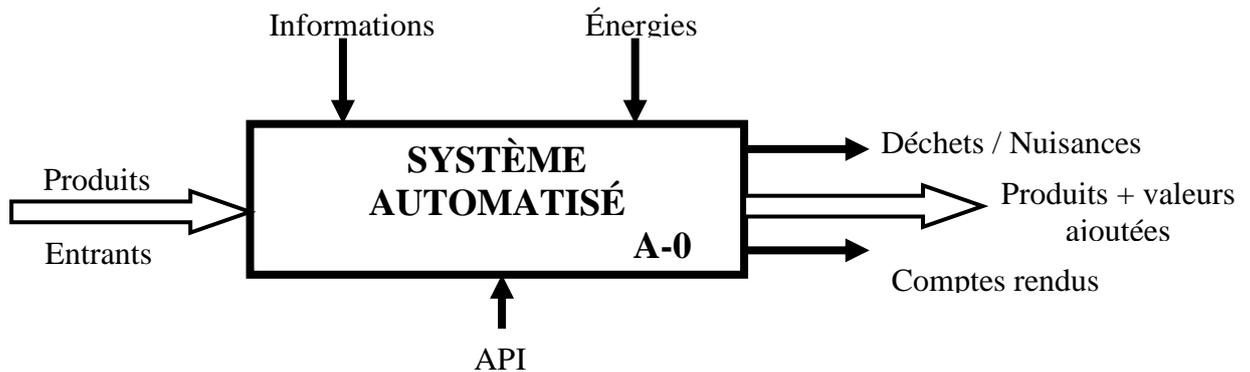


*Chapitre 3 :*                    **Automate programmable industriel API**  
**programmable logic controllers PLC**

**I) Principe d'un système automatisé**

Un système automatisé est un moyen d'assurer l'objectif primordial d'une entreprise et à la compétitivité de leurs produits. Il permet d'ajouter une valeur aux produits entrants d'où la fonction globale de tout système (figure 1) est de conférer une valeur ajoutée, à un ensemble de matières d'œuvre dans un ou un contexte donné. De plus, un système de production est dit « industriel » si l'obtention de cette valeur ajoutée, pour un ensemble de matières d'œuvre donné, a un caractère reproductible et peut être exprimée et quantifiée en termes économiques.

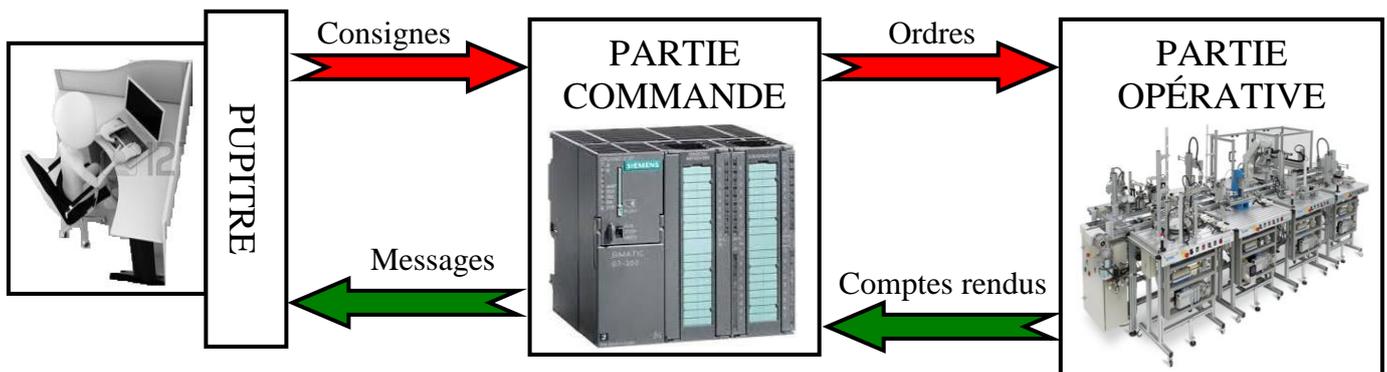


**Figure 1**

**II) Les différentes parties d'un système automatisé**

Chaque système automatisé comporte 2 parties: la Partie Opérative (PO) et la Partie Commande (PC)

- **La partie opérative (PO)** comprend les actionneurs (moteur électrique, vérins pneumatique et/ou hydraulique, ...) qui agissent directement sur le processus automatisé
- **La partie commande (PC)** coordonne les différentes actions de la partie opérative. Elle comprend essentiellement les capteurs, les boutons de commande, les pré actionneurs...



**Figure 2**

### III) Chaine d'action

#### 1) Electrique

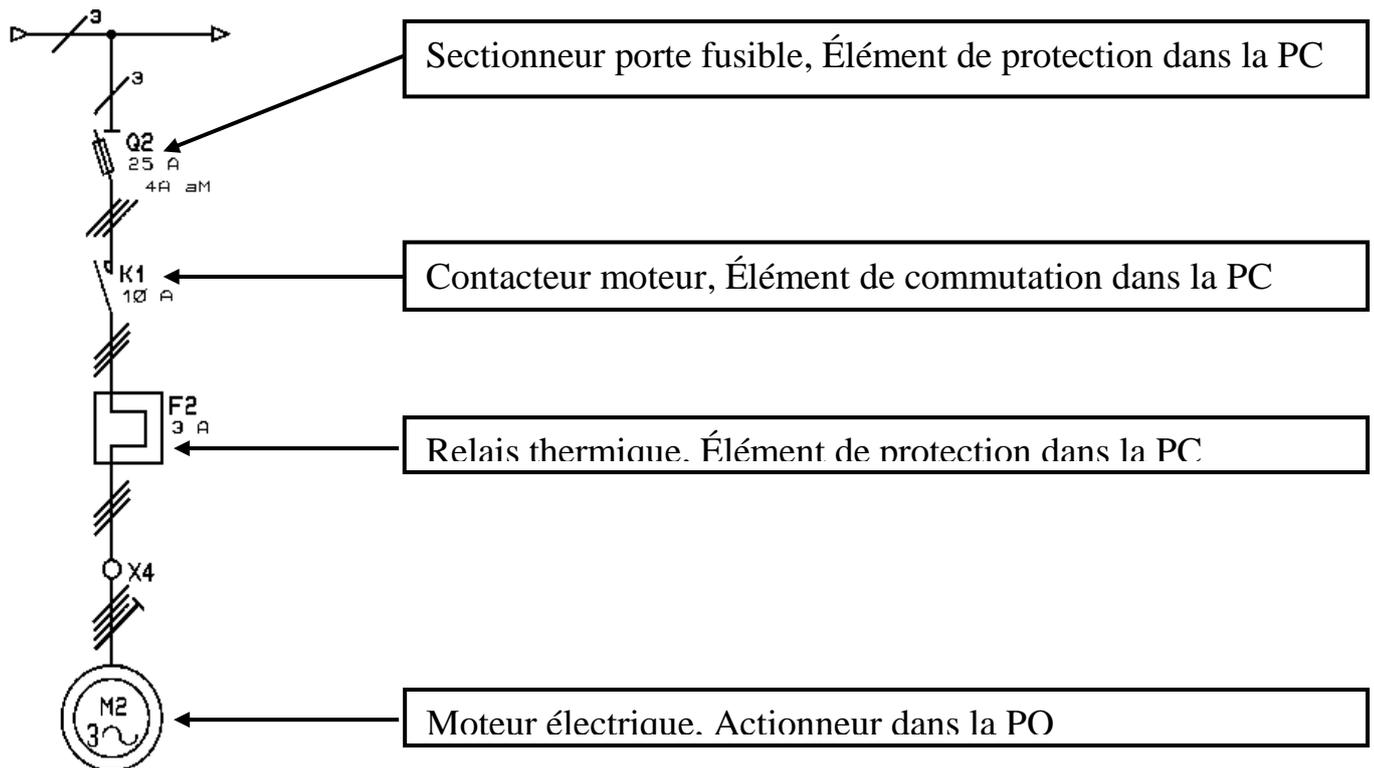


Figure 3

#### 2) Pneumatique

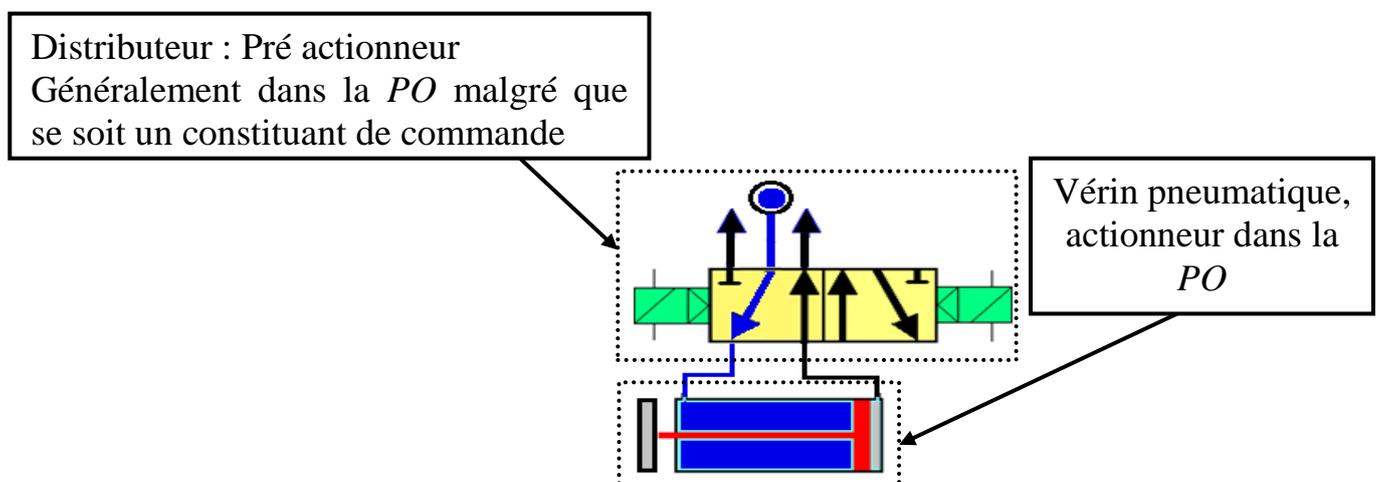


Figure 4

### IV) Structure d'un automate programmable

Il existe plusieurs types d'API différents:

- Les compacts
- Les racktables
- Les modulaires

#### Quelques exemples:

LES COMPACTS				
SIEMENS LOGO	CROUZET MILLENIUM	SCHNEIDER ZELIO	SCHNEIDER TWIDO	MOELLER PS4
				
Et d'autres marques				

LES RACKTABLES ET MODULAIRES			
SIEMENS S7-300	SCHNEIDER TSX 37	MOELLER	SCHNEIDER TSX 57
			

## V) Structure d'un automate programmable industriel

### A)- Architecture d'un API :

La structure interne d'un API peut se représenter comme suit :

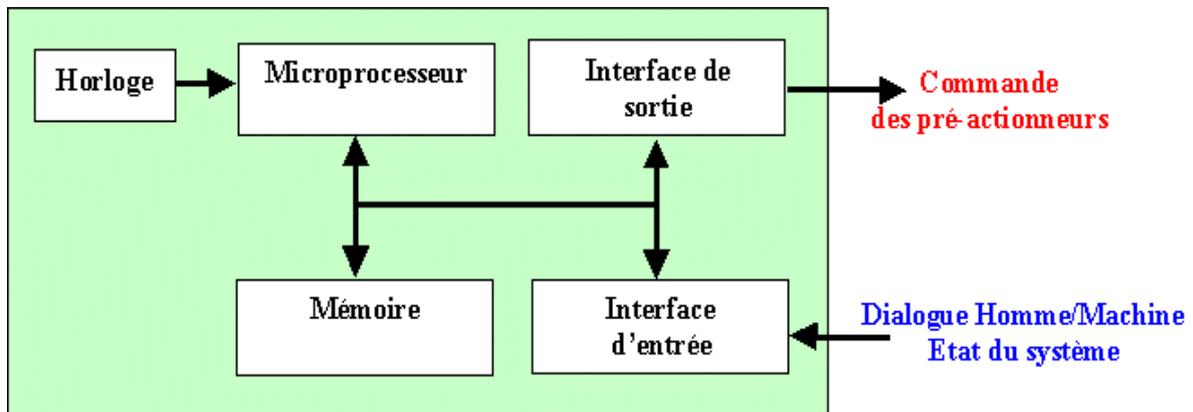


Figure 5

L'automate programmable **reçoit** les informations relatives à l'état du système et puis **commande** les prés -actionneurs suivant le programme inscrit dans sa mémoire, un API se compose donc de trois grandes parties :

- Le processeur ;
- La zone mémoire ;
- Les interfaces Entrées/Sorties

#### 1)- Le microprocesseur :

Le **microprocesseur** réalise toutes les fonctions logiques ET, OU, les fonctions de temporisation, de comptage, de calcul... à partir d'un programme contenu dans **sa mémoire**. Il est connecté aux autres

éléments (mémoire et interface E/S) par des liaisons **parallèles** appelées '**BUS**' qui véhiculent les informations sous forme binaire.

## 2)- La zone mémoires :

### a)- La Zone mémoire va permettre :

- De recevoir les informations issues des capteurs d'entrées
- De recevoir les informations générées par le processeur et destinées à la commande des sorties (valeur des compteurs, des temporisation, ...)
- De recevoir et conserver le programme du processus

### b)- Action possible sur une mémoire :

- **ECRIRE** pour modifier le contenu d'un programme
- **EFFACER** pour faire disparaître les informations qui ne sont plus nécessaires
- **LIRE** pour en lire le contenu d'un programme sans le modifier

## 3)- Les interfaces d'entrées/sorties :

Les **entrées** reçoivent des informations en provenance des **éléments de détection** et du **pupitre opérateur**. Les **sorties** transmettent des informations aux **pré-actionneurs** et aux **éléments de signalisation** du pupitre.

### a)- Interfaces d'entrées :

Elles sont destinées à :

- Recevoir l'information en provenance du capteur
- Traiter le signal en le mettant en forme, en éliminant les parasites et en isolant électriquement l'unité de commande de la partie opérative.

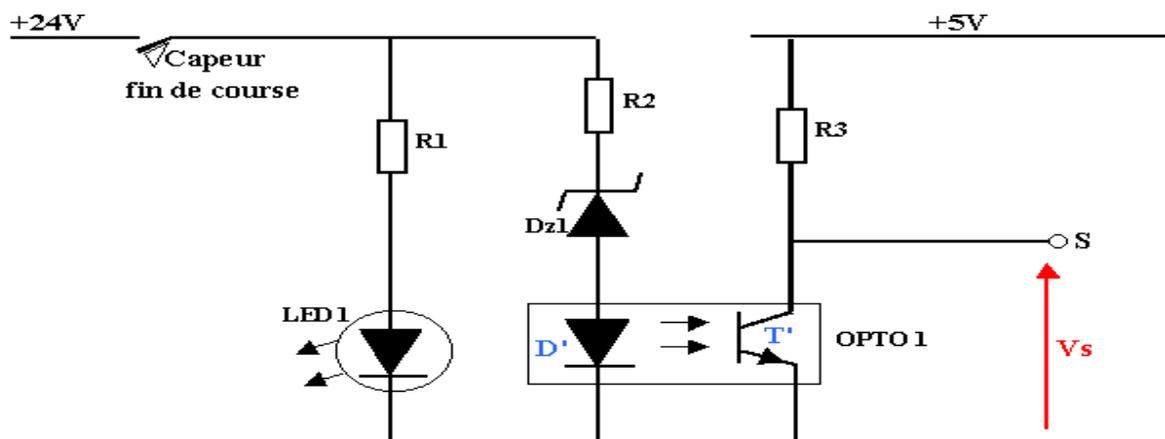


Figure 6

### b)- Interfaces de sorties :

Elles sont destinées à :

- Commander les pré-actionneurs et éléments des signalisations du système
- Adapter les niveaux de tensions de l'unité de commande à celle de la partie opérative du système en garantissant une isolation galvanique entre ces dernières.

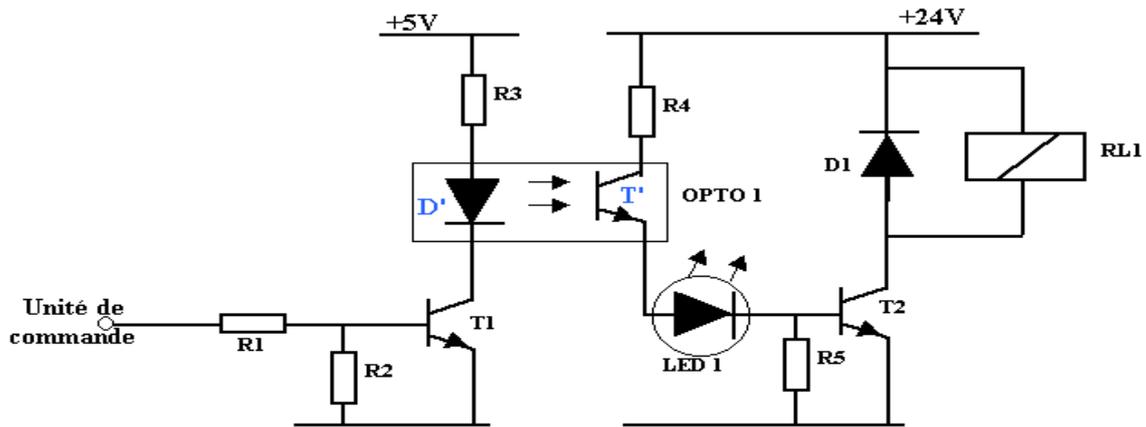
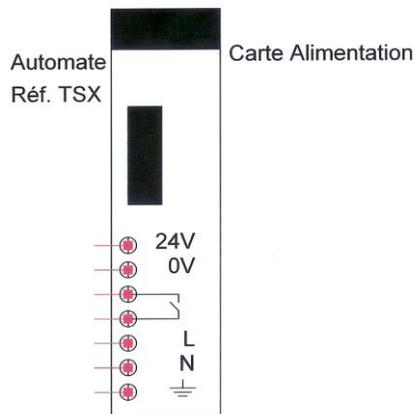


Figure 7

#### 4)- Alimentation de l'automate programmable industriel :

L'alimentation intégrée dans l'API, fournit à partir des tensions usuelles des réseaux (230 V, 24 V=) les tensions continues nécessaire au fonctionnement des circuits électroniques.



#### 5) les organes de commande :

Les signaux qui proviennent des organes de commande sont très divers :

- manuels : boutons-poussoirs, commutateurs, sélecteurs...
- automatiques : fins de course, détecteurs de proximité, cellules photoélectriques, détection de pression, de vitesse, de température pour les plus courants.

#### Nature des signaux d'entrée

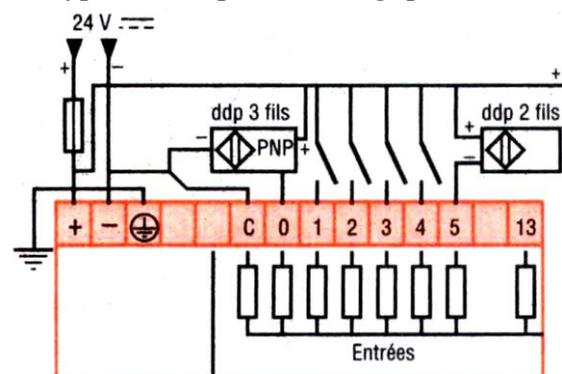
Elle peut être très diverse : de type Tout ou Rien (TOR) ou de type numérique ou analogique.

#### 6) les tensions :

Les tensions les plus couramment rencontrées sur les entrées sont :

- en courant continu : 12, 24, 48 V ;
- en courant alternatif : 24, 48, 130, 230 V

Enfin, les entrées doivent être protégées contre les parasites qui sont souvent des surtensions



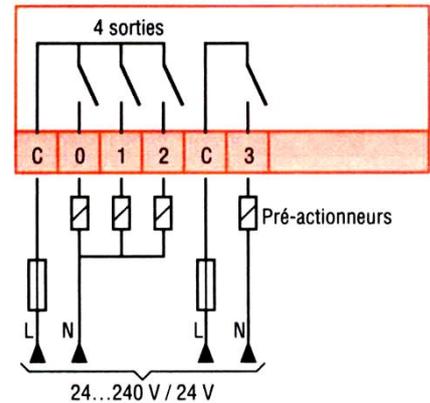
dangereuses pour l'automate programmable.

### ***Le branchement des capteurs sur des entrées 24 V à courant continu.***

#### **7) les sorties**

Les modules de sortie assurent l'interfaçage entre l'automate et les organes commandés, c'est-à-dire que le signal de sortie est au niveau demandé et il est isolé de l'électronique de l'automate.

- Les organes commandés sont en général des contacteurs, des distributeurs pneumatiques ou hydrauliques, des voyants de signalisation.
- Les modules de sortie sont soit à base de relais électromagnétiques, soit à base de transistors ou de triacs pour les sorties statiques



Les tensions applicables en sortie sont en général de 24 à 240 V en courant alternatif et de 24 ou 48 V en courant continu (sortie statique).

### ***Le branchement des bobines de contacteurs sur les sorties contacts de relais de l'automate programmable.***

#### **B)- Automate programmable industriel TSX 37 :**

Un automate est constitué d'une CPU (de l'anglais Central Processing Unit, " Unité centrale de traitement ") de plusieurs "cartes" ou "modules". Ces "cartes" ou "modules" s'encastrent dans les différents "racks".

Certains automates peuvent avoir des cartes mémoires, des piles de sauvegarde, des cartes de communication.

#### **1) Les différentes cartes**

Il existe différent type de cartes:

- TOR, Tout Ou Rien
- Sorties analogiques
- Entrées analogiques
- Communication
- Comptage

#### **2) Les racks**

L'API est constitué de plusieurs rack dans lesquels on peut installer des modules de différents types de cartes.

#### **3) Affectation & écriture des entrées / sorties**

L'affectation des entées et des sorties permet de faire l'adressage entre le matériel et l'API en fonction de sont câblage. D'un point de vue de la programmation on travaillera sur l'adressage des E/S.

Les entrées et les sorties analogiques sont notées par un mot de la façon suivante : %xy.z

x : les sorties analogiques seront notées par la lettre Q (Output)  
les entrées analogiques seront notées par la lettre I (Input)

y : c'est l'emplacement physique du module analogique

z : c'est le numéro de la voie utilisée

**Exemple :**

On a câblé un BP nommé DCY sur l'entrée n°5 de la carte d'entrée du module 3 et un voyant sur la sortie n°0 de la carte de sortie du module 2.

Donner l'écriture de l'entrée et de la sortie.

Écriture de l'entrée: %I3.5

Écriture de la sortie: %Q2.0

**Adressage**

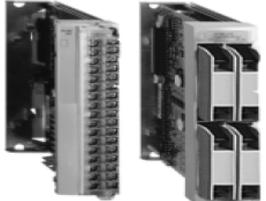
Variable	Repère	Désignation	Exemple
Entrée (I input)	%Ix.i	X : N° module i : N° de voie	%I1.4
Sortie (Q output)	%Qx.i	X : N° module i : N° de voie	%Q2.3
Mémoire bit	%M.i	i : N° du bit interne	%M25
Mémoire mot	%MWi	i : N° du mot interne	%MW11
Temporisateur	%Tmi	i : N° du temporisateur	%TM3
Compteur	%Ci	i : N° du compteur	%C4
Variable d'étape	%Xi	X : étape, i : N° étape	%X10

**X/ CÂBLAGE ENTRÉES & SORTIE TOUT OU RIEN (TOR)**

**1/ Caractéristiques de la carte**

**Exemple d'une carte DMZ28DR DE SCHNEIDER**

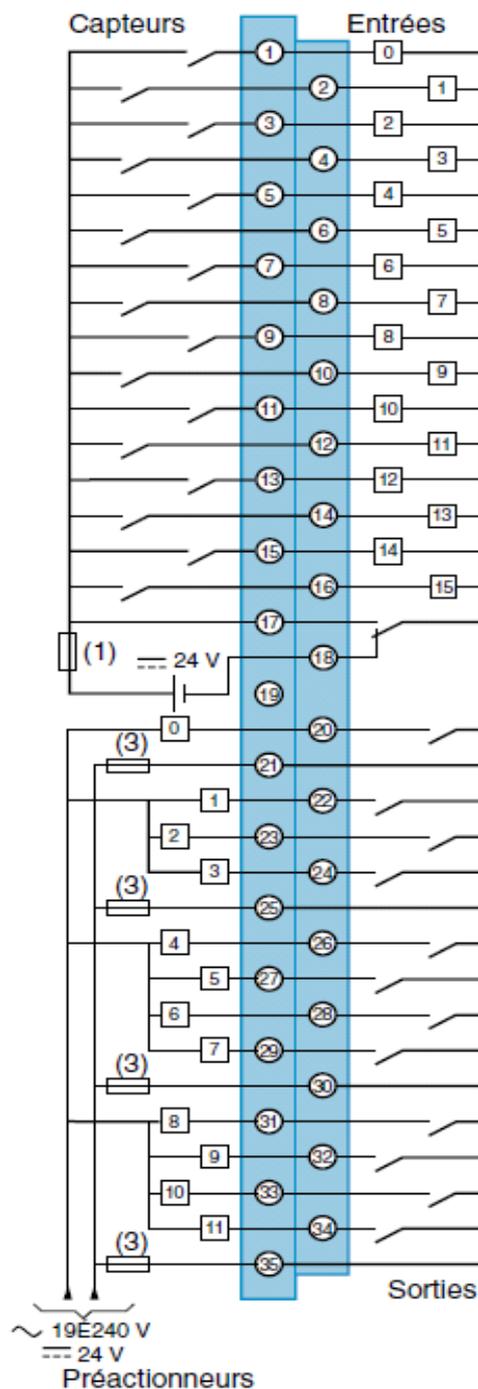
Extrait du catalogue SCHNEIDER

Modules d'entrées/sorties "Tout ou Rien"							
	Nombre d'E/S	Nombre, type d'entrées	Nombre, type de sorties	Format	Raccordement	Référence	Masse kg
 TSX DMZ 16DTK	16	8, --- 24 V (logique positive IEC type 1)	8, statiques --- 24 V/0,5 A protégées	Demi	Par connecteur type HE 10 (1) et bornier à cage	TSX DMZ 16 DTK	0,160
	28	16, --- 24 V (logique positive IEC type 1)	12, statiques --- 24 V/0,5 A protégées	Standard	Par connecteur type HE 10 (1)	TSX DMZ 28DTK	0,330
					Par bornier à vis (fourni)	TSX DMZ 28DT	0,465
 TSX DMZ 28DT TSX DMZ 64DTK	16	16, --- 24 V (logique positive IEC type 1 ou logique négative)	12, relais 50 VA non protégées	Standard	Par bornier à vis (fourni)	TSX DMZ 28DR	0,500
					Par bornier à vis (fourni)	TSX DMZ 28AR	0,500
	64	32, --- 24 V (logique positive IEC type 1)	32, statiques --- 24 V/0,1 A protégées	Standard	Par connecteur type HE 10 (1)	TSX DMZ 64DTK	0,410

(1) Module livré avec cache connecteur type HE 10.

## TSX DMZ 28DR

### Logique positive (Sink)



**Logique négative (Source) :**  
+ 24 V = borne 17  
- 24 V = borne 18 = commun

En fonction de la documentation ci-dessus, répondre aux questions suivantes.

De combien d'entrée TOR dispose cette carte?

16 entrée

De combien de sorties TOR dispose cette carte?

12 sorties

Quelle est la valeur et le type de tension des entrées?

24V DC

Quelle est le type des sorties?

Sorties à relais

### 2/ Raccordement des entrées / sorties de la carte

En fonction de la documentation ci-dessous, répondre aux questions suivantes.

Extrait du catalogue SCHNEIDER

Nous disposons de 16 entrées TOR, donner leur numéro:

Elles sont numérotées de 0 à 15

Nous disposons de 12 sorties à Relais, donner leur numéro:

Elles sont numérotées de 0 à 11

La carte doit être alimentée en 24vDC.

Donner le numéro des bornes d'alimentation et donner leur correspondance:

Elles sont numérotées de 17 & 18.

18 pour le 0V DC (-)

17 pour le 24V DC (+)

Sur quelles bornes se situe la sortie 0?

Entre les bornes 20 et 21

Sur quelles bornes se situe la sortie 1?

Entre les bornes 25 et 22

Sur quelles bornes se situe la sortie 2?

Entre les bornes 25 et 23

Sur quelles bornes se situe la sortie 3?

Entre les bornes 25 et 24

Qu'elle est la borne commune des sorties 1, 2 & 3?

C'est la borne 25

## XII/ LA PROGRAMMATION

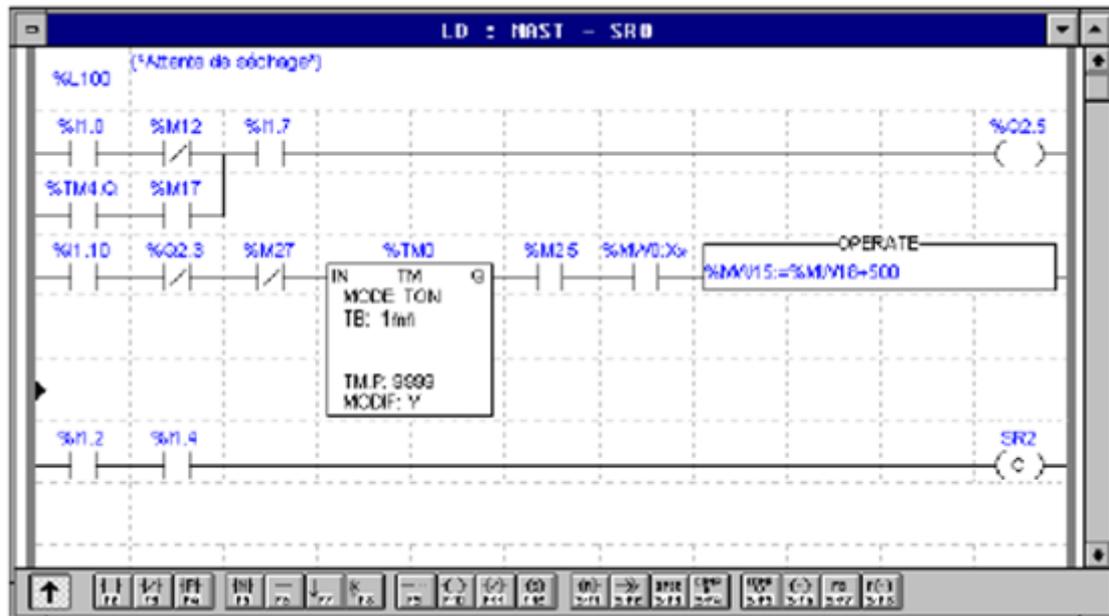
Les automates SCHNEIDER se programme à l'aide du logiciel PL7-MICRO

Il existe plusieurs types de langage:

- un langage graphique, le langage à contacts (LD) transcription de schémas à relais, est adapté au traitement combinatoire.

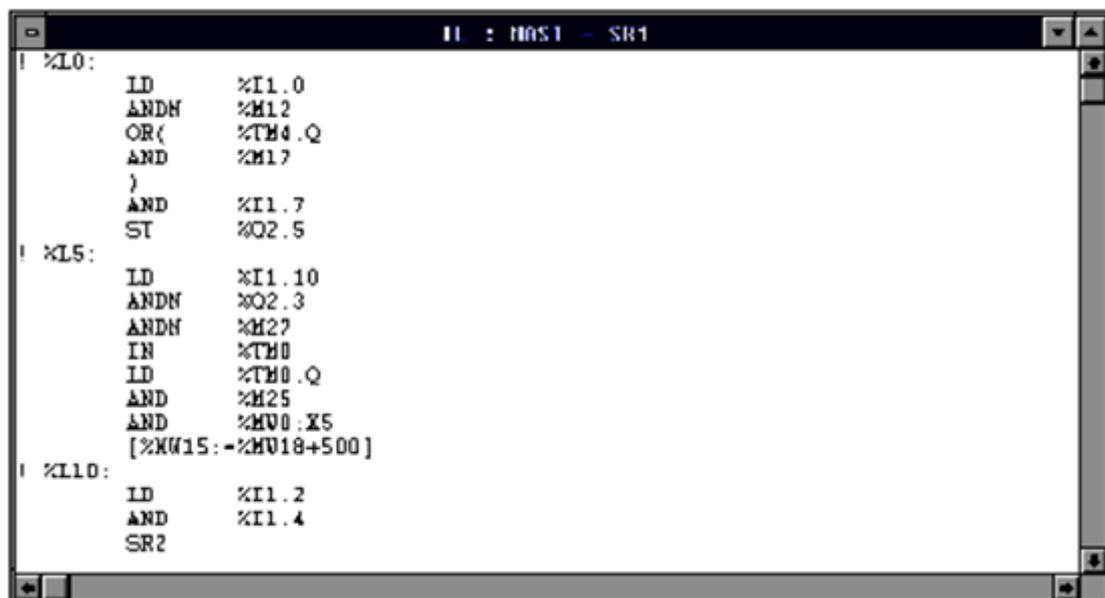
Il offre les symboles graphiques de base : contacts, bobines, blocs. L'écriture de calculs numériques est possible à l'intérieur de blocs opérations.

*Exemple:*



- le langage liste d'instructions (IL) est un langage "machine" qui permet l'écriture de traitements logiques et numériques.

*Exemple:*



- un langage littéral structuré (ST) qui est un langage de type "informatique" permettant l'écriture structurée de traitements logiques et numériques.

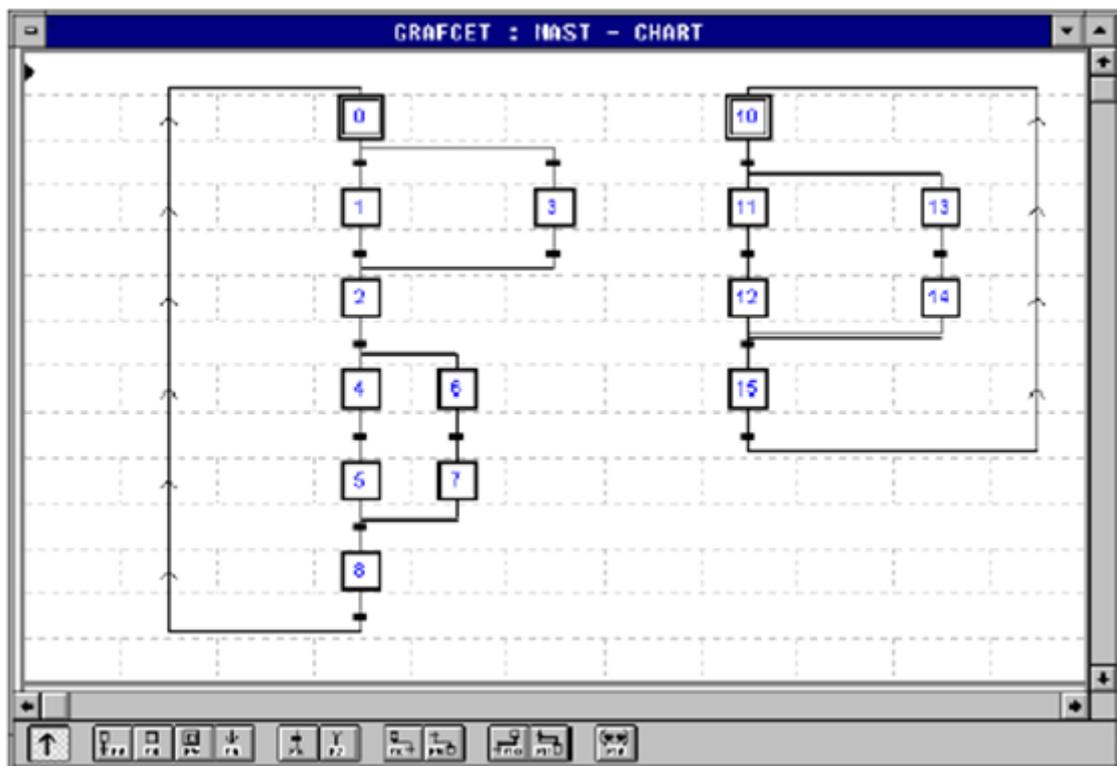
Exemple:

```
ST : MAST - SR10
(*Recherche du premier élément non nul dans un tableau de 32 mots
Détermination de sa valeur(%M10), de son rang(%M11)
Cette recherche s'effectue si %M0 est à 1
%M1 est mis à 1 si un élément non nul existe, sinon il est mis à 0 *)

IF %M0 THEN
  FOR %M99:=0 TO 31 DO
    IF %M100[%M99]<>0 THEN
      %M10:=%M100[%M99];
      %M11:=%M99;
      %M1:=TRUE;
      EXIT;          (*Sortie de la boucle FOR*)
    ELSE
      %M1:=FALSE;
    END_IF;
  END_FOR;
ELSE
  %M1:=FALSE;
END_IF;
```

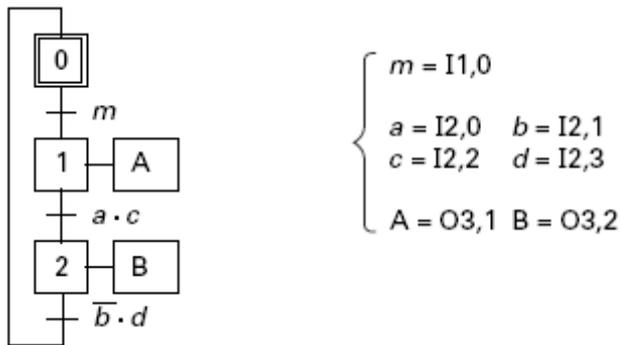
- un langage Grafcet qui permet de représenter graphiquement et de façon structurée le fonctionnement d'un automate séquentiel.

Exemple:

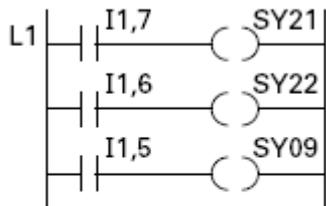


C'est ce langage que nous utiliserons pour programmer les automates TSX 37.

### Programmation "directe" en GRAFCET Exemple



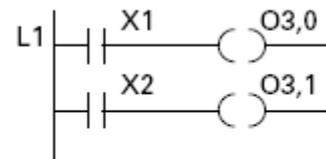
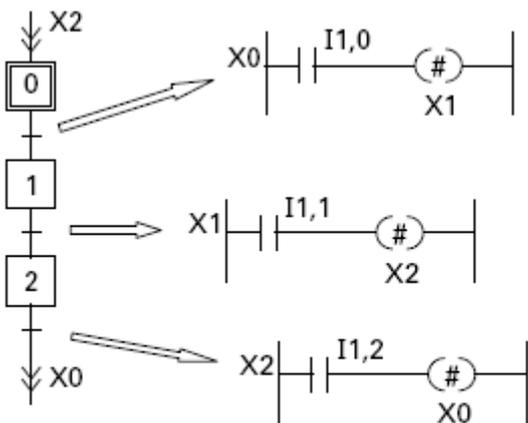
$m = I1,0$   
 $a = I2,0 \quad b = I2,1$   
 $c = I2,2 \quad d = I2,3$   
 $A = O3,1 \quad B = O3,2$



**Bits systèmes de l'automate :**

- SY21 = I1,7 : init du G7
- SY22 = I1,6 : RAZ du G7 (AUD)
- SY09 = I1,5 : mise à 0 des Out (Aud)

**a traitement préliminaire**



**c traitement postérieur**

**b traitement séquentiel**