

Lycée Technique Mohammedia

AUTOMATE PROGRAMMABLE INDUSTRIEL

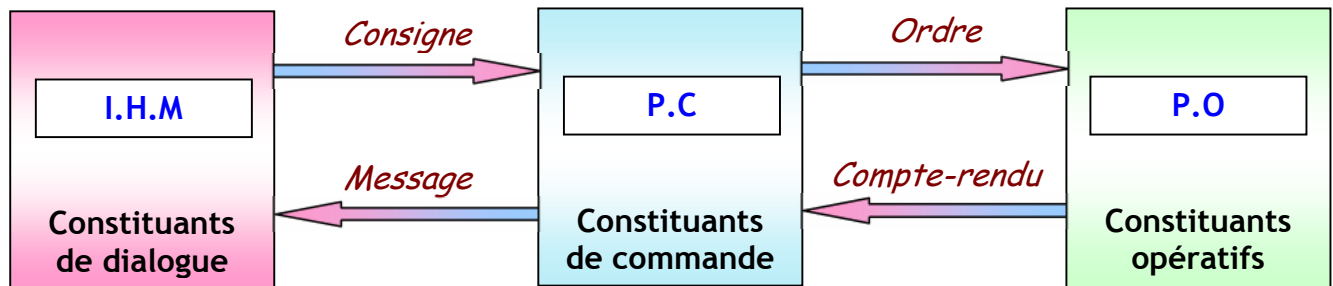
1^{ère} STE **Unité ATC**

Professeur : **MAHBAB**

1. Structure d'un système automatisé :

Un système automatisé se compose de deux parties qui coopèrent:

- ❖ une partie opérative constituée du processus à commander, des actionneurs qui agissent sur ce processus et des capteurs permettant de mesurer son état.
- ❖ une partie commande qui élabore les ordres pour les actionneurs en fonction des informations issues des capteurs et des consignes. Cette partie commande peut être réalisée par des circuits câblés, ou par des dispositifs programmables (automates, calculateurs)



2. Cahier des charges d'un automatisme logique :

Le cahier des charges décrit:

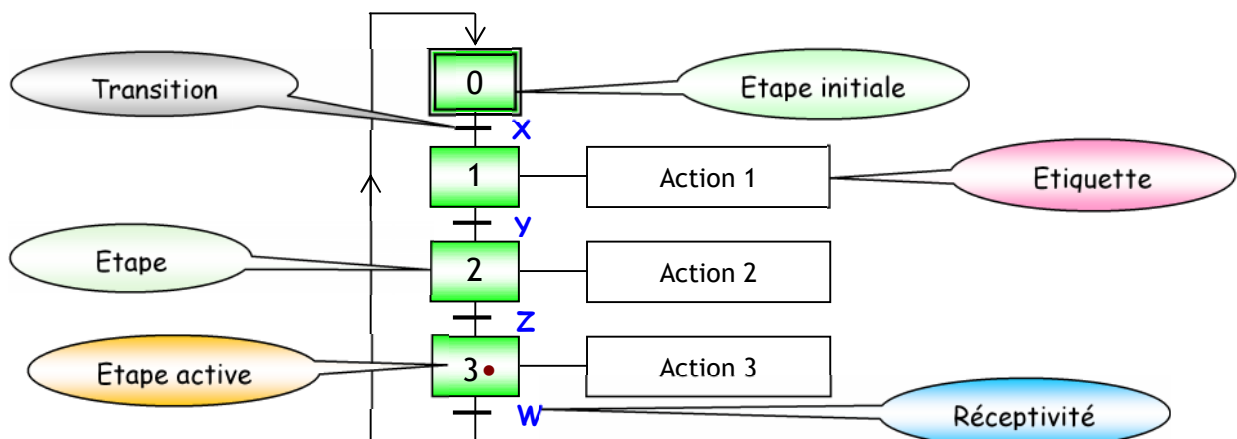
- ❖ les relations entre la partie commande et la partie opérative.
- ❖ les conditions d'utilisation et de fonctionnement de l'automatisme.

Le fonctionnement d'un automatisme séquentiel peut être décomposé en un certain nombre d'étapes. Le passage (ou transition) d'une étape à une autre étape se fait à l'arrivée d'un évènement particulier (réceptivité) auquel le système est réceptif.

3. GRAFCET :

3.1. Définition :

- ❖ Le GRAFCET (Graphe de Contrôle Etape-Transition) est un outil graphique normalisé permettant de spécifier le cahier des charges d'un automatisme séquentiel.
- ❖ Le GRAFCET est une représentation graphique alternée d'étapes et de transitions. Une seule transition doit séparer deux étapes.
- ❖ L'étape caractérise un comportement invariant du système technique au moment donnée, elle peut être active ou inactive.
- ❖ L'action associée à une étape quelconque caractérise ce que doit faire le système lorsque cette dernière est active.
- ❖ Les actions associées aux étapes sont inscrites dans les étiquettes.
- ❖ La transition indique le critère d'évolution entre deux étapes consécutives.
- ❖ Une réceptivité est une condition logique qui conditionne la transition d'une étape à la suivante.
- ❖ l'étape initiale est représentée par un double carreau.
- ❖ Les étapes qui se succèdent sont reliées par des segments orientés auxquels sont associées des transitions (trais horizontaux coupant les segments orientés).



3.2. Règles de syntaxe :

Règle N° 1 : situation initiale

L'étape initiale est considérée initialement active ce qui permet l'évolution de l'automatisme dès que la réceptivité associée à la première transition devient vraie.

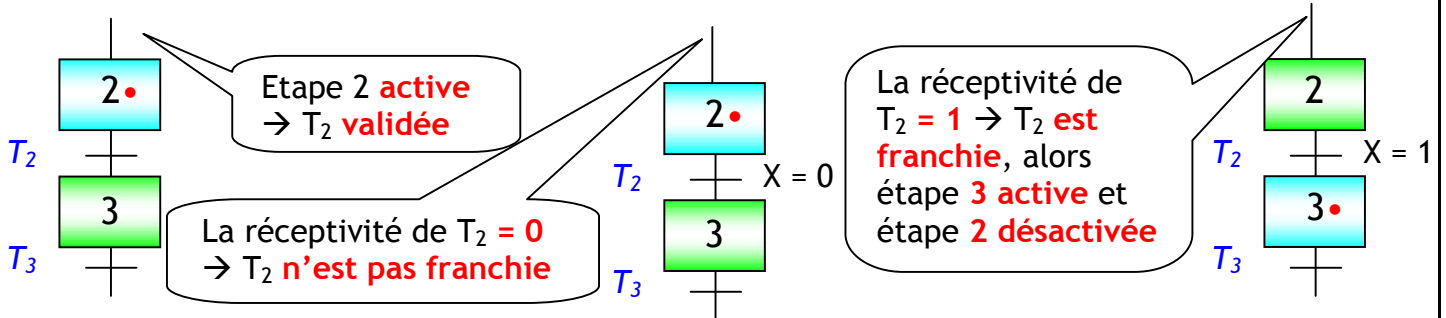
Règle N° 2 : franchissement d'une transition

Une transition est franchie lorsque l'étape associée est active et la réceptivité associée à cette transition est vraie.

Règle N° 3 : évolution des étapes actives

Le franchissement d'une transition provoque simultanément :

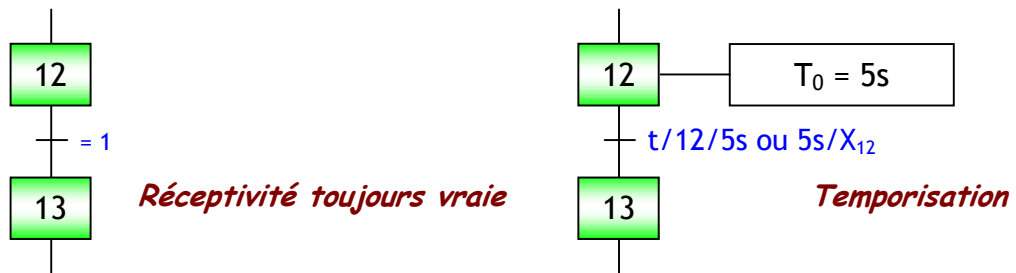
- ❖ la désactivation de toutes les étapes immédiatement précédentes reliées à cette transition,
- ❖ l'activation de toutes les étapes immédiatement suivantes reliées à cette transition.



Cas particuliers : Il y a des cas particuliers de réceptivité, on en cite 2 :

❖ **Temporisation :** Pour faire intervenir le temps dans une réceptivité, il suffit d'indiquer après le repère "t" son origine et sa durée. L'origine sera l'instant de début de l'activation de l'étape déclenchant la temporisation. La notation **t/12/5s** signifie que la réceptivité sera vraie 5 secondes après l'activation de l'étape repérée 12. La notation normalisée s'écrit **5s/X12**.

❖ **Réceptivité toujours vraie :** une telle réceptivité s'écrit "**= 1**". Le franchissement de cette transition se fera dès que la ou les étapes immédiatement antérieures seront actives sans autre condition.

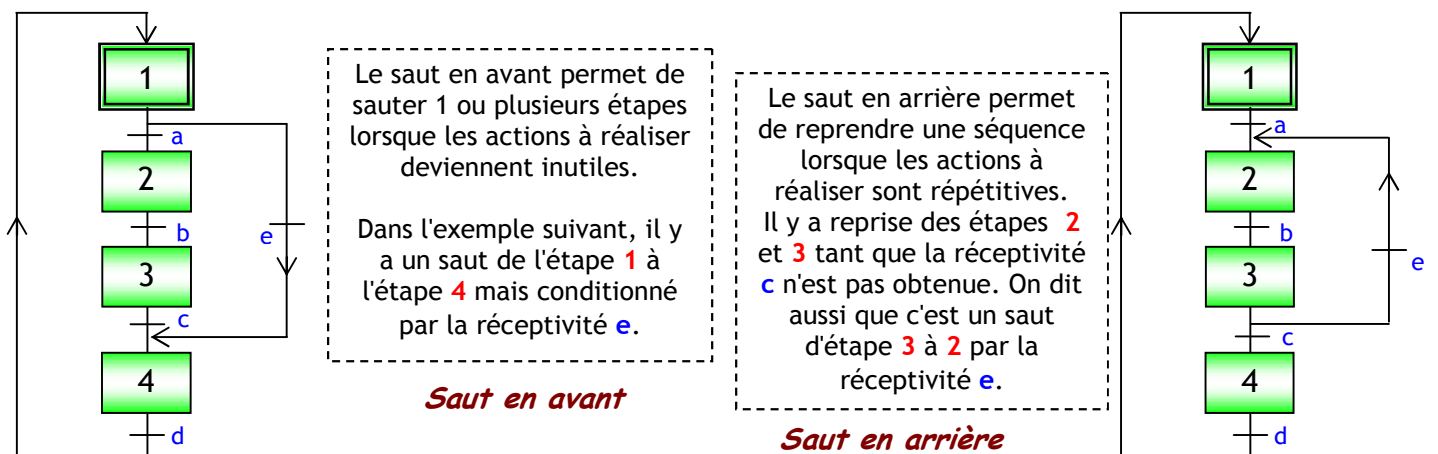


4. Structure de base d'un GRAFCET :

4.1. La séquence linéaire :

Une séquence linéaire est composée d'un ensemble d'étapes successives où chaque étape est suivie d'une seule transition et chaque transition n'est validée que par une seule étape.

4.2. Saut en avant et saut en arrière (saut de phase et reprise de phase) :



4.3. Les séquences simultanées :

Lorsque le franchissement d'une transition conduit à activer simultanément plusieurs séquences d'étapes, on obtient des séquences simultanées qui s'exécuteront parallèlement mais indépendamment. C'est-à-dire, l'évolution de chacune des séquences d'étapes dépendra des conditions d'évolution du système automatisé.

Divergence en ET : lorsque la transition T_A est franchie, les étapes 1 et 4 sont actives.

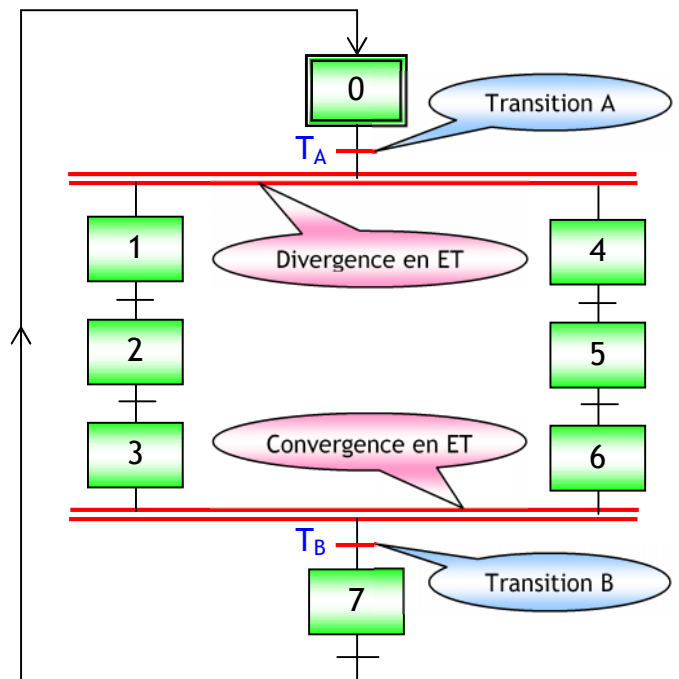
Convergence en ET : la transition T_B sera validée lorsque les étapes 3 et 6 seront actives.

Si la réceptivité associée à cette transition est vraie, alors celle-ci est franchie.

REMARQUE :

- ❖ Après une divergence en ET, on trouve une convergence en ET.
- ❖ Le nombre de branches parallèles peut-être supérieur à 2.

La réceptivité associée à la convergence peut-être de la forme = 1. Dans ce cas la transition est franchie dès qu'elle est active.



4.4. Sélection de séquences :

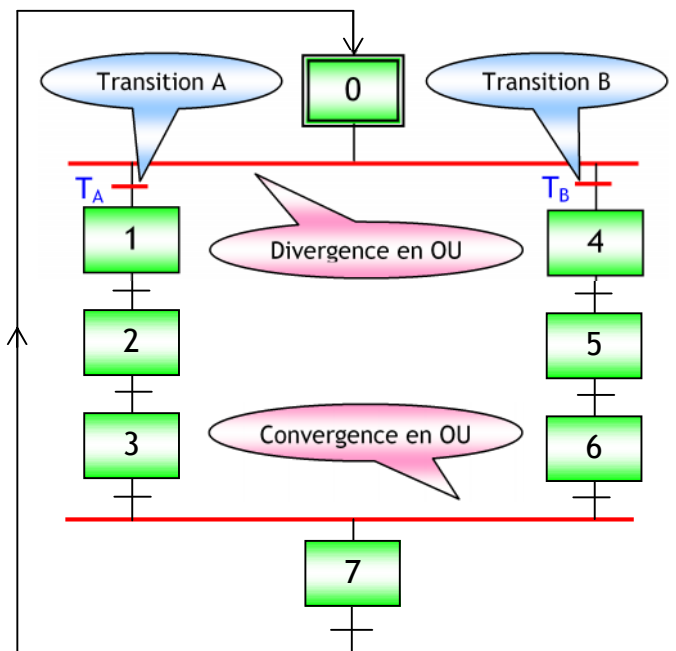
Une structure alternative permet d'effectuer un choix unique d'évolution entre plusieurs étapes en aval à partir d'une seule étape en amont.

Divergence en OU : l'évolution du système vers une branche dépend des réceptivités des transitions T_A et T_B associées aux transitions.

Convergence en OU : après l'évolution dans une branche, il y a convergence vers une étape commune.

REMARQUE :

- ❖ A et B ne peuvent être vrais simultanément (conflit).
- ❖ Après une divergence en OU, on trouve une convergence en OU.
- ❖ Le nombre de branches peut-être supérieur à 2.
- ❖ La convergence de toutes les branches ne se fait pas obligatoirement au même endroit.



5. Différents points de vue d'un GRAFCET :

Suivant les différents points de vue (utilisateur, technico-commercial, concepteur-réalisateur, etc.), on peut distinguer plusieurs 3 types de GRAFCET :

- GRAFCET d'un point de vue du système ;
- GRAFCET d'un point de vue de la partie opérative (P.O) ;
- GRAFCET d'un point de vue de la partie commande (P.C).

NB :

Le GRAFCET d'un point de vue de la P.O et le GRAFCET d'un point de vue de la P.C doivent avoir le même nombre d'étapes.

5.1. Exemple 'Poste de perçage automatique' :

On se propose d'étudier le système automatisé 'Poste de perçage' suivant, en décrivant son fonctionnement par l'outil GRAFCET.

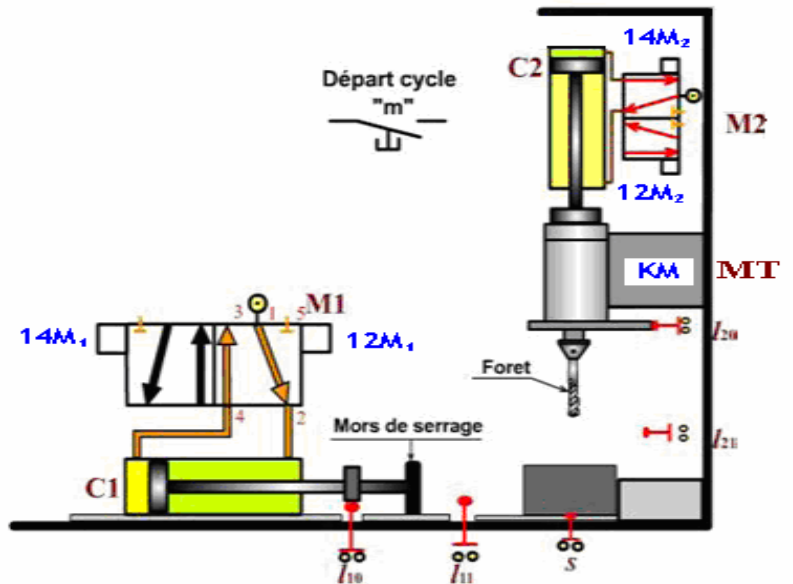
Fonctionnement :

L'appui sur le bouton Départ cycle (m) lance le cycle suivant :

- Serrage de la pièce à percer ;
- Perçage de la pièce ;
- Desserrage de la pièce.

Remarque :

Le cycle ne peut être lancé que si la pièce à percer est présente.



Fonctionnement en tenant compte des solutions technologiques utilisées :

- Le capteur (s) indique la présence de la pièce à percer ;
- L'appui sur le bouton Départ cycle (m) lance le cycle ;
- Le vérin de serrage (C₁) déplace la pièce pour la serrer; le capteur (l₁₁) indique que la pièce est serrée ;
- Le moteur supportant le foret (M_T) commence à tourner et le vérin (C₂) pousse le moteur vers le bas ;
- Le perçage de la pièce commence et le capteur (l₂₁) indique que la pièce est percée ;
- Alors le vérin (C₂) remonte ; quand le capteur (l₂₀) est actionné, cela indique que le foret est retourné ;
- Le moteur (M_T) et le vérin (C₂) sont arrêtés ;
- Le vérin (C₁) retourne dans l'autre sens ; le capteur (l₁₀) indique que la pièce est desserrée ;
- On revient alors à l'état initial.

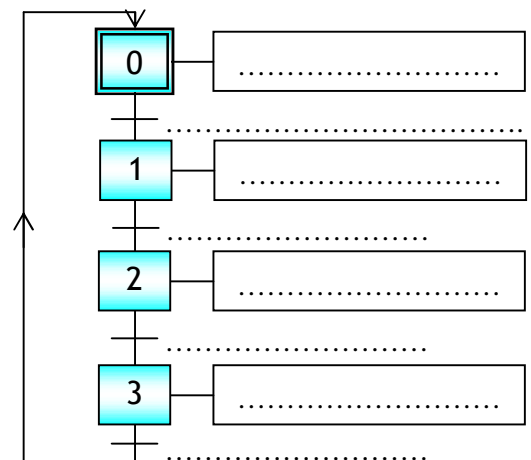
5.2. GRAFCET d'un point de vue du système :

Le GRAFCET point de vue système décrit le fonctionnement global du système, en traduisant le cahier des charges sans tenir compte de la technologie adoptée ; donc il permet de dialoguer avec des personnes non spécialistes. On le désigne aussi par "GRAFCET fonctionnel".

1. Compléter le tableau suivant :

| N° de la tâche | Description de la tâche | passage à la tâche suivante si |
|----------------|-------------------------|--------------------------------|
| 0 | | |
| 1 | | |
| 2 | | |
| 3 | | |

2. Compléter le GRAFCET du point de vue du système 'Poste de perçage'



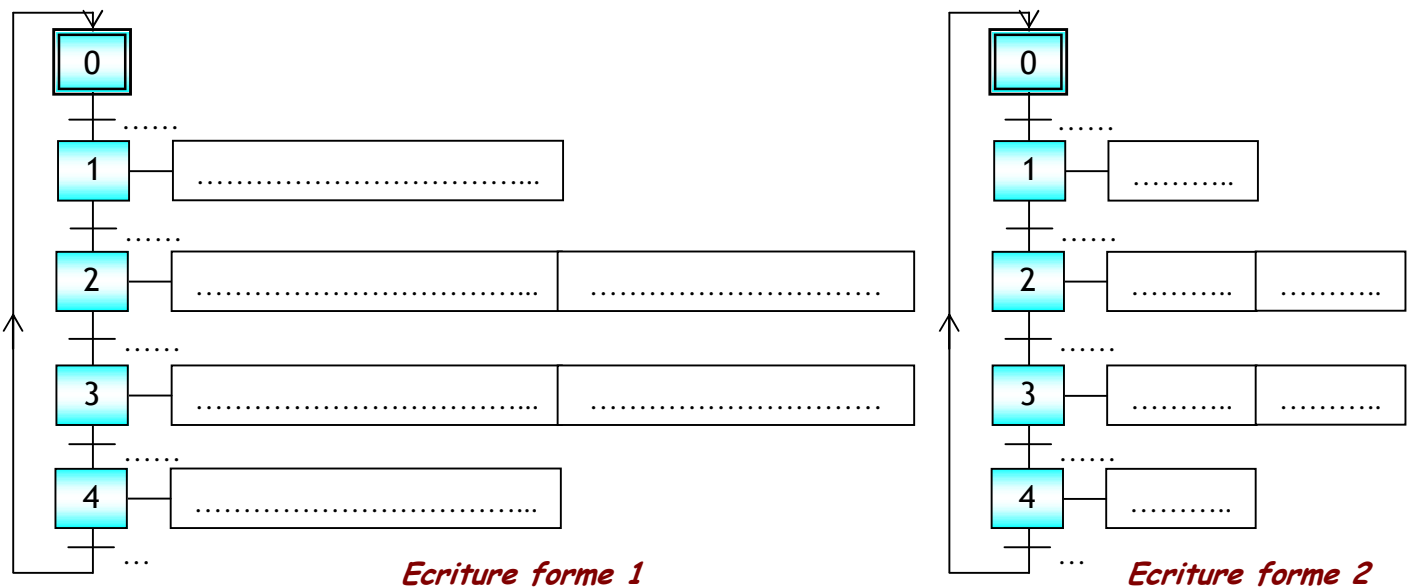
5.3. GRAFCET d'un point de vue de la partie opérative (P.O) :

Le GRAFCET d'un point de vue de la partie opérative décrit en terme clair ou par symbole les actions et les réceptivités, en tenant compte de la technologie choisie pour les éléments de la partie opérative ainsi que le type d'informations reçues (ordres) ou envoyées (comptes rendus).

1. Compléter le tableau suivant :

| N° de la tâche | Description de la tâche | Action donnée par les actionneurs | Événement de la fin de l'action |
|----------------|-------------------------|-----------------------------------|---------------------------------|
| 0 | Attendre | | |
| 1 | Serrer la pièce | | |
| 2 | Percer la pièce | | |
| | | | |
| | | | |
| | | | |
| 3 | Desserrer la pièce | | |

2. Compléter le GRAFCET du point de vue P.O 'Poste de perçage'

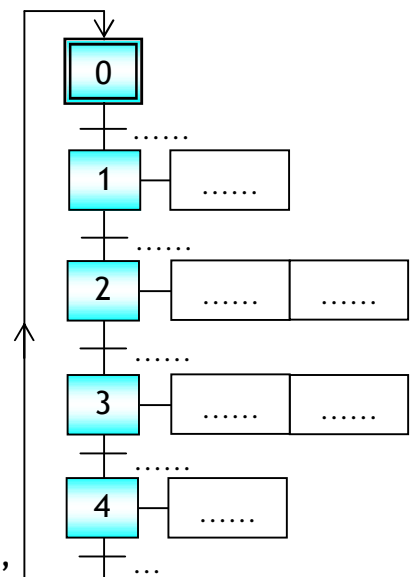


5.4. GRAFCET d'un point de vue de la partie commande (P.C) :

Le GRAFCET d'un point de vue de la partie commande décrit les échanges de la partie commande avec la partie opérative et le dialogue avec l'opérateur, en tenant compte des choix technologiques des préactionneurs et des capteurs.

1. Compléter le tableau suivant :

| Tâche N° | Description de la tâche | Préactionneurs | Actions des Préactionneurs | Capteurs et boutons |
|----------|----------------------------|----------------|----------------------------|---------------------|
| 0 | Attendre | | | |
| 1 | Avancer le mors de serrage | | | |
| 2 | Faire tourner le foret | | | |
| | Descendre le foret | | | |
| 3 | Faire tourner le foret | | | |
| | Monter le foret | | | |
| 4 | Reculer le mors de serrage | | | |



2. Compléter le GRAFCET du point de vue P.C 'Poste de perçage'

1. Définition :

Un capteur est un dispositif qui transforme une grandeur physique (lumière , température , pression ..) à qui il est soumis , en image électrique (tension , courant , impédance..).Si la grandeur physique est désignée par m comme mesurande, l'image électrique est désignée par S , on aura : $S = f(m)$.

- ❖ S : Grandeur de sortie ou réponse du capteur.
- ❖ m : Grandeur d'entrée ou mesurande.



2. Réponse d'un capteur :

La mesure de S doit permettre de connaître la valeur de m . La relation $S=f(m)$ résulte des lois physique qui régissent le capteur. Pour faciliter l'exploitation on s'efforce de réaliser le capteur ou l'utiliser en sortie qu'il établit une relation linéaire entre S de la grandeur de sortie et m de le mesurande : $\Delta S = s \Delta m$ s : sensibilité du capteur.

3. Différents type de capteurs :

Le capteur se présente vu de sa sortie :

- ❖ Soit comme un **générateur**, s étant alors une charge, une tension ou un courant, le capteur est dit alors capteur **actif**.
- ❖ soit comme une **impédance**, s étant alors une résistance, une inductance ou une capacité, le capteur est dit alors capteur **passif**.

Il existe différents types de sortie de capteur :

- ❖ Capteur tout ou rien : la sortie présente un niveau bas et un niveau haut.
- ❖ Capteur analogique : les informations acquises par le capteur sont délivrées sous forme analogique.
- ❖ Capteur numérique : les informations acquises par le capteur sont délivrées sous formes numérique et peuvent être traités directement par un calculateur.

4. Caractéristiques d'un capteur :

Certains paramètres sont communs à tous les capteurs. Ils caractérisent les contraintes de mise en œuvre et permettent le choix d'un capteur :

L'étendue de la mesure : c'est la différence entre le plus petit signal détecté et le plus grand perceptible sans risque de destruction pour le capteur.

La sensibilité : ce paramètre caractérise la capacité du capteur à détecter la plus petite variation de la grandeur à mesurer. C'est le rapport entre la variation ΔS du signal électrique de sortie pour une variation donnée Δm de la grandeur physique d'entrée : $S = \Delta S / \Delta m$

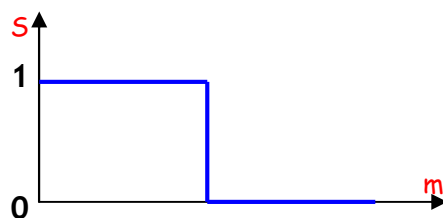
La fidélité : Un capteur est dit fidèle si le signal qu'il délivre en sortie ne varie pas dans le temps pour une série de mesures concernant la même valeur de la grandeur physique d'entrée. Il caractérise l'influence du vieillissement.

Le temps de réponse : c'est le temps de réaction d'un capteur entre la variation de la grandeur physique qu'il mesure et l'instant où l'information est prise en compte par la partie commande.

Linéarité : Le capteur est linéaire dans une plage déterminée du mesurande, si sa sensibilité est indépendante de la valeur du mesurande, alors le signal électrique tout le long de la chaîne de mesure est proportionnel à la variation du mesurande.

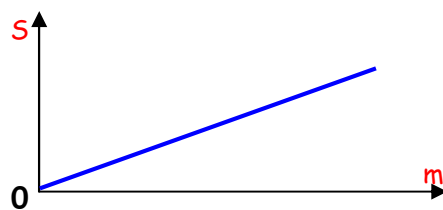
5. Différents type de signaux délivrés par un capteur :

5.1. Le signal logique :



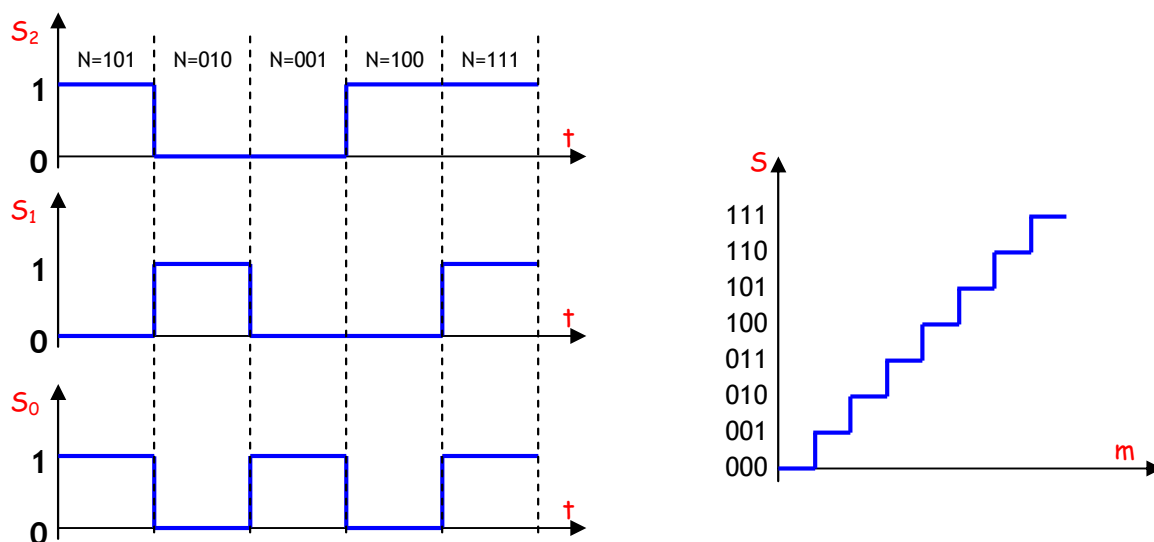
Ce signal ne peut prendre que deux valeurs binaires « 1 » ou « 0 » (vrai ou faux, présent ou absent). Il est appelé tout ou rien (T.O.R).

5.2. Le signal analogique :



Ce signal électrique est proportionnel au phénomène physique mesuré. L'utilisation d'un capteur analogique n'est pas possible avec des systèmes numériques.

5.3. Le signal numérique :



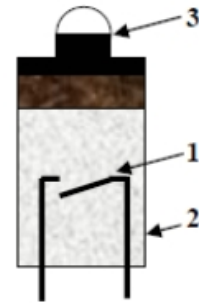
Signal numérique sur 3 bits

Ce signal est une combinaison d'état logique. A chaque combinaison, correspond une valeur en tension. Le capteur numérique présente l'avantage d'être utilisable par des systèmes numériques. La précision obtenue dépend de la résolution du capteur.

1. Détecteur (ou interrupteur) de position à action mécanique :

1.1. Structure :

- Un contact électrique (1)
 - Un corps (2)
 - Une tête de commande avec son dispositif d'attaque (3)
- Ce capteur est électriquement équivalent à un contact, avec un dispositif de commande.

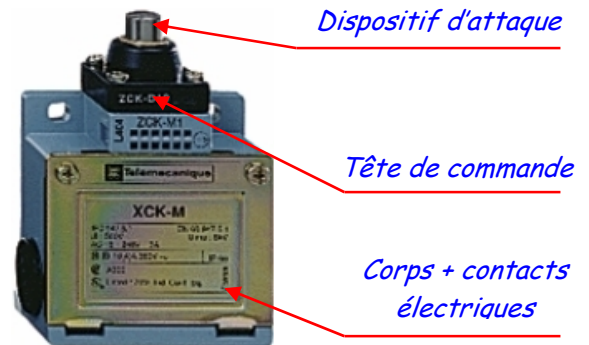


1.2. Fonctionnement :

La détection de présence est réalisée lorsque l'objet à détecter entre en contact avec la tête de commande au niveau de son dispositif d'attaque.

Le mouvement engendré sur la tête d'attaque provoque la fermeture du contact électrique situé dans le corps du capteur.

De multiples têtes sont disponibles :



à poussoir



à galet



à levier galet



à tige souple

1.3. Caractéristiques :

Symbole:



Portée nominale :

Contact direct

Tension d'alimentation :

240 V AC; 250 V DC max

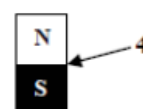
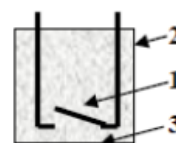
1.4. Applications :

Ils sont utilisables pour détecter des matériaux rigides, ils servent essentiellement à détecter des présences ou des passages. Exemples : Fin de course, présence pièce sur support d'usinage; passage d'un vérin ou d'un chariot sur un rail.

2. Détecteur magnétique ou interrupteur à lame souple (I.L.S) :

2.1. Structure :

- Un contact électrique (1)
- Un corps (2)
- Face sensible du capteur (3)
- Objet magnétique à détecter (4)



2.2. Fonctionnement :

Lorsqu' un champ magnétique (4) est dirigé sur la face sensible (3) du capteur, le contact s'établit entre les deux bornes du capteur.

2.3. Applications :

Ce type de détecteurs est souvent monté directement sur le corps de vérins en tant que fin de course (dans ce type de montage, le piston du vérin est magnétisé)



2.4. Caractéristiques :

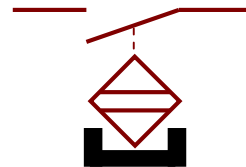
Portée nominale:

Dépend de l'amplitude du champ magnétique de l'objet à détecter

Tension d'alimentation:

10 à 30 V DC

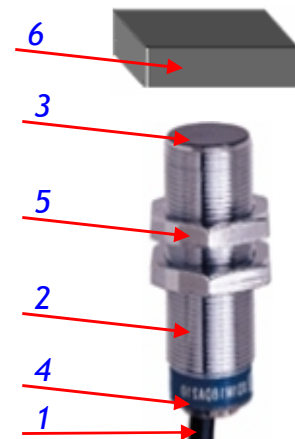
Symbole:



3. Détecteur de proximité inductif :

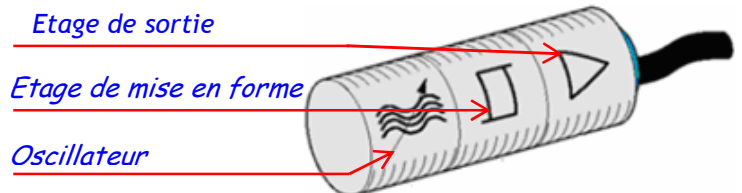
3.1. Structure :

- Câble (1)
- Corps fileté (2)
- Face active (3)
- Led de visualisation (4)
- Ecrus de fixation (5)
- Objet conducteur à détecter (6)



3.2. Composition :

Les bobinages de l'oscillateur, constituent la face sensible du capteur. A l'avant de celle-ci est créé un champ magnétique alternatif.



3.3. Fonctionnement :

Un détecteur de proximité inductif détecte sans contact tous les objets de matériaux conducteurs. A l'approche d'un objet conducteur (6), de la face sensible (3), le capteur délivre un signal.

Le signal de sortie correspondant à un contact électrique.

3.4. Caractéristiques :

Portée nominale:

2 à 5 mm (courant)

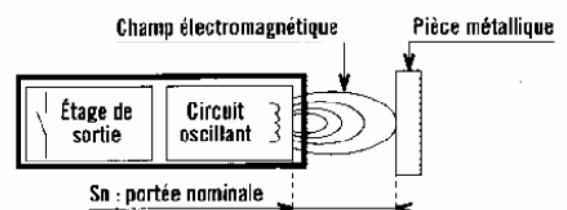
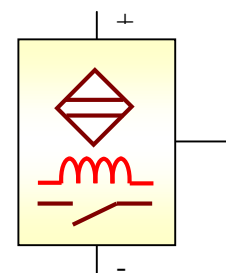
Jusqu'à 40 (performant)

Tension d'alimentation:

10 à 30 V DC

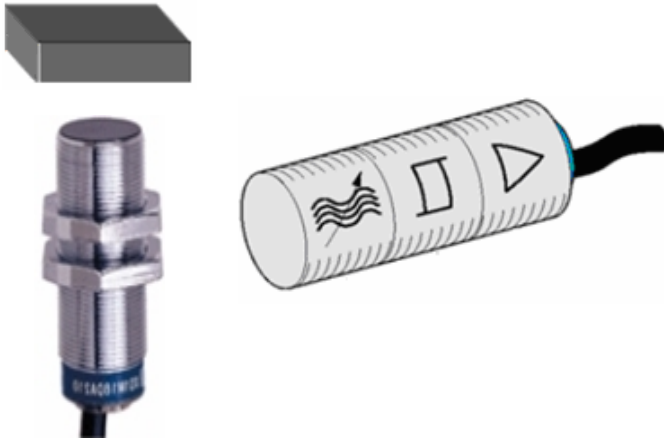
20 à 264 V AC

Symbole:



4. Détecteur de proximité capacitif :

4.1. Structure et composition :



4.2. Caractéristiques :

Portée nominale:

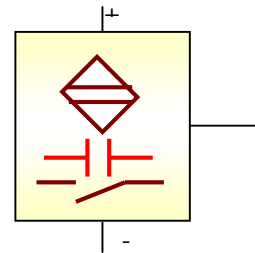
2, 5, 10, 15 ou 20 mm

Tension d'alimentation:

10 à 30 V DC

20 à 264 V AC

Symbole:

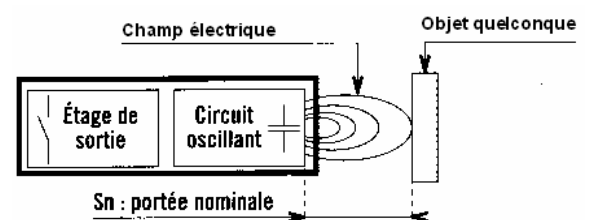


4.3. Fonctionnement :

Un détecteur de proximité capacitif détecte sans contact tous les objets de matériaux conducteurs ou isolants.

IL se compose principalement d'un oscillateur dont les condensateurs constituent la face sensible.

Le signal de sortie correspondant à un contact électrique.



5. Détecteur de proximité photoélectrique :

5.1. Fonctionnement :

Ils sont constitués d'un émetteur de lumière à diode électroluminescente et d'un récepteur de lumière à phototransistor qui convertit le signal lumineux en signal électrique. La détection est obtenue lorsque le faisceau lumineux émis, n'arrive pas au récepteur car l'objet se trouve entre les deux modules.

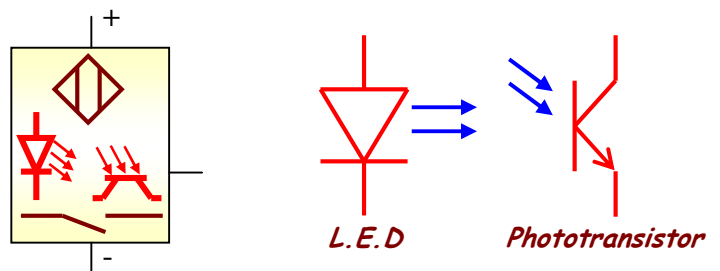
5.2. Caractéristiques :

Tension d'alimentation :

10 à 30 V DC

20 à 264 V AC

Symbole :

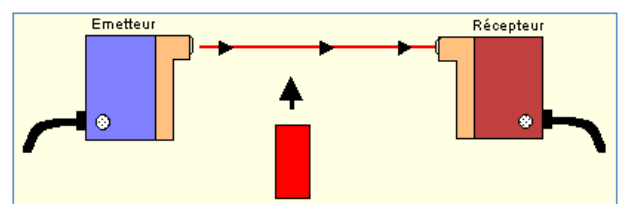


Il existe 3 systèmes de base pour détecter un objet à l'aide d'un émetteur-récepteur de lumière.

5.3. Le système BARRAGE :

L'émetteur et le récepteur sont dans 2 boîtiers séparés. L'objet est détecté lorsqu'il interrompt le faisceau lumineux.

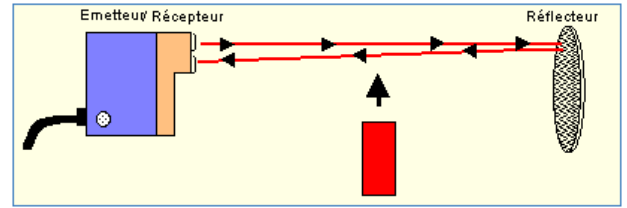
Portée nominale : jusqu'à 50 m
(100 m : laser)



5.4. Le système REFLEX :

L'émetteur et le récepteur sont dans le même boîtier. Le faisceau lumineux émis est renvoyé vers le récepteur par un réflecteur. La détection se fait par coupure du faisceau.

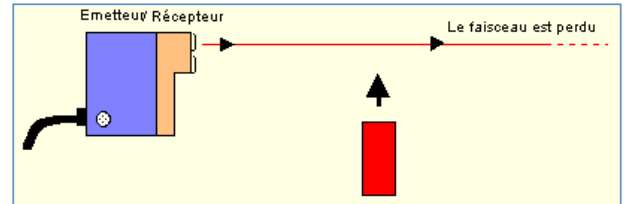
Portée nominale: jusqu'à 15 m (4 à 8 m courants)



5.5. Le système DE PROXIMITE :

L'émetteur et le récepteur sont dans un même boîtier. La détection se fait lorsque le faisceau lumineux est renvoyé par l'objet.

Portée nominale: de 0.005 à 2 m

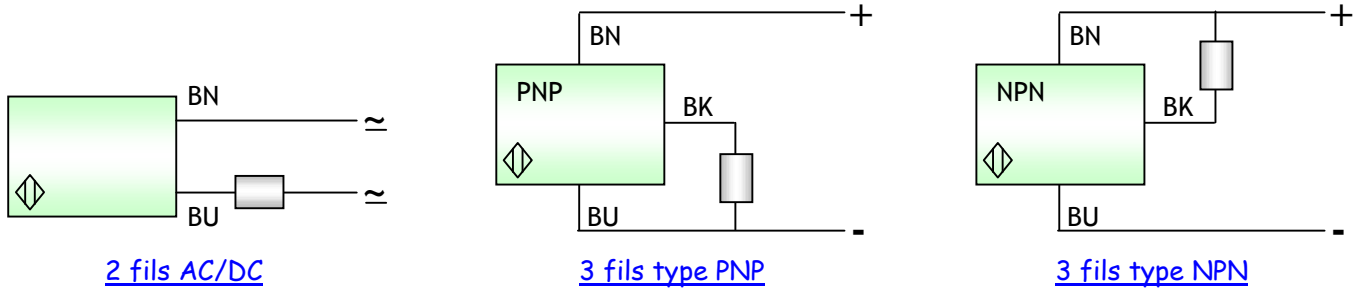


6. Choix et raccordement d'un capteur de proximité :

Le choix d'un détecteur de proximité dépend :

- de la nature du matériau constituant l'objet à détecter,
- de la distance de l'objet à détecter,
- des dimensions de l'emplacement disponible pour implanter le détecteur.

Technique de raccordement 2 fils et 3 fils.

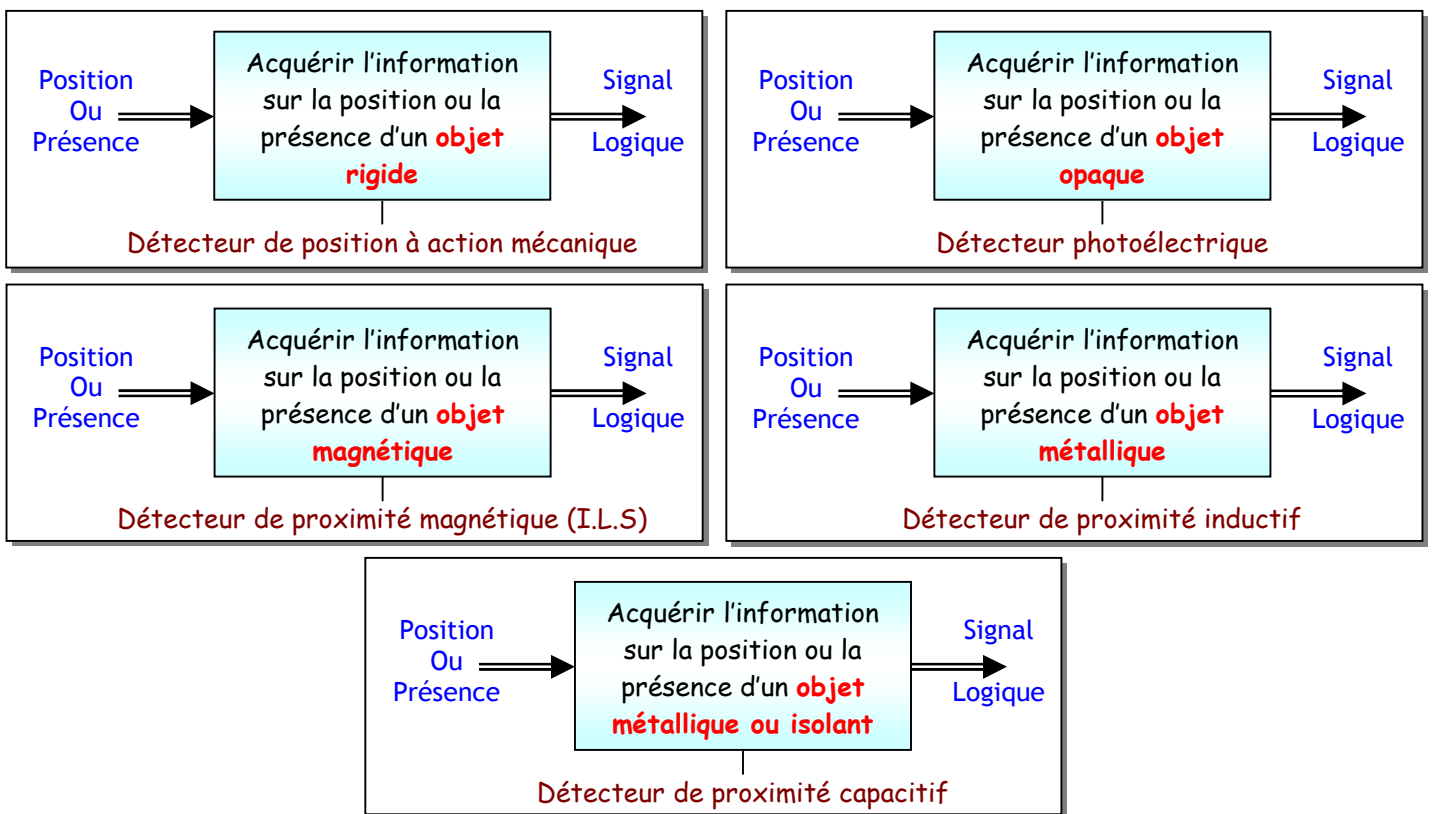


2 fils AC/DC

3 fils type PNP

3 fils type NPN

7. Conclusion :



1. La fonction TRAITER :

1.1. Présentation :

Dans la chaîne d'information, les informations (comptes rendus et consignes) issues de la fonction « acquérir » doivent être TRAITÉES puis COMMUNIQUÉES à l'environnement. A cette fin, des solutions technologiques spécifiques sont utilisées. La connaissance de la nature des informations circulant entre les divers éléments est indispensable.

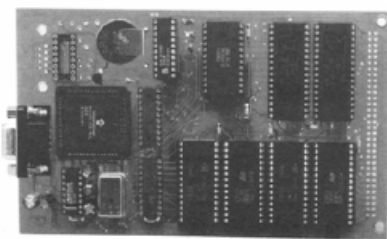
1.2. Les types de traitement des informations existant :

Logique câblée :

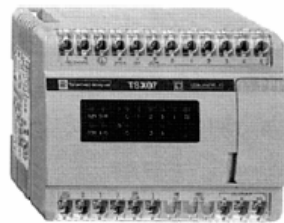
Ce type de traitement est figé et en conséquence réservé aux systèmes simples ou liés à la sécurité. Il est réalisé par des circuits électriques câblés ou des cartes électroniques.

Logique programmée :

Ce type de traitement réalisé par un programme permet des adaptations et des évolutions par programmation. Il est réalisé par des:



Module logique programmable

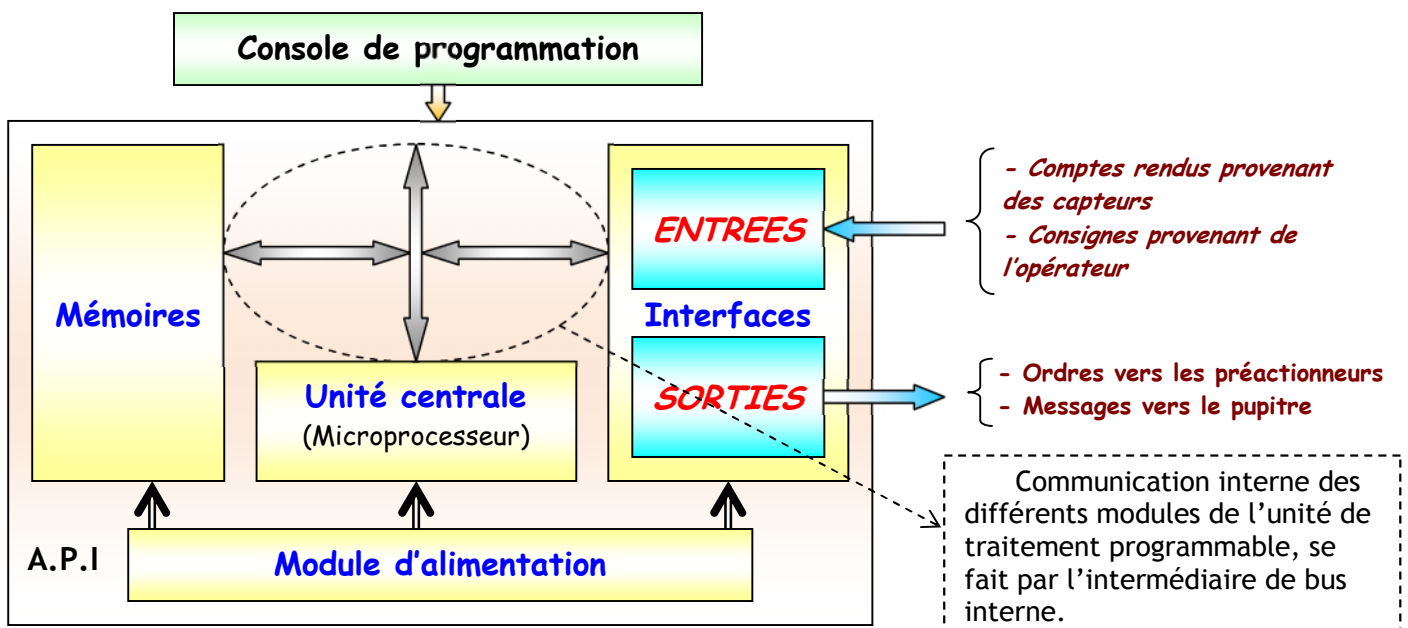


Automate programmable



Ordinateur

1.3. Structure des unités de traitement programmable :

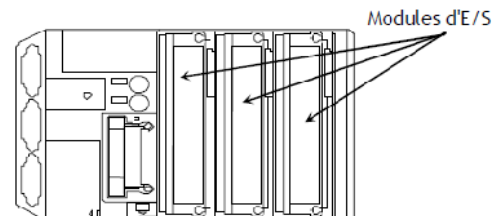


Les unités de traitement programmable sont constituées de :

- **L'unité centrale** : à base de microprocesseurs, elle réalise toutes les fonctions logiques, arithmétiques et de traitement numérique (transfert, comptage, temporisation, ...).
- **Les mémoires** : Elles permettent de stocker le système d'exploitation (ROM ou PROM), le programme (EEPROM) et les données système lors du fonctionnement (RAM). Cette dernière est généralement secourue par pile ou batterie. On peut, en règle générale, augmenter la capacité mémoire par adjonction de barrettes mémoires type PCMCIA.

- **Le module d'alimentation** : il assure la distribution de l'énergie électrique de 24V aux différents modules.
- **La console de programmation** : C'est généralement un PC où est installé qui le logiciel de programmation spécifique à l'API. Ce logiciel permet d'éditer le programme, de le compiler et de le transférer à l'automate.
- **Les interfaces d'entrées/sorties :**

Ces interfaces d'Entrée/Sortie (E/S) se présentent généralement sous forme d'interfaces modulaires qu'on ajoute selon le besoin.



- ➔ **Le module des entrées ou carte d'entrée** : c'est un circuit électronique qui reçoit les informations et les adapte pour l'unité de traitement. Modularité : **8, 16** ou **32** voies.

L'interface d'entrée a pour fonction de :

- ⊗ Recevoir les signaux logiques en provenance des capteurs et du pupitre opérateur ;
- ⊗ Traiter ces signaux en les mettant en forme, en éliminant les parasites d'origine industrielle et en isolant électriquement l'unité de commande de la partie opérative (isolation galvanique) pour la protection ;
- ⊗ Généralement les entrées sont désignées ainsi : **%i.j** où **i** est le numéro du module et **j** le numéro de l'entrée dans ce module, le signe "%" est spécifique au constructeur (ici Télémécanique). Exemple : **%I0.3** représente l'entrée **3** du module **0**.

- ➔ **Le module des sorties ou carte de sortie** : c'est un circuit électronique qui convertit les données de l'unité de traitement en ordres ou informations exploitables. Modularité : **8, 16** ou **32** voies.

L'interface de sortie a pour fonction de :

- ⊗ Commander les pré-actionneurs et éléments de signalisation du système ;
- ⊗ Adapter les niveaux de tension de l'unité de commande à celle de la partie opérative du système en garantissant une isolation galvanique entre ces dernières ;
- ⊗ Généralement les sorties sont désignées ainsi : **%Qi.j** où **i** est le numéro du module et **j** le numéro de la sortie. Exemple : **%Q1.5** représente la sortie **5** du module **1**.

2. LES AUTOMATES PROGRAMMABLES INDUSTRIELS (A.P.I) :

2.1. Historique :

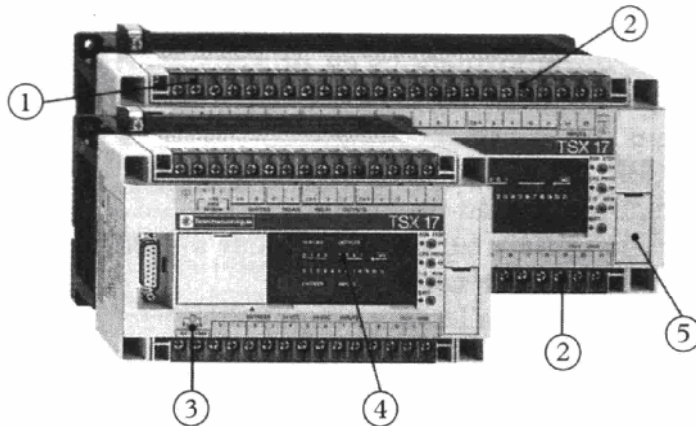
Les automates programmables industriels (A.P.I) sont apparus à la fin des années soixante, à la demande de la société américaine (GM) qui demandait plus d'adaptabilité de leurs systèmes de commande.

L'automate programmable industriel (A.P.I) **est un appareil électronique programmable, adapté au milieu industriel, qui réalise des fonctions d'automatisme pour assurer la commande de préactionneurs et d'actionneurs à partir d'informations logique, analogique ou numérique.**

2.2. Domaines d'emploi des automates :

On utilise les API dans tous les secteurs industriels pour la commande des machines (convoyage, emballage ...) ou des chaînes de production (automobile, agroalimentaire ...) ou il peut également assurer des fonctions de régulation de processus (métallurgie, chimie ...). Il est de plus en plus utilisé dans le domaine du bâtiment (tertiaire et industriel) pour le contrôle du chauffage, de l'éclairage, de la sécurité ou des alarmes.

2.3. Structure externe d'un A.P.I :



Automate Programmable TSX 17/20 (Télémechanique)

1. Alimentation
2. entrées-sorties (E/S)
3. Alimentation capteurs 24 V
4. Visualisation des E/S
5. Extension du bus E.S

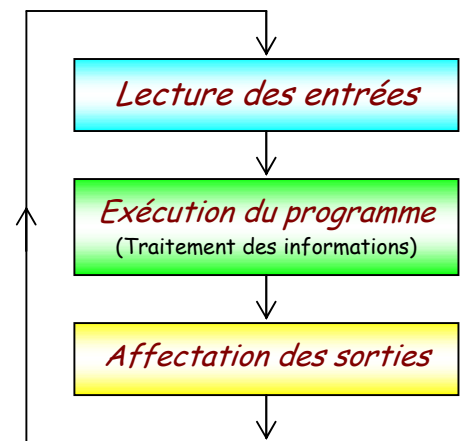
2.4. Cycle d'exécution d'un automate :

Durant son fonctionnement, un API exécute le même cycle de fonctionnement qu'on appelle "cycle automate" ; la durée de ce cycle est typiquement de 1 à 50 ms :

→ **La détection** : les signaux électriques en provenance des capteurs arrivent sur les modules d'entrée. Chaque module est repéré par une adresse d'entrée.

→ **Le traitement** : l'API calcule les équations de fonctionnement du système en fonction des entrées.

→ **L'affectation** : Les résultats sont recopiés dans les modules de sortie. Chaque module est repéré par une adresse de sortie.



2.5. Programmation de l'API :

La programmation d'un API consiste à traduire dans le langage spécialisé de l'automate, les équations de fonctionnement du système à automatiser. Parmi les langages normalisés, on cite quelques-uns des plus connus et plus utilisés :

- ⊗ Langage à contacts (LADDER) ;
- ⊗ Langage List d'instructions (Instruction List) ;
- ⊗ Langage GRAFCET (Sequential Function Chart : SFC).

1. INTRODUCTION :

Dans un système automatisé, l'unité de traitement reçoit les informations traitées principalement de :

- ❖ L'utilisateur, grâce aux organes de dialogue en entrée tel un bouton d'arrêt d'urgence ; il s'agit de l'interface Homme/Machine (IHM) ;
- ❖ Des capteurs tel un "fin de course" ; il s'agit d'interface centrée principalement sur l'isolation électrique ou galvanique et la mise en forme du signal.

L'interface Homme/Machine en entrée consiste à transmettre au système automatisé les ordres de l'opérateur qu'on désigne par "consignes". Le dialogue Homme/Machine se fait par l'utilisation de constituants regroupés dans ce qu'on appelle « pupitre » de commande.

2. INTERFACE HOMME/MACHINE (IHM):

2.1. Les boutons poussoirs :

Les boutons poussoirs sont utilisés lorsque les informations transférées vers le système sont limitées à des signaux Tout Ou Rien (TOR).



Bouton
Coup de poing



Bouton
Affleurant



Bouton
Tournant à manette



Bouton
Tournant à clé

2.2. Les claviers :

Les claviers permettent la saisie d'informations alphanumériques et la modification de données et paramètres ; comme le nombre de pièces à fabriquer.



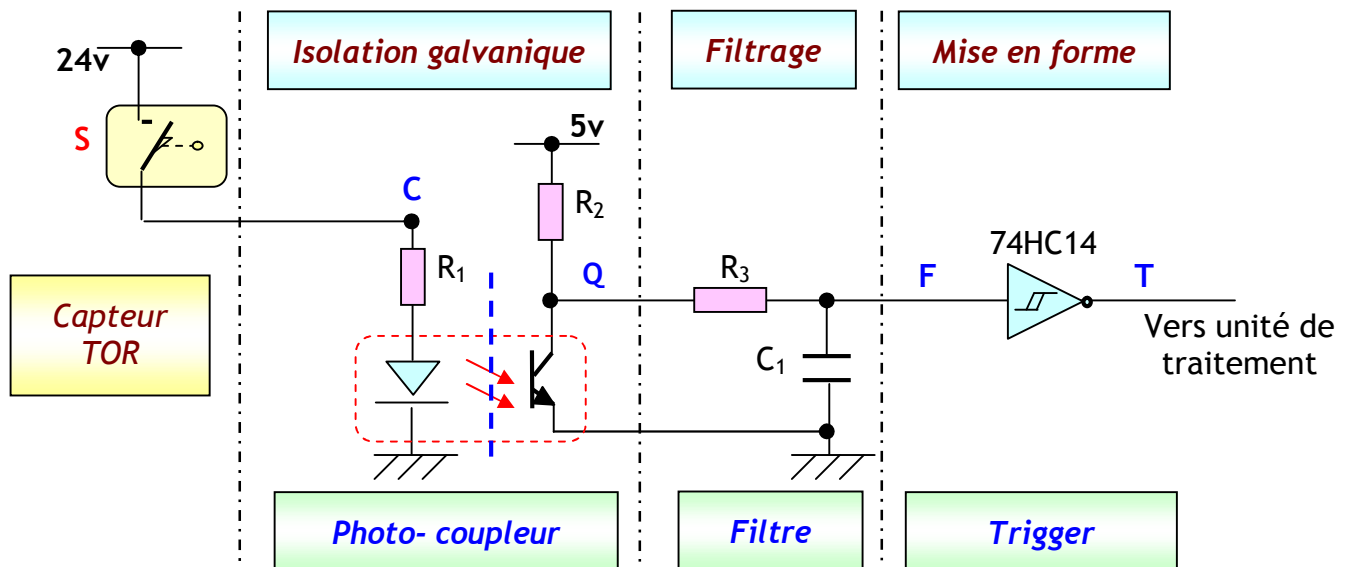
3. ISOLATION GALVANIQUE :

L'unité de traitement travaille typiquement avec une tension de 5V DC, nécessaire pour alimenter les circuits intégrés logiques. Alors qu'un capteur tel un "fin de course" fournit une tension de 24 V DC.

Pour protéger l'unité de traitement contre une éventuelle liaison directe avec tension relativement dangereuse pour elle, il faut une isolation électrique ou galvanique, ainsi qu'un filtrage et mise en forme du signal.

3.1. Schéma de principe :

Le principe est résumé par le schéma fonctionnel suivant :



Ce montage, on le trouve déjà intégré dans un API.

3.2. Fonctionnement :

Quand le capteur est actionné, son contact est fermé, il fournit du 24 V au circuit de la diode infrarouge de l'optocoupleur :

- ❖ La diode de l'optocoupleur conduit et émet de l'infrarouge ;
- ❖ Le transistor de l'optocoupleur, travaillant en commutation se sature ;
- ❖ La porte inverseuse Trigger reçoit un '0' logique et fournit à sa sortie un '1'.

Quand le capteur est non actionné, alors :

- ❖ La diode de l'optocoupleur est bloquée et n'émet pas d'infrarouge ;
- ❖ Le transistor de l'optocoupleur se bloque ;
- ❖ La porte inverseuse Trigger reçoit un '1' logique et fournit à sa sortie T un '0'.

3.3. Chronogrammes :

L'état de la porte reflète l'état du capteur ;

Le bruit affectant le signal du capteur, à cause des parasites industrielles, est filtré par le circuit RC ;

La porte Trigger met en forme le signal et envoie à l'unité de traitement un signal "propre", et donc bien compatible.



1. INTRODUCTION :

Dans un système automatisé, l'unité de traitement communique les informations traitées principalement vers :

- ❖ L'utilisateur, grâce aux organes de dialogue en sortie ? tel un voyant de signalisation ; il s'agit de l'interface Homme/Machine (IHM) ;
- ❖ Des organes de puissance tel un moteur électrique triphasé ; il s'agit d'interface de puissance centré principalement sur l'isolation électrique ou galvanique.

2. INTERFACE HOMME/MACHINE (IHM) :

2.1. Les voyants :

Les voyants sont des témoins lumineux qui constituent une interface de dialogue simple donnant à l'opérateur des informations sur l'état du système automatisé.

Les voyants possèdent un code de couleur qui permet d'orienter l'opérateur sur l'origine du message :

| COULEUR | SIGNIFICATION | EXEMPLE |
|---------|---|-------------------------------|
| ROUGE | Urgence ou condition dangereuse | Arrêt d'urgence |
| JAUNE | Condition anormale pouvant entraîner une situation dangereuse | Manque en matière d'oeuvre |
| VERT | Préparation de conditions normales | Départ de cycle |
| BLANC | Information générale | Présence de la tension réseau |

2.2. Les afficheurs numériques :

Dans certaines applications industrielles, le contrôle du système automatisé nécessite de surveiller les paramètres avec une grande précision qui n'est pas permise par les voyants. Les afficheurs numériques permettent d'effectuer une surveillance précise et informent l'opérateur des résultats de mesure (température, pression, etc.) ou de comptage (nombre de cycle, quantité de pièces produites, etc.).

2.3. Les terminaux d'exploitation :

Les terminaux d'exploitation sont des constituants de dialogue programmés permettant à l'opérateur :

- ❖ D'être informé clairement sur l'état du système automatisé ;
- ❖ D'intervenir facilement et rapidement sur les paramètres de fonctionnement du système automatisé.

On distingue deux types de terminaux d'exploitation :

- ❖ Les terminaux d'exploitation à afficheurs ;
- ❖ Les terminaux d'exploitation à écran permettant l'affichage simultané d'un nombre plus important de données en plus des représentations graphiques concernant le système automatisé.



Exemple de voyant



Exemple d'afficheur numérique



Terminal d'exploitation à afficheur Terminal d'exploitation à écran

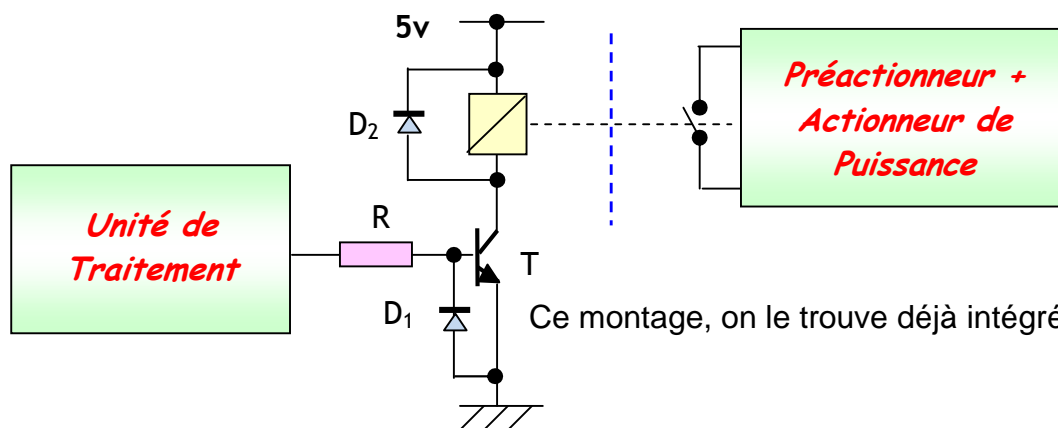
3. ISOLATION GALVANIQUE :

L'unité de traitement travaille typiquement avec une tension de 5V DC, nécessaire pour alimenter principalement les circuits intégrés logiques. Alors qu'un actionneur tel un moteur triphasé a besoin d'une forte tension (220V AC, par exemple). Pour protéger l'unité de traitement contre une éventuelle liaison directe avec la forte tension, il faut une isolation électrique ou galvanique.



3.1. Schéma de principe :

Le principe est résumé par le schéma fonctionnel suivant :



Ce montage, on le trouve déjà intégré dans un API.

3.2. Fonctionnement :

Quand l'unité de traitement communique l'ordre d'action (5 V), alors :

- ❖ Le transistor T, travaillant en commutation se sature ;
- ❖ Le relais est excité avec la tension 5v, ce qui ferme son contact ; ce contact isolé électriquement par rapport à la commande peut être utilisé pour établir le courant dans un organe de puissance (moteur, lampe, résistance chauffante, etc.) ;
- ❖ La diode D n'a aucun rôle dans cet état ; elle est bloquée.

Quand l'unité de traitement communique l'ordre de commande d'arrêt (0 V), alors :

- ❖ Le transistor Q se bloque ;
- ❖ Le relais est désexcité, ce qui ouvre son contact ; la charge est alors désalimentée.
- ❖ La diode D joue le rôle de roue libre pour protéger le transistor T contre la surtension qui apparaît aux bornes de la bobine du relais.

1. Langage LADDER :

1.1. Description :

Le langage LADDER est une succession " de réseaux de contacts " véhiculant des informations logiques depuis les entrées vers les sorties. Le résultat dépend des fonctions programmées.

La programmation du GRAFCET en langage LADDER consiste à associer à chaque étape i du GRAFCET un bit interne de l'API X_i .

Le programme est alors constitué de 2 traitements :

- ❖ **Traitement séquentiel** : cette partie du programme décrit l'évolution séquentielle des étapes en calculant l'état des bits internes X_i représentant les étapes.
- ❖ **Traitement postérieur** : cette partie détermine l'état des sorties.

1.2. Exemple 'Poste de perçage automatique' :

Dans cet exemple, on traduit le GRAFCET correspondant au 'Poste de perçage automatique' en LADDER ; l'API utilisé est le TSX 17 de télémécanique.

1.2.1. Affectations des entrées sorties de l'API :

Pour matérialiser ce GRAFCET, on doit raccorder les différents constituants à l'API, ce qui impose d'effectuer un choix d'affectation d'entrées-sorties. Ce choix permet d'attribuer :

- ❖ Un numéro d'entrée à chacun des capteurs ou boutons ;
- ❖ Un numéro de sortie à chacun des préactionneurs ou voyants.

Les entrées de l'API TSX 17 sont désignées ainsi : $\%I_{i,j}$ où i est le numéro du module et j le numéro de l'entrée dans ce module, le signe "%" est spécifique au constructeur.

Exemple : $\%I_{0,3}$ représente l'entrée 3 du module 0 (ou est connecté le capteur L_{10}).

Les sorties sont désignées ainsi : $\%Q_{i,j}$ où i est le numéro du module et j le numéro de la sortie dans ce module.

Exemple : $\%Q_{0,2}$ représente la sortie 2 du module 0 (ou est connecté le contacteur KM).

| Opération | Actionneur | Préactionneur | Sortie API |
|--------------------|--------------|---------------|------------|
| Serrer la pièce | Vérin C_1 | SC_1 | $14 M_1$ |
| Desserrer la pièce | | RC_1 | $12 M_1$ |
| Perçer la pièce | Moteur M_T | M_T | KM |
| | Vérin C_2 | SC_2 | $14 M_2$ |
| | | RC_2 | $12 M_2$ |

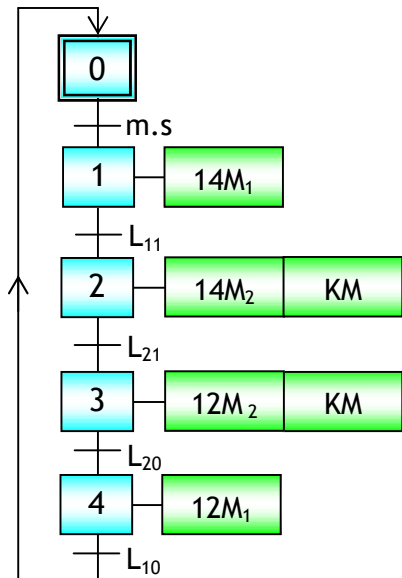
| Compte-rendu et ordre | Capteur ou bouton | Mnem. | Entrée API |
|-------------------------------|------------------------------|----------|------------|
| Départ cycle | Bouton poussoir | m | $I_{0,0}$ |
| présence de la pièce à perçer | Détecteur mécanique à levier | S | $I_{0,1}$ |
| Position du foret | Détecteur mécanique à levier | L_{11} | $I_{0,2}$ |
| | Détecteur mécanique à levier | L_{10} | $I_{0,3}$ |
| Position du mors de serrage | Détecteur mécanique à levier | L_{21} | $I_{0,4}$ |
| | Détecteur mécanique à levier | L_{20} | $I_{0,5}$ |

1.2.2. GRAFCET du point de vue P.C codé :

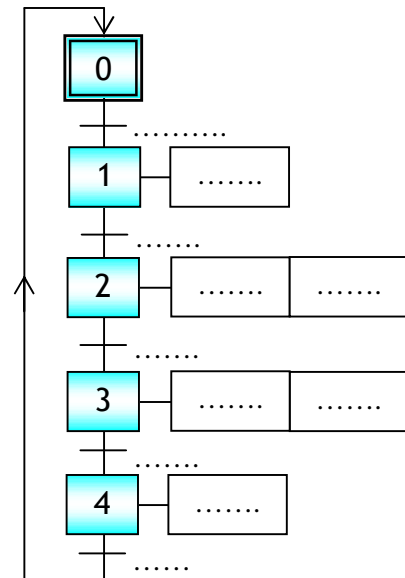
Ce GRAFCET est obtenu à partir du GRAFCET du point de vue P.C :

- ☞ On remplace les capteurs et les boutons par les entrées de l'API correspondantes;
- ☞ On remplace les préactionneurs et les voyants par les sorties de l'API correspondantes.

GRAFCET du point de vue P.C
'Poste de perçage automatique'



GRAFCET du point de vue P.C codé API
'Poste de perçage automatique'



1.2.3. Mise en équation :

On associe à chaque étape n°i du GRAFCET un bit interne de l'API Mi :

- ☞ à l'étape 0 on associe un bit X_0 .
- ☞ à l'étape 1 on associe un bit X_1 .
- ☞ à l'étape 2 on associe un bit X_2 .
- ☞ à l'étape 3 on associe un bit X_3 .
- ☞ à l'étape 4 on associe un bit X_4 .

Etape n°0 :

Mise à 1 de X_0 -Set- si l'étape 4 ($X_4 = 1$) est active et la réceptivité $I_{0,3}$ ($I_{0,3} = 1$) est vraie.

Donc $SX_0 = \dots\dots\dots$

Mise à 0 de X_0 -Reset- si l'étape 1 ($X_1 = 1$) est active.

Donc $RX_0 = \dots\dots\dots$

Etape n°2 :

Mise à 1 de X_2 -Set- si l'étape 1 ($X_1 = 1$) est active et la réceptivité $I_{0,2}$ ($I_{0,2} = 1$) est vraie.

Donc $SX_2 = \dots\dots\dots$

Mise à 0 de X_2 -Reset- si l'étape 3 ($X_3 = 1$) est active.

Donc $RX_2 = \dots\dots\dots$

Etape n°4 :

Mise à 1 de X_4 -Set- si l'étape 3 ($X_3 = 1$) est active et la réceptivité $I_{0,5}$ ($I_{0,5} = 1$) est vraie.

Donc $SX_4 = \dots\dots\dots$

Mise à 0 de X_4 -Reset- si l'étape 0 ($X_0 = 1$) est active.

Donc $RX_4 = \dots\dots\dots$

Etape n°1 :

Mise à 1 de X_1 -Set- si l'étape 0 ($X_0 = 1$) est active et la réceptivité $I_{0,0} \cdot I_{0,1}$ ($I_{0,0} \cdot I_{0,1} = 1$) est vraie.

Donc $SX_1 = \dots\dots\dots$

Mise à 0 de X_1 -Reset- si l'étape 2 ($X_2 = 1$) est active.

Donc $RX_1 = \dots\dots\dots$

Etape n°3 :

Mise à 1 de X_3 -Set- si l'étape 2 ($X_2 = 1$) est active et la réceptivité $I_{0,4}$ ($I_{0,4} = 1$) est vraie.

Donc $SX_3 = \dots\dots\dots$

Mise à 0 de X_3 -Reset- si l'étape 4 ($X_4 = 1$) est active.

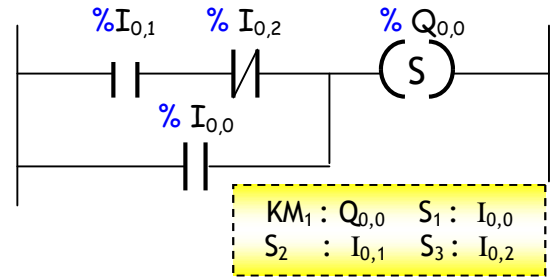
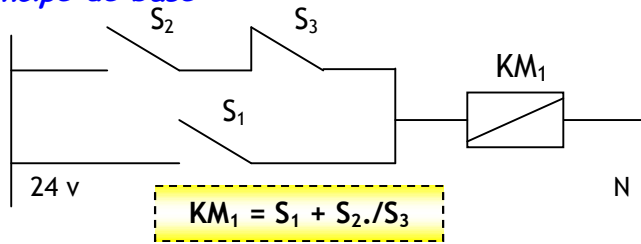
Donc $RX_3 = \dots\dots\dots$

équations des sorties :

- $Q_{0,0} = \dots\dots\dots$
- $Q_{0,1} = \dots\dots\dots$
- $Q_{0,2} = \dots\dots\dots$
- $Q_{0,3} = \dots\dots\dots$
- $Q_{0,4} = \dots\dots\dots$

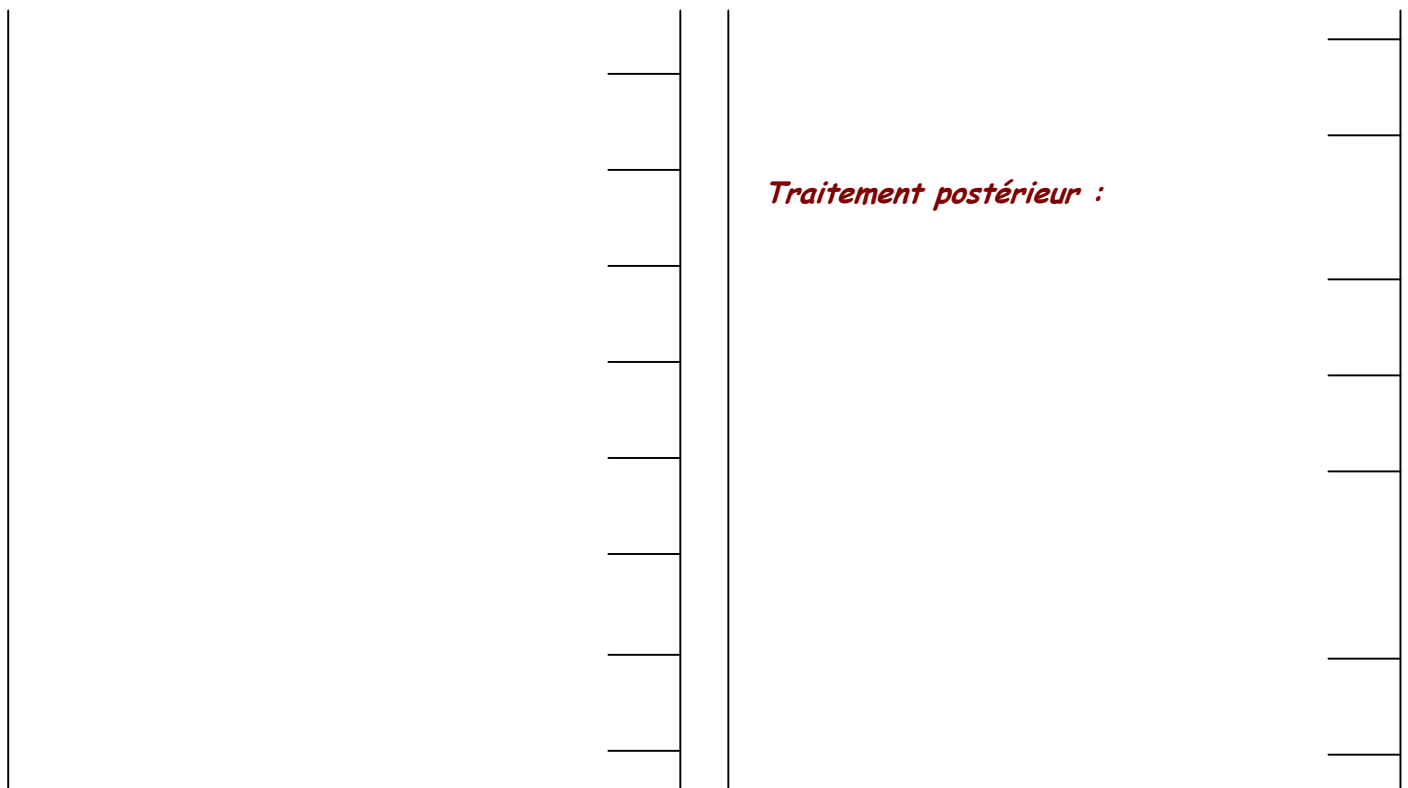
1.2.4. Programme LADDER :

Principe de base :



Programme LADDER 'Poste de perçage automatique'

Traitement séquentiel :



Traitement postérieur :

2. Liste d'instructions "IL" :

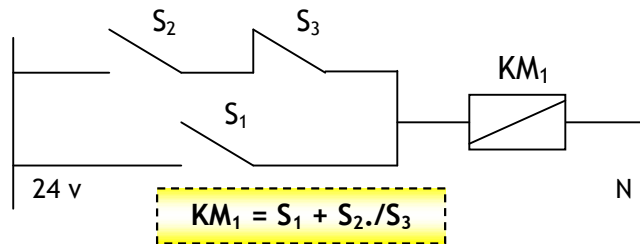
2.1. Description :

L'IL est un langage dans lequel toutes les opérations sont décrites par des instructions mnémoniques. Ce n'est pas un langage graphique.

Le tableau suivant donne une liste représentative de ce langage :

| INSTRUCTION | FONCTION |
|-------------|---|
| LD | Lire une entrée ou une variable interne. |
| LDN | Lire l'inverse d'une entrée ou d'une variable interne. |
| AND | ET logique entre le résultat de l'instruction précédente et l'état de l'opérande. |
| ANDN | OU logique entre le résultat de l'instruction précédente et l'état inverse de l'opérande. |
| OR | OU logique entre le résultat de l'instruction précédente et l'état de l'opérande. |
| ORN | ET logique entre le résultat de l'instruction précédente et l'état inverse de l'opérande. |
| N | Négation du résultat de l'instruction précédente. |
| ST | L'opérande associé prend la valeur du résultat de la zone test. |
| STN | L'opérande associé prend la valeur inverse du résultat de la zone test. |
| S | L'opérande associé est mis à 1 lorsque le résultat de la zone test est à 1. |
| R | L'opérande associé est mis à 0 lorsque le résultat de la zone test est à 1. |
| END | Fin de programme. |

2.2. Principe de base :



LD %I_{0,1} OR %I_{0,0}
 ANDN %I_{0,2} ST %Q_{0,0}

KM₁ : Q_{0,0} S₁ : I_{0,0}
 S₂ : I_{0,1} S₃ : I_{0,2}

Programme LADDER 'Poste de perçage automatique'

| | | |
|---|---|---|
| <p><i>Traitement séquentiel :</i></p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> | <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> | <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> |
| <p><i>Traitement postérieur :</i></p> <p>.....</p> <p>.....</p> <p>.....</p> | | |

3. Sequential Function Chart "SFC" :

3.1. Description :

Le SFC est le langage graphique ; par abus de langage, on l'appelle aussi le langage GRAFCET. Pour éditer un programme GRAFCET on passe par les étapes suivantes :

- ☞ On commence par construire graphiquement le GRAFCET ;
- ☞ On traduit les réceptivités dans le langage IL ou le langage LADDER ;
- ☞ La programmation des actions se fait dans le traitement postérieur en LADDER ou en "IL".
- ☞ En langage GRAFCET l'activation et la désactivation des étapes se fait automatiquement.

3.2. Principe de base :

Construction du CHART :

Exemple

Traitement postérieur :

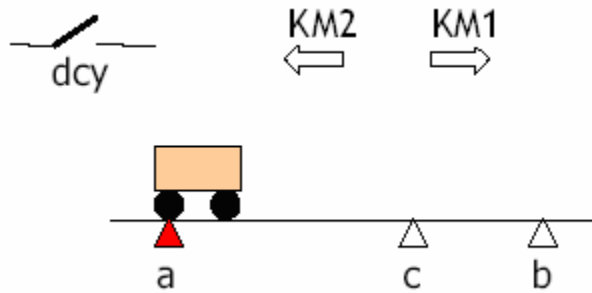
Ou en IL
LD %I_{0,0}

Ou en IL
LD %I_{0,1}

Ou en IL
LD %I_{0,2}

Ou en IL
LD % X₁ % Q_{0,0}
ST % Q_{0,0}
LD % X₂ % Q_{0,1}
ST % Q_{0,1}
END.

1. Cahier des charges :



Après l'ordre de départ cycle « dcy », le chariot part jusque b, revient en c, repart en b puis rentre en a. La commande du système est réalisée par l'A.P.I ZELIO SR3101BD.

2. Identification des entrées et des sorties :

| Mouvement | Actionneur | Ordres | Sortie API |
|------------------------------|------------|--------|----------------|
| Déplacer le chariot à droite | Moteur | KM1 | Q ₁ |
| Déplacer le chariot à gauche | Moteur | KM2 | Q ₂ |

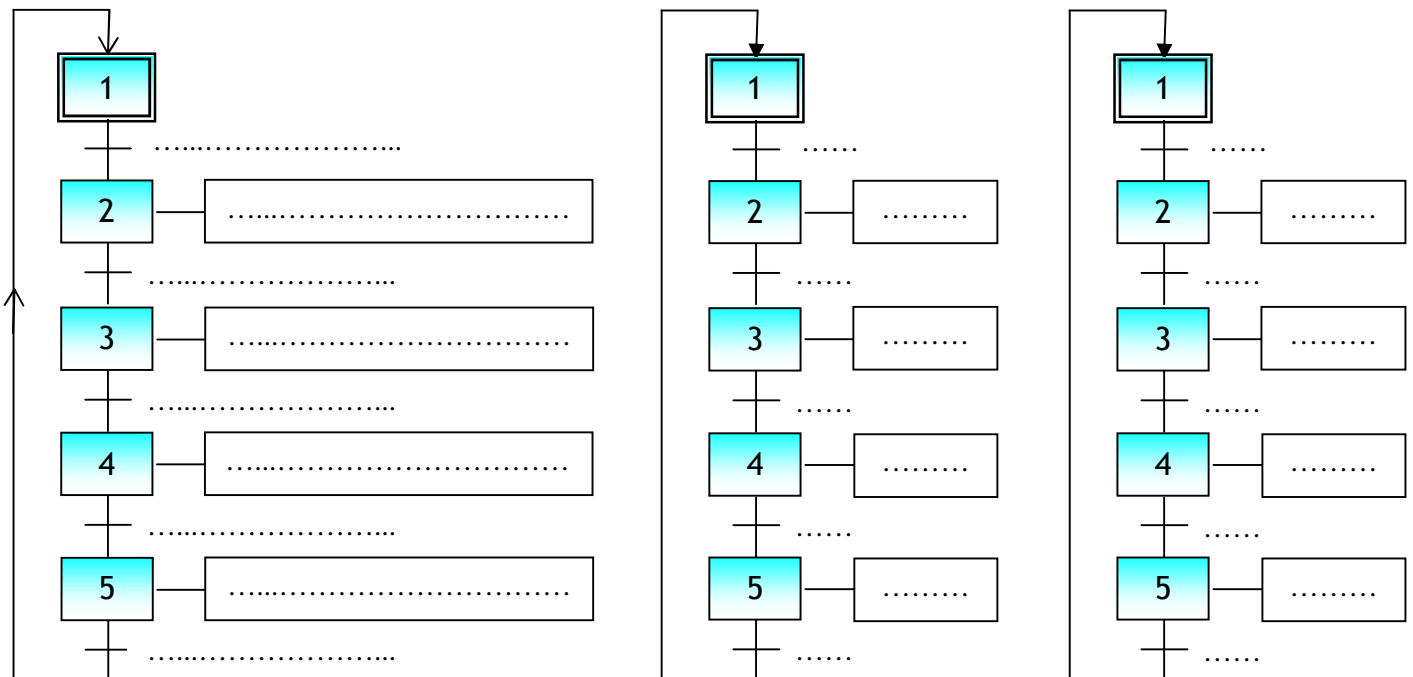
| Compte-rendu | Capteur | Mnem. | Entrée API |
|-------------------------|------------------------------|-------|----------------|
| chariot à en position a | Détecteur mécanique à levier | a | I ₁ |
| chariot à en position b | Détecteur mécanique à levier | b | I ₂ |
| chariot en position c | Détecteur mécanique à levier | c | I _B |

| Consigne | Constituant | Mnem. | Entrée API |
|--------------|-----------------|-------|----------------|
| départ cycle | Bouton poussoir | Dcy | I _c |

GRAFSET P.O

GRAFSET P.C

GRAFSET P.C codé API

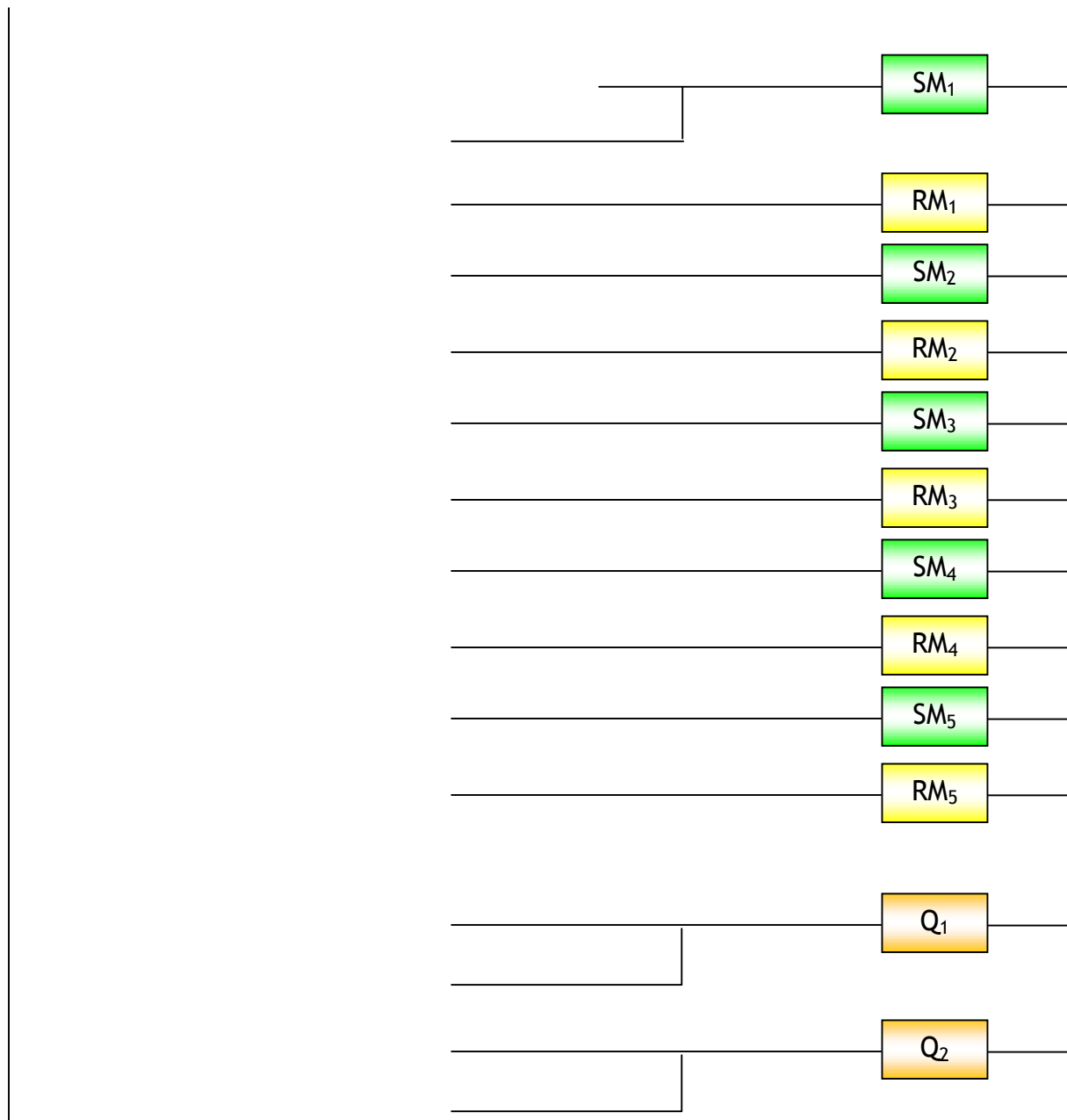


Mise en équation :

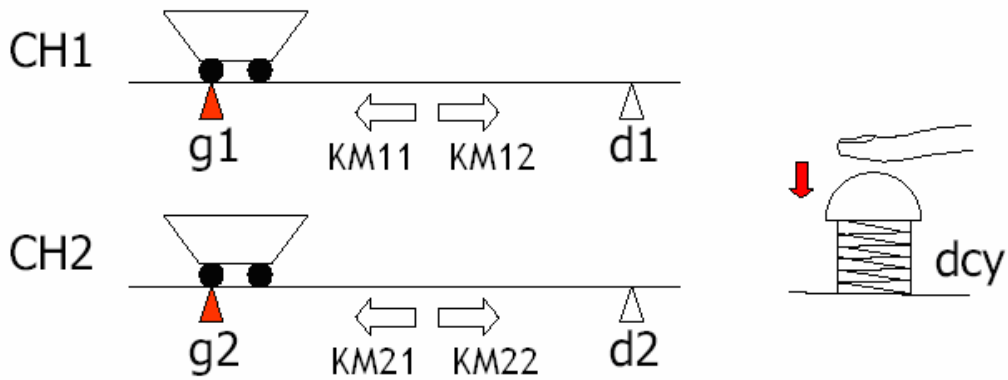
| Etapas | Activation | Désactivation |
|--------|-------------------------|-------------------------|
| 1 | SM ₁ = | RM ₁ = |
| 2 | SM ₂ = | RM ₂ = |
| 3 | SM ₃ = | RM ₃ = |
| 4 | SM ₄ = | RM ₄ = |
| 5 | SM ₅ = | RM ₅ = |

| | |
|---------|--|
| Actions | Q ₁ = KM ₁ = |
| | Q ₂ = KM ₂ = |

Programme LADDER :



1. Cahier des charges :



Après appui sur départ cycle « dcy », les chariots partent pour un aller retour.
 Un nouveau départ cycle ne peut se faire que si les deux chariots sont à gauche.
 La commande du système est réalisée l'A.P.I ZELIO SR3101BD.

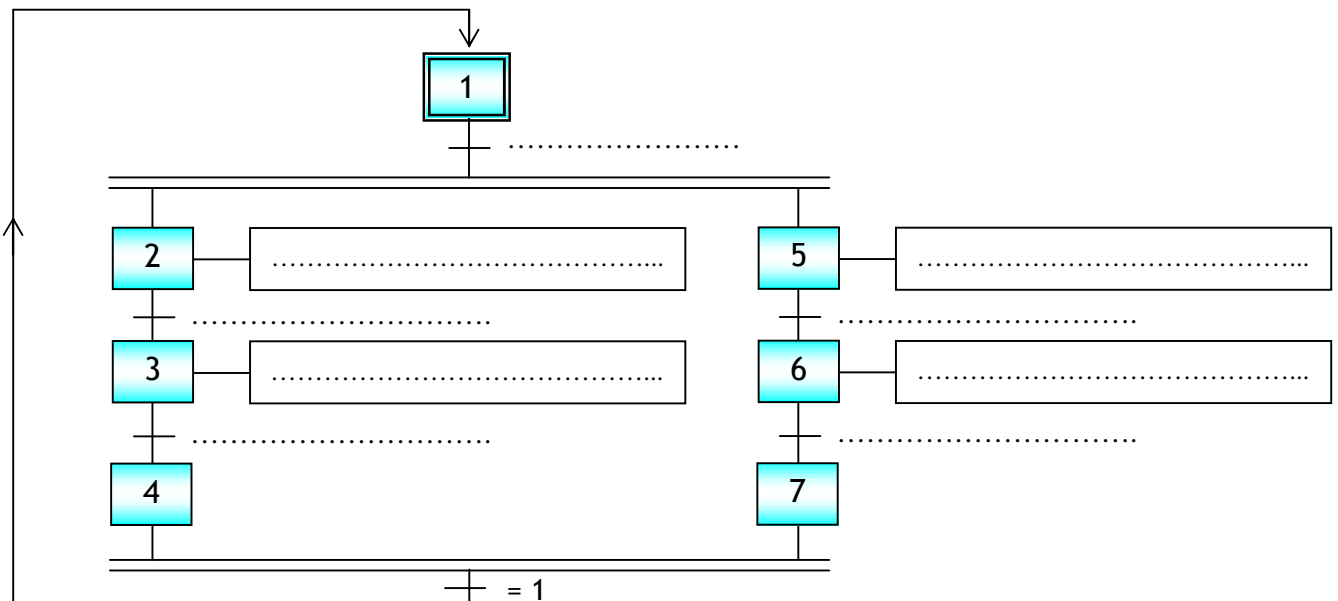
2. Identification des entrées et des sorties :

| Mouvement | Actionneur | Ordres | Sortie A.P.I |
|--------------------------------|------------|------------------|----------------|
| Déplacer le chariot 1 à droite | Moteur 1 | KM ₁₂ | Q ₁ |
| Déplacer le chariot 1 à gauche | Moteur 1 | KM ₁₁ | Q ₂ |
| Déplacer le chariot 2 à droite | Moteur 2 | KM ₂₂ | Q ₃ |
| Déplacer le chariot 2 à gauche | Moteur 2 | KM ₂₁ | Q ₄ |

| Compte-rendu | Capteur | Mnem. | Entrée A.P.I |
|--------------------|------------------------------|----------------|----------------|
| chariot 1 à gauche | Détecteur mécanique à levier | g ₁ | I ₁ |
| chariot 1 à droite | Détecteur mécanique à levier | d ₁ | I ₂ |
| chariot 2 à gauche | Détecteur mécanique à levier | g ₂ | I _B |
| chariot 2 à droite | Détecteur mécanique à levier | d ₂ | I _C |

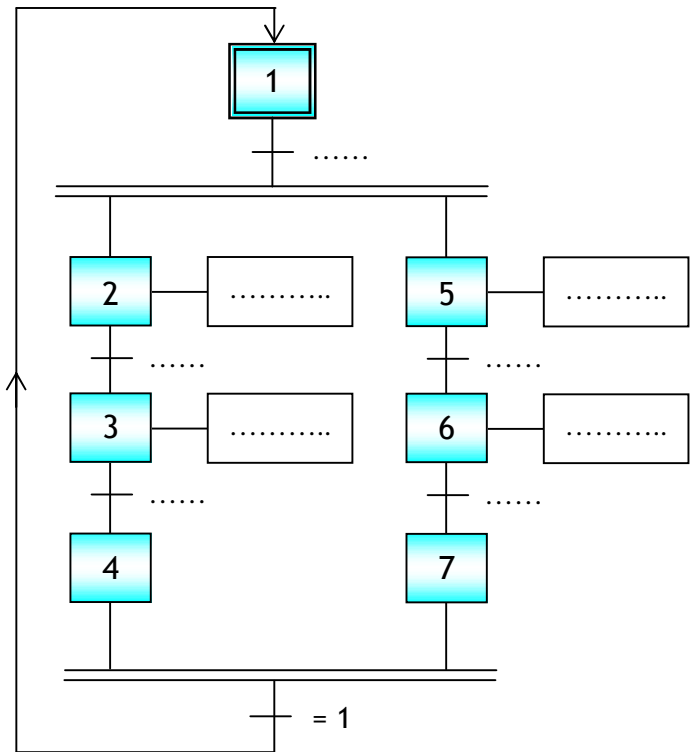
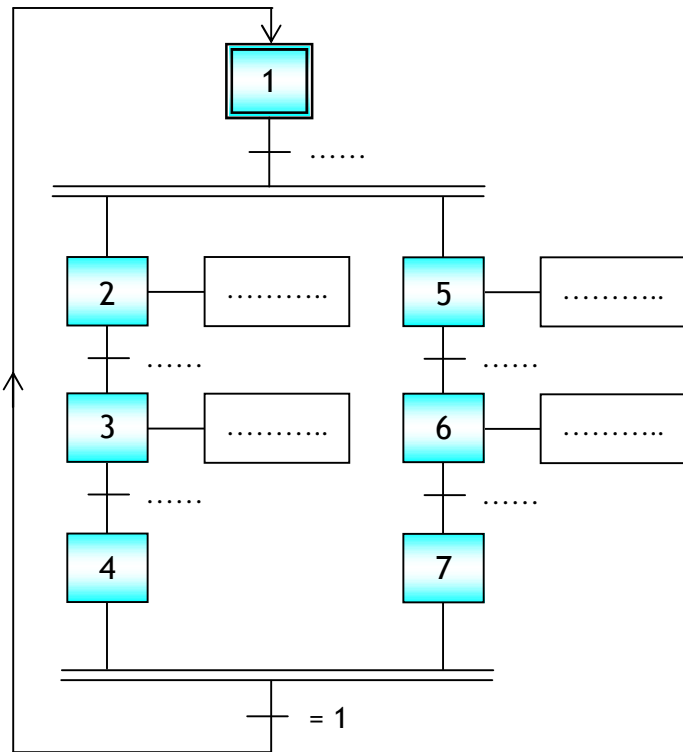
| Consigne | Constituant | Mnem. | Entrée A.P.I |
|--------------|-----------------|-------|----------------|
| départ cycle | Bouton poussoir | Dcy | I _d |

GRAFSET du point de vu P.O :



GRAFCET du point de vu P.C :

GRAFCET du point de vu P.C codé A.P.I :

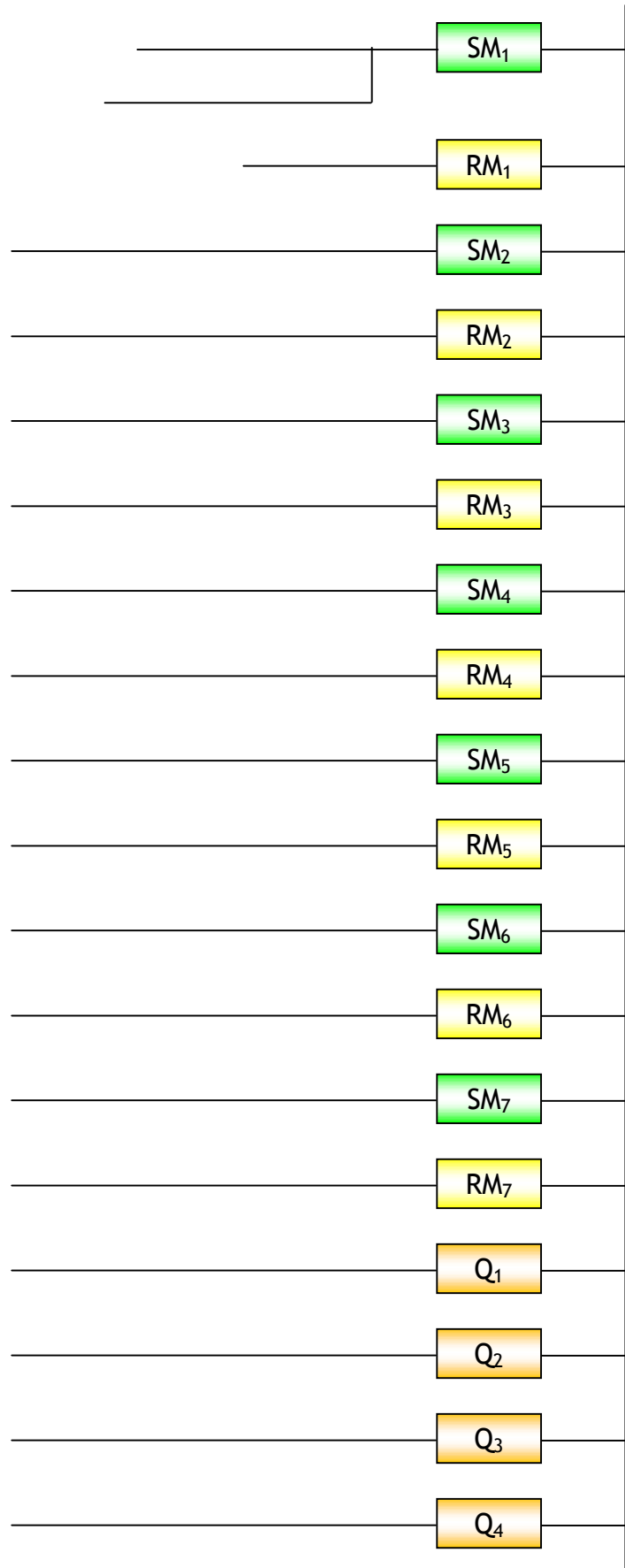


Mise en équation :

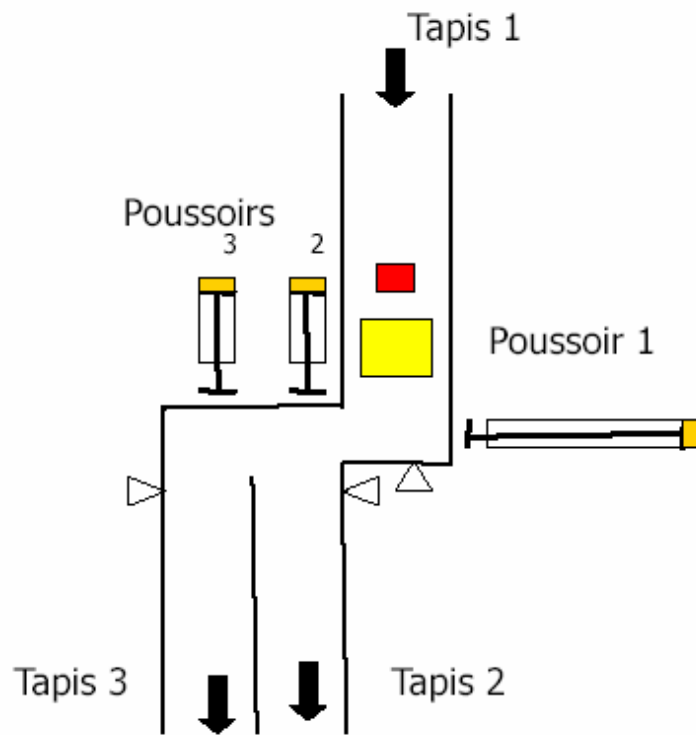
| Etapes | Activation | Désactivation |
|--------|-------------------------|-------------------------|
| 1 | SM ₁ = | RM ₁ = |
| 2 | SM ₂ = | RM ₂ = |
| 3 | SM ₃ = | RM ₃ = |
| 4 | SM ₄ = | RM ₄ = |
| 5 | SM ₅ = | RM ₅ = |
| 6 | SM ₆ = | RM ₆ = |
| 7 | SM ₇ = | RM ₇ = |

| | |
|---------|---|
| Actions | Q ₁ = KM ₁₂ = |
| | Q ₂ = KM ₁₁ = |
| | Q ₃ = KM ₂₂ = |
| | Q ₄ = KM ₂₁ = |

Programme LADDER :



1. Cahier des charges :



Un dispositif automatique destiné à trier des caisses de deux tailles différentes se compose d'un tapis amenant les caisses, de trois poussoirs et de deux tapis d'évacuation.

Le poussoir 1 pousse les petites caisses devant le poussoir 2 qui, à son tour, les transfère sur le tapis d'évacuation 2, alors que les grandes caisses sont poussées devant le poussoir 3, ce dernier les évacuant sur le tapis 3. Pour effectuer la sélection des caisses, un dispositif de détection placé devant le poussoir 1 permet de reconnaître sans ambiguïté le type de caisse qui se présente.

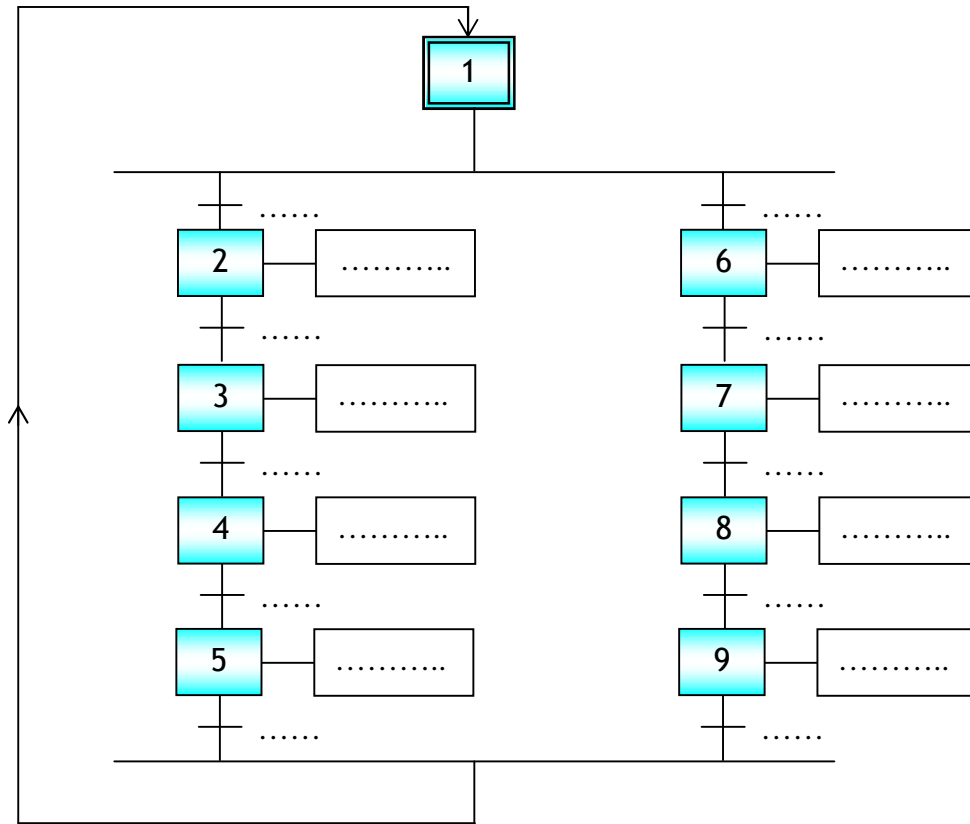
La commande est faite un A.P.I de la famille ZELIO à 12 entrées et 6 sorties. Les poussoirs 2 et 3 ne peuvent avancer que lorsque le poussoir 1 est en arrière.

2. Identification des entrées et des sorties :

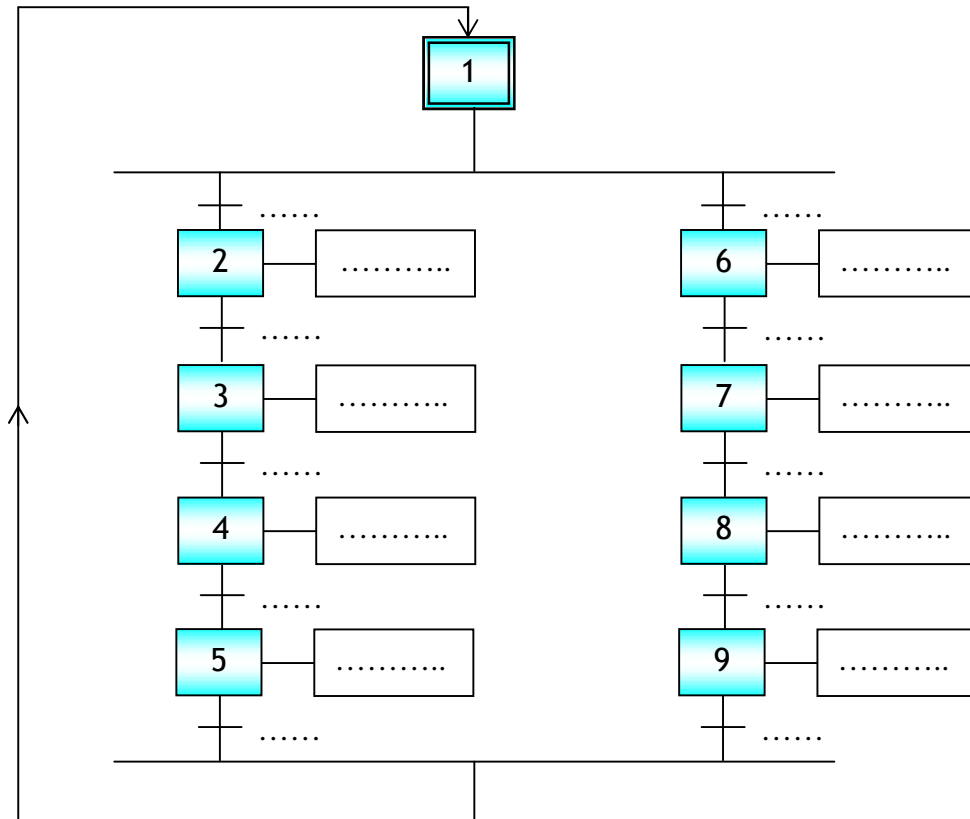
| Mouvement | Actionneur et action | | Ordres | Sortie API |
|----------------------------|-----------------------|-----------------|-------------------|----------------|
| Sortir la tige du vérin 1 | Poussoir 1 (Vérin) | SC ₁ | 14 M ₁ | Q ₁ |
| Rentrer la tige du vérin 1 | | RC ₁ | 12 M ₁ | Q ₂ |
| Sortir la tige du vérin 2 | Poussoir 2 (Vérin) | SC ₂ | 14 M ₂ | Q ₃ |
| Rentrer la tige du vérin 2 | | RC ₂ | 12 M ₂ | Q ₄ |
| Sortir la tige du vérin 3 | Poussoir 3 (Vérin) | SC ₃ | 14 M ₃ | Q ₅ |
| Rentrer la tige du vérin 3 | | RC ₃ | 12 M ₃ | Q ₆ |

| Compte-rendu | Capteur | Mnem. | Entrée API |
|--------------------|------------------------------|------------------|----------------|
| P1 en arrière | Détecteur mécanique à levier | L ₁₀ | I ₁ |
| P2 en arrière | Détecteur mécanique à levier | L ₂₀ | I ₂ |
| P3 en arrière | Détecteur mécanique à levier | L ₃₀ | I _B |
| Caisse sur tapis 2 | Détecteur mécanique à levier | L ₂₁ | I _C |
| Caisse sur tapis 3 | Détecteur mécanique à levier | L ₃₁ | I _D |
| Caisse devant P2 | Détecteur mécanique à levier | L ₁₁₂ | I _E |
| Caisse devant P3 | Détecteur mécanique à levier | L ₁₁₃ | I _F |
| Grande caisse | Détecteur mécanique à levier | G | I _G |
| Petite caisse | Détecteur mécanique à levier | P | I _H |

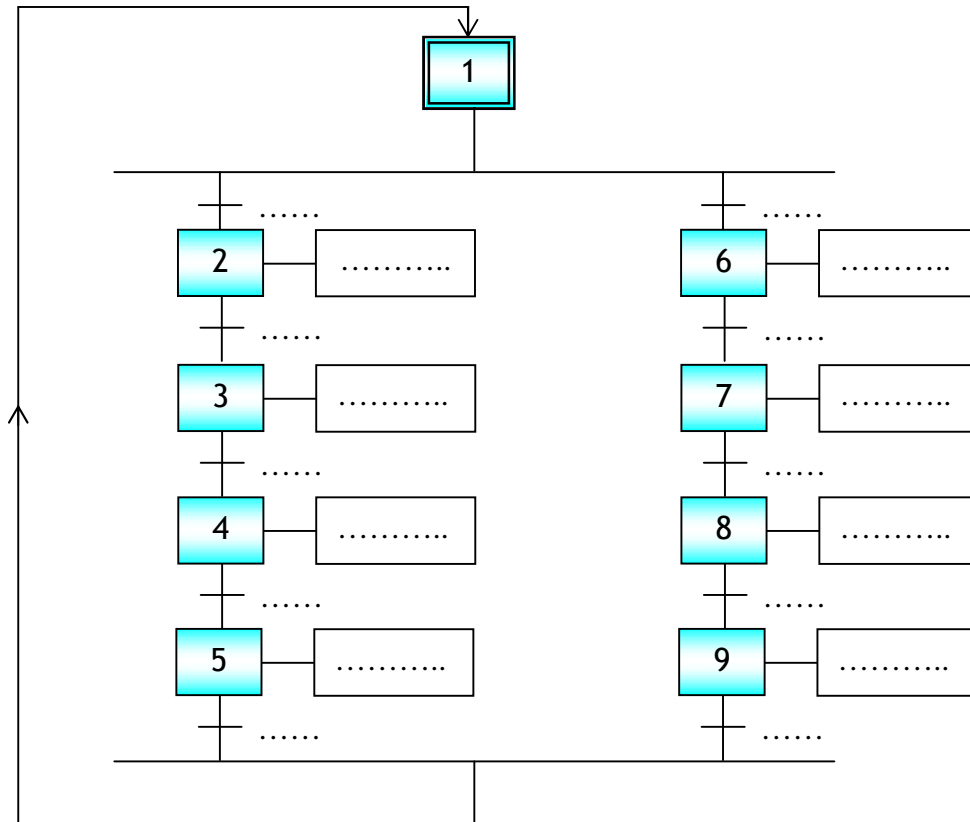
GRAFCET du point de vu P.O :



GRAFCET du point de vu P.C :



GRAFCET du point de vu P.C codé A.P.I :

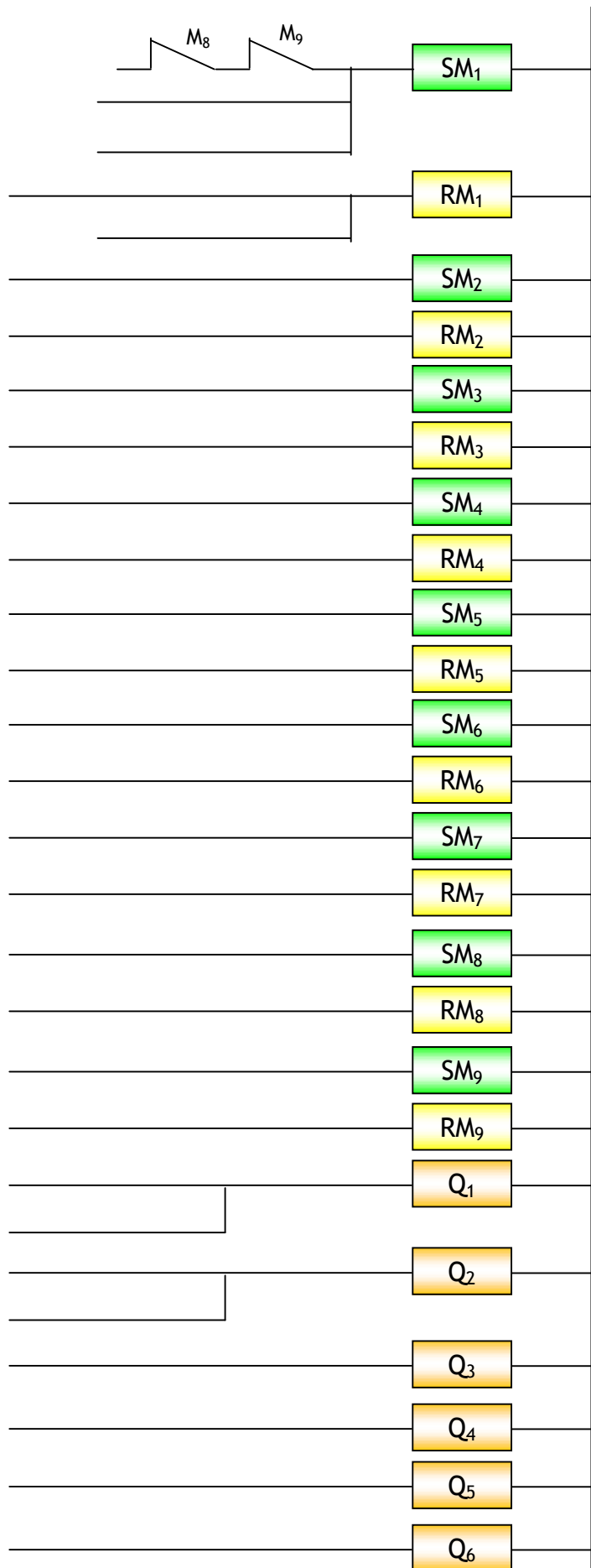


Mise en équation :

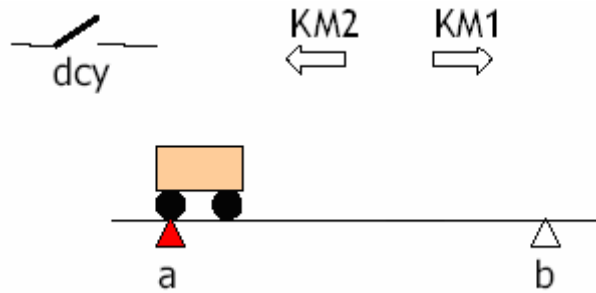
| Etapes | Activation | Désactivation |
|--------|-------------------------|-------------------------|
| 1 | SM ₁ = | RM ₁ = |
| 2 | SM ₂ = | RM ₁ = |
| 3 | SM ₃ = | RM ₂ = |
| 4 | SM ₄ = | RM ₃ = |
| 5 | SM ₅ = | RM ₄ = |
| 6 | SM ₆ = | RM ₅ = |
| 7 | SM ₇ = | RM ₆ = |
| 8 | SM ₈ = | RM ₇ = |
| 9 | SM ₉ = | RM ₉ = |

| | | |
|--------|---|---|
| Action | Q ₁ = 14M ₁ = | Q ₄ = 12M ₂ = |
| | Q ₂ = 12M ₁ = | Q ₅ = 14M ₃ = |
| | Q ₃ = 14M ₂ = | Q ₆ = 12M ₃ = |

Programme LADDER :



1. Cahier des charges :



Après l'ordre de départ cycle « dcy », le chariot part jusque b, reste en b un temps T_0 de 15s puis rentre en a. La commande du système est réalisée l'A.P.I ZELIO SR3101BD.

2. Identification des entrées et des sorties :

| Mouvement | Actionneur | Ordres | Sortie API |
|------------------------------|------------|-----------------|----------------|
| Déplacer le chariot à droite | Moteur | KM ₁ | Q ₁ |
| Déplacer le chariot à gauche | Moteur | KM ₂ | Q ₂ |

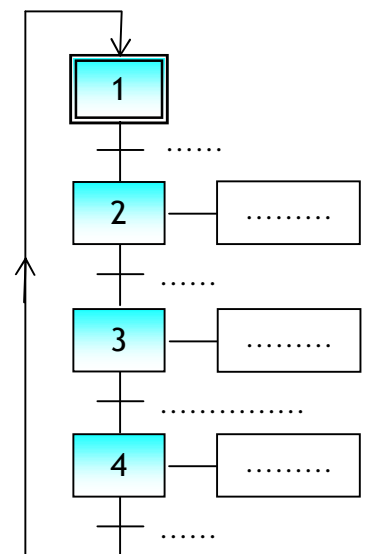
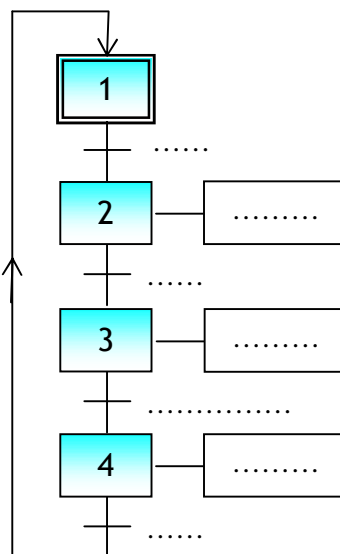
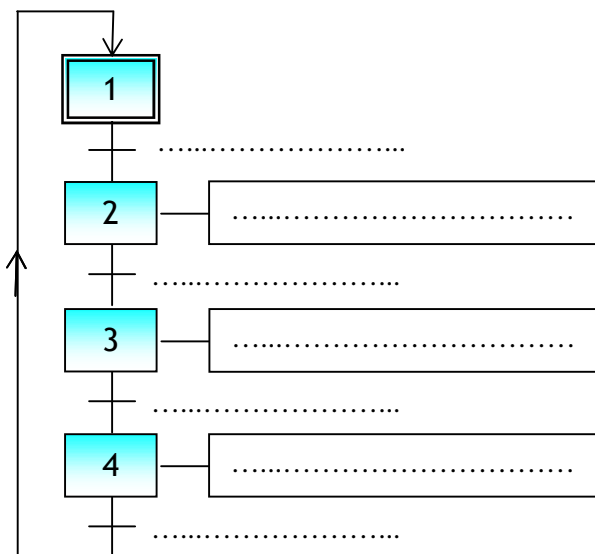
| Compte-rendu | Capteur | Mnem. | Entrée API |
|-------------------------|------------------------------|-------|----------------|
| chariot en position a | Détecteur mécanique à levier | a | I ₁ |
| chariot à en position b | Détecteur mécanique à levier | b | I ₂ |

| Consigne | Constituant | Mnem. | Entrée API |
|--------------|-----------------|-------|----------------|
| départ cycle | Bouton poussoir | Dcy | I _c |

GRAFSET P.O

GRAFSET P.C

GRAFSET P.C codé API



Mise en équation :

| Etapes | Activation | Désactivation |
|--------|-------------------------|-------------------------|
| 0 | SM ₁ = | RM ₁ = |
| 1 | SM ₂ = | RM ₂ = |
| 2 | SM ₃ = | RM ₃ = |
| 3 | SM ₄ = | RM ₄ = |

| | |
|---------|--|
| Actions | Q ₁ = KM ₁ = |
| | Q ₂ = KM ₂ = |
| | T ₀ = |

Programme LADDER :

