

Master PAIP Mécatronique 1

Automatisme et Supervision, Premier cours 25/1/21

0 - présentation : Cours et TD en distanciel sur BBB. 4 TP en présentiel au Hall à Illkirch. Rappel des pré-requis (mais on les révise rapidement, je vous donne des liens vers mes cours si nécessaire) : [numérique](#) (binaire, hexa), [algèbre de Boole](#), [tableaux de Karnaugh](#), [portes logiques...](#)

1 - vocabulaire : <http://ptrau.free.fr/autom/autom01.htm> : PO/PC, E/S, actionneurs, capteurs, ToR/analogique/numérique, combinatoire/séquentiel).

2 – Combinatoire : machine distributeur à café : qu'on traitera au TP 1

Nous allons étudier un distributeur de café. Les produits sont tous en poudre. Trois produits principaux sont disponibles : café (noté K), café décaféiné (D), chocolat (C). Toute distribution fournira un de ces trois produits principaux. En plus, deux produits secondaires sont proposés : du lait (L) et du sucre (S). Le chocolat est toujours servi avec du lait et du sucre. Le "Capuccino" contient une dose K et une dose C (avec obligatoirement une dose de lait ainsi que de sucre).

La machine dispose d'un clavier à 16 touches, délivrant 5 signaux tout ou rien : le premier (nommé "ta" pour "touche appuyée") vaut 1 quand une touche valide est appuyée (0=pas de touche, ou plusieurs en même temps...); les 4 autres correspondent au numéro de la touche (en binaire, nommées t_3 , t_2 , t_1 et t_0). Le tableau de commande de notre distributeur propose les 10 options ci-contre :

produit	numéro de touche
café non sucré	0
café sucré	1
café au lait	2
café au lait sucré	3
décaféiné non sucré	4
décaféiné sucré	5
décaféiné au lait	6
décaféiné au lait sucré	7
chocolat	8
capuccino	9

Déterminez les équations de cinq sorties (nommées K, D, C, L et S) définissant les produits nécessaires à la préparation choisie, en fonction des 5 entrées.

Pour cela commencez par faire une table de vérité : 10 lignes, 4 colonnes pour les entrées t_3 à t_0 (pour moi, indice 0 pour l'unité, 1 pour la « deuzaine »... Analysez chaque sortie séparément en fonction des entrées. Nous ne traiterons pas les cas impossibles sur les entrées, car dans ce cas là ta ne serait pas validé.

N	ta	t3	t2	t1	t0	K	D	L	S	C
0	1	0	0	0	0	1	0	0	0	0
1	1	0	0	0	1	1	0	0	1	0
2	1	0	0	1	0	1	0	1	0	0
3	1	0	0	1	1	1	0	1	1	0
4	1	0	1	0	0	0	1	0	0	0
5	1	0	1	0	1	0	1	0	1	0
6	1	0	1	1	0	0	1	1	0	0
7	1	0	1	1	1	0	1	1	1	0
8	1	1	0	0	0	0	0	1	1	1
9	1	1	0	0	1	1	0	1	1	1

Tableau de Karnaugh de « K »

		t3 t2			
		00	01	11	10
t1 t0	K	00	01	11	10
	00	1	0	x	0
	01	1	0	x	1
	11	1	0	x	x
	10	1	0	x	x

$$K = ta.t2.(t3+t0)$$

Tableau de Karnaugh de « D »

		t3 t2			
		00	01	11	10
t1 t0	D	00	01	11	10
	00	0	1	x	0
	01	0	1	x	0
	11	0	1	x	x
	10	0	1	x	x

$$D = ta.t2$$

Tableau de Karnaugh de « L » « S » « C »

		t3 t2			
		00	01	11	10
t1 t0	L	00	01	11	10
	00	0	0	x	1
	01	0	0	x	1
	11	1	1	x	x
	10	1	1	x	x

$$L = ta.(t3+t1)$$

		t3 t2			
		00	01	11	10
t1 t0	S	00	01	11	10
	00	0	0	x	1
	01	1	1	x	1
	11	1	1	x	x
	10	0	0	x	x

$$S = ta.(t3+t0)$$

		t3 t2			
		00	01	11	10
t1 t0	C	00	01	11	10
	00	0	0	x	1
	01	0	0	x	1
	11	0	0	x	x
	10	0	0	x	x

$$C = ta.t3$$

personnellement, j'aurais plutôt mis t3 t2 à gauche et t1 t0 en haut, car c'est dans cet ordre que je les lis. Mais bien sûr le résultat est le même. Au niveau du résultat, la mise en facteur de t2 dans K n'est pas donnée par le tableau de Karnaugh, mais une simplification vue par après.

3 – mise en œuvre sur automates Siemens (en Step7)

Voir ma [doc Step7](#) sur moodle. Le langage CONT (contacts, appelé plus généralement Ladder) : des contacteurs (normaux ou inversés) mis en série (ET) ou parallèle (OU), sortie sur une bobine. Je définis au préalable mes « mnémonique » : affectation de noms (t0,t1,t2,t3,ta,K,D,L,C,S) aux E/S réelles (Ex.y entrées, Ax.y sorties)

Contenu de : 'Environnement\Interf...

Nom
TEMP

Obl : "Main Program Sweep (Cycle)"

Commentaire :

☐ Réseau 1 : calcul de D (Déca)

☐ Réseau 2 : calcul de L (Lait)

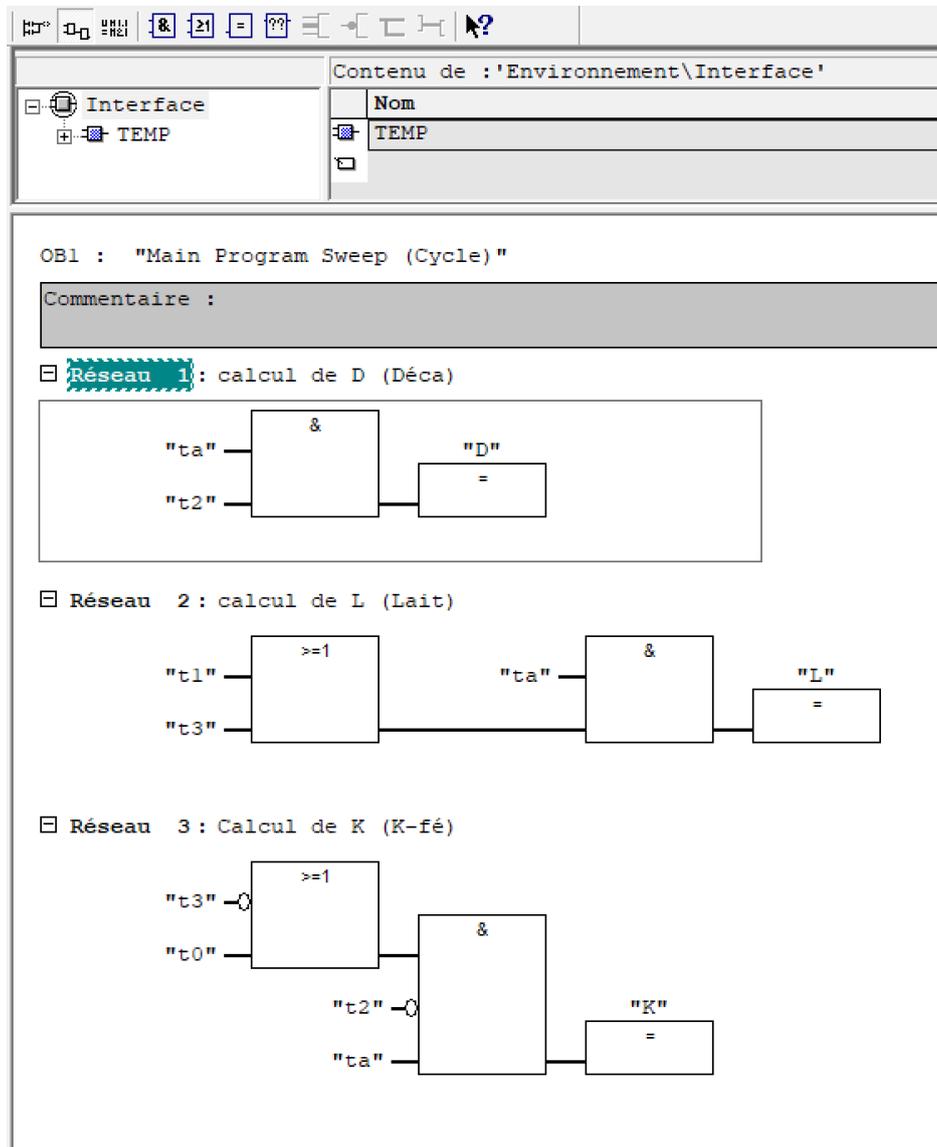
☐ Réseau 3 : Calcul de K (K-fé)

notez qu'au dessus de la fenêtre se trouvent les éléments les plus courants du langage : contacts, bobines, descente pour un « OU ».

Attention, un réseau par sortie. Si vous écrivez 2 réseaux pour la même sortie (en combinatoire), alors seul le dernier réseau déterminera l'état de la sortie. Comme en informatique, où écrire x=5 puis x=3 met en fin de compte 3 dans x. En effet, un automate scrute (lit) toutes les entrées, mémorise leur état, puis exécute le programme, puis affecte (écrit) les sorties en fonction de leur état final. Puis il recommence le cycle. Le temps de cycle, sur un automate, se compte en millisecondes. Si vous voulez aller plus vite, c'est possible, surtout sur un ordinateur (microsecondes voire nanosecondes su à 1 GHz), comme par exemple un raspberry.

Le langage CONT est bien adapté au ToR (tout ou rien) combinatoire, avec uniquement ET, OU, complément.

Autre solution : le langage LOG (portes logiques)



repérez et, ou, complément, =. Ici aussi, un réseau par sortie. Le langage LOG est adapté au ToR avec des fonctions plus complexes (ou exclusif, bascules...)

Dernière solution : le LIST (langage mnémonique) : c'est le « langage machine » de l'automate. Sa forme est textuelle, avec des lignes comportant toujours un opérateur suivi d'un opérande. Chaque calcul se fait entre le résultat précédent et l'opérande. Sur une calculatrice ce serait :

- entrer 5
- additionner 2
- soustraire 4
- multiplier par 2
- stocker dans la mémoire 1

ici les opérateurs en ToR combinatoire sont U (Und, ET), O (Oder, OU), UN (et pas), ON (ou pas), = (stocker). On peut utiliser des parenthèses, ou organiser le calcul dans le bon ordre pour les éviter. On peut séparer en réseaux, mais ça ne permet que d'améliorer la lisibilité.

☐ Réseau 1 : calcul de D (Déca)

```
U    "ta"  
U    "t2"  
=    "D"
```

☐ Réseau 2 : Calcul de L (lait)

```
O    "t3"  
O    "t1"  
U    "ta"  
=    "L"
```

☐ Réseau 3 : Calcul de K (K-fé)

```
U(  
ON  "t3"  
O   "t0"  
)  
UN  "t2"  
U   "ta"  
=   "K"
```