Logique séquentielle

Eric SIMON

Ludovic MACAIRE

Mars 2015

Table des matières

Chapitre 1 GENERALITES SUR LES SYSTEMES SEQUENTIELS

Réf [0]

1. Rappel

1.1. Système combinatoire

Un **système logique** est un système composé de plusieurs ..... binaires et de plusieurs ..... binaires. Le système est dit ..... si, après un temps fini, les sorties (ou l'état) sont stables et déterminées de manière univoque par la combinaison des ..... Les relations entres les variables de sortie et les variables d'entrée sont donc définies par des fonctions binaires. Soient n le nombre d'entrées et m le nombre de sorties, on peut écrire :

.....

schéma:

Remarque : la propagation de l'information des entrées vers les sorties s'effectue à vitesse ..... (c'est-à-dire qu'elle n'est pas négligeable). Il existe donc un ..... Δt associé à chaque sortie entre le moment où on établit une combinaison nouvelle des entrées, et le moment où la sortie Sj prend sa valeur définitive. Nous considérerons ici un même retard pour toutes les sorties.

Le système combinatoire peut alors être considéré comme un circuit idéal, (pas de retard de propagation), suivi d'éléments de retard :

1.2. Système séquentiel

Dans un système séquentiel, les ..... (ou l'état) ne dépendent pas uniquement de la combinaison des entrées à un moment donné mais de la ..... des combinaisons précédentes de ces entrées et de son ..... Un système séquentiel possède donc une fonction .....

Pour différencier ces états, il faut introduire un certain nombre de variables supplémentaires, appelées ..... $v_1...v_p$, l'état du système est fixé par la combinaison ($e_1...e_n$, $v_1...v_p$). Le système séquentiel peut alors être vu comme un système combinatoire avec les entrées e1...en, $v_1...v_p$. Ce circuit est appelé circuit .....

Remarque : les variables internes $v_1 ... v_p$ sont elles-même fonction de l'état du système. Par conséquent les systèmes séquentiels sont des systèmes ..... Notons également qu'il faut là aussi prendre en compte le temps de calcul des variables internes si le circuit combinatoire a un temps de calcul moyen Δt. On obtient le schéma suivant:

Conclusion : c'est le retard Δt qui constitue ..... du système. Notons que cette mémoire est .....

1.3. Systèmes séquentiels asynchrones et synchrones

Suivant les conditions de changement des variables de ....., on distingue deux types de système séquentiel:

- les systèmes séquentiels asynchrones : l'état des sorties évolue spontanément à la suite d'un changement de configuration des variables d'entrée. Cette évolution ne dépend que de la ..... dont le nombre et la durée peut être variable.
- Les systèmes séquentiels synchrones : l'évolution des sorties dépend d'une .... généralement appelée ..... Il n'y a plus d'états transitoires multiples dus aux changements des variables internes. Le système est synchronisé sur le signal d'horloge.

Remarque : les systèmes ..... sont les moins fiables car ils sont plus sensibles aux parasites sur les entrées. Ces systèmes sont donc de moins en moins utilisés.

2. Éléments de base de la logique séquentielle : Les bascules

La bascule est l'élément de base de la logique séquentielle. On trouve les deux types de fonctionnement synchrones et asynchrones. Dans le mode synchrone, l'autorisation qui conditionne le changement d'état peut être donné de trois façon différentes :

- synchronisation sur ..... : il faut appliquer un niveau donné appelé niveau actif pour que la sortie de la bascule puisse changer.
- synchronisation sur ..... : la sortie pourra changer au moment d'un front actif. Dans ce cas, la durée de l'autorisation est réduite à son minimum, c'est-à-dire le temps pour que le signal d'horloge passe d'un niveau à un autre.
- synchronisation .....: une impulsion est composée de deux fronts. La sortie pourra évoluer au moment d'une impulsion.

2.1. La bascule RS (bascule asynchrone)

2.1.1. Symbole

2.1.2. Table de fonctionnement

Remarque : contrairement à la table de vérité, la table de fonctionnement fait intervenir la notion de temps.

Exemple de fonctionnement :

2.1.3. Réalisation d'une bascule RS

Comme tous les systèmes séquentiels asynchrones, on peut mettre le schéma de la bascule sous la forme d'un circuit ..... et d'un circuit ..... Notons que le nombre de variables internes est lié au nombre de possibilités différentes des ..... pour une .....

Ici, il existe deux valeurs possibles pour la combinaison d'entrée R = S = 0 suivant la valeur déjà en mémoire. Par conséquent une variable interne suffit pour distinguer les deux états précédents correspondants. On obtient le schéma suivant :

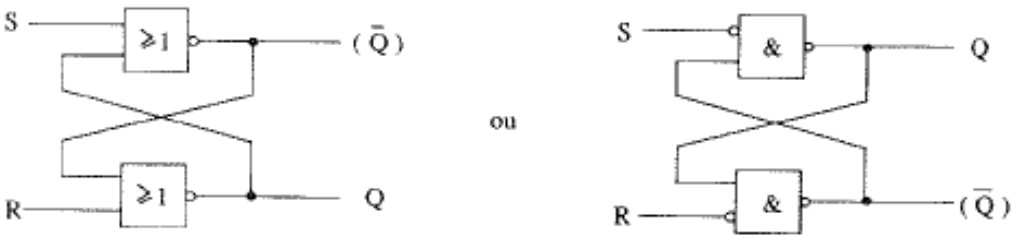
On peut maintenant définir la table de vérité du circuit combinatoire idéal (sans retard) :

puis le tableau de Karnaugh correspondant :

Après simplification, on obtient l'expression de la fonction d'excitation X ainsi que son schéma de réalisation avec des portes idéales (sans retard de propagation) :

2.1.4. Réalisation pratique

En pratique, on utilise la technologie NOR ou la technologie NAND, ce qui donne :



Remarque : on intègre le retard dans les opérateurs, ce qui correspond au cas réel.

2.2. Bascule RS synchrone ou RST

2.2.1. Symbole

2.2.2. Fonctionnement

la bascule RST est la bascule RS dont on a synchronisé les ..... avec des .... Elle est synchronisée par le niveau ..... de l'horloge (T) et elle comporte deux fonctionnements distincts :

T = 0 : la sortie ne change pas quelles que soient les entrées R et S. C'est le fonctionnement ..... La bascule n'est pas synchronisée.

T = 1 : la bascule est alors synchronisée. Sa sortie suit la table de fonctionnement de la bascule RS.

On obtient la table de fonctionnement suivante :

Exemple de fonctionnement :

2.2.3. Réalisation d'une bascule RST

Étant donné que la bascule RST est en fonctionnement mémoire pour les deux combinaisons .....,et ...., il suffit d'ajouter une porte ..... qui permet de bloquer les commandes R et S tant que T=0. Le schéma est donc le suivant :

2.3. Bascule D (Delay) ou Mémoire

2.3.1. Schéma

Cette bascule dispose d'une seule entrée appelée l'entrée D. Le signal de synchronisation peut être actif soit sur un ..... (la bascule est alors appelée D latch) soit sur un ..... (edge triggered).

2.3.2. Fonctionnement

Comme il n'y a qu'une entrée, il ne peut y avoir que ..... modes de fonctionnement. Les modes de fonctionnement sont les suivants:

- Le signal de synchronisation est ....., la sortie recopie l'entrée D
- Le signal de synchronisation ..... la sortie est inchangée.

2.3.3. Réalisation d'une bascule D latch

Pour réaliser une bascule D latch, on part d'une bascule RST, et on commande les entrées R et S par le même signal en utilisant un inverseur sur R. Le schéma peut être le suivant :

Remarque : la bascule D est le composant de base d'une mémoire d'ordinateur

Chapitre 2 DESCRIPTION D'UN SYSTEME SEQUENTIEL : introduction au Grafcet

Le GRAFCET (GRAphe Fonctionnel de Commande Etape / Transition) est un diagramme fonctionnel dont le but est de décrire graphiquement, suivant un cahier des charges, les différents comportements de l'évolution d'un automatisme séquentiel.

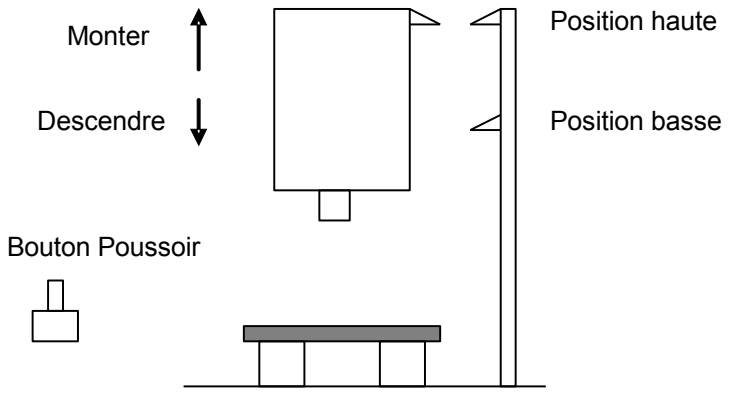
Pourquoi le Grafcet?

Lorsque certaines spécifications sont exprimées en langage courant, il y a un risque permanent d'incompréhension. Certains mots sont peu précis, mal définis ou possèdent plusieurs sens. Le langage courant est donc mal adapté pour décrire précisément les systèmes séquentiels.

Le Grafcet fut donc crée pour représenter de façon symbolique et graphique le fonctionnement d'un automatisme séquentiel. Il permet une meilleure compréhension par tous les intervenants.

1. Premier exemple: la poinçonneuse semi-automatique

Une poinçonneuse semi-automatique se compose d'une table fixe recevant la tôle à poinçonner et d'un poinçon mobile.



Au repos, le poinçon est en position haute (cf figure). Lorsque l'opérateur appuie sur le bouton poussoir (BP), le poinçon descend en position basse puis remonte en position haute et retourne au repos. Ce cycle se répète à chaque pression du bouton poussoir.

La poinçonneuse a donc décrit un cycle. Le cycle est composé de trois comportements

différents:

- La poinçonneuse est au repos ou encore en position haute.
- Le poinçon descend
- Le poinçon remonte.

Chacun de ces comportements est appelé .....

Il faut ensuite préciser ce qui provoque un changement de comportement de la machine, c'est-à-dire les conditions logiques qui déterminent le passage d'une étape à une autre.

Le passage d'une étape à une autre est appelé .....

Le passage de l'état de repos à la descente du poinçon s'effectue si:

- l'opérateur fournit l'information marche par appui sur le BP,
- et si le poinçon est en position haute.

Ces deux informations constituent la condition de transition de l'étape 1 à l'étape 2. Cette condition est appelée ..... associée à la transition étape 1-étape 2.

On peut donc représenter le comportement automatique de cette poinçonneuse par un Grafcet. Celui-ci est basé sur une succession d'étapes auxquelles sont associées des ....., et de transitions auxquelles sont associées des .....

Cette première représentation prend en compte uniquement les spécifications fonctionnelles et fait abstraction de toute réalisation technologique. Ce type de grafcet est appelé grafcet fonctionnel ou grafcet de niveau 1. Nous verrons au chapitre 4 les autres niveaux de description d'un grafcet, qui tiennent compte des spécifications technologiques.

Ce Grafcet fait intervenir:

- des étapes auxquelles sont associées un comportement ou une action à obtenir,
- des transitions auxquelles sont associées des informations permettant le franchissement des étapes. Ces informations sont représentées sous forme d'une condition logique appelée réceptivité.

2. Architecture générale d'un système automatisé

Cette partie situe la place du grafcet dans l'univers plus général des systèmes automatisés de production.

Un système à automatiser se décompose en deux parties qui communiquent entre elles: une …..... et une ….....

La partie opérative (PO) est le processus physique à automatiser.

La partie commande (PC) est l'automatisme qui élabore en sortie des …..... destinés au processus.

La figure suivante montre la structure générale des liens entre la P.C. et la P.O. pour la réalisation d'une fonction particulière du système (correspondant à une seule étape d'un grafcet).

Il n'est pas nécessaire de prévoir un interfaçage particulier entre les capteurs et la carte d'entrées de l'automate. En effet les sorties des capteurs sont en général directement compatibles avec les entrées des automates (au niveau du type et du niveau d'énergie, par exemple énergie électrique en 24V continu). On peut donc brancher directement le capteur sur la carte d'entrées de l'automate.

Contrairement à la chaîne d'acquisition, il faut prévoir un interfaçage entre la carte de sortie de l'automate et les actionneurs. En effet, en général les actionneurs utilisent des énergies de type et de niveau différents des cartes de sorties des automates (exemple : vérin pneumatique). La liaison entre ces deux composants va donc nécessiter une conversion du type ou du niveau d'énergie, ce qui va être réalisé par le pré-actionneur (distributeur pneumatique, …).

La partie commande peut être réalisée avec:

- des solutions câblées,
- des solutions programmées.

Parmi les solutions programmées, on trouve le micro-contrôleur, l'automate programmable, le micro-ordinateur ou des solutions hybrides. Notre domaine d'étude étant limité aux systèmes industriels, nous traiterons uniquement l'automate programmable.

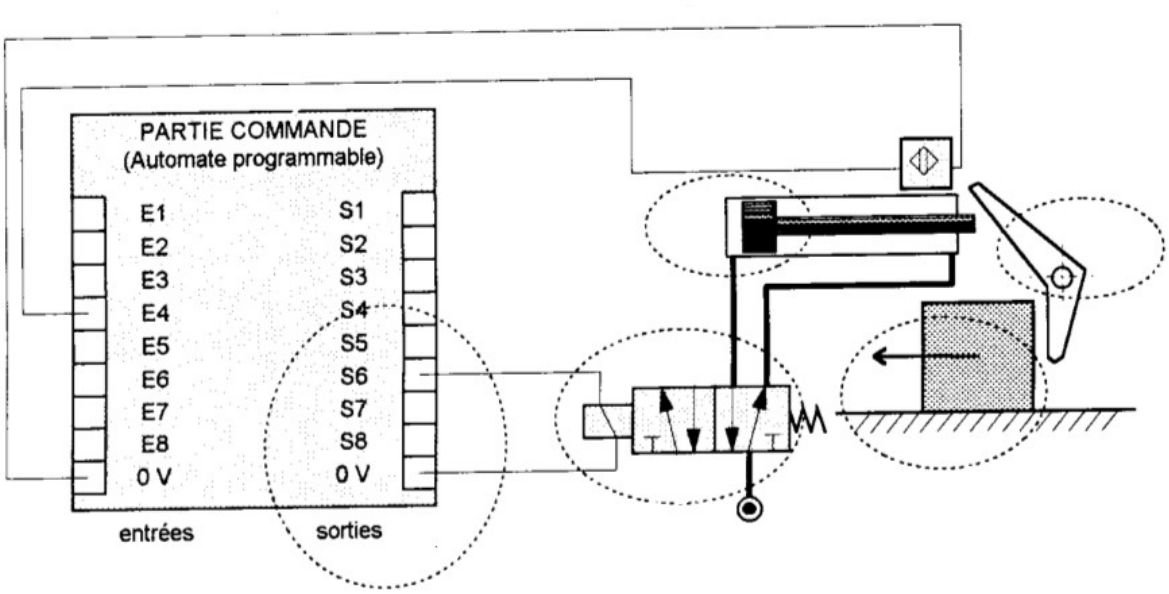
3. Niveau de précision d'un Grafcet

Nous avons vu au paragraphe précédent que la chaîne ..... comportait un certain nombre d'éléments, en partant de la P.O. avec l'effectueur, jusqu'à la sortie de la P.C. Il existe donc plusieurs niveaux de description d'une ....., et par conséquent d'un grafcet puisque c'est cette description de l'action qui va être utilisée pour remplir le carré correspondant dans le grafcet.

Il y a donc une dimension « finesse » d'un grafcet, caractérisant le niveau de détail dans la description d'une action, d'un niveau global (la fonction à réaliser) à un niveau plus détaillé ou toutes les informations élémentaires sont prises en compte.

4. Exemple d'une caisse à déplacer

On prend l'exemple simple d'une caisse qu'il faut déplacer à un moment donné d'un cycle automatisé de palettisation. L'action de déplacer la caisse correspond donc à une étape donnée d'un grafcet décrivant ce cycle de palettisation.



[Réf 1]

La description de cette étape du grafcet peut donc se réaliser à différents niveaux de la chaîne d'action. De plus, la caractérisation du niveau nécessite de prendre en compte deux points de vue : le point de vue de .... et le point de vue du ..... Pour le point de vue de l'observateur extérieur, on utilise un verbe au ..... pour expliquer que l'action est en train de se produire, et pour le point de vue système on utilise un verbe à ..... pour signifier que l'action correspond à un ordre émis par le système. Les différents niveaux de description sont les suivants :

1. **Au niveau de la fonction à réaliser : déplacement de la caisse**

Point de vue observateur extérieur

Point de vue système

2. **Au niveau de l'effectueur**

Point de vue observateur extérieur

Point de vue système

3. Au niveau de l'actionneur

Point de vue observateur extérieur

Point de vue système

4. Au niveau du pré-actionneur

Point de vue observateur extérieur

Point de vue système

5. Au niveau de la sortie de l'automate

Point de vue observateur extérieur

Point de vue système

Chapitre 3 NOTIONS ESSENTIELLES DU GRAFCET

1. Les éléments du Grafcet

1.1. Etapes

Définition d'une étape:

L'étape symbolise une situation dans laquelle l'état du système est invariant (pas d'évolution sur les sorties).

A chaque étape est associée une (ou plusieurs) ....., c'est-à-dire un ordre vers la partie opérative du système.

Représentation d'une étape:

L'étape se représente par un carré.

L'entrée se fait par le haut du carré et la sortie par le bas.

On numérote chaque étape par un entier positif, mais pas nécessairement croissant par pas de 1. Deux étapes différentes ne doivent jamais avoir le même numéro.

Actions associées à l'étape:

L'action associée à l'étape est définie dans un rectangle à droite de l'étape. Les actions à effectuer sont décrites de manière littérale ou symbolique. Les actions peuvent être de nature diverse:

Etat d'une étape

Une étape possède deux états distincts, elle peut être ..... ou .....**.**

Une étape est dite ..... lorsqu'elle effectue l'action (ou les actions) qui lui est associée, c'est-à-dire lorsque le programme est pointé dessus.

Remarque: on montre les étapes actives à un instant précis en plaçant un point dans la partie inférieur du carré.

Variables d'étape et variables de temporisation

On peut associer à chaque étape une variable ..... exprimant son activité. Cette variable est booléenne et vaut 1 lorsque l'étape associée est ....., et 0 lorsque celle-ci est ..... Cette variable est notée X suivie du numéro de l'étape

Exemple:

On définit également les variables de temporisation **:**

De manière générale, la variable de temporisation s'écrit

.....

où i est le numéro de l'étape de lancement et *durée* est le retard apporté au front montant de la variable Xi. Par exemple, 3s / X1 signifie que le signal passe à 1 3 secondes après le début de l'activation de l'étape 1.

Exemple :

Définition: situation d'un grafcet

La situation d'un grafcet à un instant donné est définie comme la liste de ses étapes actives à cet instant.

La situation est représentée par un vecteur de la forme {i, j, …} où i, j, sont les étapes actives. La situation est dite vide lorsqu'aucune étape n'est active.

Etape initiale:

Une étape peut être **initiale,** elle est alors active au début du processus de commande (les étapes non initiales sont alors inactives)

On repère une étape initiale par un doublement du symbole d'étape.

L'ensemble de ces étapes initiales caractérise le comportement initial de la partie commande.

1.2. Transitions

Définition

Les transitions indiquent les possibilités d'évolution d'une étape à la suivante.

A chaque transition on associe une condition logique (appelée .....) qui détermine la condition nécessaire pour passer d'une étape à une autre.

Représentation

Une transition se représente par un tiret horizontal.

On note à droite la réceptivité.

1.3. Réceptivité

Cette condition de ..... de la transition est une fonction combinatoire calculée à partir d'informations telle que:

- états des capteurs,
- action de bouton poussoir par l'opérateur,
- état courant des étapes du grafcet (variable d'étape Xn)...

Comme pour toute fonction combinatoire, on utilise les opérateurs logique ....., ..., ... pour combiner ces informations.

Remarque: une réceptivité toujours vraie est écrite : .....

1.4. Les liaisons orientées

Définition:

Une liaison est un arc orienté permettant de relier une ..... à une ..... ou une ..... à une ..... Une liaison ne peut être parcourue que dans un sens.

Représentation:

Une liaison se représente par un trait, vertical ou horizontal. Dans le cas général, les liaisons qui se font du haut vers le bas ainsi que de la droite vers la gauche ne comportent pas de flèches. Dans les autres cas il faut utiliser une flèche.

Résumé:

Le contenu d'une étape fait référence aux ..... (sorties de la PC)

Le contenu d'une réceptivité fait référence aux ..... (entrées de la PC)

Entre deux étapes, il y a toujours une et une seule .....

2. Règles d'évolution du grafcet

Le grafcet est basé sur une succession ..... séparées par des ..... L'évolution d'un grafcet correspond aux franchissements successifs de ces transitions. Cette évolution est décrite de manière très précise avec les règles suivantes:

On rappelle qu'une étape est dite **active** lorsque le programme est pointé sur cette étape (dans le cas de séquences simultanées, plusieurs étapes sont actives en même temps).

Une transition est soit ....., soit **non** .....

Elle est ..... lorsque toutes les étapes immédiatement précédentes sont .....

Règle 1: .

Exemple:

Règle 2:

Résumé:

On dit qu'une étape est ...../.....

Une transition est ...../non .....

Une réceptivité est ...../.....

Pour franchir une transition, il faut que:

- les étapes immédiatement précédentes soient .....,
- et que la réceptivité associée à la transition soit .....

Le franchissement d'une transition entraîne:

- l'activation des étapes immédiatement .....,
- et la désactivation de toutes les étapes immédiatement .....

3. Structures de base du grafcet

Les structures sont les différentes manières d'organiser les liaisons orientées. Les structures les plus utilisées sont décrites ci-dessous. Cette énumération n'est pas exhaustive.

3.1. Divergence et convergence en OU: sélection de séquences ou aiguillage

3.1.1. Début d'aiguillage:

La **divergence en OU** permet de choisir entre plusieurs séquences (aiguillage vers plusieurs séquences). Elle est aussi appelée structure à .....

Si l'on ne souhaite sélectionner qu'une seule séquence, il est indispensable que toutes les réceptivités associées aux transitions validées en même temps soient exclusives. Cette exclusion peut être:

- Soit d'ordre ..... (incompatibilité mécanique ou temporelle)
- Soit d'ordre ..... dans l'écriture des réceptivités

3.1.2. Fin d'aiguillage

3.1.3. Cas particulier: saut d'étapes et reprise de séquence

La reprise de séquence est une structure qui permet de reprendre une ou plusieurs fois la même séquence tant qu'une condition n'est pas obtenue.

Le saut conditionel est un aiguillage particulier en OU permettant de sauter une ou plusieurs étapes lorsque les actions associées deviennent inutiles.

3.2. Divergence et convergence en ET: séquences simultanées ou parallélisme

3.2.1. Début de parallélisme :

La divergence en ET permet de réaliser simultanément plusieurs séquence. Si la réceptivité est vraie, ce sont toutes les étapes suivantes qui sont activées

Lorsque l'étape 7 est active, la transition 7-(10, 20) est ..... Lors du franchissement de la transition (lorsque la réceptivité devient vraie), il y a ..... de l'étape 7 et ..... des étapes 10 et 20.

3.2.2. Fin de parallélisme :

Remarque sur l'évolution du grafcet en fin de parallélisme :

On rappelle d'abord la la règle 1 de l'évolution des grafcets :

Règle 1: .....

.....

De plus, on rappelle également qu'une transition est **validée** lorsque .....

.....

Par conséquent, il faut que les étapes 17 et 24 soient ..... pour que la transition (17, 24)-5 soit ..... En d'autres termes, il faut attendre que les deux séquences parallèles soient terminées pour pouvoir passer à la suite.

En pratique, les étapes de fin de parallélisme ne comporte pas ..... De plus, la transition de fin de parallélisme est souvent imposée à '=1'. Les étapes sans action permettent alors de ..... la fin des différents cycles en amont et lorsqu'elles sont ....., le franchissement de la transition est automatique.

Exemple de synchronisation:

Cahier des charges 1 : Le problème est le suivant : soient deux chariots 1 et 2 se trouvant à gauche de la piste à la mise en marche du système. Lorsqu'on appuie sur le bouton m, les deux chariots vont à droite jusqu'en b1, b2 puis retournent à gauche jusqu'en a1, a2. On ne peut les relancer que lorsqu'ils sont tous deux arrivés à gauche. Les deux pistes ont des tailles différentes. De plus, chaque chariot aura sa propre séquence. Comme les deux pistes n'ont pas la même distance, cela va nécessiter de synchroniser les deux séquences avant de relancer.

Donner le grafcet de commande correspondant à ce cahier des charges.

Cahier des charges 2 : on ajoute une contrainte au cahier des charges précédent. Les deux chariots doivent s'attendre à droite avant de repartir à gauche.

Réaliser le grafcet de commande correspondant à ce cahier des charges.

Cahier des charges 3 : chaque chariot possède maintenant son propre bouton marche (m1 et m2). On peut donc les lancer de manière indépendante. Pour le reste, on garde la contrainte d'attente à droite avant qu'ils puissent repartir automatiquement à gauche.

Réaliser le grafcet de commande correspondant à ce cahier des charges. La solution précédente avec une seule étape initiale ne marche plus, car maintenant les deux chariots peuvent être lancés de manière indépendante. Proposer une solution avec deux étapes initiales, une par chariot, synchronisées par un parallélisme. Ce type de parallélisme s'appelle du parallélisme de synchronisation.

4. Règles de syntaxe

Pour terminer ce chapitre, nous résumons les règles de syntaxe à respecter.

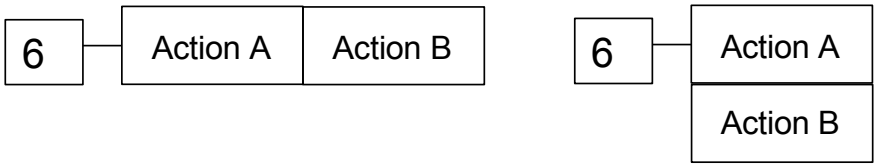
Deux étapes ne doivent jamais être reliées directement: elles doivent toujours être séparées par une transition. Deux transitions ne doivent jamais être reliées directement: elles doivent toujours être séparées par une étape.

Voici quelques exemples d'erreurs de syntaxe fréquentes:

Chapitre 4 NOTIONS COMPLEMENTAIRES

1. Sur les actions

Rappel: Une action est une ..... de la partie commande qui permet d'envoyer un ..... vers la partie ..... Une ou plusieurs actions peuvent être associées à une étape. Dans le cas où plusieurs actions sont associées à une étape, on les représente soit à coté, soit l'une en dessous de l'autre:



Par défaut, les actions sont produites tant que les étapes auxquelles elles sont associées sont ..... Ce sont les actions ..... Mais le Grafcet possède d'autes types d'actions. On distingue ainsi :

- les actions conditionnelles,
- les actions mémorisées,
- les actions limitées dans le temps ou au contraire retardées.

Dans le cadre de ce cours, nous nous limiterons aux actions conditionnelles et aux actions mémorisées.

1.1. Les actions continues

La durée de l'action est identique à la durée d'activation de l'étape associée.

1.2. Actions continues avec condition d'assignation (C pour conditionned)

L'action est produite tant que l'étape associée est active et que la condition est vraie.

1.3. Actions mémorisées

Pour que l'action reste maintenue lorsque l'étape qui l'a commandée vient d'être désactivée, il faut utiliser une action mémorisée, ce qui est alors spécifié par les symboles de début « S » et de fin d'action « R ».

2. Sur les réceptivités

Rappel: la réceptivité peut être vraie (état 1) ou fausse (état 0). C'est elle qui conditionne l'évolution du grafcet.

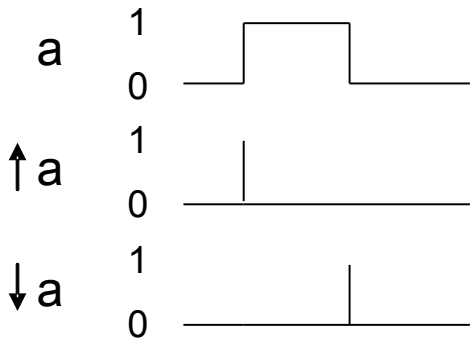
L'évolution du grafcet peut dépendre soit du niveau maintenu du signal (état 1, 0), soit du changement d'état du signal (front montant, front descendant).

Dans le cas d'un niveau maintenu, on dit que l'évolution est liée à une .....

Dans le cas d'un changement d'état, on dit que l'évolution est liée à un .....

La notation ….... représente l'apparition ou le front montant de l'information « a » (passage de l'état logique 0 à l'état logique 1) et la notation ….. la disparition ou le front descendant de l'information « a » (passage de l'état logique 0 à l'état logique 1)

Représentation des fronts:



Nous présentons maintenant un exemple d'application des fronts :

Exemple : allumage et extinction d'une lampe :

Commentaires sur l'exemple : détecter l'apparition d'une information, c'est d'abord vérifier que antérieurement cette information était …..... et détecter la disparition d'une information, c'est vérifier qu'elle était …..... antérieurement.

La reconnaissance d'un front s'effectue donc en deux temps:

- vérification de l'état …..... de l'information

- détection du ….....de l'information

les deux réceptivités opposées sont séparées par une étape supplémentaire car elles correspondent bien à un changement d'état.

3. Source et puits

Etape source : on appelle étape source une étape non reliée à une transition ..... Elle ne peut être activée que si elle est initiale ou si elle est soumise à un ordre de forçage par un grafcet de niveau supérieur.

Etape puits : on appelle étape puits une étape non reliée à une transition ..... Seul un ordre de forçage peut modifier son état.

La transition source n'est pas reliée à une étape amont. Par convention, elle est toujours validée et deviendra franchissable lorsque la réceptivité est vraie.

La transition puits n'est pas reliée à une étape aval.

4. Programmation d'un Grafcet

4.1. Description du programme

Un grafcet peut être décomposé en deux parties comme le montre la figure ci-dessous. On a d'une part le grafcet proprement dit, à savoir la succession d'étapes-transitions (réceptivités comprises) et d'autre part les actions qui sont associées aux étapes. Les actions dépendent directement des états des étapes. Le programme doit donc mettre à jour les étapes du grafcet dans un premier temps, et dans un deuxième temps seulement, produire les actions en fonction de la nouvelle situation du grafcet.

Quelque soit le langage de l'automate, on réalise l'algorithme général suivant (sauf si le langage de l'automate intègre la représentation graphique du grafcet):

-partie grafcet :

-partie actions :

Le grafcet ayant évolué, la deuxième partie du programme doit produire les actions associées aux étapes actives. La programmation est donc simple :

4.2. Description de l'automate programmable

4.2.1. Les éléments de l'automate programmable

L'automate programmable est constitué de [1]:

cartes d'entrées-sorties:

Afin de piloter la PO de manière automatique, il est nécessaire de recueillir des informations pour calculer les réceptivités du grafcet, et de transmettre des ordres pour faire fonctionner la PO.

variables internes:

Les variables internes sont stockées dans l'unité centrale. Une temporisation est une variable interne.

unité centrale

L'unité centrale est elle même composée des éléments suivants:

- le jeu d'instructions
- la mémoire programme utilisateur (RAM) qui contient le programme écrit par l'utilisateur.
- la mémoire programme système (ROM), non accessible par l'utilisateur.
- Les mémoires images des entrées et des sorties: cette zone mémoire est « l'image » des entrées et des sorties. Ces mémoires sont accessibles par le programme utilisateur et par le programme système.
- L'horloge interne: elle cadence l'exécution du programme

4.2.2. Principe de fonctionnement

Le programme est constitué d'une liste d'instruction écrite par le programmeur. Ce programme doit être capable de réagir en temps réel en fonction des sollicitations extérieures et des états des variables internes.

Pour permettre ces réactions en temps réel, le déroulement du programme repose sur le principe suivant [1]:

- Les instructions sont automatiquement exécutées les unes après les autres
- Lorsque toutes les instructions du programme ont été lues, le programme se déroule de nouveau depuis le ….....
- Cette relecture cyclique est ininterrompue, on parle de .....

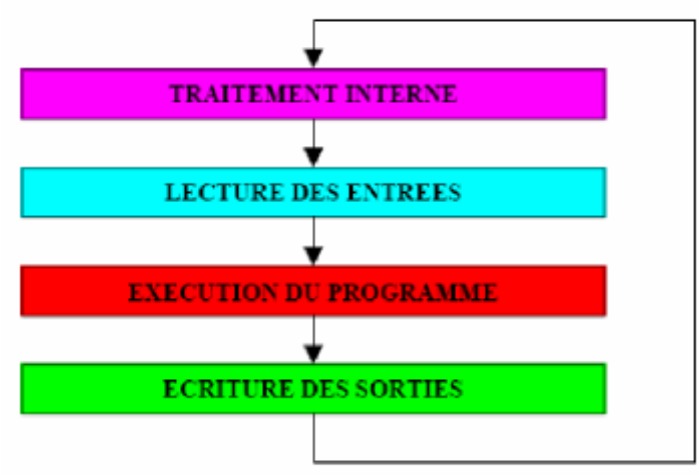
Remarque: la durée d'une scrutation est de l'ordre de quelques dizaines de millisecondes (en fonction des programmes).

Notons que les changements d'état des signaux délivrés par les capteurs sont indépendants du déroulement du programme, ce qui pourrait conduire à une certaine instabilité dans le fonctionnement du programme. En effet le programmeur développe son programme en considérant qu'une entrée est soit fausse, soit vraie à un moment donné mais non les deux à la fois. Par conséquent on va rendre stable les états des entrées pour la durée d'exécution d'un seul cycle. Le principe est le suivant :

- En début de scrutation, les états des entrées physiques telles qu'elles sont vues par les cartes d'entrées sont mémorisées dans les mémoires images correspondantes,
- pendant l'exécution du programme, toute instruction qui requiert l'état d'une entrée provoque en réalité la lecture de l'image de cette entrée,
- les mémoires images des entrées sont rafraîchies en début de chaque nouvelle scrutation.

4.2.3. Résumé

Pour fonctionner correctement, l'automate effectue en plus au début de chaque scrutation un traitement interne. Le schéma global du fonctionnement de l'automate est le suivant :

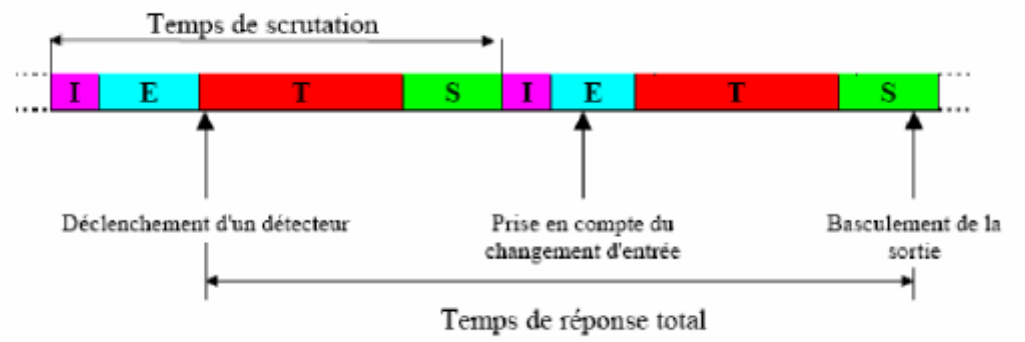


- **Traitement interne** : L'automate effectue des opérations de contrôle et met à jour certains paramètres systèmes (détection des passages en RUN / STOP, ...).
- **Lecture des entrées** : L'automate lit les entrées (de façon synchrone) et les recopie dans la mémoire image des entrées.
- **Exécution du programme** : L'automate exécute le programme instruction par instruction et écrit les sorties dans la mémoire image des sorties.
- **Ecriture des sorties** : L'automate bascule les différentes sorties (de façon synchrone) aux positions définies dans la mémoire image des sorties.

Ces quatre opérations sont effectuées continuellement par l'automate (fonctionnement cyclique).

On appelle scrutation l'ensemble des quatre opérations réalisées par l'automate et le temps de scrutation est le temps mis par l'automate pour traiter la même partie de programme. Ce temps est de l'ordre de la dizaine de millisecondes pour les applications standard.

Le temps de réponse total (TRT) est le temps qui s'écoule entre le changement d'état d'une entrée et le changement d'état de la sortie correspondante :



http://niedercorn.free.fr/iris/iris1

Bibliographie

[0] M. Gindre, D. Roux « Logique séquentielle » McGraw-Hill

[1] B. Reeb, «Développement des grafcets, des machines simples aux cellules flexibles, du cahier des charges à la programmation », ellipses

[2] R. Gourdeau, G. Cloutier « Introduction à l'automatisme, le Grafcet », cours polycopié à l'école polytechnique de Montréal, septembre 2001.

[3] M. Fryziel « Automates programmables industriels, le langage Grafcet », cours polycopié à l'UT A de Lille 1 – Licence RT

[4] S. Moreno, E. Peulot « Initiation au Grafcet, Memento à l'usage des élèves de lycées professionnels et technologiques », Casteilla