

TP AUTOMATISME ET SUPERVISION

TP1 : Machine distributeur à café - combinatoire en STEP 7

Table des matières

I.	Combinatoire : Machine distributeur à café :	2
A.	Equation des sorties en fonction des entrées :	3
B.	Configuration	4
C.	Programme LOG	4
D.	Programme CONT	5
II.	Séquentiel :	9
IV.	Numérique	11
A.	LIST	11
V.	ANNEXES	12
A.	Programme Langage LOG	12
B.	Programme Langage CONT	16
C.	Programme Langage LIST	20

I. Combinatoire : Machine distributeur à café :

L'objectif ici est de réaliser un programme automate qui puisse réaliser les fonctions suivantes, suivant un clavier 16 touches. Celui-ci fonctionnera en tout ou rien.

Pour le moment le comportement est combinatoire c'est-à-dire que les sorties ne dépendent que des entrées.

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

- On a alors la table de vérité suivante : Avec « ta » la touche de validation

ta	E3	E2	E1	E0		K	D	C	L	S
1	0	0	0	0		1	0	0	0	0
1	0	0	0	1		1	0	0	0	1
1	0	0	1	0		1	0	0	1	0
1	0	0	1	1		1	0	0	1	1
1	0	1	0	0		0	1	0	0	0
1	0	1	0	1		0	1	0	0	1
1	0	1	1	0		0	1	0	1	0
1	0	1	1	1		0	1	0	1	1
1	1	0	0	0		0	0	1	1	1
1	1	0	0	1		1	0	1	1	1

A. Equation des sorties en fonction des entrées :

K	E1E0				
E3E2	00	01	11	10	
00	1	1	1	1	
01	0	0	0	0	
11	X	X	X	X	
10	0	1	X	X	

$$K = ta. \overline{E2}(\overline{E3} + E0)$$

L	E1E0				
E3E2	00	01	11	10	
00	0	0	1	1	
01	0	0	1	1	
11	X	X	X	X	
10	1	1	X	X	

$$L = ta. (\overline{E3}.E1 + E3.\overline{E1}.\overline{E2})$$

S	E1E0				
E3E2	00	01	11	10	
00	0	1	1	0	
01	0	1	1	0	
11	X	X	X	X	
10	1	1	X	X	

$$S = ta. (\overline{E3}.E0 + E3.\overline{E1}.\overline{E2})$$

Pour la sortie C, elle n'est active que lorsque l'entrée E3 ainsi que la validation sont actives :

$$C = ta. E3$$

Pour la sortie D, elle n'est active que lorsque l'entrée E2 ainsi que la validation sont actives :

$$D = ta. E2$$

B. Configuration

- Au préalable définir le matériel à l'aide des références figurantes sur les façades des modules sur le rack Siemens et configurer les réseaux en fonction de notre numéro de valise (tutoriel du TP).
 - Pour programmer nous allons utiliser le langage LOG et programmer dans l'OB1 (Organisation Block) l'OB1 est cyclique et déclenche ainsi le traitement cyclique du programme utilisateur. Le traitement de programme cyclique constitue le traitement normal pour les automates programmables. Pour les programmes plus complexes nous utilisons l'OB1 pour appeler d'autres blocs.
 - Pour le premier chargement il faut également charger la configuration matérielle que nous avons établie, dans ce TP cela ne nous posera pas de problème mais il faut faire attention car le chargement d'une configuration matérielle fait passer l'automate en « STOP ».
 - Nous avons aussi renseigné les mnémoniques avec des noms explicites qui facilitent la programmation et la manipulation des E/S dans le programme. Nous avons également ajouté des commentaires pour expliquer la fonction de chaque entrée et sortie.
 - Nous allons travailler sur STEP7 un logiciel permettant de programmer des automates de la marque Siemens.
- Les entrées TOR se notent E a.b (E= Eingang)
- Les sorties TOR se notent A a.b (A= Ausgang)
- Avec a l'adresse du module, b étant le numéro du bit dans l'octet (entre 0 et 7)

C. Programme LOG

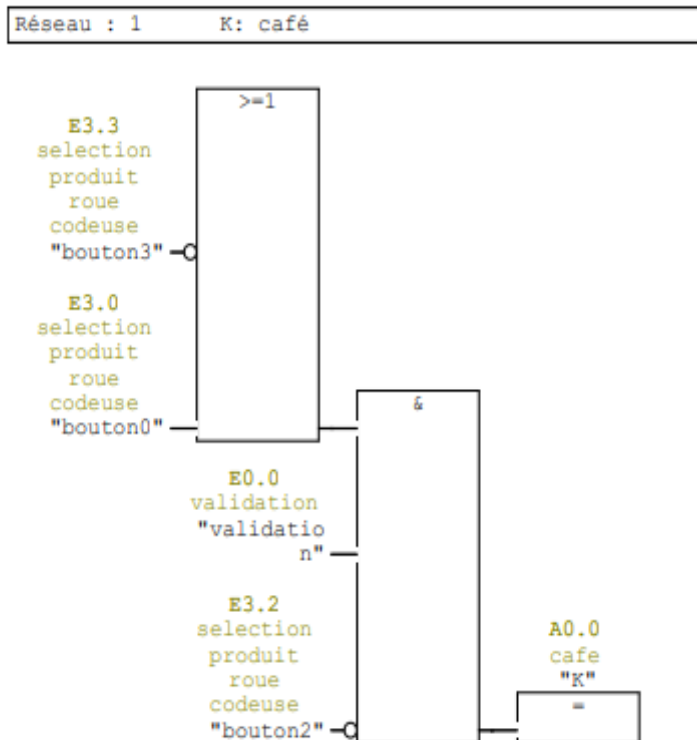
Nous pouvons commencer à programmer ses sorties en langage LOG.

- Le LOG est un type de programmation proche des symboles utilisés par les électroniciens, il utilise les portes logiques (ET, OU etc.)
- Pour cela on s'appuiera sur les équations de sorties trouvées à l'aide des tableaux de Karnaugh.
- Il faut prendre la précaution de ne pas faire figurer une sortie plusieurs fois dans le programme. Un automate sous Siemens scrute le programme de haut en bas et donc ce sera la sortie la plus basse dans le programme qui sera prise en compte.
- Le programme réalisé se déroule étape par étape, à la manière d'un Grafcet, mais en moins performant. Ainsi qu'une programmation plus laborieuse.

D. Programme CONT

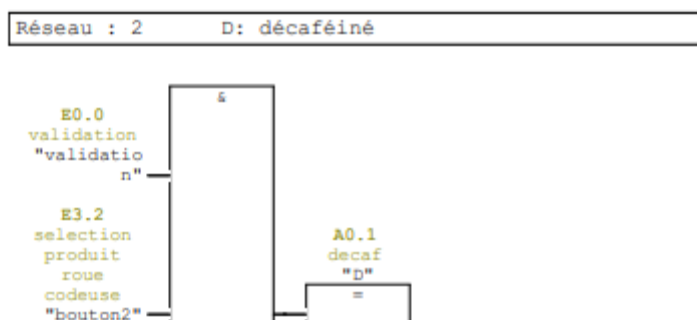
- Le CONT aussi appelé LADDER est un type de programmation proche des symboles utilisés par les électriciens, Il utilise des symboles rappelant la structure d'un schéma électrique, ce qui rend sa lecture relativement graphique et simple.
- Les entrées sont représentées par des interrupteurs -||- (Contact normalement ouvert) ou -||/- si l'entrée est inversée (contact normalement fermés), les sorties par des bobines -() ou des bascules -(S) -(R).
- C'est une suite de réseaux parcourus séquentiellement.
- On relie les éléments en série pour la fonction ET et en parallèle pour le OU.
- On peut utiliser des bits internes pour stocker un résultat intermédiaire. On peut les utiliser comme des bobines ou des interrupteurs.
- Il faut prendre la précaution de ne pas faire figurer une sortie plusieurs fois dans le programme. Un automate sous Siemens scrute le programme de haut en bas et donc ce sera la sortie la plus basse dans le programme qui sera pris en compte.

Pour la sortie K :

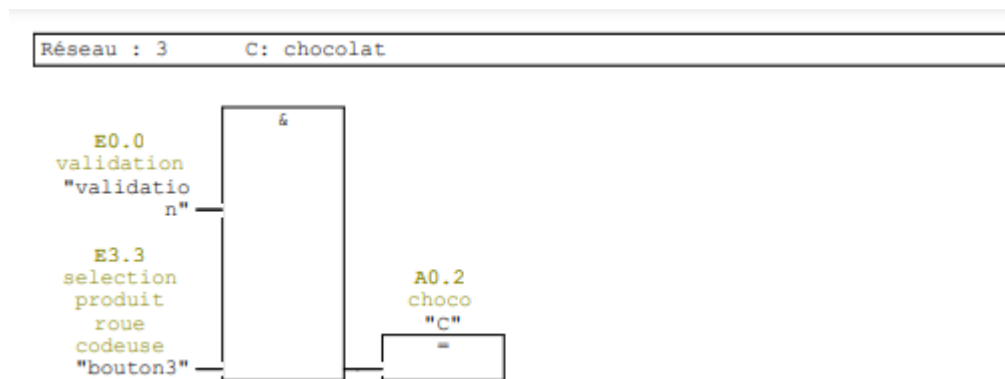


- « validation » symbolise « ta » dans les équations du [tableau de Karnaugh](#).

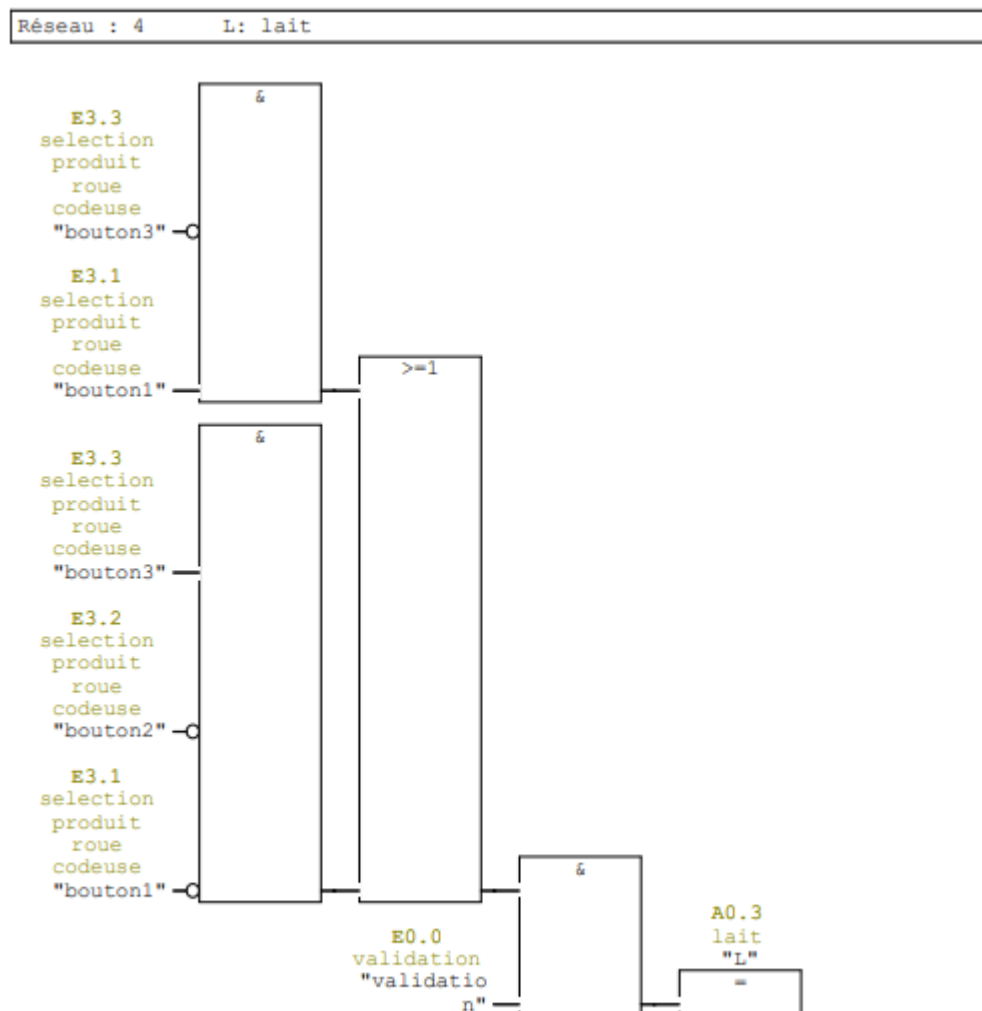
Pour la sortie D :



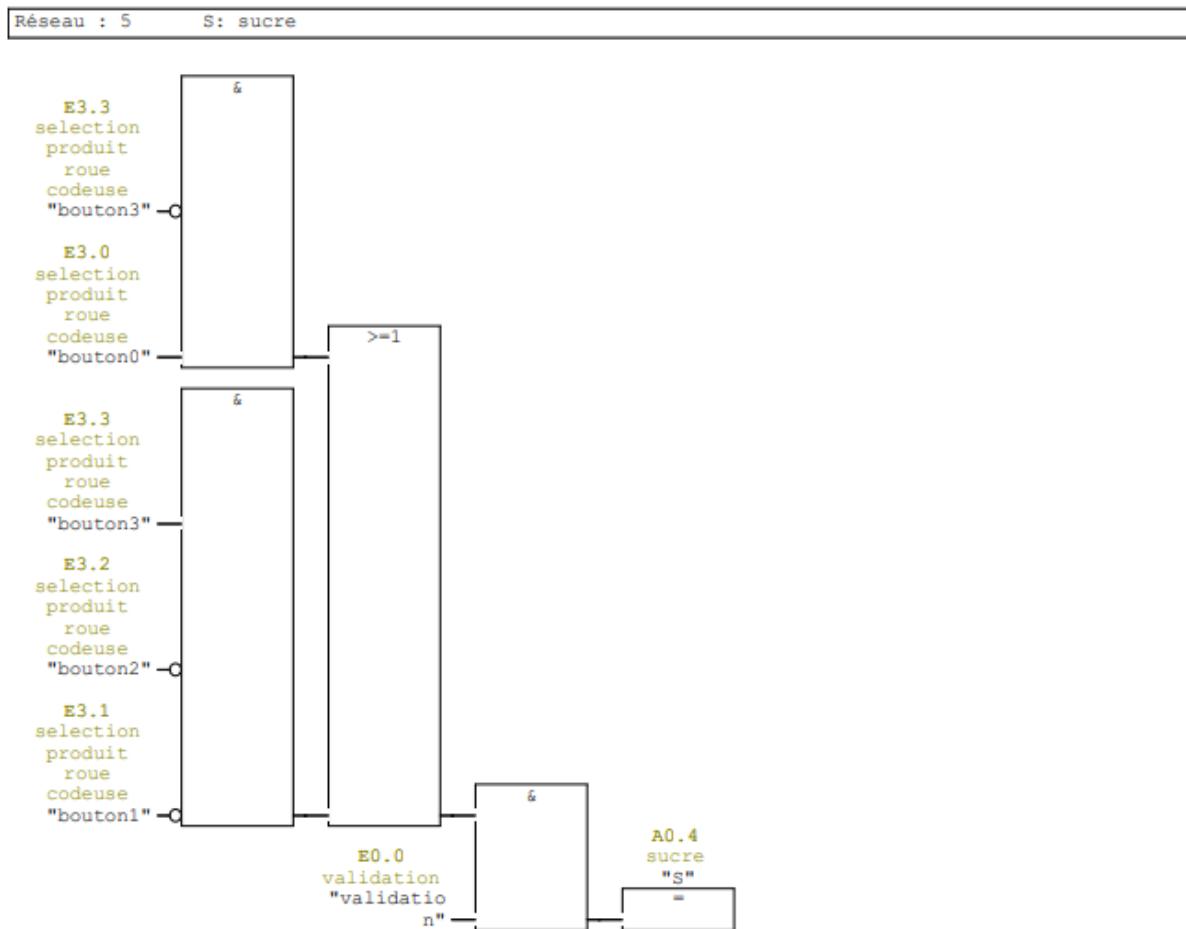
Pour la sortie C :



Pour la sortie L :

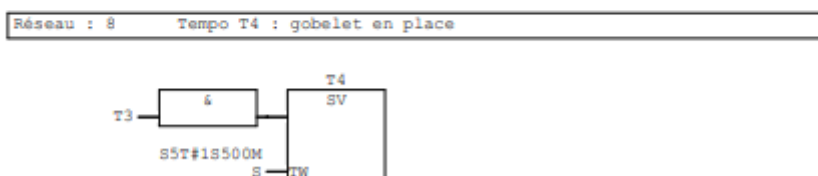
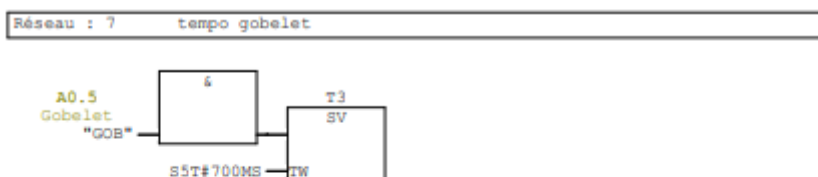
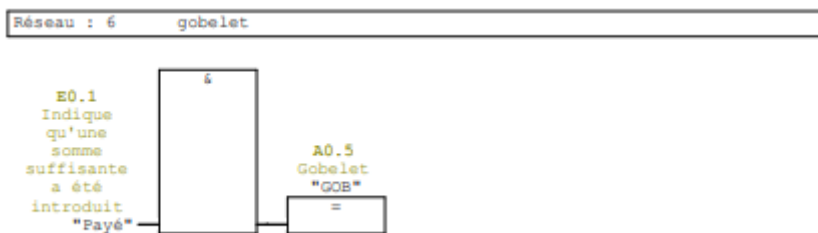


Pour la sortie S :



II. Séquentiel :

- Le comportement est séquentiel c'est-à-dire que les sorties dépendent à la fois des entrées mais aussi des sorties.
- Dans cette partie on ajoute les sorties VK+, VD+, VC+, VL+, VS+ et VE+ désignant les vannes d'eau chaude et de café, décaféiné, chocolat, lait et sucre. Ici on se contentera de la vanne d'eau chaude et de café, car la programmation pour les autres vannes reste la même. On ajoute aussi une sortie gobelet et une entrée paiement.
- L'entrée paiement permet de démarrer le déroulement du programme de l'automate :

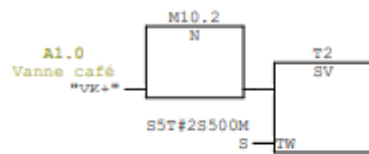


- Ensuite, on peut ouvrir pendant 0.4s la vanne de café lorsque l'entrée café est activée.

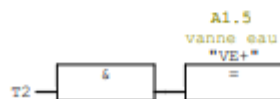


- Une fois les 0.4 secondes passées, on ouvre la vanne d'eau chaude pendant 2.5s. Une fois cela fait, on considère le cycle comme étant fini.

Réseau : 11



Réseau : 12



- Nous avons eu recours à une temporisation de type SV avec détection de front descendant (ce qui permet à l'utilisateur de maintenir le bouton sans démarrer la temporisation. Il existe bien évidemment d'autre solution pour réaliser cette temporisation par exemple avec des compteurs mais nous avons décidé d'utiliser cette méthode car cela permet d'explorer les outils proposer par STEP7.

IV. Numérique

Pour le numérique nous utiliserons le langage « LIST » c'est un programme proche de l'assembleur, il permet de réaliser des calculs et des actions plus complexes, mais au prix d'une interface moins visuelle surtout en visualisation dynamique.

Le système sait toujours traduire du CONT ou LOG en LIST mais pas forcément l'inverse.

A. LIST

Dans notre programme nous utilisons l'octet interne MB20 car nous avons déjà utilisé MB10 plus tôt dans le programme.

Réseau : 13 comptage credit

```
U      "insertion 5cts"  E0.2
FN     M      10.5
SPBN   X1
L      "unité"          MB20      -- comptage crédit
L      1
+I
T      "unité"          MB20      -- comptage crédit
```

- Le réseau 13 permet sur un front descendant d'incrémenter les crédits, le front descendant est obligatoire sinon les crédits seraient incrémentés à la vitesse du temps de cycle de l'automate.
- Le programme se lit :
 - Lorsque l'automate programmable détecte un front descendant au contact E 0.2, RLG = 1 (U venant de l'allemand UND).
 - Saut au repère X1 si RLG (résultat logique) égale 0.
 - La séquence de programme se poursuit ici si le saut ne s'exécute pas :
Charger valeur « unité » de comptage en cours comme entier.
 - Charger valeur « 1 » de comptage en cours comme entier.
 - Additionne 1 à la mémoire MB20
 - Transfère le résultat dans l'octet de memento MB20

Réseau : 14 monnayeur

```
x1:  L      "unité"  MB20      -- comptage crédit
      L      7
      >I
      =      "paye2"  M21.0    -- memoire paiement
```

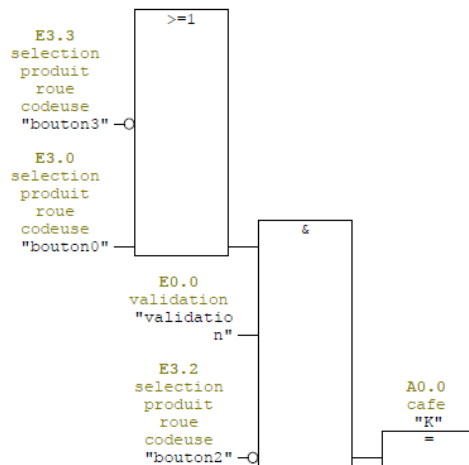
- Charger valeur « unité » de comptage en cours comme entier.
- Charger valeur « 7 » de comptage en cours comme entier.
- Comparer si « unité » est supérieur à « 7 » (Note : il faudrait remplacer « >I » par « >=I » pour que le prix du café soit bien de 35 cts car sinon il serait de 40 cts)
- Paye2 = 1 si « unité » >= « 7 »

V. ANNEXES

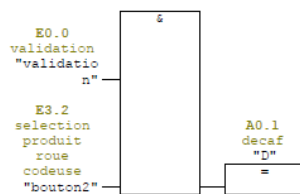
A. Programme Langage LOG

Bloc : OB1 "Main Program Sweep (Cycle)"

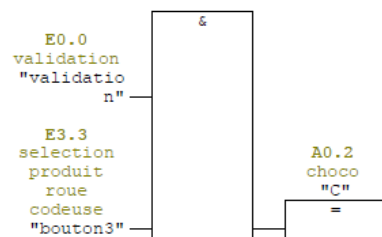
Réseau : 1 K: café



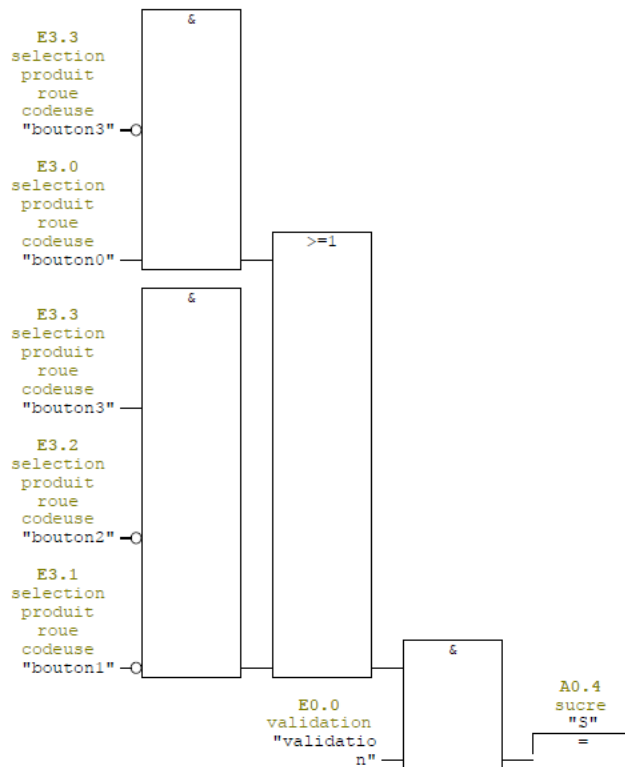
Réseau : 2 D: décaféiné



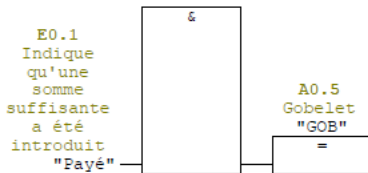
Réseau : 3 C: chocolat



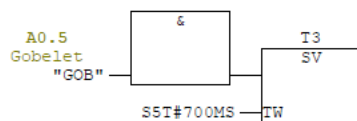
Réseau : 5 S: sucre



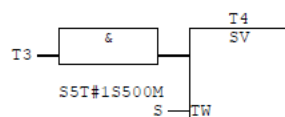
Réseau : 6 gobelet



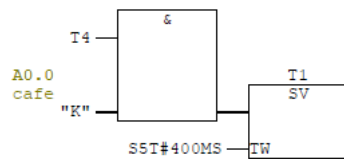
Réseau : 7 tempo gobelet



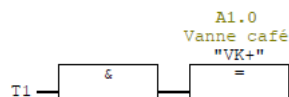
Réseau : 8 Tempo T4 : gobelet en place



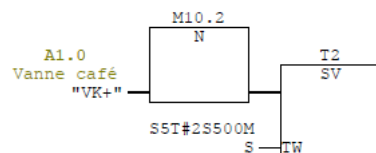
Réseau : 9 Tempo café



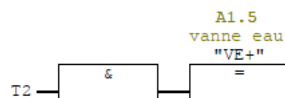
Réseau : 10 Vanne café



Réseau : 11



Réseau : 12



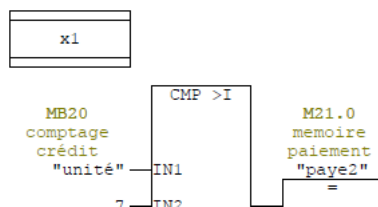
Réseau : 13 comptage credit

```

U      "insertion Scts"  E0.2
FN      M      10.5
SPBN    X1
L      "unité"          MB20      -- comptage crédit
L      1
+I
T      "unité"          MB20      -- comptage crédit

```

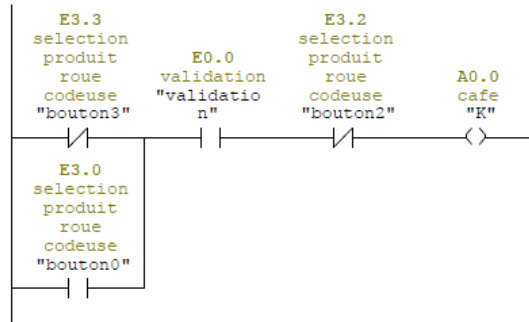
Réseau : 14 monnayeur



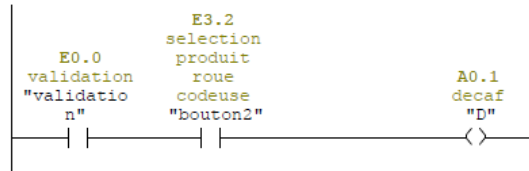
B. Programme Langage CONT

Bloc : OB1 "Main Program Sweep (Cycle)"

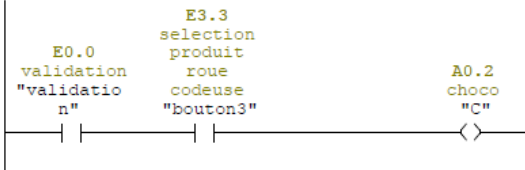
Réseau : 1 K: café



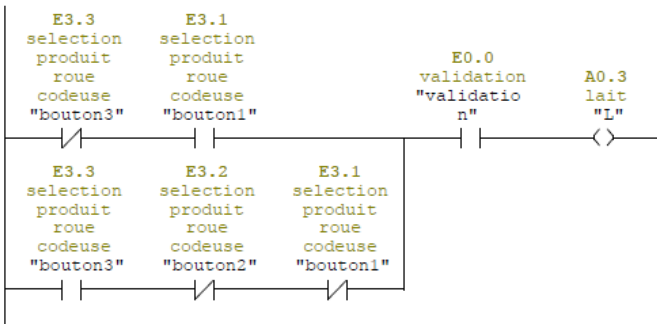
Réseau : 2 D: décaféiné



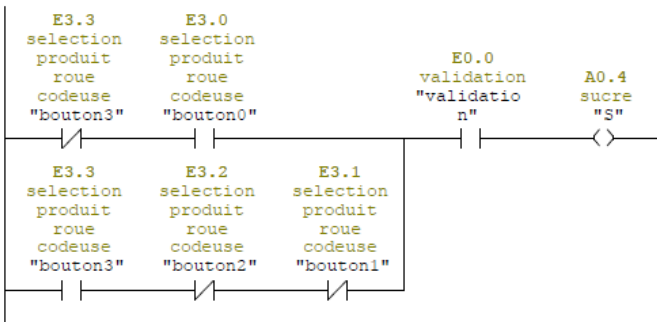
Réseau : 3 C: chocolat



Réseau : 4 L: lait



Réseau : 5 S: sucre



Réseau : 6 gobelet



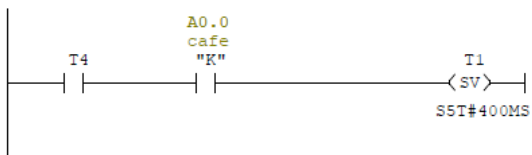
Réseau : 7 tempo gobelet



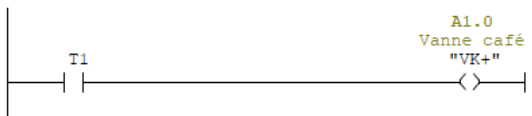
Réseau : 8 Tempo T4 : gobelet en place



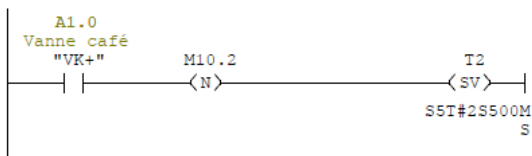
Réseau : 9 Tempo café



Réseau : 10 Vanne café



Réseau : 11



Réseau : 12

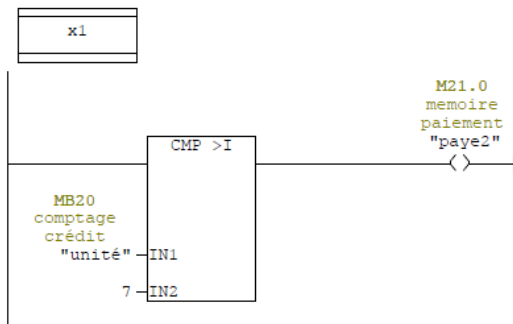


Réseau : 13 comptage credit

```

U      "insertion 5cts"  E0.2
FN     M      10.5
SPBN   X1
L      "unité"          MB20      -- comptage crédit
L      1
+I
T      "unité"          MB20      -- comptage crédit
  
```

Réseau : 14 monnayeur



C. Programme Langage LIST

Bloc : OB1 "Main Program Sweep (Cycle) "

Réseau : 1 K: café

```
U(  
ON  "bouton3"    E3.3      -- selection produit roue codeuse  
O   "bouton0"    E3.0      -- selection produit roue codeuse  
)  
U   "validation" E0.0      -- validation  
UN  "bouton2"    E3.2      -- selection produit roue codeuse  
=   "K"          A0.0      -- cafe
```

Réseau : 2 D: décaféiné

```
U   "validation" E0.0      -- validation  
U   "bouton2"    E3.2      -- selection produit roue codeuse  
=   "D"          A0.1      -- decaf
```

Réseau : 3 C: chocolat

```
U   "validation" E0.0      -- validation  
U   "bouton3"    E3.3      -- selection produit roue codeuse  
=   "C"          A0.2      -- choco
```

Réseau : 4 L: lait

```
U(  
UN  "bouton3"    E3.3      -- selection produit roue codeuse  
U   "bouton1"    E3.1      -- selection produit roue codeuse  
O  
U   "bouton3"    E3.3      -- selection produit roue codeuse  
UN  "bouton2"    E3.2      -- selection produit roue codeuse  
UN  "bouton1"    E3.1      -- selection produit roue codeuse  
)  
U   "validation" E0.0      -- validation  
=   "L"          A0.3      -- lait
```

Réseau : 5 S: sucre

```
U(
UN  "bouton3"  E3.3      -- selection produit roue codeuse
U   "bouton0"  E3.0      -- selection produit roue codeuse
O
U   "bouton3"  E3.3      -- selection produit roue codeuse
UN  "bouton2"  E3.2      -- selection produit roue codeuse
UN  "bouton1"  E3.1      -- selection produit roue codeuse
)
U   "validation" E0.0    -- validation
=   "S"        A0.4      -- sucre
```

Réseau : 6 gobelet

```
U   "Payé"  E0.1      -- Indique qu'une somme suffisante a été introduit
=   "GOB"   A0.5      -- Gobelet
```

Réseau : 7 tempo gobelet

```
U   "GOB"    A0.5      -- Gobelet
L   SST#700MS
SV  T        3
```

Réseau : 8 Tempo T4 : gobelet en place

```
U   T        3
L   SST#1S500MS
SV  T        4
```

Réseau : 9 Tempo café

```
U   T        4
U   "K"      A0.0      -- cafe
L   SST#400MS
SV  T        1
```

Réseau : 10 Vanne café

```
U   T        1
=   "VK+"    A1.0      -- Vanne café
```

Réseau : 11

```
U   "VK+"    A1.0      -- Vanne café
FN  M        10.2
L   SST#2S500MS
SV  T        2
```

Réseau : 12

```
U   T        2
=   "VE+"    A1.5      -- vanne eau
```

Réseau : 13 comptage credit

```
U   "insertion 5cts" E0.2
FN  M        10.5
SPBN X1
L   "unité"        MB20      -- comptage crédit
L   1
+I
T   "unité"        MB20      -- comptage crédit
```

Réseau : 14 monnayeur

```
x1: L   "unité"  MB20      -- comptage crédit
    L   7
    >I
    =   "paye2"  M21.0     -- memoire paiement
```