

Les langages de programmations.

Introduction :

L'écriture d'un programme consiste à créer une liste d'instructions permettant l'exécution des opérations nécessaires au fonctionnement du système.
Il existe différents types de langage de programmation.

Le langage booléen (**Logigramme**).

Le langage à contact (**Ladder**)

Le langage **GRAFCET**

Le langage LIST*

Le langage Littéral structuré*

L'API traduit le langage de programmation en langage compréhensible directement par le microprocesseur. Ce langage est propre à chaque constructeur, il est lié au matériel mis en œuvre.

Chaque instruction du programme est composée:

-de l'**opération** à effectuer (la nature de l'opération est codée 1 ou 0).

-de la **variable** sur laquelle l'opération va être effectué (variable de sortie, variable d'entrée, variable interne,...)

-de la nature de la variable (binaire, numérique, texte, ...)

Chaque instruction est écrite dans une partie de la mémoire appelée **adresse** ou **label**.

**: non étudier*

La programmation

Elle s'effectue au moyen d'un terminal de programmation qui permet également la mise en service ainsi que le réglage des variables internes.

Ce terminal de programmation peut être utilisé:

En **mode local** : Préparation du programme en bureau d'étude.

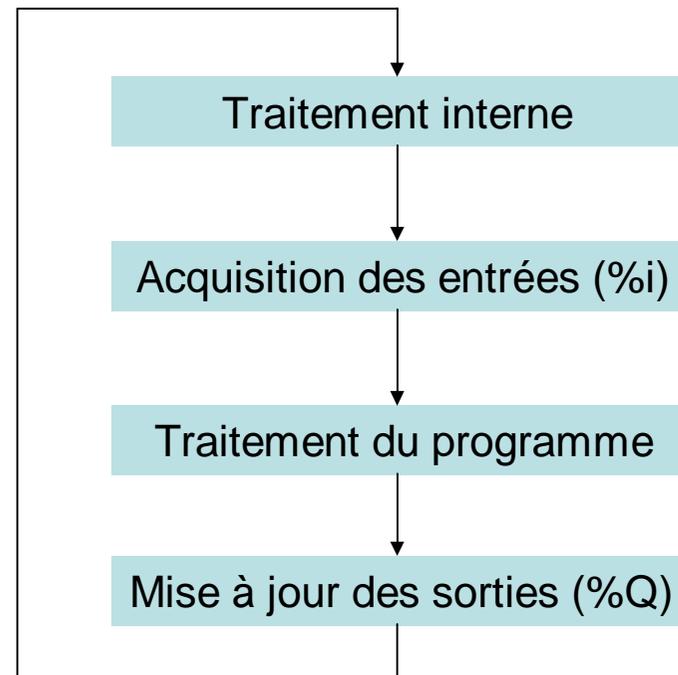
Ou **relié** à l'API afin de transférer le programme et de réaliser la mise en service.

Lecture du programme

En fonctionnement, le microprocesseur de l'API lit en permanence le programme et exécute les instructions les unes à la suite des autres. L'ensemble de ces tâches réalisées par le microprocesseur s'appelle une **scrutation**.

Une sortie spécifique de l'automate qui change d'état lorsqu'une défaillance de l'automate survient permet de couper le circuit d'arrêt d'urgence du système, cette sortie de sécurité est appelée **chien de garde**.

La lecture du programme est conditionné par la fonction **RUN / STOP** de l'API. (Marche/Arrêt)



Le langage Booléen (Logigramme)

Ce langage est basé sur l'écriture d'équation logique qui utilise quatre opérateurs de base

Nom	Schéma à contacts	Table de vérité	Equation	Logigramme															
Oui		<table border="1"> <tr><td>a</td><td>S</td></tr> <tr><td>0</td><td>0</td></tr> <tr><td>1</td><td>1</td></tr> </table>	a	S	0	0	1	1	$S=a$										
a	S																		
0	0																		
1	1																		
Non		<table border="1"> <tr><td>a</td><td>S</td></tr> <tr><td>0</td><td>1</td></tr> <tr><td>1</td><td>0</td></tr> </table>	a	S	0	1	1	0	$S = \bar{a}$ (ou /a)										
a	S																		
0	1																		
1	0																		
OU		<table border="1"> <tr><td>a</td><td>b</td><td>S</td></tr> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> </table>	a	b	S	0	0	0	0	1	1	1	1	1	1	0	1	$S = a + b$	
a	b	S																	
0	0	0																	
0	1	1																	
1	1	1																	
1	0	1																	
ET		<table border="1"> <tr><td>a</td><td>b</td><td>S</td></tr> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>0</td></tr> </table>	a	b	S	0	0	0	0	1	0	1	1	1	1	0	0	$S = a \cdot b$	
a	b	S																	
0	0	0																	
0	1	0																	
1	1	1																	
1	0	0																	

Ecriture d'un équation en langage Booléen

Il s'agit de l'écriture logique du fonctionnement. Il faut utiliser les quatre opérateurs de base.
La construction de cette équation logique peut se réaliser à partir de trois sources:

- **A partir d'un texte:** La description qui est faite permet normalement d'écrire l'équation.

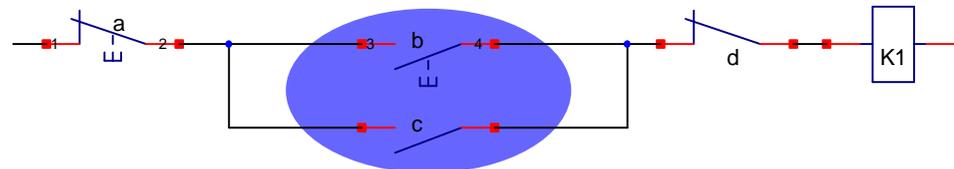
*Ex: La barrière d'un parking ne pourra s'ouvrir(KM1) que si une voiture est présente (S10) et si le ticket d'entrée n'est plus dans l'appareil (S7). Les mots importants sont soulignés.
L'équation du contacteur qui provoque l'ouverture de la barrière est:*

$$KM1 = S10 . /S7$$

(le point est le symbole de la fonction ET, le / est celui du complément n'est plus)

- **A partir d'un schéma électrique** dans lequel il faudra traduire le montage série en fonction ET (.) et les montages parallèles en fonction OU(+).

En regardant le schéma, il est possible de reconnaître la fonction OU (b+c) et la fonction ET avec a en série avec (b+c) en série avec d

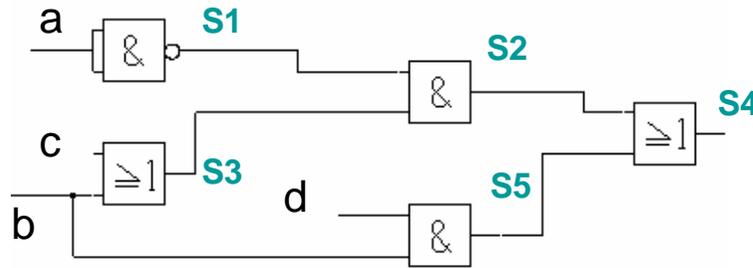


L'équation peut alors s'écrire : $K1 = a . (b + c) . d$

Ecriture d'un équation en langage Booléen

- A partir d'un logigramme:

Il est nécessaire de partir de la dernière fonction (généralement placée à droite du dessin) et d'écrire le résultat de chaque fonction logique du coté des entrées.



Exercice: $S4 = S2 + S5$

$$S2 = S1 \cdot S3$$

$$S5 = b \cdot d$$

$$S1 = \neg a$$

$$S3 = c + d$$

Donc

$$S2 = \neg a \cdot (c + d)$$

$$S4 = \neg a \cdot (c + d) + b \cdot d$$

Transformation d'équation et exercices

Commutativité / Associativité / Distributivité voir [logiciel](#)

Théorème De Morgan voir [logiciel](#)