

Automate programmable industriel

- Objectifs de la séance :
- d'analyser le fonctionnement d'un système simple
- de définir le concept de logique programmée
- de citer les caractéristiques fonctionnelles d'un système séquentiel
- de décoder le schéma de raccordement d'un automate programmable de décoder un programme.



SOMMAIRE

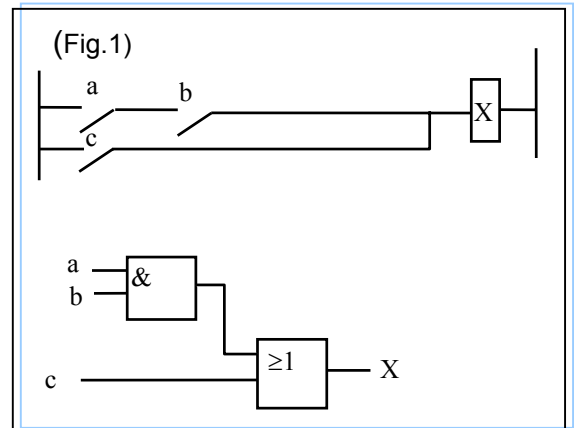
Page n° 1	Page de garde
Page n° 2	Logique câblée – logique programmée
Page n° 3	L'automate programmable
Page n° 4	Détail de l'architecture
Page n° 5	Raccordement de l'automate
Page n° 6	Programmation
Page n° 7	Programmation
Page n° 8	Exemple de programmation
Page n° 9	Exercice
Page n° 10	Annexe1
Page n° 11	Annexe 2

Avant propos : les automatismes nécessitent un grand nombre d'entrées sorties le système de relayage devenant trop complexe, on le remplace par un système de logique programmée appelée automate programmable.

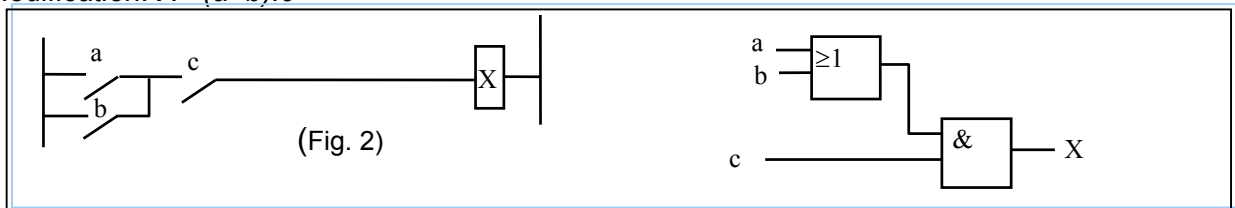
1. La logique câblée :

Exemple: $X = (a.b) + c$

L'automatisme est obtenu en reliant entre eux les différents constituants de base ou fonctions logiques par câblage. La logique câblée correspond à un traitement parallèle de l'information. Plusieurs constituants peuvent être sollicités simultanément.

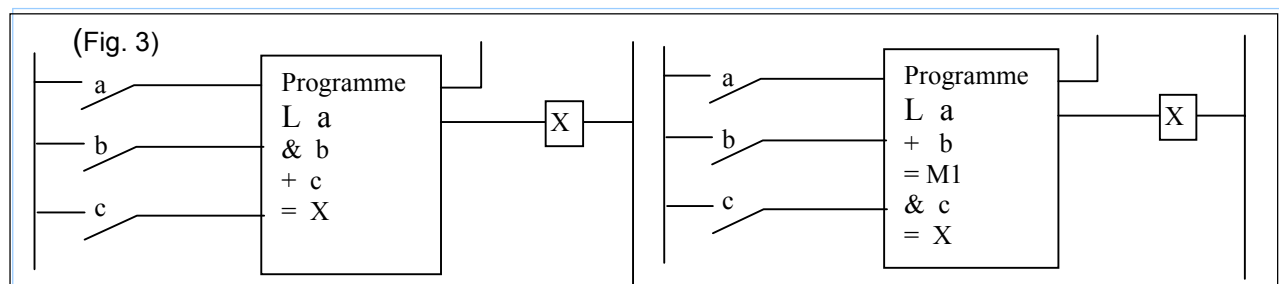
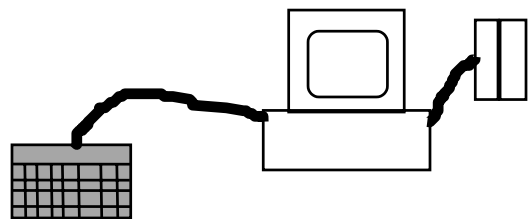


Modification: $X = (a+b).c$



2. La logique programmée.

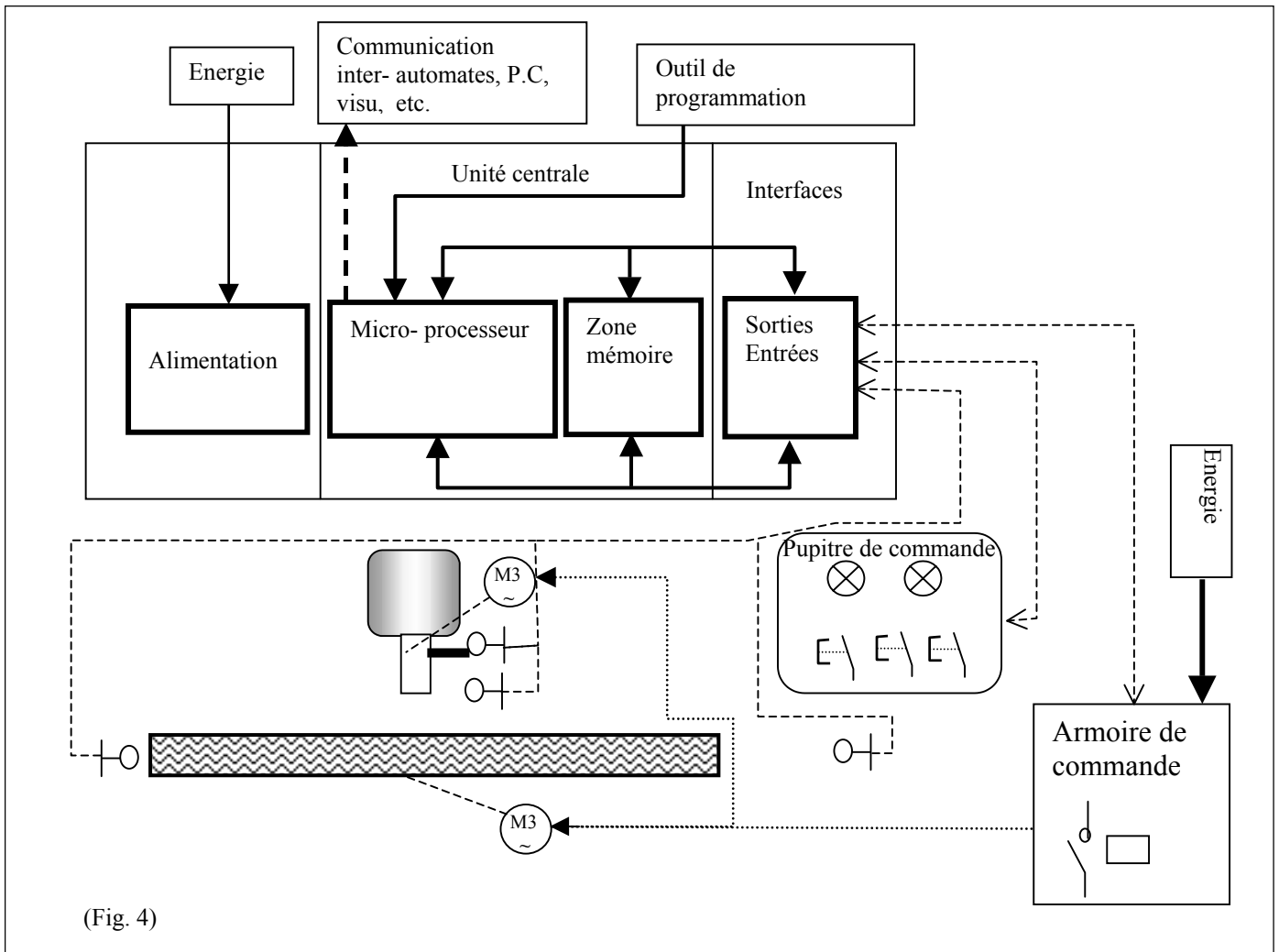
Elle correspond à une démarche séquentielle, seule une opération élémentaire est exécutée à la fois, c'est un traitement série. Le schéma électrique est transcrit en une suite d'instruction qui constitue le programme. En cas de modification des équations avec les mêmes accessoires, l'installation ne comporte aucune modification de câblage seul le jeu d'instructions est modifié.



Ici il suffit de modifier le jeu d'instruction pour obtenir un fonctionnement différent.

3. L'AUTOMATE PROGRAMMABLE.

3.1. Architecture et insertion dans un système automatisé.



3.2. Fonctionnement :

L'automate programmable **reçoit** les informations relatives au système, **il traite** ces informations en fonction **du jeu d'instruction** et **modifie** l'état de ses sorties qui **commandent** les pré-actionneurs.

Recevoir : nécessité d'informations d'entrées.

Traiter : notion de programme et de microprocesseur.

Jeu d'instructions : notion de stockage donc de mémoire.

Commander : notion de sortie afin de donner des ordres.

3.3. Détail de l'architecture.

Alimentation de l'automate : intégrée ou indépendante de l'automate programmable elle doit fournir les tensions usuelles 240 V ~, 24V~ ou 24V =

L'unité centrale : le microprocesseur réalise les fonctions logiques ou d'autres fonctions intégrées telles que les temporisations, le comptage le calcul etc..

- Le microprocesseur est connecté aux autres éléments (mémoires et interfaces entrées sorties) par des liaisons parallèles appelées bus qui véhiculent les informations sous **forme binaire**.

- Le microprocesseur possède également des liaisons avec l'extérieur pour le dialogue avec l'outil de programmation, raccordement sur terminal ou sur réseau inter automates.

La zone mémoire : elle contient toutes les données nécessaires au fonctionnement de l'automate ainsi que la liste ou le jeu d'instruction qui constitue le programme.

Classification des mémoires :

- RAM (random access memory) mémoire vive dans laquelle on peut lire ou écrire.
- ROM (read only memory) mémoire morte dans laquelle on peut lire.
- EEPROM mémoire morte reprogrammable avec un effacement électrique.
- EPROM ou UVPRM mémoire morte reprogrammable avec un effacement aux U V.

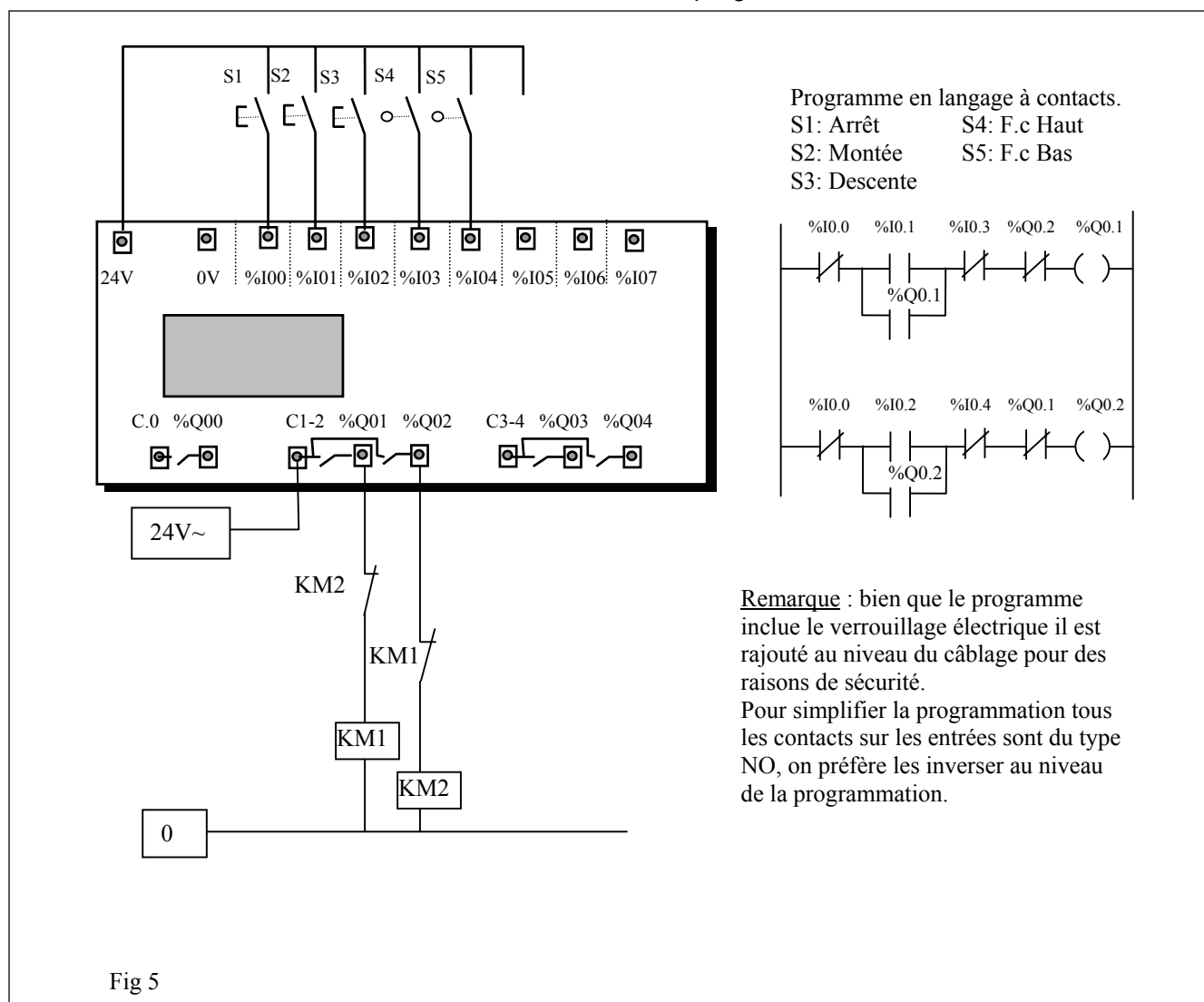
La capacité d'une mémoire se donne en mots de 8 bits (**binary digits**)
capacité d'une mémoire une mémoire qui contient 8 kilos octet contient
 $8 * 1024 \text{ octets} = 8194 \text{ octets}$.

3.4. Interfaces d'entrées / sorties.

- Les entrées reçoivent les informations depuis les capteurs ou boutons poussoirs du système. Elles transitent par un organe d'isolation galvanique pour aller vers le microprocesseur (ex : opto-coupleur). Elles sont scrutées autant de fois qu'il est nécessaire, au moins une fois par cycle
- Les sorties reçoivent les informations dictées par le microprocesseur et stocké en mémoire elles sont rafraîchies au fur et à mesure du déroulement du programme.

4. Raccordement d'un automate programmable.

Réalisation d'une montée descente avec un automate programmable.



LA PROGRAMMATION.

Langage à contacts (LD)

Structure d'un programme (section, SR ou tâche événementielle)

Un programme en langage à contacts est composé d'une suite de réseaux de contacts exécutée de façon séquentielle par l'automate.

Chaque réseau de contacts peut être :- Repéré par une étiquette.

- Complété par un commentaire de 222 caractères.

Un réseau de contacts est constitué de 7 lignes sur TSX Micro et de 16 lignes sur TSX Premium, de 11 colonnes soit au maximum 10 contacts et une bobine par ligne.

Editeur de programmes : langage à contacts

L'éditeur du langage à contacts offre de nombreux outils assurant la construction des réseaux de contacts de façon conviviale : Une palette d'éléments graphiques permet d'accéder directement, par la souris ou à l'aide du clavier, aux différents symboles graphiques du langage : contacts, fil booléen, bobines, blocs opérations, blocs fonctions prédéfinis...



Le dessin du réseau peut être réalisé sans avoir à renseigner chaque élément.

Les objets du langage peuvent être indifféremment saisis et visualisés sous forme symbolique ou repère.

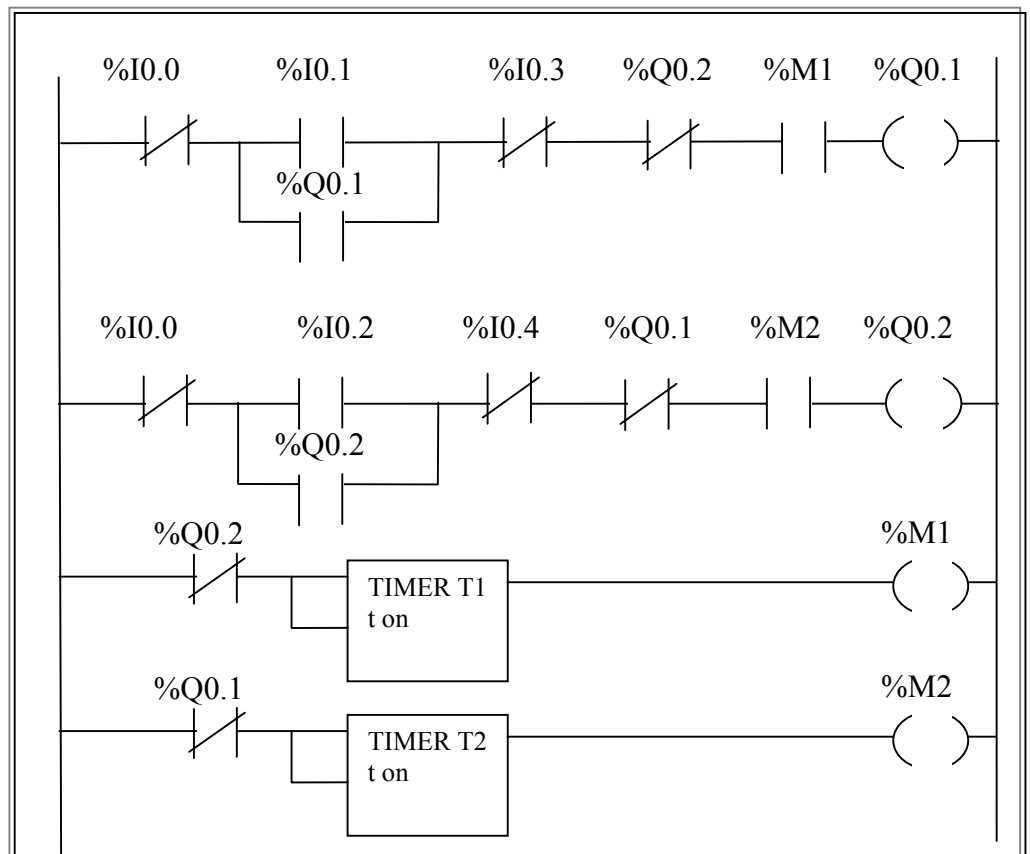
Le symbole et le repère de chaque objet peuvent être visualisés simultanément.

La construction du réseau de contacts s'effectue simplement en sélectionnant le symbole dans la palette graphique et en le plaçant à l'endroit voulu dans la grille présentée à l'écran.

L'éditeur du langage à contacts permet l'appel immédiat à des fonctions d'aide à la saisie : Accès aux bibliothèques de fonctions. Accès à l'éditeur de variables. Couper, copier, coller.

(Fig. 6)
Exemple de programmation d'un "marche avant marche arrière" en langage à contact.

La programmation s'effectue comme le schéma électrique en prenant soins de donner les bonnes affectations aux différentes variables. Les %M sont des variables internes, et les timers des temporisations



Langage Grafcet (SFC)

Le langage Grafcet permet de décrire de manière simple et graphique la partie séquentielle d'automatismes.

Il correspond au langage "Diagramme fonctionnel en séquence" SFC décrit dans la norme IEC 1131-3.

Structure de la section dans la tâche maître.

Le langage Grafcet SFC s'utilise uniquement dans une section de la tâche maître. Celle-ci est alors structurée en trois traitements (Traitement : préliminaire, chart ou graphique, postérieur)

Les programmes écrits en langage Grafcet SFC se composent :

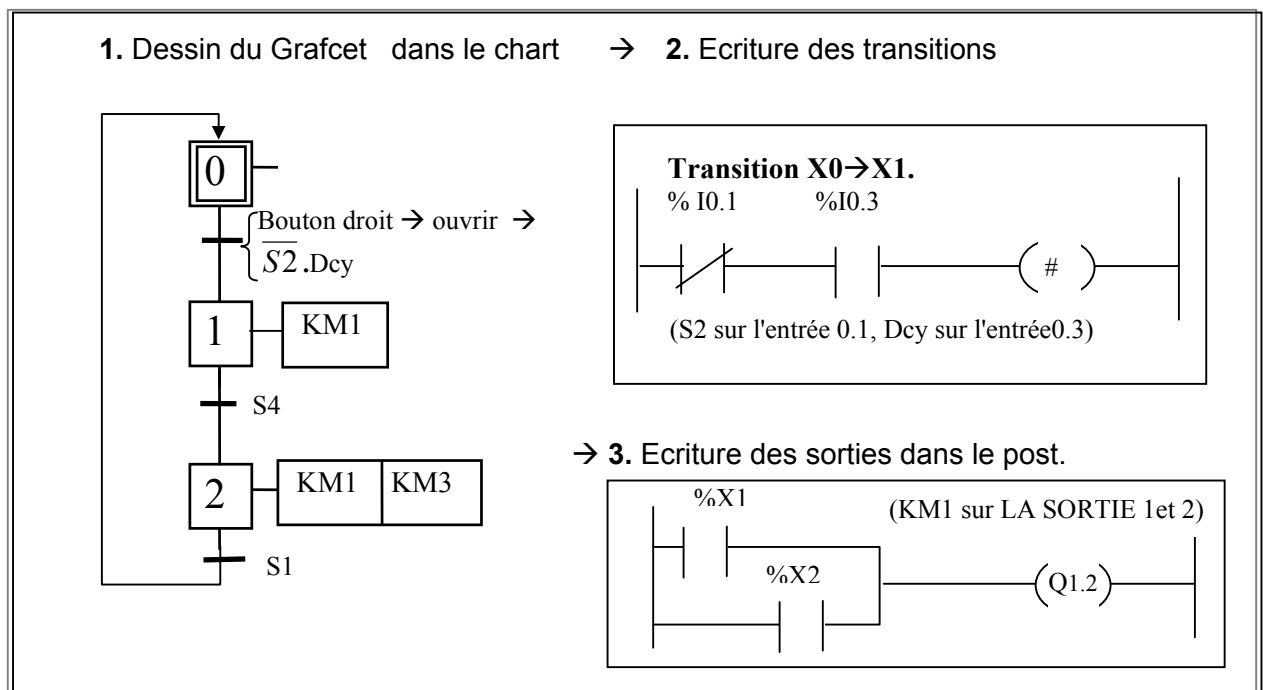
- Des macro-étapes (1) qui sont la représentation unique d'un ensemble d'étapes et de transition.
- Des étapes auxquelles sont associées les actions à effectuer.
- Des transitions auxquelles sont associées les conditions (réceptivités).
- Des liaisons orientées reliant les étapes et les transitions.

Les actions (continues, impulsionnelles à l'activation ou à la désactivation) et les réceptivités sont programmables en langage soit à contacts, soit en littéral, soit en liste d'instructions.

Editeur de programmes : langage Grafcet SFC.

- L'éditeur graphique propose 8 pages composées de 11 colonnes de 14 lignes, soit 154 cellules par page.
- Une palette d'objets graphiques permet l'accès direct à chaque symbole graphique (macro-étapes, étapes, transitions, aiguillages, activations / désactivations simultanées et renvois).
- La programmation des réceptivités et des actions s'effectue simplement en cliquant sur l'élément du graphe désiré.
- Dans une page Grafcet, des commentaires de 64 caractères maximum peuvent être saisis dans n'importe quelle cellule.
- Des aides à la saisie, telles que : couper, copier, coller...sont mises à la disposition de l'utilisateur.

EXEMPLE DE PROGRAMMATION : (Fig. 7)



Editeur de variables.

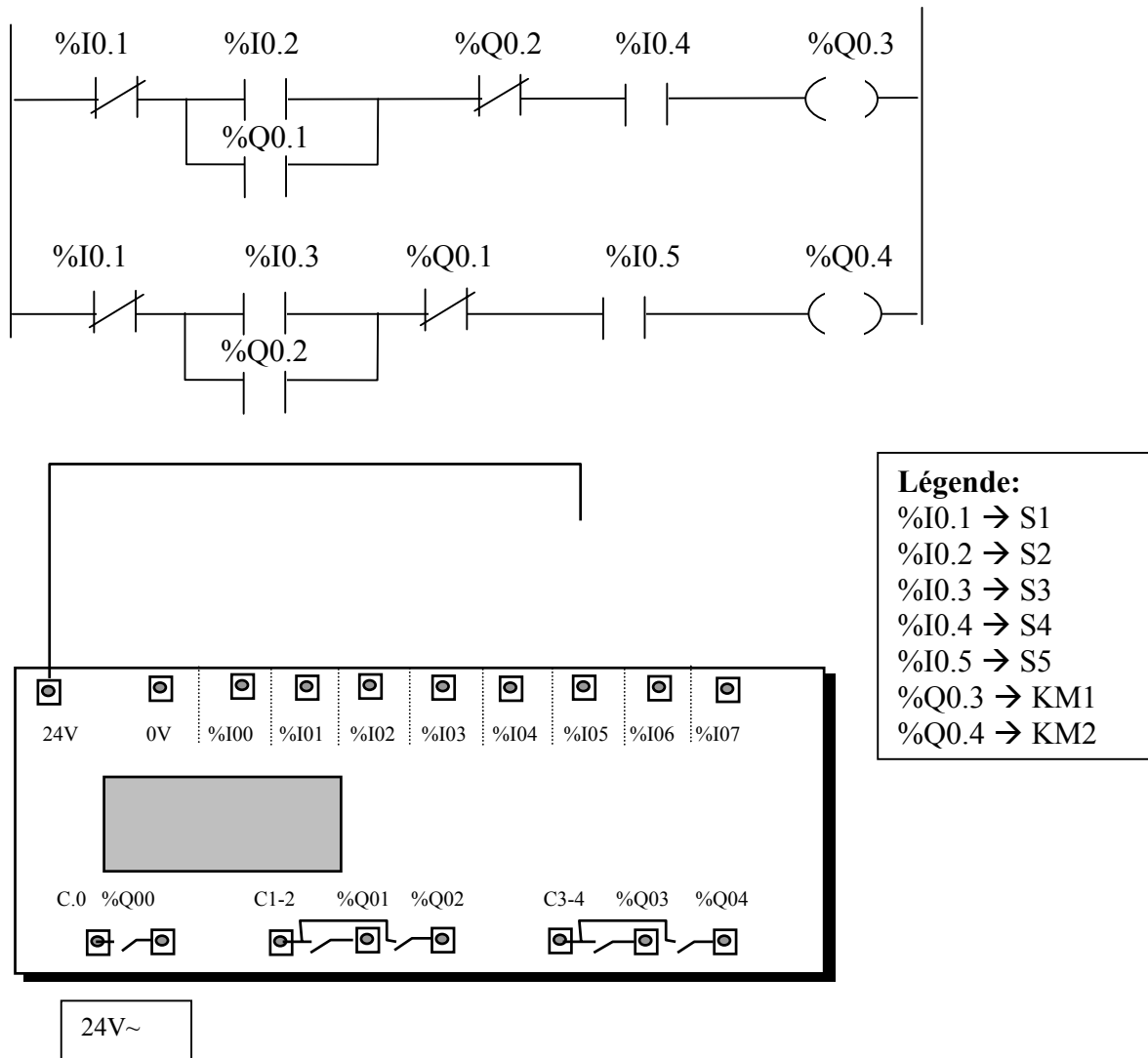
L'éditeur de variables permet de :

- Symboliser les différents objets de l'application (bits, mots, blocs fonctions, entrées/sorties...).
- Paramétrer les blocs fonctions prédéfinis (temporisateurs, compteurs, registres...).
- Saisir les valeurs des constantes avec choix de la base d'affichage (décimal, binaire, hexadécimal, flottant, message).
- Paramétrer les blocs fonctions utilisateur DFB.

Chaque symbole (32 caractères maximum, caractères accentués autorisés) peut être accompagné d'un commentaire (508 caractères maximum).

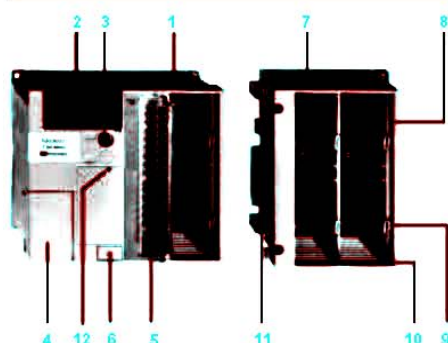
Des fonctions de recherche par famille et par type de tri facilitent l'accès aux objets.

Exercice: sur la base du programme ci desous en vous aidant de la figure (5) câbler les entrées et les sorties de l'automate, décoder le programme en schéma traditionnel.



TSX 37-10

Présentation, description, choix



Présentation

Les automates TSX 37-10 compacts et modulaires se différencient par leur tension d'alimentation et le type de module d'entrées/sorties "Tout ou Rien" implanté de base dans le premier emplacement.

Chaque configuration TSX 37-10 comprend un bac intégrant une alimentation (± 24 V ou $\sim 100/240$ V), un processeur incluant une mémoire RAM de 14 K mots (programme, données et constantes), une mémoire de sauvegarde Flash EPROM, un module d'entrées/sorties "Tout ou Rien" (28 ou 64 E/S) et un emplacement disponible.

Un mini bac d'extension TSX RKZ 02 permet d'augmenter le nombre d'emplacements de 2.

Chaque emplacement disponible peut recevoir :

- 1 module d'entrées/sorties TOR au format standard de tout type
- 2 modules demi-format de type entrées/sorties TOR, sécurisé, entrées/sorties analogiques et comptage.

Description

Les automates TSX 37-10 et le mini bac d'extension TSX RKZ 02 comprennent :

- 1 un bac de base à 2 emplacements
- 2 un bloc de visualisation centralisée
- 3 une prise terminal repérée TER
- 4 une trappe d'accès aux bornes d'alimentation
- 5 un module 28 ou 64 entrées/sorties "Tout ou Rien" positionné dans le premier emplacement (positions 1 et 2)
- 6 une trappe d'accès à la pile optionnelle
- 7 un mini bac d'extension à 2 emplacements disponibles (positions 5 à 8)
- 8 un voyant de présence de tension ± 24 V
- 9 des bornes d'alimentation protégées par un cache amovible, pour le raccordement d'une alimentation auxiliaire ± 24 V dans le cas des automates alimentés en $\sim 100/240$ V
- 10 une borne de masse
- 11 des connecteurs de raccordement à l'automate de base
- 12 un bouton de réinitialisation.

Choix

Choix des bases automates TSX 37-10

alimentation	module d'E/S TOR intégré dans le 1 ^{er} emplacement				raccordement		référence
	nombre d'entrées	nombre d'entrées	nombre de sorties	statiques	connecteur	bornier	
± 24 V	16	16	12	relais			TSX 37 10 128DT1
	16	16	12				TSX 37 10 128DTK1
	32	16	32				TSX 37 10 128DR1
	32	16	32				TSX 37 10 164DTK1
$\sim 110/240$ V	16	16	12				TSX 37 10 028AR1
	16	16	12				TSX 37 10 028DR1

Choix possible

Choix des modules à implanter

(3 emplacements disponibles soit 6 modules maximum)

type de module à implanter	nombre de modules maximum (1)				raccordement	
	1	2	4	6	format	bornier
entrées/sorties "Tout ou Rien"					stand	
8 E						
12 E						
32 E			(2)			
4 S						
8 S						
32 S			(2)			
16 E/S						
28 E/S			(2)			
64 E/S		(2)				
module de sécurité Preventa						
bus AS-I ou extension d'E/S	(3)					
entrées/sorties analogiques						
4 E et 8 E						
2 S et 4 S						
voies de comptage						
1 voie						
2 voies						

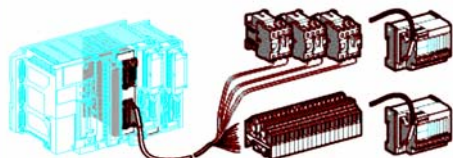
Possibilité d'implantation

(1) Avec mini bac d'extension TSX RKZ 02

(2) Y compris le module format standard positionné dans le 1^{er} emplacement de l'automate

(3) Les modules extension d'entrées/sorties TOR à distance et le coupleur bus AS-I s'intègrent à la position 4 ce qui rend leur utilisation exclusive

Caractéristiques : pages B60 et B61
Références : pages B62 et B63
Encombrements, montage : page B64

Raccordement des entrées/sorties sur connecteurs type HE 10 avec torons de fils prééquipés (section 0,324 mm²), nappes (section 0,08 mm²) ou câbles (section 0,324 mm²).				
				
16 entrées/12 sorties entrées $\sim$ 24 V sorties relais 3 A (lth)	entrées $\sim$ 100...120 V sorties relais 3 A (lth)	entrées $\sim$ 24 V sorties $\sim$ 24 V/0,5 A	entrées $\sim$ 24 V sorties $\sim$ 24 V/0,5 A	32 entrées/32 sorties entrées $\sim$ 24 V sorties $\sim$ 24 V/0,1 A
			raccordement par connecteur HE 10, 20 contacts	
			8, 12 ou 16 voies, avec ou sans DEL, avec commun ou 2 bornes par voie	
			entrées : 16 voies $\sim$ 5 V TTL, $\sim$ 24 V, $\sim$ 48 V, $\sim$ 48 V, $\sim$ 115 V ou 230 V, 2 bornes par voie	
			sorties : 8 ou 16 voies à relais 1 "F", 1 ou 2 "OF" ou statiques, $\sim$ 5...48V, $\sim$ 24 V, $\sim$ 24...240 V 1 ou 2 bornes par voie	
surveillance tension capteurs	surveillance tension capteurs	surveillance tension capteurs		
type 1	type 2	type 1		
positive/Négative		positive		
$\sim$ 2 fils,	$\sim$ 2 fils, $\sim$ 2 fils	$\sim$ 2 fils,		
$\sim$ 3 fils PNP/NPN		$\sim$ 3 fils PNP		
repli configurable des sorties		surveillance tension préactionneurs		
oui		repli configurable des sorties		
non protégées		protégées		
		positive		
TSX DMZ 28 DR	TSX DMZ 28 AR	TSX DMZ 28 DT	TSX DMZ 28 DTK	TSX DMZ 64 DTK
B76				

FICHE DE TRAVAUX PRATIQUES :BEP		Repère : 5.2
Tache attribuée pour les travaux pratiques de : automatismes automate programable		
Système:	Lieu d'activité : salle de classe	
Définition des tâches confiées à l'élève à l'occasion de la séquence de travaux pratiques.		Enoncé des objectifs de formation associées aux tâches définies.
<u>PREREQUIS:</u> Connaissance des équations logiques, de la fonction mémoire et des relais bistables. <u>EN AYANT A VOTRE DISPOSITION:</u> Des descriptifs de fonctionnement de machines. Le dossier technique des machines. <u>ON VOUS DEMANDE:</u> De réaliser le schéma de raccordement d'un automate De réaliser les schémas de commande. De vérifier le fonctionnement d'un système en se servant des indications automate. <u>CRITERES D'EVALUATION:</u> Graficets exacts. Schéma de commande exacts. Notice de fonctionnement exacte.		<u>Fonctions:</u> <u>ETUDE:</u> exploiter les documents relatif à un Système <u>Maintenance.</u> Diagnostiquer à l'aide d'un raisonnement logique une cause probable de défaillance
		<u>Domaines:</u> S5 Commande de systèmes. S 5.2. automatisme
		<u>Connaissances:</u>
		<u>Etre capable de:</u> Analyser le fonctionnement d'un système simple. Exploiter les outils pour traduire les caractéristiques fonctionnelles d'un système décoder un programme <u>Compétences terminales:</u> C1. S'informer C12 interpréter. C3. Intervenir C33 maintenir en état.
<u>Evaluation proposée par le professeur responsable:</u>	<u>Note:</u>	<u>Prévu:</u> <u>Temps:</u> <u>Passé:</u>
		<u>Nom de l'élève:</u>