'impulsos



### 3.3.2. Funcions especials

Tots els autòmats tenen moltes funcions especials; d'aquestes en destaquem dues, que corresponen als multivibradors monoestables, i que en l'autòmat S7-200 de Siemens envien els senyals mitjançant els registres interns SM, que són:

- SM0.0: registre intern, que sempre té valor 1.
- SM0.1: es posa a 1 durant 1 *scan* sempre que el PLC passa de *stop* a *run*.

El temps de *scan* d'un programa normal d'un PLC pot ser de 30 ms.

La funció que fa el registre intern SM0.0 és molt útil quan es necessita fer un pont a un element que està avariats, connectant un contacte de SM0.0 en paral·lel amb aquest element, o quan interessa que una bobina no funcioni per fer tasques de manteniment, connectant un contacte negat de SM0.0 en sèrie amb la bobina.

El contacte negat del registre SM0.0 sempre tindrà valor 0.

En la figura 45 podeu veure l'esquema de lògica cablejada i el programa en esquema de contactes corresponent a un circuit en què s'utilitzen les dues funcions especials.

El registre SM0.1 s'utilitza per inicialitzar dades cada cop que es posa l'au-tòmat en *Run*. Aquesta és una opció molt interessant, per exemple per carregar valors de temporitzadors o comptadors amb preselecció en un procés, o per posar els comptadors sense preselecció a zero.

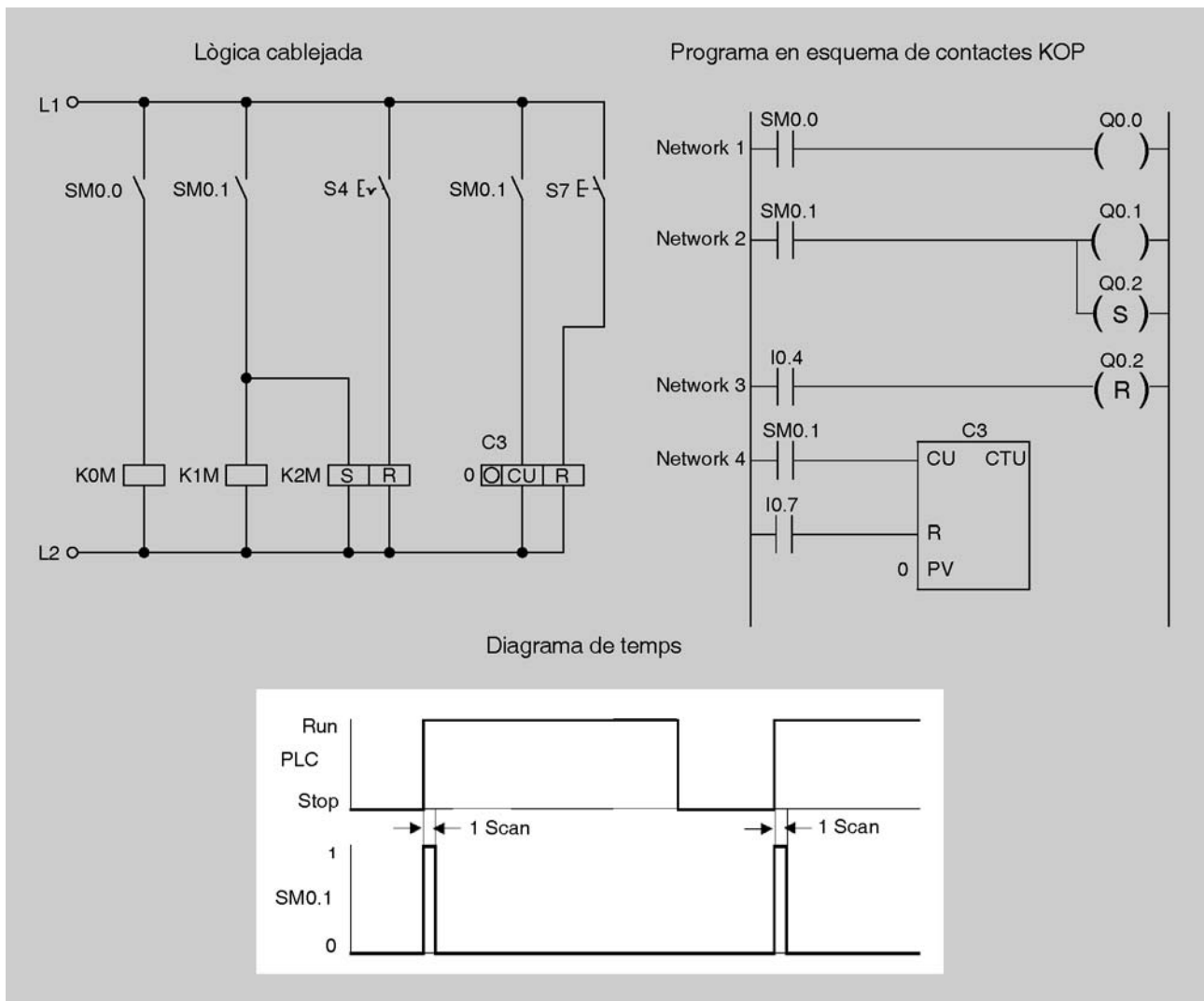
El funcionament del programa és el següent:

- La sortida Q0.0 està connectada permanentment, ja que el contacte SM0.0 sempre està a 1.
- En el *network 2* el contacte SM0.1 es posarà a 1 durant 1 *scan* i activarà les sortides Q0.1 i Q0.2. La Q0.1 es connectarà només durant 1 *scan* i la Q0.2, com que està programada mitjançant la instrucció *set*, romandrà connectada fins que es doni l'ordre de *reset* programada en el *network 3*.
- En el *network 4*, el comptador C3 comptarà les vegades que el PLC es posa a *run* mitjançant el contacte SM0.1 connectat a l'entrada CU.

**!!**

Ara feu l'activitat "Programació SM0.0 i SM0.1", que trobareu en la secció "Activitats" del web d'aquest mòdul.

Figura 45. Programació SM0.0 i SM0.1



### 3.4. Programació estructurada

El programa de l'autòmat programable corresponent al procés d'automatització d'una instal·lació es pot executar de manera lineal o estructurada.

- **Programació lineal.** Tot el programa està escrit en un únic bloc, de manera que el microprocessador del PLC, quan executa el programa, llegeix totes les instruccions programades.
- **Programació estructurada.** El programa està dividit en parts, de manera que el microprocessador del PLC, quan executa el programa, llegeix les parts del programa que en aquell moment han d'intervenir en el funcionament de la instal·lació.

En una instal·lació petita no és gaire important el nombre d'instruccions que ha de llegir la CPU en l'execució del programa, però en una de gran heu de tenir en compte que el temps de cicle augmenta en funció del nombre d'instruccions i, per tant, si hi ha parts del programa que no estan actives, no cal que la CPU dediqui temps a llegir-les. Per exemple, imagineu una instal·lació que ha de treballar en manual i en automàtic; és clar que quan col·loqueu el selector en manual, no circula corrent pel circuit d'automàtic i a l'inrevés i, per tant, quan la CPU llegeix el programa corresponent al funcionament en manual no és necessari que llegeixi les instruccions corresponents al funcionament en automàtic.

Els elements més comuns a l'hora de fer una programació estructurada són els **salts** i les **subrutines**.

#### 3.4.1. Instruccions de salt

Les instruccions de salt permeten interrompre l'execució lineal del programa i continuar en un punt diferent. Estan sempre associades a una etiqueta, anomenada normalment **meta**, i es poden realitzar cap endavant o cap endarreja, però sempre dintre del mateix bloc. L'autòmat S7-200 de Siemens identifica la instrucció d'inici del salt amb les sigles *JMP*, i la de la destinació del salt o meta amb *LBL*, i relaciona les dues instruccions amb un nom que pot estar format per lletres o xifres amb un màxim de quatre dígits.

Els salts poden ser condicionals o incondicionals, i els primers només s'executen si es compleix la condició de salt, com, per exemple, que un contacte estigui accionat, mentre que els segons s'executen sempre que la CPU llegeix la instrucció.

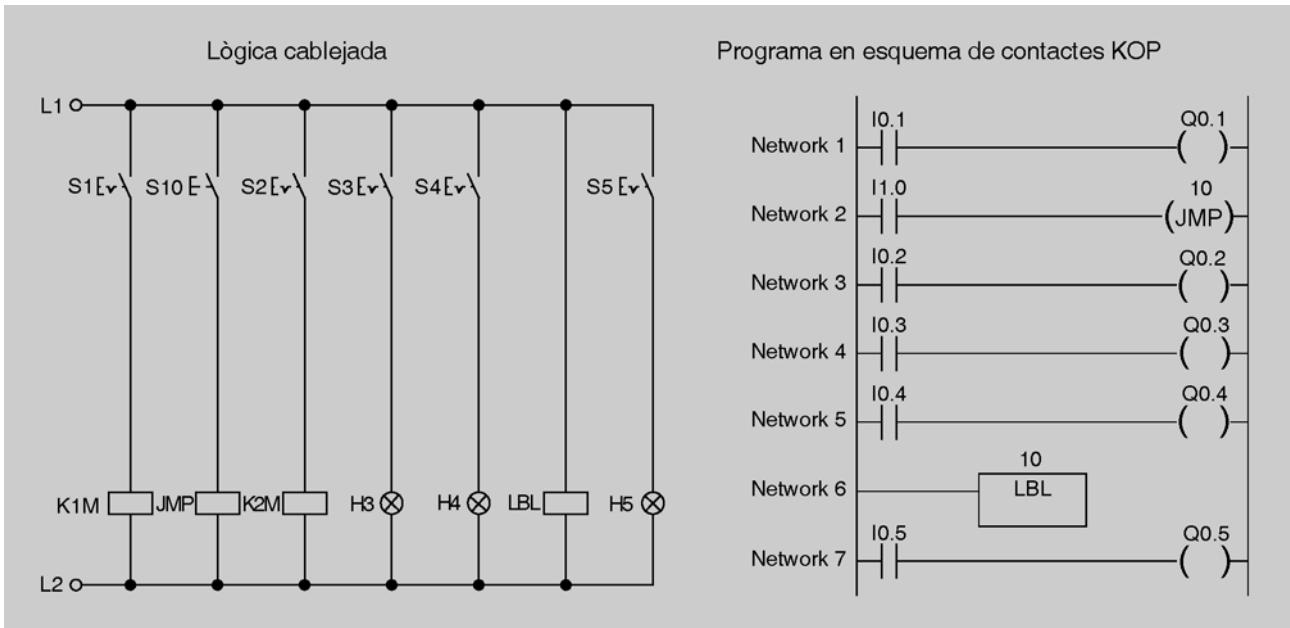
En la figura 46 podeu veure l'esquema de lògica cablejada i el programa en esquema de contactes corresponent a un circuit en què, en funció de l'estat de l'entrada I1.0, la CPU de l'autòmat no llegirà les instruccions corresponents als *networks* 3, 4 i 5.

No es pot programar un salt en un bloc i la meta en un altre bloc del programa.

#### Instruccions de salt

Les instruccions *JMP* i *LBL* les trobareu en la icona *Control de programa* del menú de navegació del programari MicroWin.

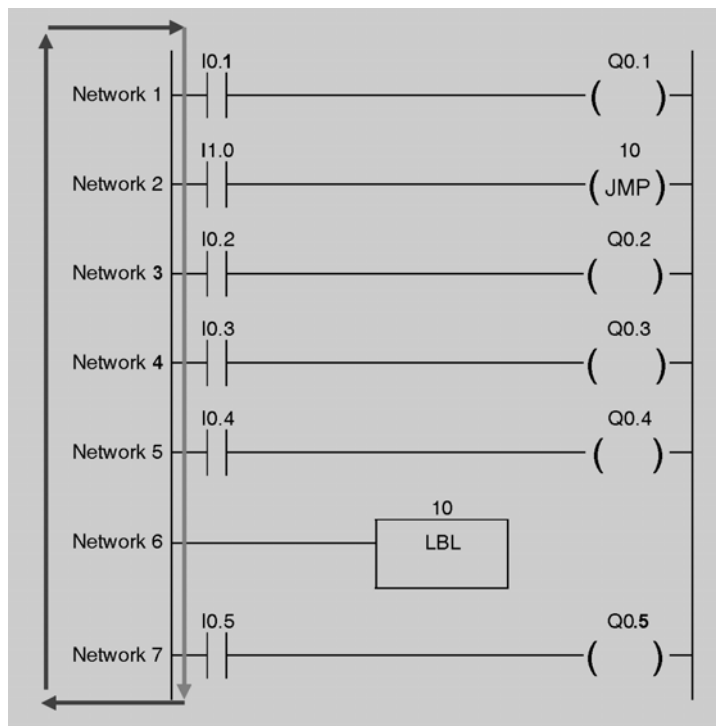
Figura 46. Salt condicional



Que la CPU **no llegeix les instruccions** vol dir que no executa les operacions programades i, per tant, encara que es connectin les entrades que han d'activar una sortida, aquesta no es connectarà.

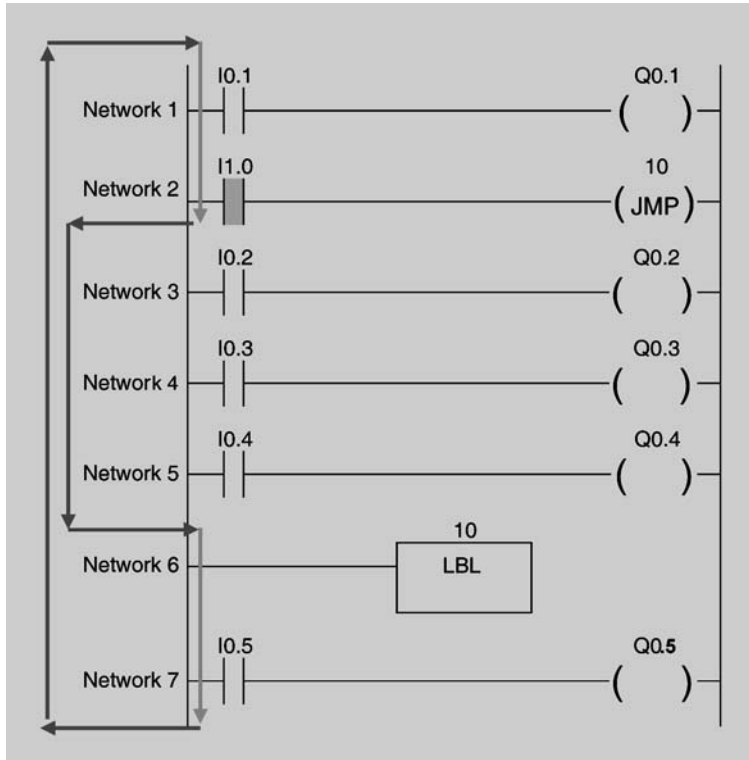
En la figura 47 teniu representat com llegiria la CPU el programa si l'entrada I1.0, que és la condició de salt, estigués a 0. Podeu veure que s'executen les instruccions de tots els *networks* i, per tant, es poden accionar totes les sortides programades mitjançant les entrades corresponents.

Figura 47. Lectura del programa sense execució del salt



En la figura 48 teniu representat com llegiria la CPU el programa si l'entrada I1.0 estigués a 1; observeu que les instruccions corresponents als *networks* 3, 4 i 5 no s'executen, ja que la CPU salta del *network* 2 al 6.

Figura 48. Lectura del programa amb execució de salt



Quan una instrucció no s'executa, no vol dir que no funcioni, sinó que com que la CPU no llegeix la instrucció, no assigna cap valor a les sortides i, per tant, aquestes mantindran el valor que tenien abans de l'execució del salt.



Ara feu l'activitat "Salt condicional", que trobareu en la secció "Activitats" del web d'aquest mòdul.

Mirant la figura 48 podeu comprovar, que mentre s'executa el salt, si poseu a 1 les entrades I0.2, I0.3 o I0.4, no es connectaran les sortides Q0.2, Q0.3 i Q0.4.

### 3.4.2. Subrutines

Tots els autòmats disposen de diferents blocs per introduir els programes. L'autòmat S7-200 de Siemens, igual que la majoria d'autòmats, els anomena **subrutines**. Es pot dividir el programa en parts i programar cada part en una subrutina diferent, per simplificar-ne l'execució.

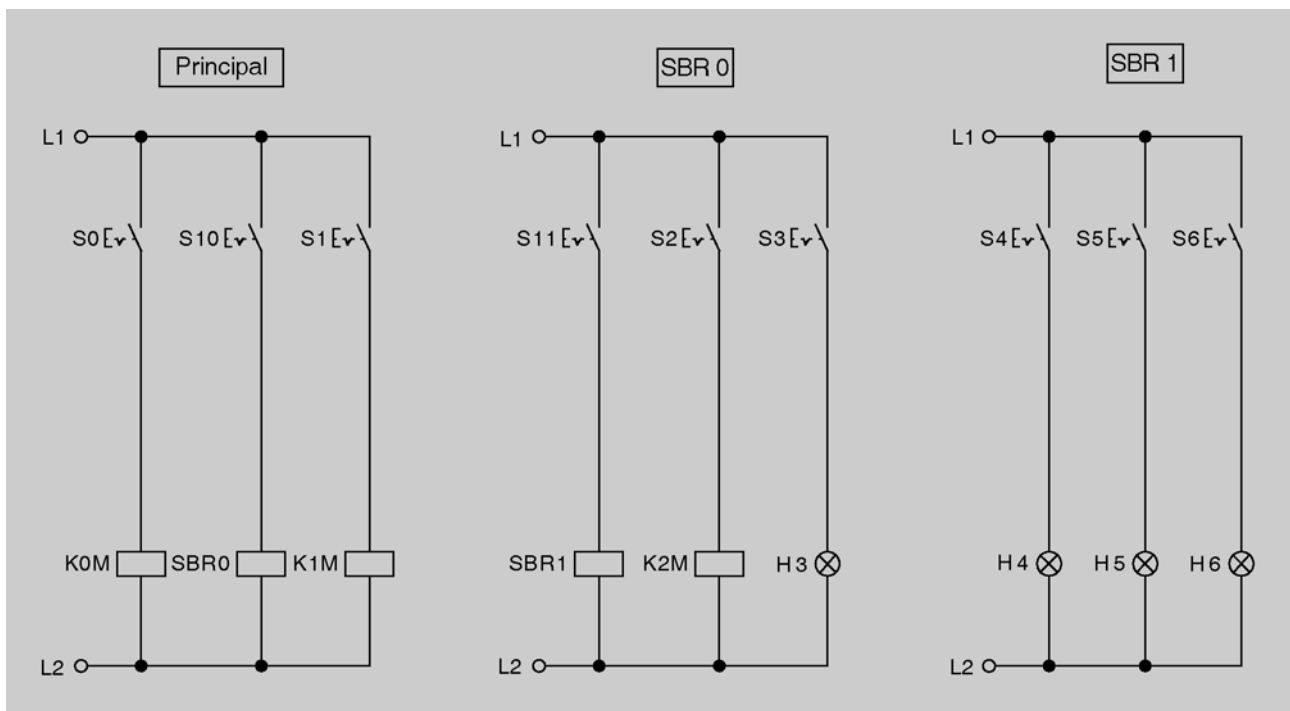
Hi ha diferents tipus de subrutines; les més comunes són les destinades a la introducció dels programes normals per part de l'usuari, però n'hi ha d'altres, com les **subrutines de sistema**, que estan creades pel fabricant

per a tasques concretes, o **subrutines d'interrupció**, que s'executen fora del cicle normal de funcionament quan es produeix algun esdeveniment concret, etc. Però per moltes subrutines que tingui un PLC, la CPU només en llegeix una, que és la **subrutina principal**, i que cada fabricant d'autòmats l'anomena de manera diferent; la resta de subrutines s'han de cridar mitjançant alguna instrucció perquè s'executin.

La instrucció més freqüent per fer la crida a una subrutina és *call*.

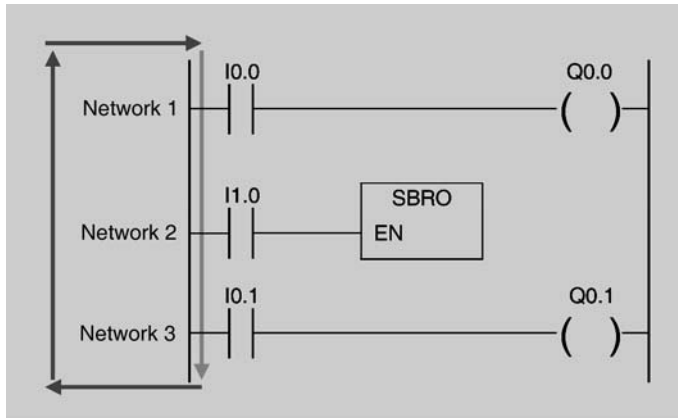
En la figura 49 podeu veure l'esquema de lògica cablada en esquema de contactes corresponent a un circuit que està dividit en tres parts: una corresponent als contactors K0M i K1M, que està programat a la subrutina principal i que, per tant, funcionaran sempre; una altra part, corresponent al contactor K2M i a la bombeta H3, programada a la subrutina 0, i que només funcionaran quan es cridi aquesta subrutina des de la principal mitjançant l'interruptor S10; i una altra part, programada en la subrutina 1 corresponent a les bombetes H4, H5 i H6, que només funcionaran quan aquesta subrutina sigui cridada des de la subrutina 0 mitjançant l'interruptor S11.

Figura 49. Programació en diferents subrutines



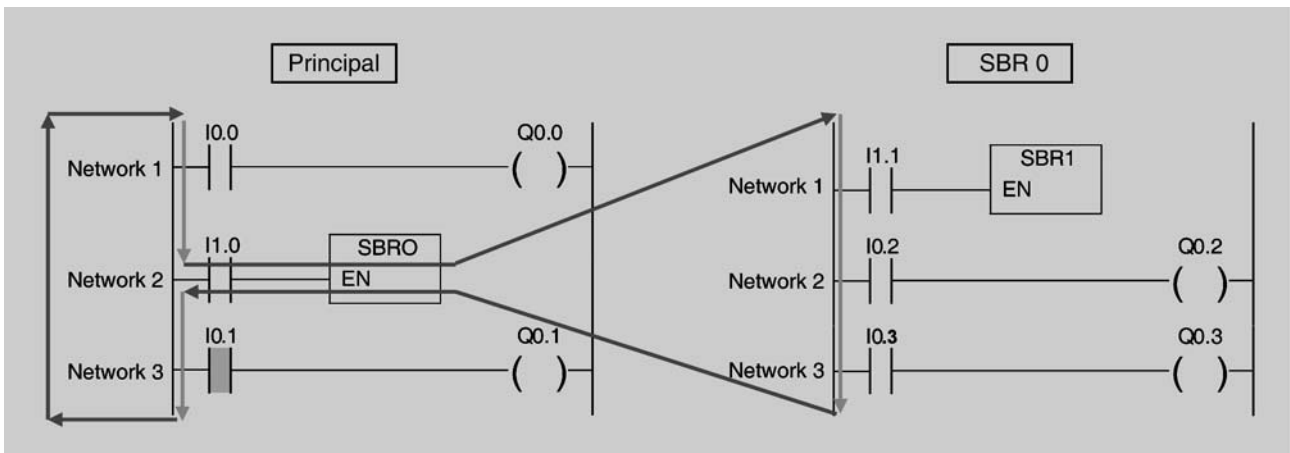
En la figura 50 podeu veure l'execució del programa si l'entrada I1.0 està oberta (a zero); per tant, no es fa la crida de la subrutina 0. Observeu que només s'executen les instruccions corresponents als *networks* 1, 2 i 3 de la subrutina principal. Per tant, amb les entrades I0.0 i I0.1 es poden connectar les sortides Q0.0 i Q0.1, respectivament.

Figura 50. Execució del programa amb l'entrada I1.0 = 0



En la figura 51 podeu veure l'execució del programa si l'entrada I1.0 està activada (està a 1); per tant, crida la subrutina 0 i s'executen les instruccions programades a les subrutines principal i SBR0. Observeu l'ordre en què la CPU executa les instruccions: comença el cicle en la principal, llegeix els *networks* 1 i 2 i des d'aquest últim salta a la subrutina 0, la llegeix sencera i torna a la principal al *network* 3, que és la instrucció següent des de la qual havia fet el salt.

Figura 51. Execució del programa amb l'entrada I1.0 = 1 i I1.1 = 0



En la figura 52 podeu veure l'execució del programa si les entrades I1.0 i I1.1 estan activades; per tant, ara sí que s'executen totes les instruccions de totes les subrutines.

En la taula 13 es mostra l'ordre en què la CPU de l'autòmat programable executa les instruccions corresponents al programa de la figura 52 quan estan accionades les entrades I1.0 i I1.1.

Taula 13. Ordre d'execució d'instruccions amb crides condicionals a subrutines

Subrutina	Network
Principal	1
Principal	2

**!!**

Ara feu l'activitat "Programació en diferents subrutines", que trobareu en la secció "Activitats" del web d'aquest mòdul.

Subrutina	Network
0	1
1	1
1	2
1	3
0	2
0	3
Principal	3
Principal	1

Figura 52. Execució del programa amb l'entrada I1.0 = 1 i I1.1 = 1

