

Semestre2 – Automatismes1

Introduction – Programmation des automates

- ❖ **1 Introduction**
- ❖ **2 Le langage LADDER**
- ❖ **3 Le langage GRAFCET**
- ❖ **4 Le langage structuré**

- ❖ **1 Introduction**
- ❖ 2 Le langage LADDER
- ❖ 3 Le langage GRAFCET
- ❖ 4 Le langage structuré

1.1 Objets manipulés

En fonction de sa puissance, un API possède

Des signaux d'entrées/sorties physiques

- Des entrées logiques (accessibles bit par bit ou par mot)
- Des sorties logiques (accessibles bit par bit ou par mot)
- Des entrées analogiques
- Des sorties PWM
- Des entrées de comptage
-

De la mémoire interne pour le stockage de données
(transition, étapes, calculs intermédiaires...)

- Bits ou mots « système » (ex First_Cycle)
- Bits ou mots réservés pour certains périphériques (ex FinTimer)
- Bits ou mots utilisables pour les données, mémorisés ou non suite à une coupure d'alimentation.

1.2 Table des entrées/sorties et mémoire interne

La Table des E/S permet d'associer à chaque symbole un emplacement physique ou en mémoire.

Name	Type	Address	\	Comment
VC4	EB00L	%Q0.3.9		Voyant cabine niveau 4
VC3	EB00L	%Q0.3.15		Voyant cabine niveau 3
VC2	EB00L	%Q0.3.10		Voyant cabine niveau 2
VC1	EB00L	%Q0.3.11		Voyant cabine niveau 1
VA4	EB00L	%Q0.3.12		Voyant d'appel niveau 4
VA3	EB00L	%Q0.3.14		Voyant d'appel niveau 3
VA2	EB00L	%Q0.3.13		Voyant d'appel niveau 2
VA1	EB00L	%Q0.3.8		Voyant d'appel niveau 1
I10_11	BOOL			
p1	EB00L	%I0.1.4		Presence niveau 1
N	INT			
DP4	EB00L	%I0.1.0		Presence niveau 4
DP3	EB00L	%I0.1.1		Presence niveau 3
DP2	EB00L	%I0.1.2		Presence niveau 2
Cde_Mont	EB00L	%Q0.3.1		Commande de montée du moteur cabine
Cde_Desc	EB00L	%Q0.3.0		Commande de descente du moteur cabine
Bp_Manu	EB00L	%I0.1.12		Bp marche manu
BP_bl	EB00L	%I0.2.0		Bouton Blanc pupitre
Bp_Auto	EB00L	%I0.1.11		Bp marche Auto
BP4d	EB00L	%I0.1.5		Appel niveau descente 4
BP4	EB00L	%I0.1.14		Appel Cabine niveau 4
BP2m	EB00L	%I0.1.6		Appel niveau montage 2

Sortie 1 bit

Bit interne

Mot interne

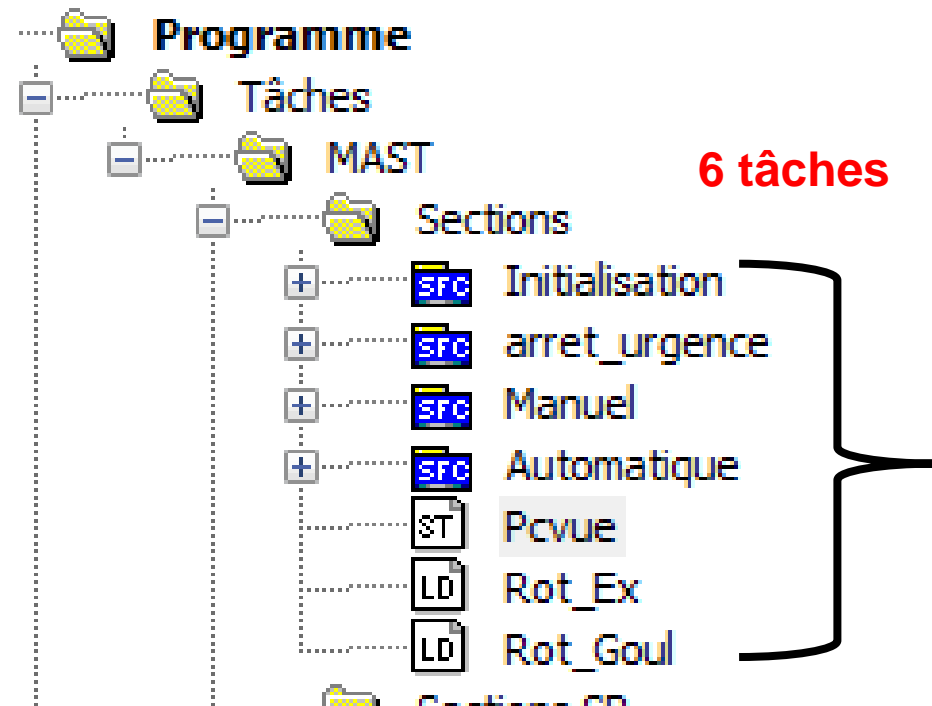
Entrée 1 bit

1.3 Comment l'API exécute-il les programmes ?

- Le programme automate est organisé en une liste de tâches, écrites dans un des langages autorisés

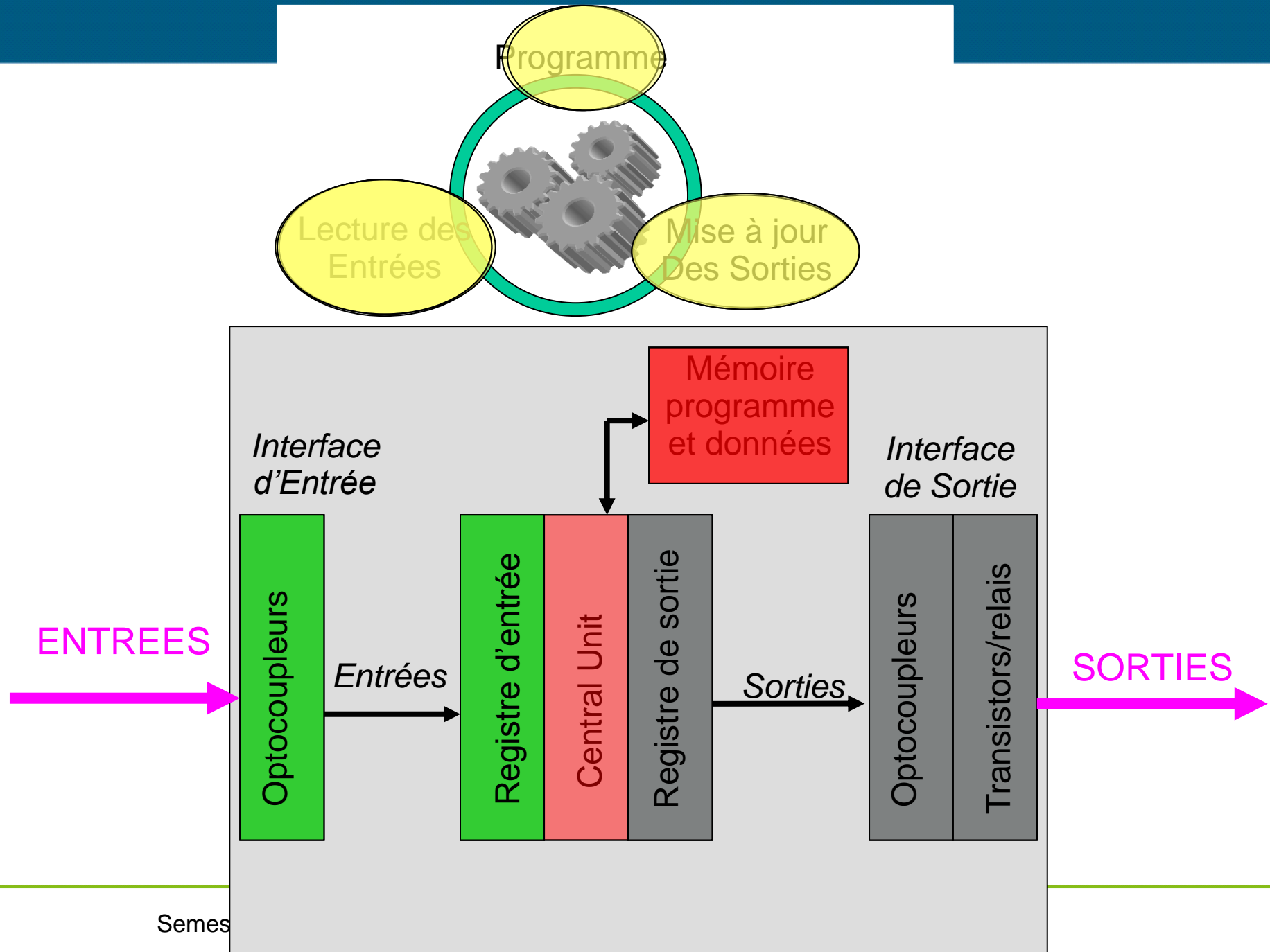
```
(*Capteurs*)
%M7 := %IO.1.6; (*fin de course goulotte*)
%M8 := %IO.1.7; (*fin de course exterieur*)
%M9 := %IO.1.8; (*pince fermée*)
%M10 := %IO.1.9; (*pince ouverte*)
%M11 := %IO.1.10; (*rotation horaire*)
%M12 := %IO.1.11; (*rotation anti horaire*)
%M13 := %IO.1.12; (*verin vertical rentrée*)
```

Une tâche



- L'automate exécute les instructions séquentiellement une tâche après l'autre,
 - pour une tâche donnée, dans l'ordre des instructions
 - une tâche après l'autre, dans l'ordre d'écriture des tâches
- Puis il recommence**
- Le cycle dure entre 1ms et 10ms
- Au temps de cycle près, on considère que les tâches sont exécutées « en même temps »

1.4 Le cycle de l'automate

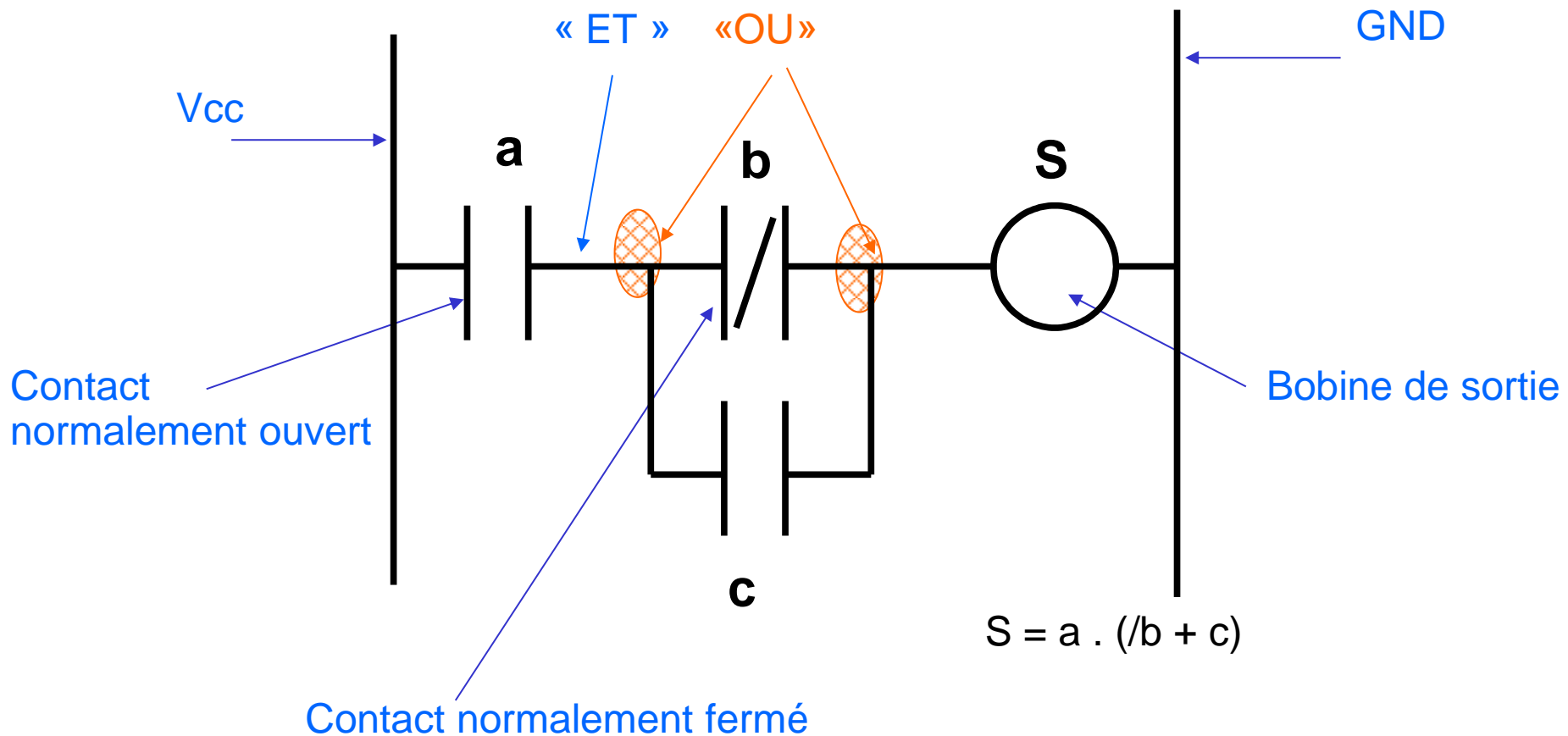


Le Langage LADDER

- ❖ 1 Introduction
- ❖ 2 Le langage LADDER
- ❖ 3 Le langage GRAFCET
- ❖ 4 Le langage structuré

2.1 Le langage LADDER

Diagramme à contacts, utilisé pour programmer des éléments combinatoires



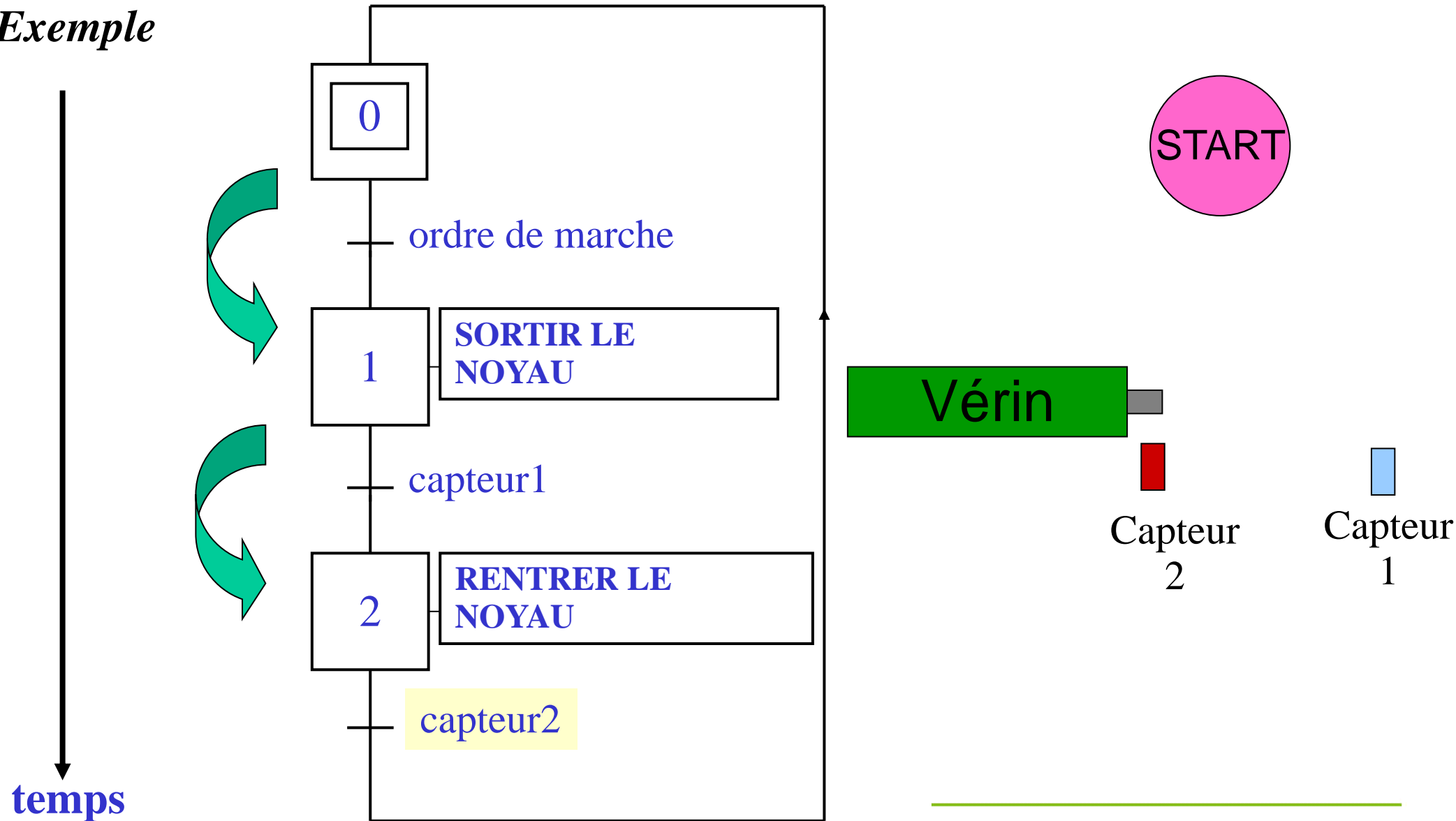
Voir le cours précédent

Le langage GRAFCET

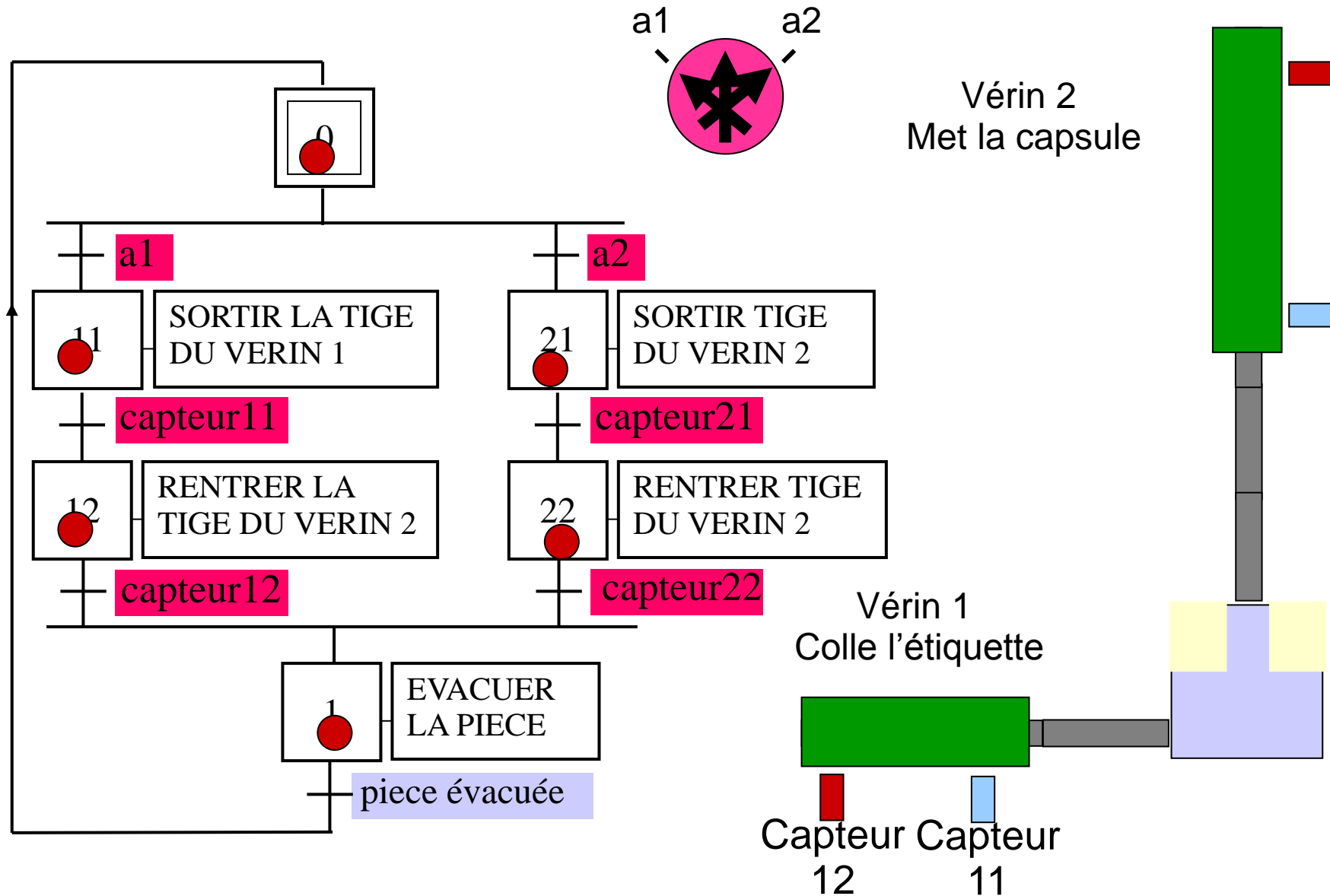
- ❖ 1 Introduction
- ❖ 2 Le langage LADDER
- ❖ 3 Le langage **GRAFCET**
- ❖ 4 Le langage structuré

3.1 Langage GRAFCET (1)

Exemple

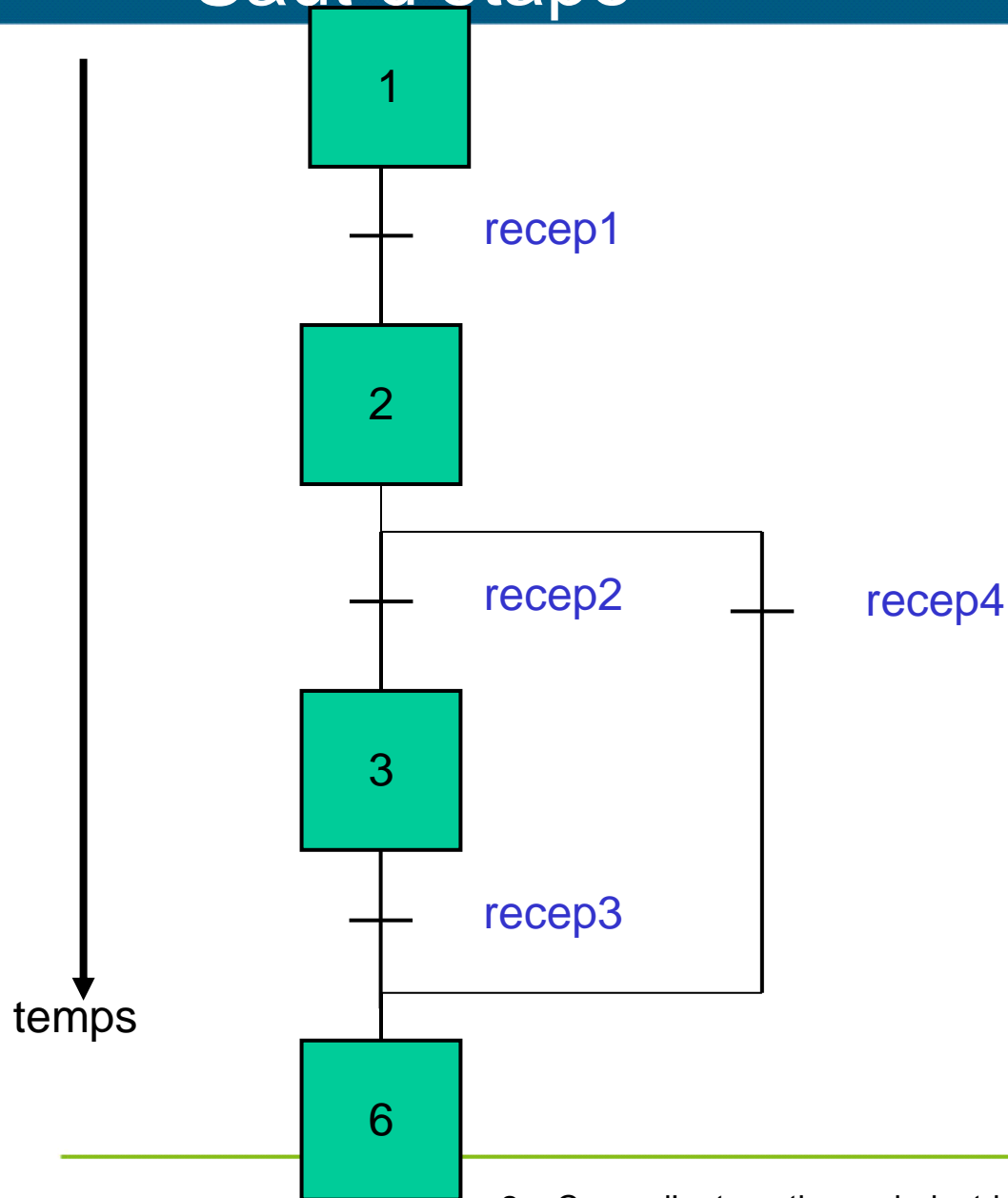


3.2 Séquences exclusives (1)

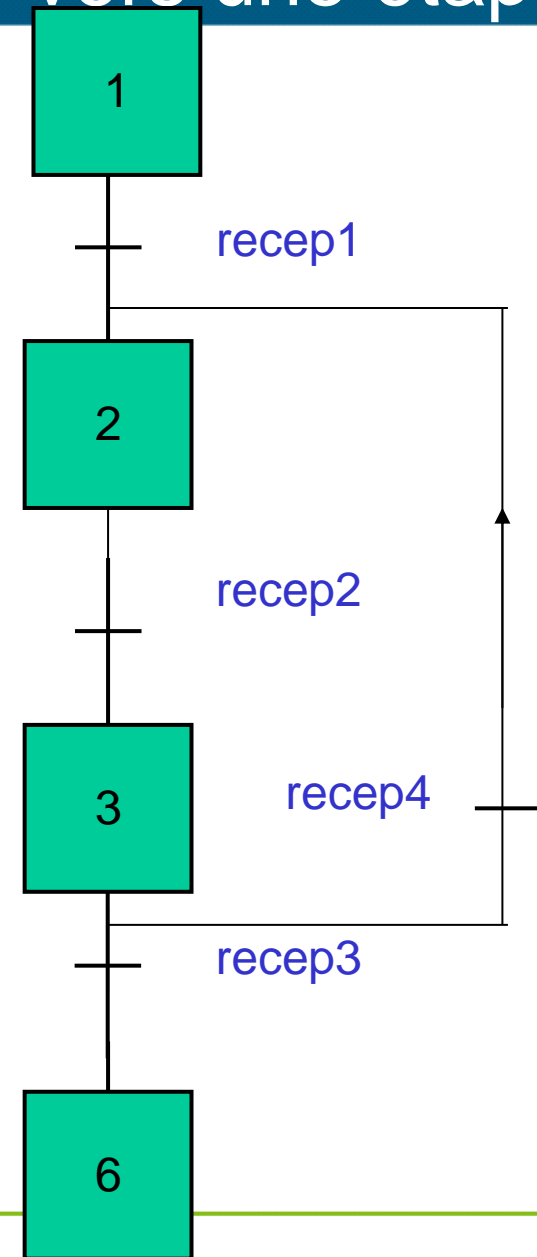


3.2 Séquences exclusives (2)

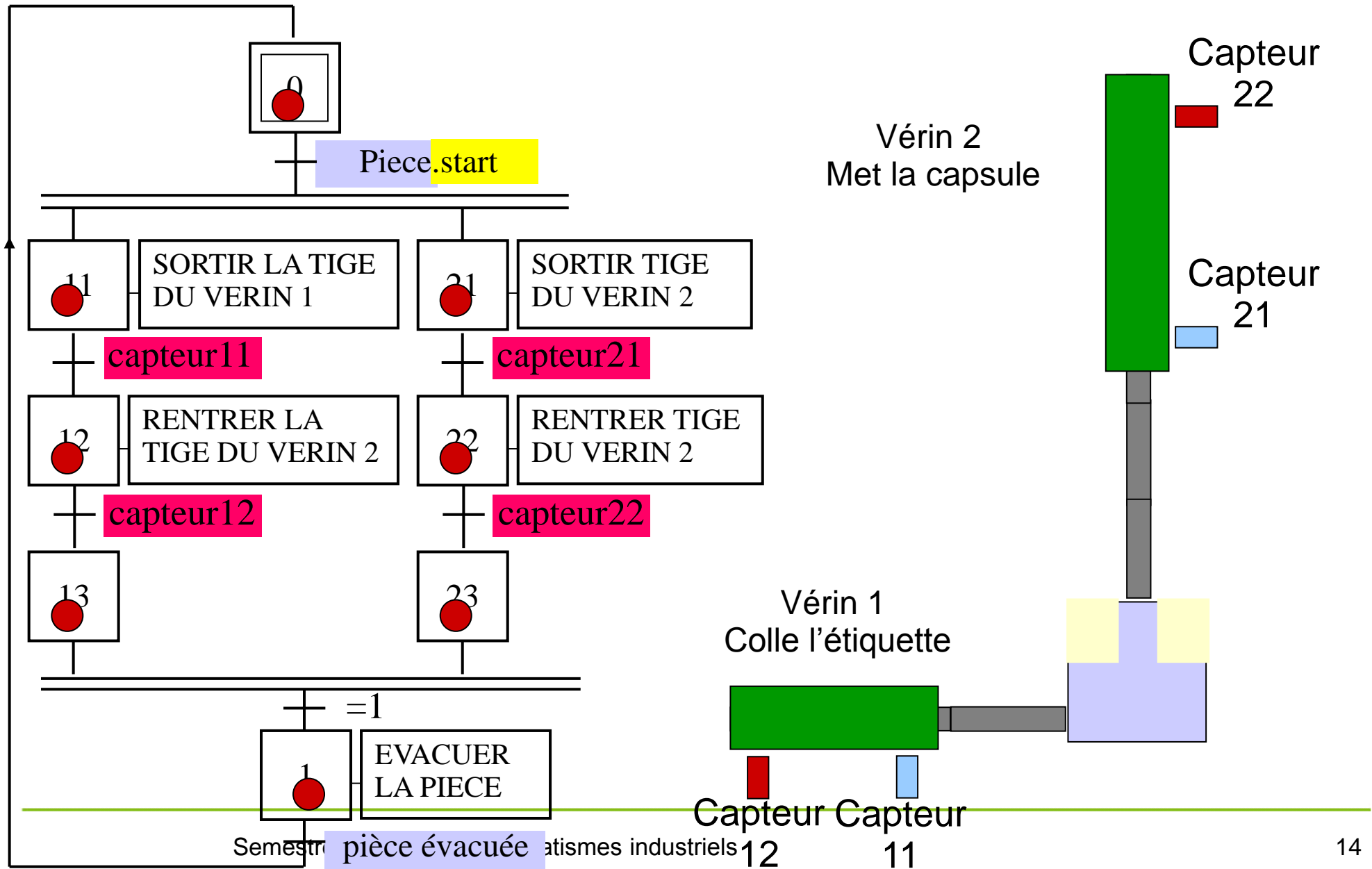
Saut d'étape



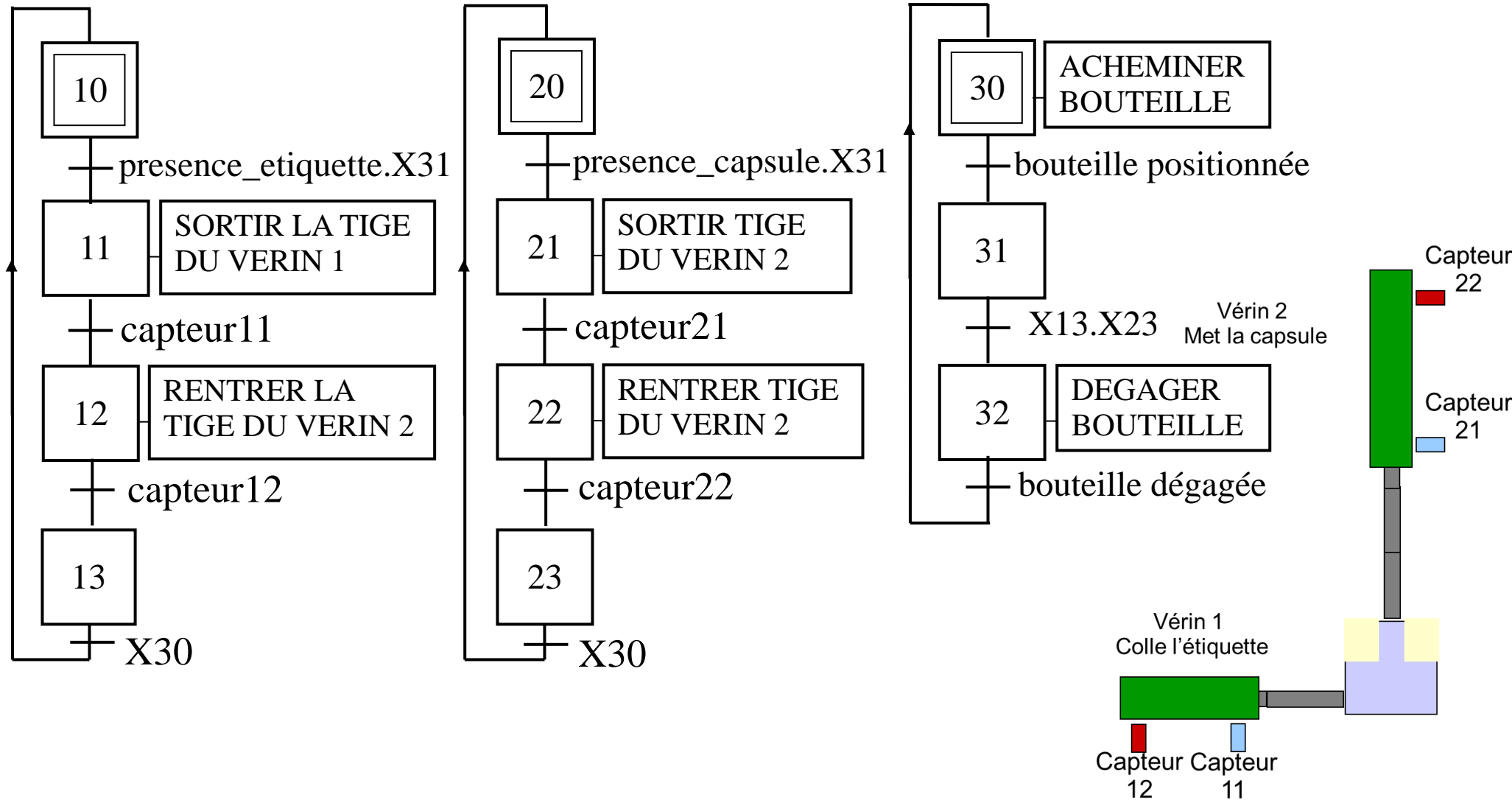
Retour vers une étape



3.3 Séquences simultanées



3.4 Synchronisation de tâches



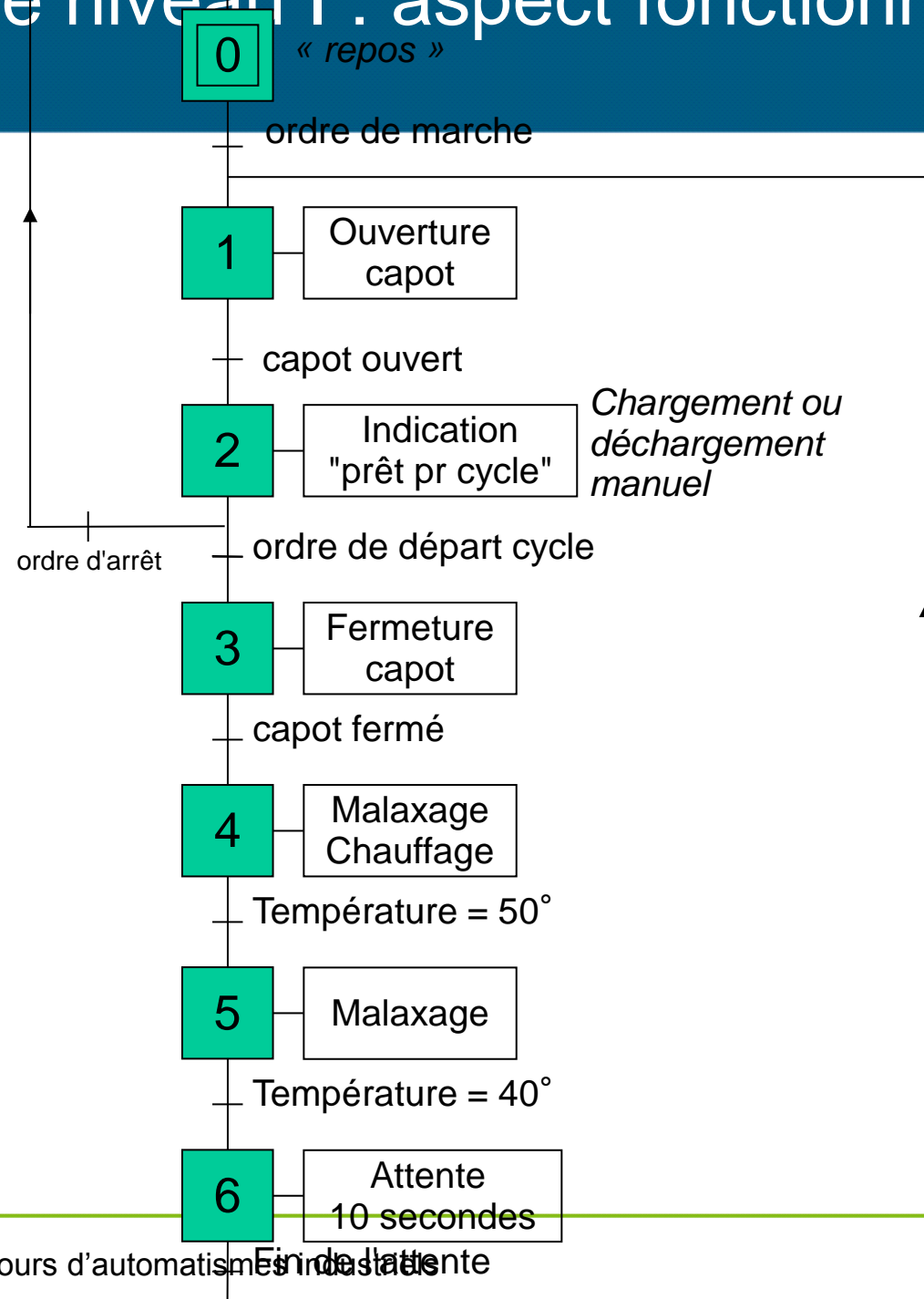
3.5 GRAFCET de niveau I : aspect fonctionnel (1)

Le GRAFCET est aussi un langage de description fonctionnelle.

Cahier des charges :

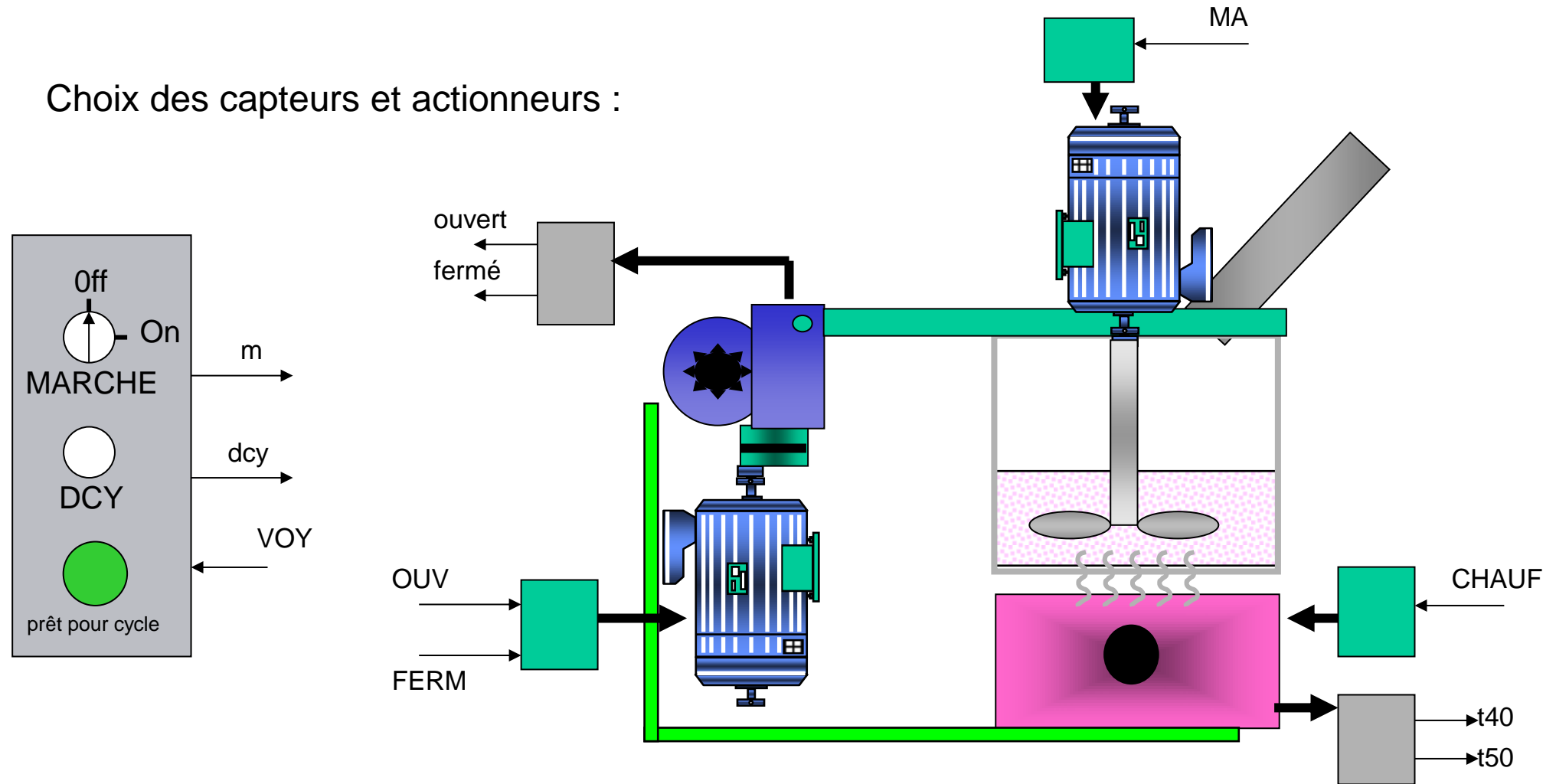
- A la mise sous tension (étape 0), Le système reste au repos.
- Si l'opérateur donne un ordre de marche, le capot est ouvert. En fin d'ouverture, une indication "prêt pour cycle" est donnée à l'opérateur. (celui-ci doit assurer manuellement le remplissage de la machine). L'opérateur doit donner l'ordre de départ cycle qui consiste en
 - ✓ La fermeture du capot
 - ✓ Le malaxage et le chauffage jusqu'à 50° C
 - ✓ Le malaxage jusqu'à ce que la température redescende à 40° C
 - ✓ L'ouverture du capot après 10 secondes
- L'opérateur doit alors vidanger manuellement l'appareil et éventuellement le remplir de nouveau pour démarrer un nouveau cycle.
- Chaque fois que le capot est ouvert, l'opérateur peut arrêter la production en annulant l'ordre de marche.

3.5 GRAFCET de niveau I : aspect fonctionnel (2)



3.6 GRAFCET de niveau II : partie opérative

Choix des capteurs et actionneurs :

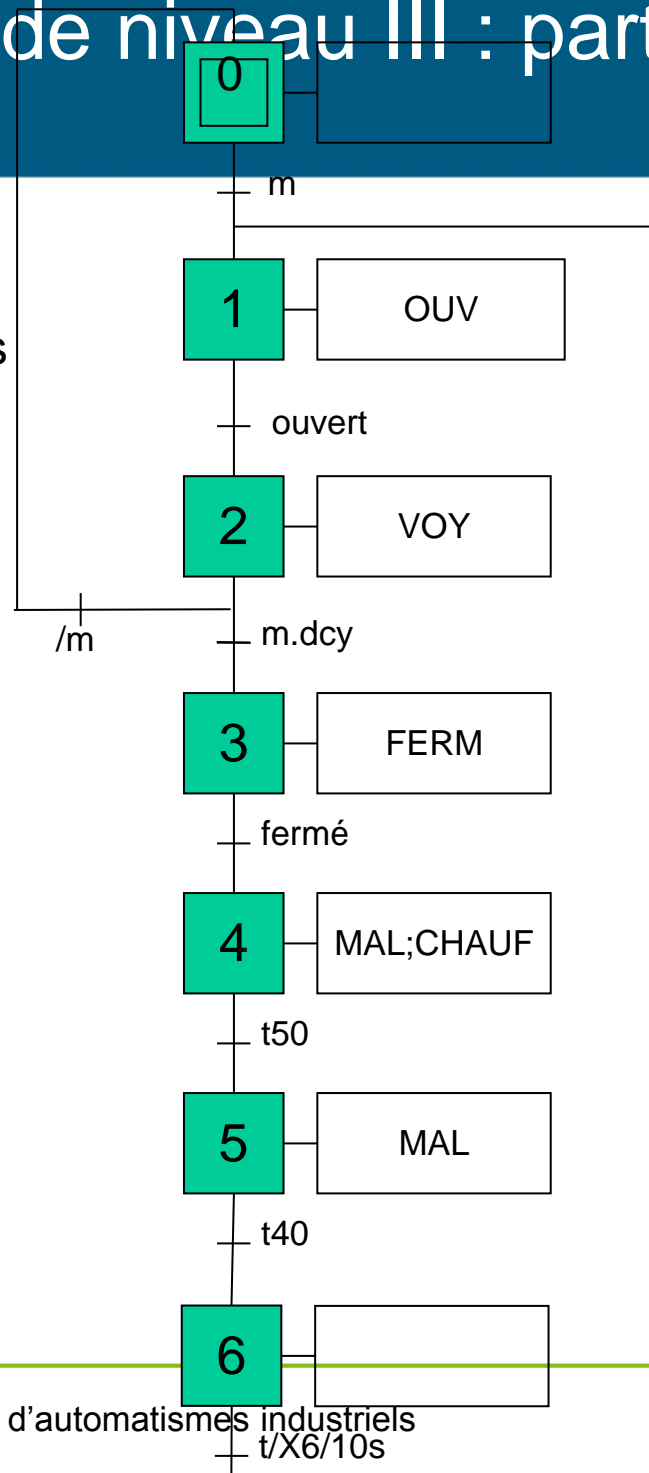
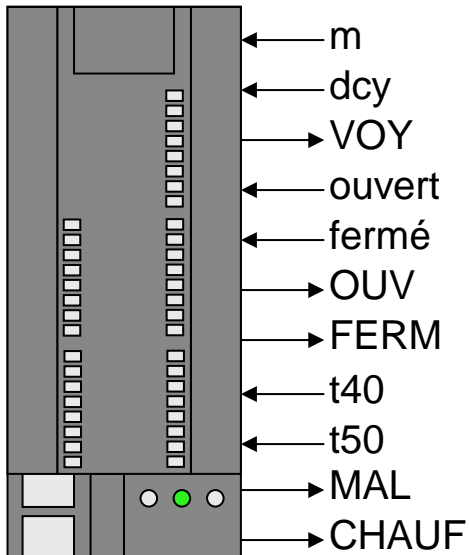


3.6 Le GRAFCET de niveau II : partie opérative (2)



3.7 Le GRAFCET de niveau III : partie commande

Choix des variables automates



Le niveau II est le grafcet point de vue partie opérative, utilisée par les mécaniciens pour décrire le fonctionnement du système, il est très proche du grafcet de niveau III

- ❖ 1 Introduction
- ❖ 2 Le langage LADDER
- ❖ 3 Le langage GRAFCET
- ❖ 4 Le langage structuré

4.1 Introduction contexte

- Langage proche du C
- Résultat plus lisible
- Mise au point et mise à jour des programmes plus rapide
- Suite d'instructions, séparées par le symbole ;
- Suite d'instructions exécutées séquentiellement et en boucle

4.2 Affectation et tests

(* Ceci est un commentaire *)

(* Ceci est une affectation *)

VOYANT := TRUE;

Compteur := 100;

(* Ceci est un test *)

Temperature > 50

Position <> 0

Fcourse = TRUE

Affectation :=

Opérateurs booléens

NOT, AND, OR

Opérateurs arithmétiques

+, -, *, /

Tests

= < > <>

4.3 Structure IF

```
IF Capteur3 = TRUE THEN  
    N := 3 ;  
END_IF;
```

```
IF Compteur > 100 THEN  
    VOYANT := TRUE;  
  
ELSE  
    VOYANT := FALSE;  
  
END_IF;
```

```
IF test THEN  
    .....;  
    .....;  
ELSE  
    .....;  
    .....;  
END_IF;
```


4.4 Instruction CASE

Exemple : estimation de la qualité d'une pièce usinée.
La pièce doit faire 3 mm d'épaisseur.

CASE epaisseur_piece_mm OF

```
1:
    QUALITE = MAUVAISE;
2 :
    QUALITE = MOYENNE;
3 :
    QUALITE = BONNE;
4 :
    QUALITE = MOYENNE;
5 :
    QUALITE = MAUVAISE;
ELSE :
    QUALITE = TRES_MAUVAISE;

END_CASE;
```

CASE selecteur OF
val1, val2

.....;
.....;

val3 :

.....;
.....;

ELSE

.....;
.....;

END_CASE;