

Chapitre 4

Le GRAFCET

Le GRAFCET est un outil qui a été élaboré par l'AFCET (Association Française pour la Cybernétique Économique et Technique) en 1977. Le mot GRAFCET est l'acronyme de GRaphe Fonctionnel de Commande d'Étapes-Transitions.

Le GRAFCET est diffusé par l'ADEPA (Agence nationale pour le développement de la productique appliquée à l'industrie). Le GRAFCET a fait l'objet d'une norme française en juin 1982 (NFC 03-190). Puis, il a été normalisé au niveau international en 1988 par la Commission électrotechnique internationale sous la norme IEC 848. Il est aussi intégré depuis mars 1993 sous la norme IEC 1131.3. Cette norme définit cinq langages de programmation standards pour les automates programmables.

1 Le pourquoi du GRAFCET

Pour une machine donnée, la description et la compréhension des cycles automatiques doivent être claires pour toutes les personnes chargées d'intervenir. Or lorsque certaines spécifications sont exprimées en langage courant, il y a un risque d'incompréhension. D'une part, certains mots sont peu précis, mal définis ou possèdent plusieurs sens. D'autre part, le langage courant est mal adapté pour décrire précisément les systèmes séquentiels.

Le GRAFCET est un langage graphique pour décrire, étudier, réaliser et exploiter les automatismes. Il est composé d'un ensemble d'étapes et de transitions représentant le déroulement du cycle de l'automatisme. Cette représentation graphique permet une meilleure compréhension de l'automatisme par tous les intervenants.

Un GRAFCET est établi pour chaque machine lors de sa conception, puis utilisé tout au long de sa vie : réalisation, mise au point, maintenance, modifications, réglages.

Le langage GRAFCET doit donc être connu de toutes les personnes concernées par les automatismes, depuis leur conception jusqu'à leur exploitation.

Le GRAFCET est inclut dans une norme internationale du Comité International d'Électrotechnique depuis 1988 (*IEC 848*). La norme *IEC 1131.3* définit les cinq langages de programmation standards pour les automates programmables. Le GRAFCET est nommé en anglais le « *Sequential Function Chart* » ou SFC. Sa pratique s'est déjà généralisée dans de nombreuses industries et dans l'enseignement.

Les avantages de ce mode de représentation sont :

- Il est indépendant de la *matérialisation technologique* ;
- Il traduit de façon cohérente le *cahier des charges* ;
- Il est bien adapté aux *systèmes automatisés*.

2 Les deux niveaux de représentation du GRAFCET

Pour aborder de façon progressive l'étude d'un automatisme, l'analyse GRAFCET est divisée en deux niveaux. Le premier niveau s'attarde aux spécifications fonctionnelles. Le second aux spécifications technologiques.

2.1 Le GRAFCET de niveau 1

Lors de l'analyse des *spécifications fonctionnelles*, le premier souci de l'automaticien est de comprendre le fonctionnement de l'automatisme. Il faut qu'il soit en mesure d'identifier le comportement de la Partie Commande par rapport à la Partie Opérative.

Pour faciliter ce premier niveau d'analyse, il ne faut pas se soucier de la technologie des actionneurs et des capteurs. Le GRAFCET de niveau 1 permet donc de représenter la séquence de fonctionnement souhaitée. Si les pinces d'un manipulateur doivent se fermer ou s'ouvrir, on le mentionne tel quel dans le GRAFCET de niveau 1. On ne doit pas se poser la question à savoir si la pince est commandée par un vérin hydraulique ou un moteur électrique. On doit seulement savoir à quel moment la pince doit s'ouvrir et à quel autre moment elle doit se fermer. La description des actions et de la séquence de l'automatisme est littérale. Enfin, la figure 4-1 montre un exemple de GRAFCET de niveau 1.

Le GRAFCET de niveau 1 permet d'identifier les *fonctions* que doit remplir l'automatisme. Pour chacune de ces fonctions, il faut déduire quelles sont les *actions* à faire, les *informations* assurant que les actions soient complétées et les précautions à prendre du point de vue sécurité, indépendamment de la matérialisation technologique.

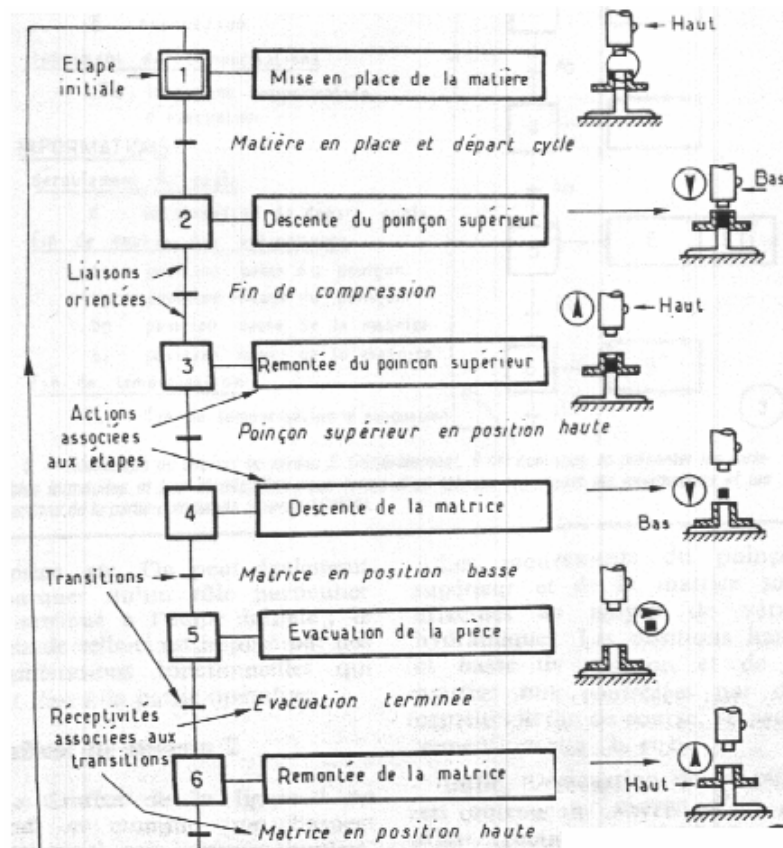


Figure 4-1 : Plateau tournant.

2.2 Le GRAFCET de niveau 2

Lors de l'analyse des *spécifications technologiques*, l'automaticien utilisera l'analyse faite avec le GRAFCET de niveau 1 pour choisir les actionneurs et les capteurs nécessaires pour générer les actions et obtenir les informations nécessaires pour remplir les fonctions.

Le choix technologique est donc fait à cette étape. Par exemple, c'est à ce niveau que l'on décide d'utiliser un vérin hydraulique pour ouvrir ou fermer la pince d'un manipulateur. Chaque capteur et actionneur sont identifiés par un nom logique que l'on utilisera dans le GRAFCET de niveau 2 et lors de la mise en équations du GRAFCET.

Donc le GRAFCET de niveau 2 est celui qui prend en compte la technologie des capteurs et actionneurs. Il pourrait mener à la programmation d'un automate ou à un séquenceur câblé. En pratique, ce GRAFCET sera ultérieurement modifié pour tenir compte des spécifications opérationnelles. La figure 4-2 montre un GRAFCET de niveau 2.

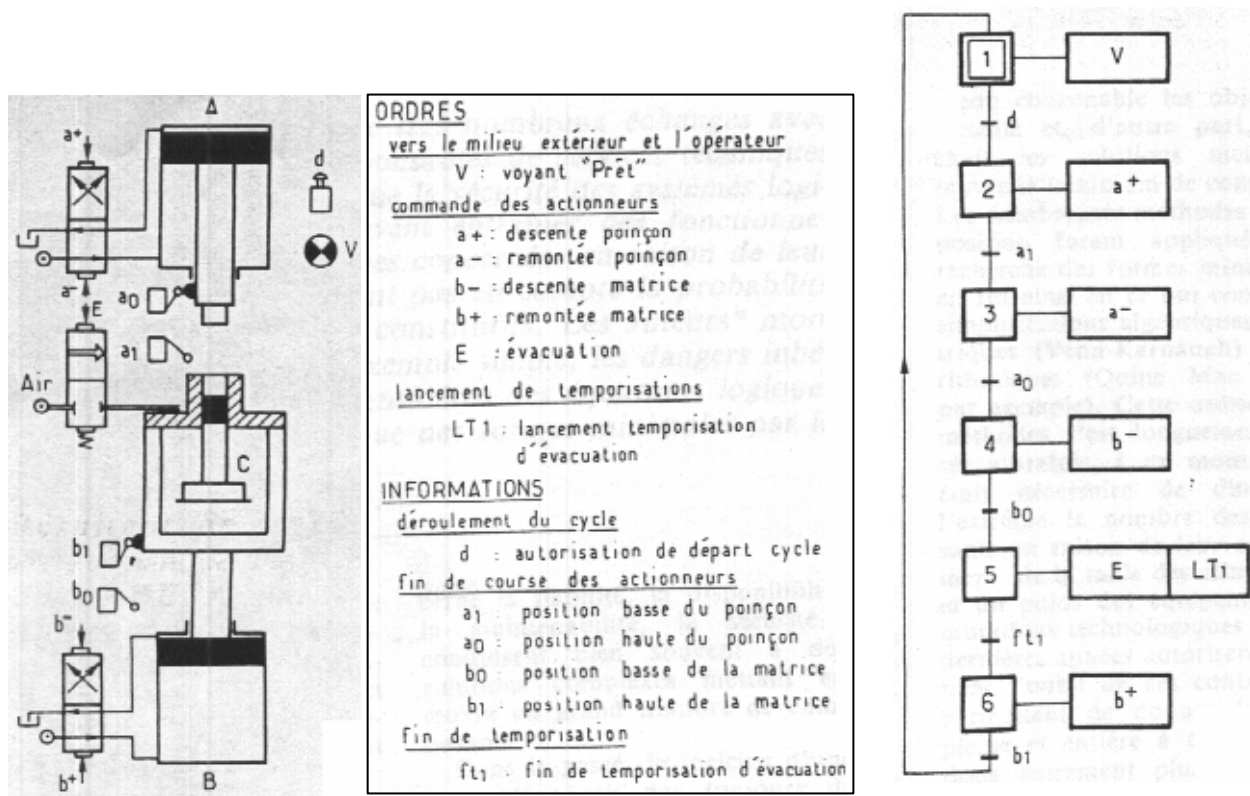


Figure 4-2 : GRAFCET de niveau 2.

En effet, les GRAFCET de niveau 1 et de niveau 2 ne s'attardent qu'au fonctionnement normal de l'automatisme. Dans ce fonctionnement normal, il est assumé que l'automatisme ne manquera jamais de matière première, ne subira jamais d'arrêt d'urgence, ne sera jamais défaillant. Donc les divers modes de marches et d'arrêts ne sont pas pris en compte. Ces modes sont introduits par l'outil méthode « GEMMA » qui est abordé au chapitre suivant.

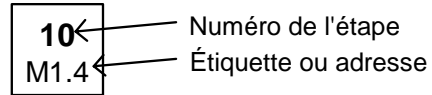
3 Les éléments de base du GRAFCET

Les exemples de GRAFCET des figures 4-1 et 4-2 montrent bien les constituantes d'un GRAFCET. Les paragraphes suivants expliqueront chacun de ces éléments.

3.1 L'étape

L'étape correspond à une situation dans laquelle le comportement du système par rapport à ses entrées et ses sorties est invariant.

Une *étape* est représentée par un carré numéroté.



Le numéro de l'étape est inscrit dans la moitié supérieure du carré, tandis qu'une étiquette ou l'adresse du bit représentant l'état de l'étape dans l'automate est inscrit dans la moitié inférieure si nécessaire (car c'est optionnel). L'étiquette ou l'adresse du bit peuvent être inscrits à côté du carré si l'espace dans le carré est restreint.

Une *étape initiale* est représentée par un carré double.



Une étape initiale doit être activée lors de la mise en énergie de la Partie Commande d'un automatisme.

Une *étape initialisable* est représentée par un carré double dont le carré intérieur est en pointillés (ou traits courts). Une telle étape peut être activée par un ordre de forçage généré par un autre GRAFCET.

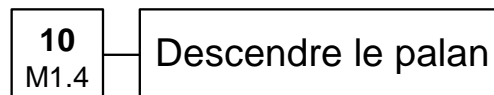


Une *étape* est soit active ou inactive. D'un point de vue logique, l'étape i sera représentée par une variable booléenne x_i . Donc $x_i = 1$ quand l'étape est active et $x_i = 0$ quand l'étape est inactive

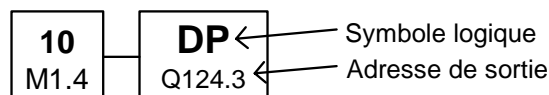
3.2 L'action

L'action sert à décrire les actions à effectuer lorsque l'étape à laquelle elle est associée est active. Une action est représentée par un rectangle qui est relié horizontalement à l'étape (carré) correspondante.

Dans un GRAFCET de niveau 1, l'action est décrite de façon littérale. On recommande fortement l'utilisation de verbes infinitifs (ouvrir, fermer, monter, chauffer, ...).



Dans un GRAFCET de niveau 2, l'action est identifiée par son symbole logique. Cela permet d'avoir une représentation plus compacte. Toutefois il faut prévoir l'ajout d'un tableau donnant la correspondance entre le symbole logique et l'action exécutée (pour que l'on sache que le symbole DP correspond à « Descendre le palan »).

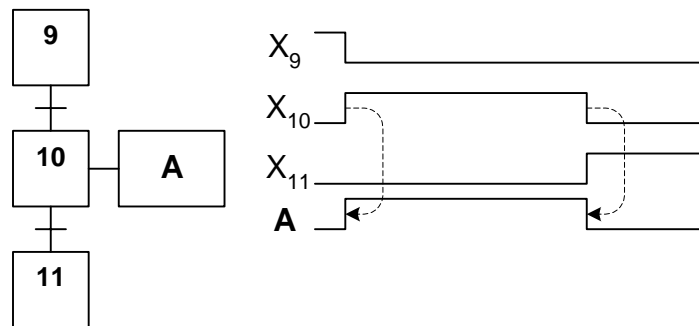


3.3 Les types d'actions

Les actions peuvent être classées selon plusieurs types.

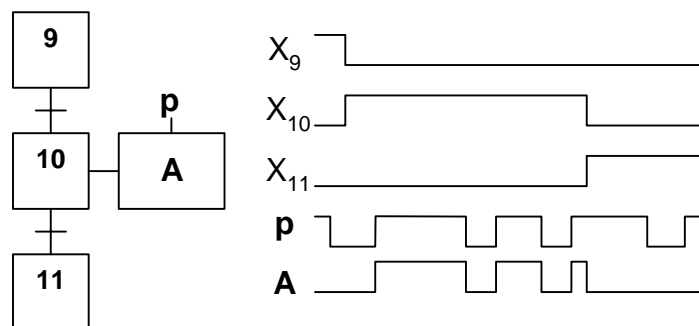
3.3.1 Action continue

L'action continue est un ordre qui dure tant que l'étape à laquelle elle est associée reste active. La figure ci-dessous montre la représentation de l'action sur le GRAFCET (à gauche) et le chronogramme (à droite). Donc on peut conclure qu'au niveau logique $A = X_2$ (en supposant que l'action A ne se produit pas à d'autres étapes).



3.3.2 Action conditionnelle

L'action conditionnelle est un ordre dont l'exécution a lieu à une étape donnée et qui est en plus soumise à la réalisation d'une condition logique. La figure ci-dessous montre la représentation de l'action sur le GRAFCET et le chronogramme. La condition logique appliquée sur l'action est montrée par un petit trait à côté duquel la condition logique est inscrite.

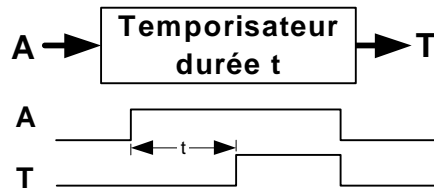


Donc on peut conclure qu'au niveau logique $A = p$ et X_2 (en supposant que l'action A ne se produit pas à d'autres étapes).

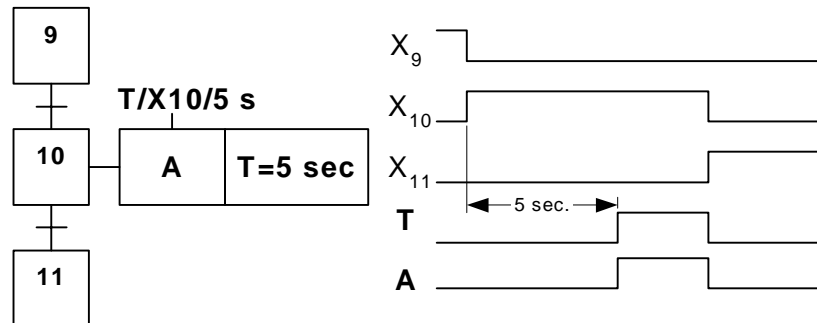
Il est à noter qu'au GRAFCET de niveau 1, l'action littérale serait écrite sous la forme : « action si condition ». Par exemple : « Éjecter au rebut si pièce défectueuse ».

3.3.3 Action temporisée

Une action temporisée est une action conditionnelle dans laquelle le temps intervient comme condition logique. L'action temporisée est obtenue par l'utilisation d'une unité de temporisation (ou temporisateur). Un temporisateur est un système qui retarde la montée à 1 du signal de sortie d'une durée de temps t démarré après la montée à 1 du signal d'entrée. Le retour à 0 du signal de sortie se produit au même moment que celui de l'entrée. Le chronogramme ci dessous montre le fonctionnement du temporisateur.



La notation utilisée pour désigner un signal de sortie d'un temporisateur est : $T/i/q$, où i est le numéro de l'étape comportant l'action de temporisation et q est la durée écoulée depuis l'activation de l'étape i . Voici un exemple :



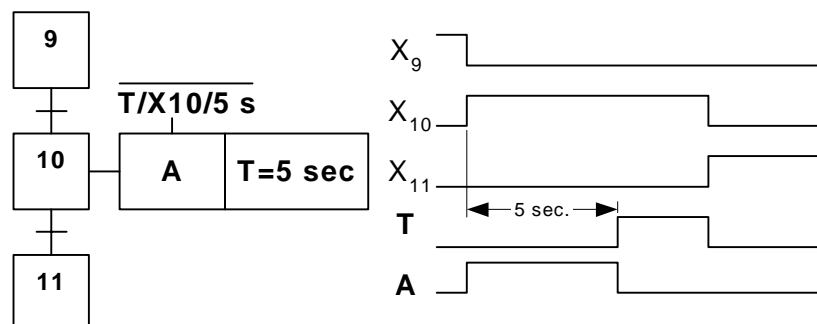
Remarquez qu'il y a deux actions associées à l'étape 10. Le temporisateur de 5 secondes démarre dès que l'étape 10 s'active. L'action A étant conditionnelle à l'état de la sortie du temporisateur, elle ne se produit pas. Dès que le délai de 5 secondes est terminé, la sortie $T/X10/5 \text{ sec}$ monte au niveau logique 1, et l'action A est lancée. La fin (ou désactivation) de l'étape 10 marque la fin de l'action du temporisateur et de l'action A.

Une remarque très importante s'impose ici. Si la durée de l'étape 10 est inférieure à 5 secondes, le temporisateur n'aura pas le temps de monter à 1 et en conséquence, l'action A n'aura pas lieu. La prochaine fois que l'étape 10 s'activera, le délai de 5 secondes se produira avant l'apparition de l'action A.

3.3.4 Action impulsionnelle

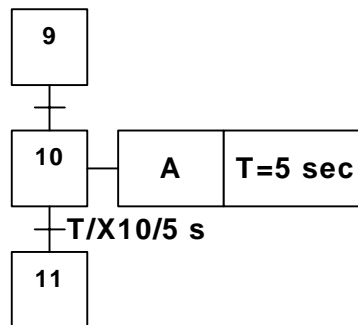
C'est le même principe que l'action temporisée, sauf que cette fois-ci, l'action est active pendant la durée d'une impulsion générée par le temporisateur.

La notation utilisée pour désigner un signal de sortie d'un temporisateur est : $\overline{T}/i/q$, où i est le numéro de l'étape comportant l'action de temporisation et q est la durée écoulée depuis l'activation de l'étape i . Voici un nouvel exemple :



Une remarque similaire à la remarque faite pour l'action temporisée peut être faite. Si la durée de l'étape 10 est inférieure à 5 secondes, le temporisateur n'aura pas le temps de

monter à 1 et en conséquence, l'action A aura une durée inférieure à 5 secondes. Si la durée de 5 secondes est obligatoire, le GRAFCET suivant peut être utilisé :



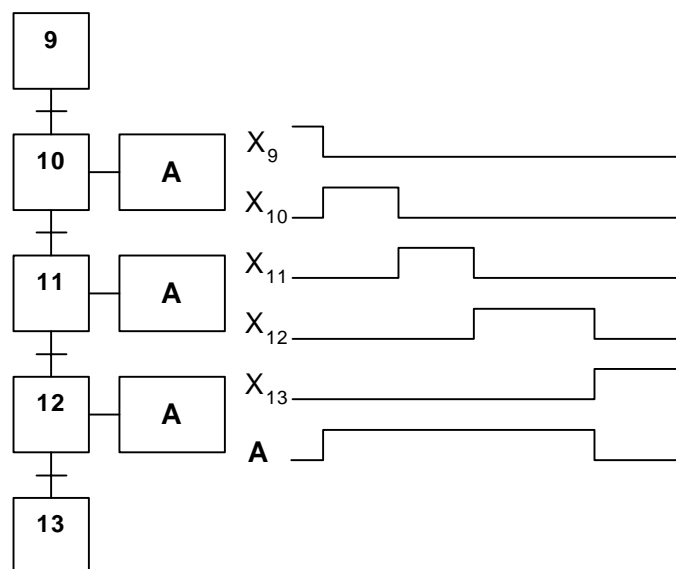
Le temporisateur activera alors la réceptivité de la transition entre les étapes 10 et 11, limitant la durée de l'étape 10 à 5 secondes.

Les quatre types d'actions vues précédemment sont celles de base, celles que l'on rencontre le plus souvent dans l'évolution du GRAFCET. Par contre, il en existe plusieurs autres dont, l'action maintenue et l'effet maintenu par action mémorisée.

3.3.5 Action maintenue

L'action maintenue est une action qui se poursuit tant qu'au moins une des étapes à laquelle elle est associée est active. L'action maintenue est donc associée à une suite d'étapes successives.

La figure ci-dessous montre que l'action A a lieu pendant les étapes 10, 11 ou 12. Au niveau logique, l'équation de l'action A est $A = X_{10} + X_{11} + X_{12}$ (en supposant que l'action A ne se produit pas à d'autres étapes).

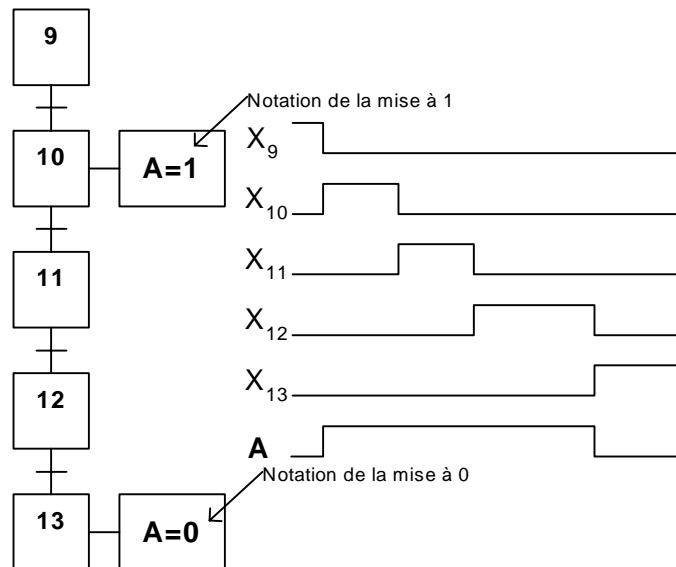


3.3.6 Action à effet maintenue par une action mémorisée

L'action mémorisée est une action qui est s'active à l'étape à laquelle une demande de mise à 1 est faite et désactivée à l'étape ou une demande de mise à 0 est faite.

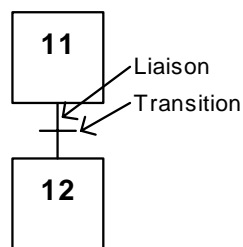
La figure ci-dessous montre que l'action A est lancée lors de l'étape 10 et qu'elle dure jusqu'à la désactivation faite à l'étape 13. La mise à 1 de l'action A est notée sur le GRAFCET « A = 1 » et la mise à 0 est notée « A = 0 ».

Au niveau logique, on écrit la demande de mise à 1 (en anglais « SET ») par « SET(A) = X₂ » et la demande de mise à 0 (en anglais « RESET ») par « RST(A) = X₄ » (en supposant que l'action A ne se produit pas à d'autres étapes). Remarquez l'évolution du niveau logique de A sur le chronogramme.



3.4 Liaison

Ce sont des lignes qui relient les étapes entre elles. Les liaisons entre les étapes sont toujours verticales ou horizontales à moins d'indications contraires (flèche sur la liaison).

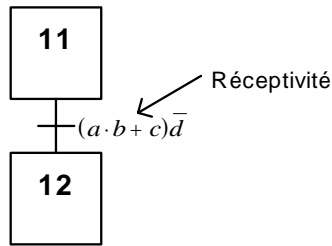


3.5 Transition

Ce sont des barrières entre les étapes qui peuvent être franchies selon certaines conditions. Une transition est représentée par une barre perpendiculaire à la liaison entre deux étapes.

3.6 Réceptivité

Ce sont les conditions logiques qui doivent être remplies pour franchir la transition. La réceptivité est inscrite à la droite de la barre représentant la transition.



3.7 Les types de réceptivité

3.7.1 Réceptivité à niveau

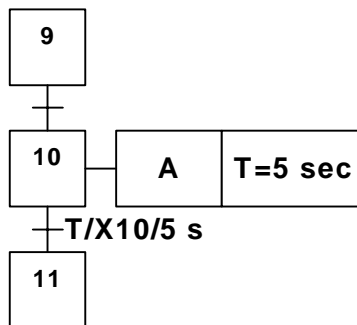
C'est une réceptivité faisant intervenir une condition logique (Ex. \overline{CBA}). La figure ci-dessus en est un exemple.

3.7.2 Réceptivité toujours vraie

C'est une réceptivité dont la valeur logique est toujours 1. On représente la réceptivité par la condition logique « =1 ». La transition associée à cette réceptivité est toujours franchie dès sa validation.

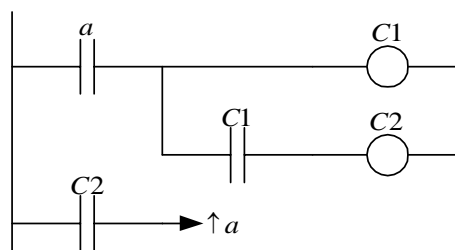
3.7.3 Réceptivité temporisée

C'est une réceptivité faisant intervenir le temps comme condition particulière. Elle implique l'utilisation d'un temporisateur. La notation utilisée pour désigner ce genre de réceptivité est : $T/i/q$ ou $\overline{T/i/q}$, où i est le numéro de l'étape comportant l'action de temporisation et q est la durée écoulée depuis l'activation de l'étape i .



3.7.4 Réceptivité Impulsionnelle:

C'est une réceptivité faisant intervenir un changement d'étape sur une transition soit un front montant ou un front descendant. La notation d'une réceptivité sur front montant est : $\uparrow a$ et celle d'une réceptivité sur front descendant est : $\downarrow a$. La génération d'un front montant peut être obtenue en faisant un câblage comme suit :



Une réceptivité est donc une proposition logique qui peut renfermer diverses variables booléennes qui peuvent être :

- des informations extérieures
 - capteurs ;
 - directives de l'opérateur ;
- des variables auxiliaires
 - compteur ;
 - temporisateurs ;
- de l'état d'autres étapes
 - attentes, interdictions ;
- des transitions sur d'autres variables
 - fronts montants ou descendants.

4 Les règles d'évolution du GRAFCET

4.1 La règle 1 : L'initialisation

Il existe toujours au moins une étape active lors du lancement de l'automatisme. Ces étapes activées lors du lancement sont nommées « Étapes Initiales ».

4.2 La règle 2 : La validation

Une transition est soit validée ou non-validée. Une transition est validée lorsque toutes les étapes immédiatement précédentes sont *actives*. Elle ne pourra être franchie que lorsqu'elle est validée ET que la réceptivité associée est vraie. Des exemples montrés ci-dessous montrent comment interpréter cette règle.

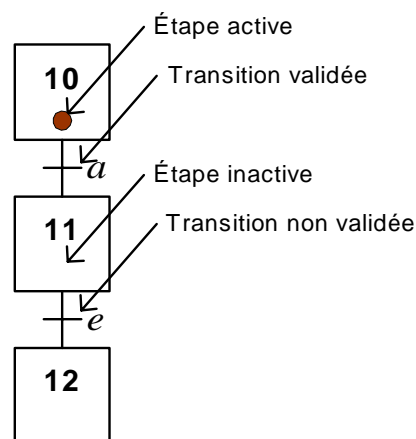


Figure 4-3 : Exemple avec GRAFCET à séquence unique.

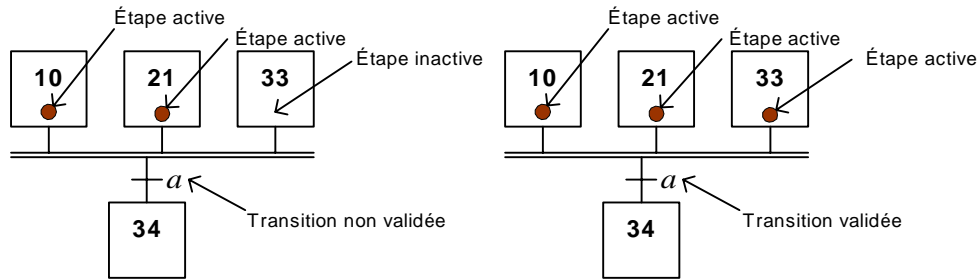


Figure 4-4 : Exemple avec GRAFCET à séquences simultanées.

4.3 La règle 3 : Le franchissement

Une transition est franchie lorsqu'elle est validée et que la *réceptivité* associée à la transition est *vraie*. Le franchissement entraîne l'activation de toutes les étapes immédiatement suivantes et la désactivation de toutes les étapes précédentes. Des exemples montrés ci-dessous montrent comment interpréter cette règle en visualisant la situation avant et après le franchissement.

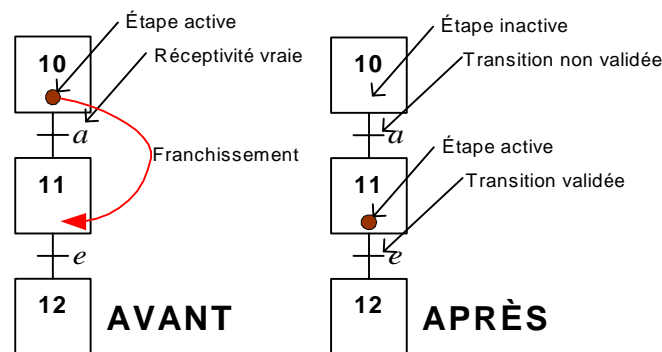


Figure 4-5 : Exemple avec GRAFCET à séquence unique.

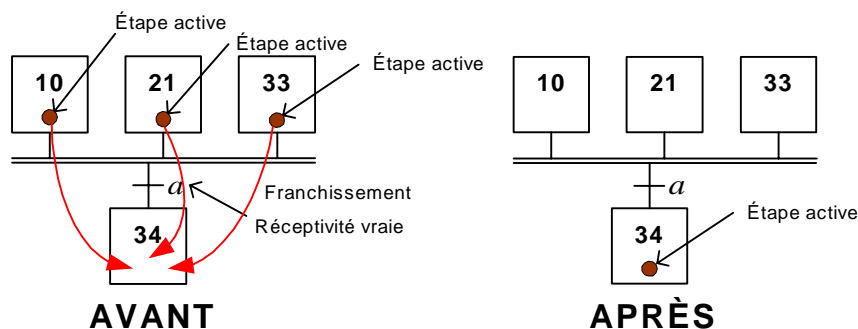


Figure 4-6 : Exemple avec GRAFCET à séquences simultanées.

4.4 La règle 4 : Le franchissement de plusieurs transitions

Plusieurs transitions simultanément franchissables sont simultanément franchies.

4.5 La règle 5

Si au cours du fonctionnement une même étape doit être activée et désactivée simultanément, elle reste activée. Cela est nécessaire à la cohérence théorique du GRAFCET.

Note : En plus de ces règles, l'alternance étape-transition, étape-transition, étape-transition, doit toujours être respectée.

5 Les structures de base du GRAFCET

5.1 La séquence unique

Le GRAFCET le plus simple, c'est la séquence unique qui donne un GRAFCET très linéaire. La figure en page 4-3 en donne un exemple.

5.2 Les sélections de séquences

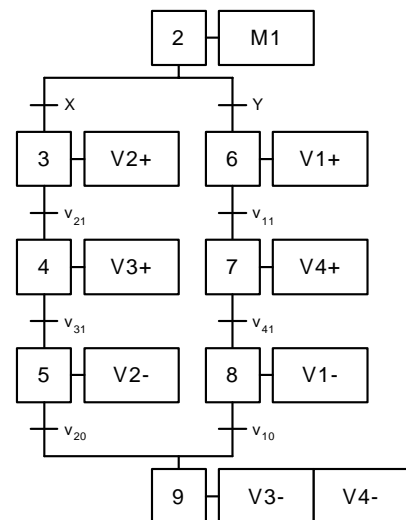
Lorsque l'automatisme doit avoir un comportement qui peut changer en fonction de l'état de certains capteurs, on parle alors de sélection de séquences. Il existe trois types de sélection de séquences.

5.2.1 Séquences exclusives

Les séquences exclusives permettent de choisir entre deux séries d'étapes celle qui sera exécutée.

En fonction de l'état des capteurs « X » et « Y », le GRAFCET montré à gauche passera de l'étape 2 à l'étape 3 ou à l'étape 6 respectivement. Une seule séquence est donc choisie, ce qui explique le terme de « séquences exclusives ».

Les valeurs logiques « X » et « Y » doivent être mutuellement exclusives.

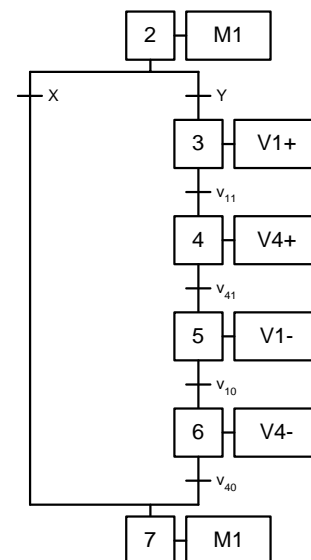


5.2.2 Saut d'étapes

Le saut d'étape sert à permettre l'exécution ou non (on parle alors de saut) d'une série d'étapes.

Dans l'exemple montré ci-contre à gauche, les étapes 3, 4, 5 et 6 sont exécutées si « Y » est activé. Si « X » est activé, on saute ces quatre étapes en passant de l'étape 2 à l'étape 7.

Il faut que les variables logiques « X » et « Y » soient mutuellement exclusives.

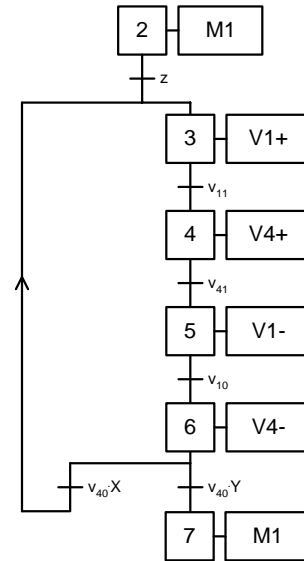


5.2.3 Reprises d'étape

La reprise d'étape permet de reprendre ou non une série d'étapes à plusieurs reprises.

Dans l'exemple montré ci-contre à gauche, les étapes 3, 4, 5 et 6 ne seront faites qu'une seule fois si la variable « Y » est active lors de l'étape 6. Si la variable « X » est active à l'étape 6, l'automatisme reprendra les étapes 3, 4, 5 et 6.

Il faut que les variables logiques « X » et « Y » soient mutuellement exclusives.



! Il est important de noter que les réceptivités mutuellement exclusives ne doivent jamais être vraies en même temps. Il faut donc avoir la certitude que cette condition soit remplie. Ce peut être parce que physiquement les réceptivités ne peuvent être actives en même temps (par exemple, un réservoir ne peut être plein et vide en même temps). Ce peut être en modifiant les réceptivités en les forçant à être mutuellement exclusives (par exemple en modifiant dans les exemples X et Y par $X\bar{Y}$ et $\bar{X}Y$).

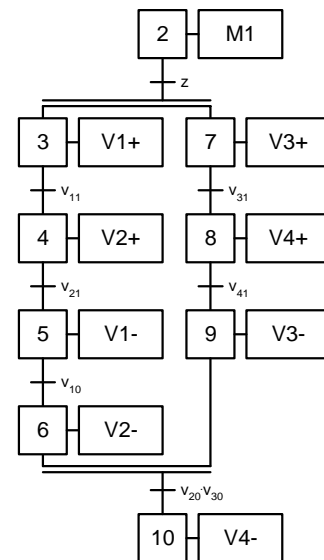
5.3 Les séquences simultanées

Lorsque l'automatisme est en mesure de faire certaines opérations simultanément (par exemple, remplir une bouteille et en boucher une autre), il est possible d'utiliser le parallélisme structural pour faire des séquences simultanées.

Dans l'exemple montré ci-contre à gauche, lorsque l'étape 2 est active et que la réceptivité « z » est vraie, les deux séquences démarrent en même temps. Donc les étapes 3, et 7 s'activent en même temps.

Ensuite, chaque séquence évolue de façon indépendante à son propre rythme. Pour passer à l'étape qui suit les séquences simultanées, il faut que les dernières étapes de toutes les séquences soient actives et que la réceptivité commune soit vraie.

Donc on passera à l'étape 10, si les étapes 6 et 9 sont actives et que la réceptivité $v_{20}v_{30}$ est vraie.

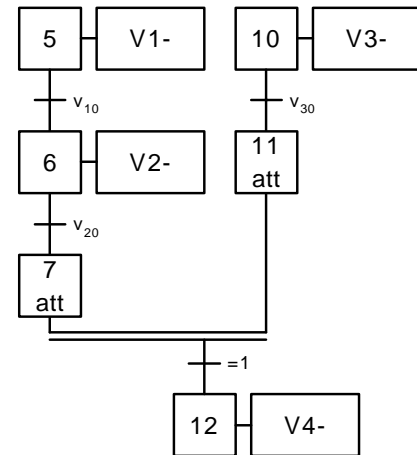


! Dans certaines applications, il est très fortement recommandé de terminer chaque séquence avec une étape d'attente. C'est particulièrement vrai avec les actionneurs électriques. Supposez qu'à l'étape 6, l'action V2- représente le déplacement d'un pont roulant. Si le pont roulant arrive vis-à-vis le capteur v_{20} , mais que l'autre séquence n'est pas à l'étape 10, alors on ne passe pas à l'étape 12 et l'étape 6 reste active. Donc le pont roulant

poursuit son mouvement et s'il devait s'arrêter lorsque v_{20} devient vrai, c'est raté. Si en plus v_{20} était un capteur indiquant que l'on arrive à la fin du rail, le problème est grave (quoique l'on doit toujours prévoir un interrupteur de surcourse, « limit switch »).

L'ajout d'une étape d'attente évite ce problème, puisque lors de l'étape d'attente il ne se produit pas d'action. Ainsi, dans la section de GRAFCET de droite, lorsque l'étape 6 est active, dès que v_{20} devient vrai on passe à l'étape 7 et l'action $V2-$ n'a plus lieu. Ainsi, si on reprend l'exemple du pont roulant, celui-ci s'arrêtera maintenant au bon endroit.

Comme les réceptivités indiquant que les séquences sont terminées se font avant les étapes d'attente, alors il suffit pour passer à l'étape suivante de vérifier que toutes les séquences sont complétées. La réceptivité qui suit les séquences simultanées se terminant par une étape d'attente sur chaque branche est identifiée par « =1 » qui signifie qu'elle est toujours vraie. Ce qui est normal, car toutes les séquences sont en attente.

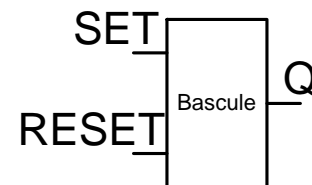


6 La mise en équation du GRAFCET

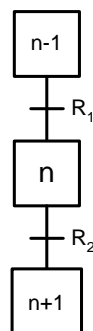
Malheureusement, ce ne sont pas tous les automates qui se programment en GRAFCET directement. Mais, généralement ils peuvent être programmés en « diagramme échelle ».

Il faut donc pouvoir transformer le GRAFCET qui est la meilleure approche qui existe pour traiter les systèmes séquentiels en « diagramme échelle » qui est le langage le plus utilisé par les automates.

Quatre variantes de transformation existent et sont très équivalentes. Elles reposent sur l'utilisation de bascules. Une bascule est une fonction logique ayant deux entrées « SET » et « RESET » et une sortie « Q ». Lorsque l'entrée « SET » est active, la sortie « Q » est mise à 1 et reste à 1 tant que l'entrée « RESET » n'est pas active. Si l'entrée « RESET » est active, la sortie « Q » est mise à 0 et reste à 0 tant que l'entrée « SET » n'est pas active. Si les deux entrées sont actives en même temps, l'état de la sortie « Q » dépendra du design interne de la bascule (ce qui explique le nombre de variantes).



Pour montrer comment le GRAFCET se transforme en diagramme échelle, supposons une suite de trois étapes tel que montré ci-dessous :



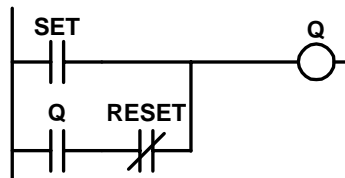
Pour trouver l'équation logique de X_n , la variable logique représentant l'étape « n », il faut appliquer les règles du GRAFCET.

L'étape « n » s'activera lorsque la réceptivité R_1 sera franchie. Cette réceptivité est franchie si l'étape « n-1 » est active et si la condition logique R_1 est vraie. L'étape « n » s'activera alors et désactivera l'étape « n-1 ». L'équation logique de la mise à 1 de X_n sera : $X_{n-1}R_1$.

L'étape « n » se désactivera lorsque la réceptivité R_2 sera franchie. Lorsque le franchissement se fera, l'étape « n+1 » s'activera et l'étape « n » se désactivera. L'équation logique de la mise à 0 de X_n sera : $\overline{X_{n+1}}$.

6.1 La variante #1

La première variante est l'utilisation de bascules ayant une priorité à l'activation. Dans cette variante, la bascule mettra la sortie « Q » à 1 si les deux entrées sont activées simultanément. En diagramme échelle, le schéma de cette bascule est :

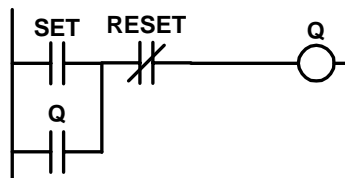


Si on applique cette représentation pour écrire l'équation logique de X_n , on aura que :

$$X_n = (X_{n-1}R_1 + X_n \overline{X_{n+1}})$$

6.2 La variante #2

La seconde variante est l'utilisation de bascules ayant une priorité à la désactivation. Dans cette variante, la bascule mettra la sortie « Q » à 0 si les deux entrées sont activées simultanément. En diagramme échelle, le schéma de cette bascule est :



Si on applique cette représentation pour écrire l'équation logique de X_n , on aura que :

$$X_n = (X_{n-1}R_1 + X_n) \overline{X_{n+1}}$$

Cette variante viole la règle 5 du GRAFCET et ne devrait pas être utilisée.

6.3 La variante #3

La troisième variante utilise la fonction bascule de l'automate. Dans cette application on aura deux équations logiques.

L'équation logique de SET X_n est : SET $X_n = X_{n-1}R_1$ et de RESET X_n est : RESET $X_n = X_{n+1}$.

6.4 La variante #4 (A UTILISER OBLIGATOIREMENT)

Dans la quatrième variante les transitions sont évaluées indépendamment des étapes du GRAFCET, ce qui permet d'éviter d'avoir deux étapes actives en même temps lors des transitions d'une étape à l'autre. Ce problème d'étapes actives en même temps devient dangereux lorsque le temps de scrutation de l'automate devient long.

Chaque transition sera identifiée par Y_n . Ainsi, la transition entre l'étape $n-1$ et n sera identifiée Y_{n-1} , puis celle entre l'étape n et $n+1$ sera identifiée Y_n , et ainsi de suite...

L'équation logique du franchissement de la transition Y_n est : $Y_n = X_n R_2$. Elle donne un niveau logique 1 lorsque l'étape qui la précède X_n est active et lorsque la réceptivité R_2 est vraie.

L'équation logique de l'étape X_n est : $X_n = Y_{n-1} + X_n \overline{Y_n}$. L'étape X_n s'active si la transition précédente Y_{n-1} est franchie et se désactive si la transition Y_n suivante est franchie.

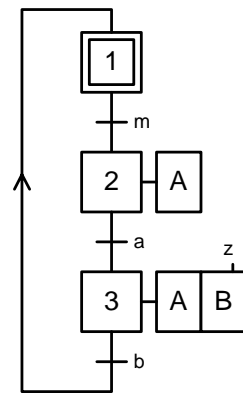
La variante #4 devra être utilisée puisque qu'elle élimine bien des problèmes. Elle est supérieure à toutes les autres variantes.

6.5 Exemples de mise en équation

Exemple #1 :

Voici la mise en équations du petit GRAFCET de droite :

$$\begin{aligned} Y_1 &= X_1 m \\ Y_2 &= X_2 a \\ Y_3 &= X_3 b \\ X_1 &= (Y_3 + X_1 \overline{Y_1}) + \text{Init} \\ X_2 &= Y_1 + X_2 \overline{Y_2} \\ X_3 &= Y_2 + X_3 \overline{Y_3} \\ A &= X_2 + X_3 \\ B &= X_3 z \end{aligned}$$



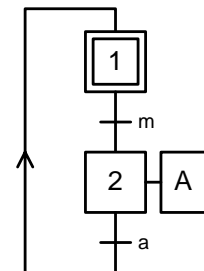
Dans cet exemple deux concepts sont introduits. Le premier, c'est qu'un GRAFCET converti en équations fait apparaître les équations des étapes et les équations des actions.

Le second, c'est le concept d'initialisation. Un GRAFCET comporte une *étape initiale* qui doit être activée lors de la mise sous tension de l'automatisme. Généralement, un automate génère une *impulsion d'initialisation* lors de sa mise en marche (lorsqu'il passe en mode « RUN »). Cette impulsion identifiée « Init » sert à activer la première étape de notre GRAFCET en forçant l'équation de l'étape initiale à 1 lors du démarrage de l'automatisme.

Exemple #2 :

Voici la mise en équations du petit GRAFCET de droite en utilisant la variante #2 :

$$\begin{aligned} X_1 &= (X_2 a + X_1) \overline{X_2} + \text{Init} \\ X_2 &= (X_1 m + X_2) \overline{X_1} \\ A &= X_2 \end{aligned}$$



Ces équations font apparaître des contradictions de taille. Par exemple, l'équation de X_2 peut se lire comme suit : « Pour activer l'étape 2, il faut être à l'étape 1 et la réceptivité « m » doit être vraie, et pour désactiver, il faut être à l'étape 1 ». Or la désactivation étant prioritaire, il est impossible de d'activer l'étape 2. Cela vient du fait que cette variante viole la règle 5 du GRAFCET.

On règle ce problème en utilisant la première variante, ce qui donne les équations suivantes ou l'activation est prioritaire :

$$X_1 = (X_2 a + X_1 \overline{X_2}) + \text{Init}$$

$$X_2 = (X_1 m + X_2 \overline{X_1})$$

$$A = X_2$$

On suggère toutefois d'utiliser la « variante 4 » qui mènerait à ce petit programme :

$$Y_1 = X_1 m$$

$$Y_2 = X_2 a$$

$$X_1 = (Y_2 + X_1 \overline{Y_1}) + \text{Init}$$

$$X_2 = Y_1 m + X_2 \overline{Y_2}$$

$$A = X_2$$

Exemple #3 :

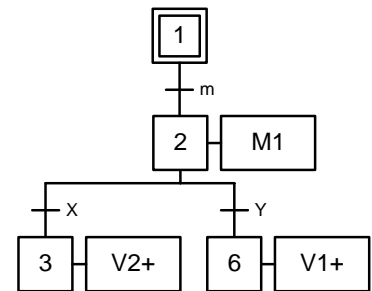
Voici l'équation de l'étape 2 du GRAFCET de droite :

$$Y_1 = X_1 m$$

$$Y_{2a} = X_2 X$$

$$Y_{2b} = X_2 Y$$

$$X_2 = Y_1 + X_2 (\overline{Y_{2a}} + \overline{Y_{2b}}) = X_1 m + X_2 \overline{Y_{2a}} \overline{Y_{2b}}$$



L'utilisation de la loi de Morgan est nécessaire, car la fonction logique non-ou n'est pas implantable en diagramme échelle.

Exemple #4 :

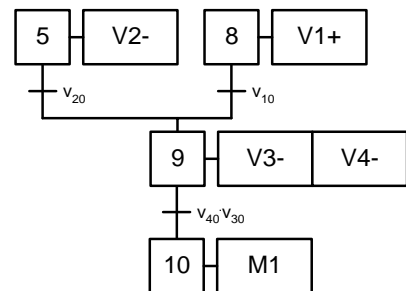
Voici l'équation de l'étape 9 du GRAFCET de droite :

$$Y_5 = X_5 v_{20}$$

$$Y_8 = X_8 v_{10}$$

$$Y_9 = X_9 v_{40} v_{30}$$

$$X_9 = (Y_5 + Y_8 + X_9 \overline{Y_9})$$



Comme le GRAFCET peut arriver à l'étape 9 de l'étape 5 ou de l'étape 8 puisqu'il comporte deux séquences exclusives, cela apparaît dans la partie activation de l'équation.

Exemple #5 :

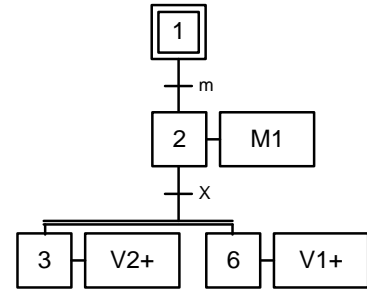
Voici l'équation de l'étape 2 du GRAFCET de droite :

$$Y_1 = X_1 m$$

$$Y_2 = X_2 X$$

$$X_2 = (Y_1 + X_2 \overline{Y_2})$$

L'équation reste simple puisque l'on utilise la transition Y_2 pour désactiver l'étape 2.



Exemple #6 :

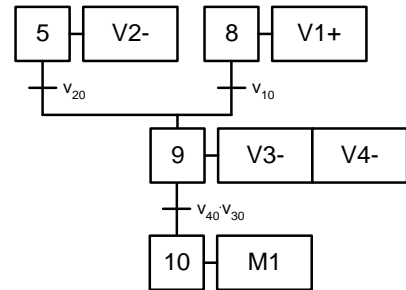
Voici l'équation de l'étape 9 du GRAFCET de droite :

$$Y_{5-8} = X_5 X_8 v_{10} v_{20}$$

$$Y_9 = X_9 v_{40} v_{30}$$

$$X_9 = Y_{5-8} + X_9 \overline{Y_9}$$

Le GRAFCET exige pour activer à l'étape 9 que les étapes 5 et 8 soient actives et que la réceptivité soit vraie, tel que montré par la partie activation de l'équation.



7 Les extensions du modèle GRAFCET

7.1.1 Étapes sources

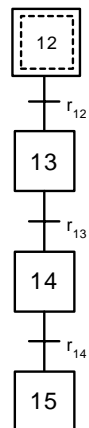
On appelle étape source une étape non reliée à une transition amont. Cette étape ne peut être activée que par un ordre de forçage, puisque aucune étape et transition ne la précède (exemple : l'étape 12 du GRAFCET de gauche).

Une étape initiale sans transition amont est une étape source activée inconditionnellement à la mise sous tension

Pour montrer la mise en équation d'une étape source, supposons que l'étape 12 est forcée par l'étape 5 d'un autre GRAFCET, alors l'équation sera :

$$Y_{12} = X_{12} r_{12}$$

$$X_{12} = X_5 + X_{12} \overline{Y_{12}}$$



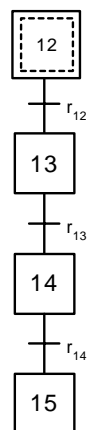
7.2 Étapes puits

On appelle étape puits une étape non suivie d'une transition, cette étape ne peut être désactivée que par un ordre de forçage (exemple : l'étape 15 du GRAFCET de droite).

Supposons que l'étape 15 soit désactivée par l'étape 9 d'un autre GRAFCET, alors l'équation sera :

$$Y_{14} = X_{14} r_{14}$$

$$X_{15} = Y_{14} + X_{15} \overline{X_9}$$



7.3 Transitions sources

Une transition source est une transition non précédée d'une étape (exemple : celle qui précède l'étape 13).

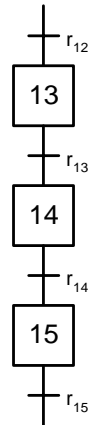
Par convention cette transition est toujours validée (malgré l'absence d'étape précédente), pour la franchir, il suffit que la réceptivité soit vraie.

La réceptivité associée à une transition source est généralement une réceptivité sur front (sinon l'étape suivante est activée en permanence).

Ainsi, l'équation de l'étape 13 serait :

$$Y_{13} = X_{13} r_{13}$$

$$X_{13} = r_{12} + X_{13} \overline{Y_{13}}$$



7.4 Transitions puits

Une transition puits est une transition non suivie d'une étape (exemple : celle qui suit l'étape 15). Les règles de franchissement s'applique à ce type de transition, lors du franchissement l'étape précédente est désactivée, aucune autre étape n'est activée.

Ainsi l'équation de l'étape 15 sera :

$$Y_{14} = X_{14} r_{14}$$

$$Y_{15} = X_{15} r_{15}$$

$$X_{15} = Y_{14} + X_{15} \overline{Y_{15}}$$

8 Hiérarchie et forçage de situation

8.1 Hiérarchisation

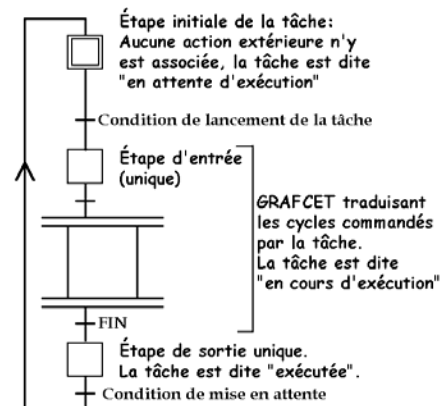
Dans la plupart des applications, il est plus simple d'utiliser plusieurs GRAFCET travaillant ensemble pour contrôler un automate. Cette notion devient plus évidente lorsque l'étude GEMMA est faite et que l'on doit obtenir un GRAFCET final intégrant le GRAFCET de niveau 2 du fonctionnement normal et les modes de marches et d'arrêts générés par le GEMMA. Mais parfois, même en analysant simplement le fonctionnement normal on peut se rendre compte que l'utilisation de plusieurs GRAFCET simplifie la tâche. Cela introduit deux nouvelles notions, celle de « tâche » et celle de « macro-étape ».

8.2 Les GRAFCET de tâches

Les GRAFCET de tâches sont l'équivalents de sous-programmes pour décrire le fonctionnement de sous-ensembles de la machine (ou sous machines).

La structure d'une tâche est simple. Elle comporte une étape initiale unique. Lorsque la tâche est à cette étape, la tâche est dite « en attente ».

La réceptivité qui suit cette étape initiale est la « condition de lancement de la tâche ». Cette réceptivité est soit une étape d'un GRAFCET de conduite, soit une étape de fin de



tâche d'une autre tâche.

La tâche est ensuite décrite par les étapes suivantes de ce GRAFCET. Lorsque la tâche est dans ces étapes, la tâche est en « cours d'exécution ». Lorsqu'elle se termine, la tâche se retrouve à une étape finale indiquant que la tâche est « exécutée ». Un signal de mise en attente retourne cette tâche à son étape initiale, prête à être exécutée de nouveau.

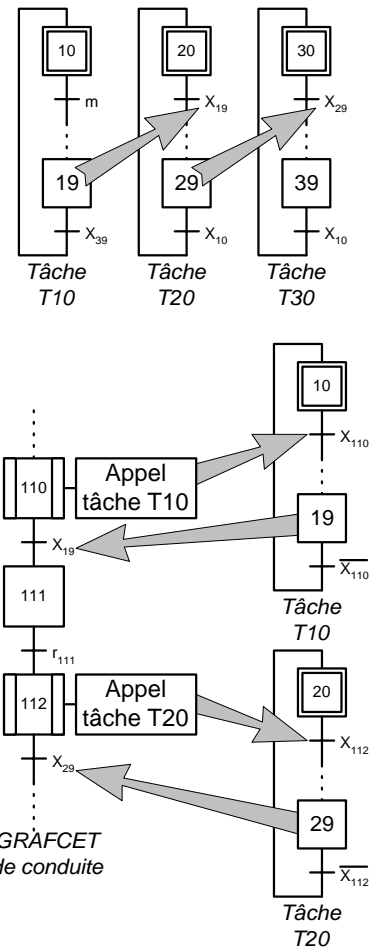
Ces GRAFCET de tâches peuvent être structurés entre-eux selon une coordination horizontale. Les tâches ont toutes le même niveau hiérarchique et une seule tâche à la fois est en cours d'exécution. Pensez à des coureurs à relais, seul un coureur à la fois transporte le relais.

Ces GRAFCET de tâches peuvent aussi être structurées selon une coordination verticale introduisant une hiérarchie. Un GRAFCET maître dirige un ensemble de GRAFCET esclaves qui sont des tâches.

Le GRAFCET de conduite appelle donc des tâches et on utilise un symbole spécial pour identifier les étapes d'appel de tâches (exemple : étapes 110 et 112).

Il est à noter les conditions logiques de lancement et de fin de tâches et la synchronisation qui se produit alors entre les GRAFCET. La figure de gauche montre clairement ce qu'il en est.

Lorsque le GEMMA sera abordé, il vous sera présenté des exemples montrant une telle structure de GRAFCET.



8.3 Les macro-étapes

Une macro-étape sert à représenter un ensemble d'étapes et de transitions. Elle se substitue à une étape de GRAFCET.

Les trois règles suivantes s'appliquent :

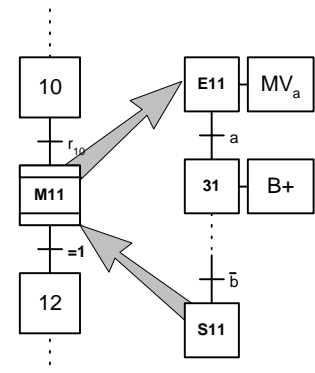
1. L'expansion de la macro-étape comporte une étape d'entrée repérée « E » et une étape de sortie repérée « S ». La macro-étape elle-même est identifiée par « M » et par le symbole montré à gauche;
2. Tout franchissement de la transition précédant la macro-étape (r_{10} dans l'exemple) active l'étape d'entrée. Ainsi, dans l'exemple, l'étape E11 s'active en même temps que la macro-étape M11;
3. L'étape de sortie participe à l'activation de la transition suivant la macro-étape. Donc l'équation de l'étape 12 serait :

$$Y_{11} = X_{M11} X_{S11}$$

$$Y_{12} = X_{12} r_{12}$$

$$X_{12} = Y_{11} + X_{12} \overline{Y_{12}}$$

(Notez la présence de X_{S11}).



Il est important d'indiquer qu'il est préférable d'éviter toute action aux étapes d'entrée et de sortie.

La macro-étape permet donc de simplifier la représentation du GRAFCET. Une macro-étape n'est pas une sous routine comme une tâche, mais une expansion du GRAFCET.

8.4 Le forçage

Le forçage est une approche simple pour obliger un GRAFCET à changer de comportement. Cette méthode est souvent la résultante de l'analyse GEMMA qui introduit des modes de marches et d'arrêts qui changent le comportement d'un automate. Un exemple, c'est l'arrêt d'urgence qui nécessite souvent une intervention rapide pour inhiber le fonctionnement normal et entamer des séquences d'urgence.

Le forçage se définit donc comme le passage imposé de la situation actuelle du GRAFCET en une situation déterminée différente de celle qu'il aurait atteint avec une évolution normale.

L'utilisation du forçage implique l'utilisation de tâches. Les tâches peuvent être hiérarchisées ou non. Dans le cas des tâches hiérarchisées, ce sont des GRAFCET de niveau hiérarchique supérieur qui forcent des GRAFCET de niveau hiérarchique inférieur.

Le forçage doit suivre les règles suivantes :

1. Le forçage est un ordre interne au GRAFCET consécutif à une évolution. Pour une situation comportant plusieurs ordres de forçages, les GRAFCET forcés prendront immédiatement et directement la ou les situations imposées.
2. Toute apparition d'une situation de forçage est prioritaire par rapport à toute autre activité du GRAFCET.
3. Les GRAFCET forcés sont maintenus dans la situation de forçage tant que l'ordre de forçage est valide.
4. Un GRAFCET ne peut être forcé que par un GRAFCET hiérarchiquement supérieur.
5. A tout instant un GRAFCET ne peut être forcé que dans une situation et une seule à partir d'un et un seul GRAFCET supérieur.

Le forçage est représenté par une action ayant la syntaxe suivante :

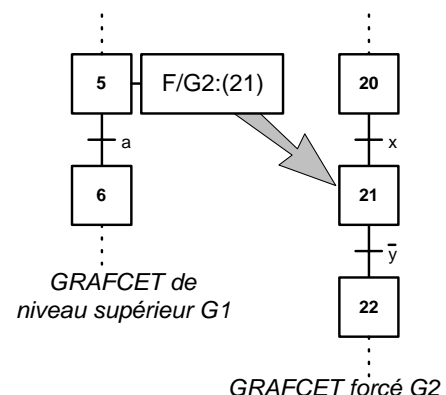
F/nom du GRAFCET : (Situation)

Ainsi, dans l'exemple ci-contre, lorsque le GRAFCET G1 est à l'étape 5, le GRAFCET G2 est forcé à être à l'étape 21. Pour que le GRAFCET G2 évolue, il faut que le GRAFCET G1 passe à l'étape 6, sinon, le GRAFCET G2 reste forcé à l'étape 21.

L'équation représentant l'étape 21 du GRAFCET G2 est :

$$\begin{aligned} Y_{20} &= X_{20} \cdot x \\ Y_{21} &= X_{21} \cdot \bar{y} \\ X_{21} &= [Y_{20} + X_{21} \cdot \bar{Y}_{21}] + X_5 \end{aligned}$$

Si la situation est vide, i.e., que le forçage est identifié par « F/G2:() », toutes les étapes du GRAFCET G2 se désactivent. Il faut bien sûr réactiver au moins une étape de ce GRAFCET pour qui puisse se remettre en route. Si le figeage de l'étape 5 eu été vide, l'équation de l'étape 21 du GRAFCET G2 aurait été :



$$Y_{20} = X_{20} \cdot x$$

$$Y_{21} = X_{21} \cdot \bar{y}$$

$$X_{21} = [Y_{20} + X_{21} \cdot \bar{Y}_{21}] \cdot \bar{X}_5$$

Le figeage d'un GRAFCET est un cas particulier d'un forçage. Le GRAFCET figé voit son évolution bloquée, quels que soient les réceptivités. Le forçage est représenté par une action ayant la syntaxe suivante :

F/nom du GRAFCET : (*)

⚠ Le figeage doit se faire de façon prudente, car les actions se poursuivent même si le GRAFCET est figé. Il faut donc souvent bloquer les actions en même temps que l'on fige un GRAFCET.

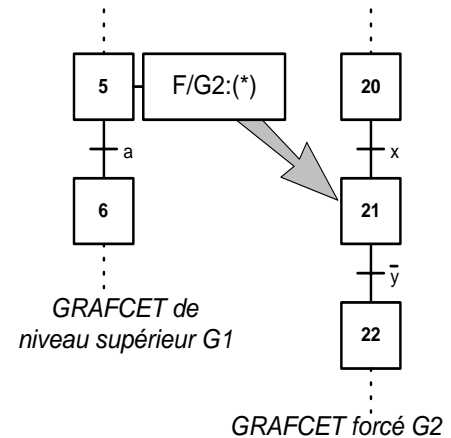
L'équation représentant l'étape 21 du GRAFCET figé est :

$$Y_{20} = X_{20} \cdot x \cdot \bar{X}_5$$

$$Y_{21} = X_{21} \cdot \bar{y} \cdot \bar{X}_5$$

$$X_{21} = Y_{20} + X_{21} \cdot \bar{Y}_{21}$$

Les diverses applications du forçage et du figeage résultent de l'utilisation du GEMMA et le chapitre suivant présentera quelques exemples.



9 Exemples

9.1 La machine à remplir et à boucher

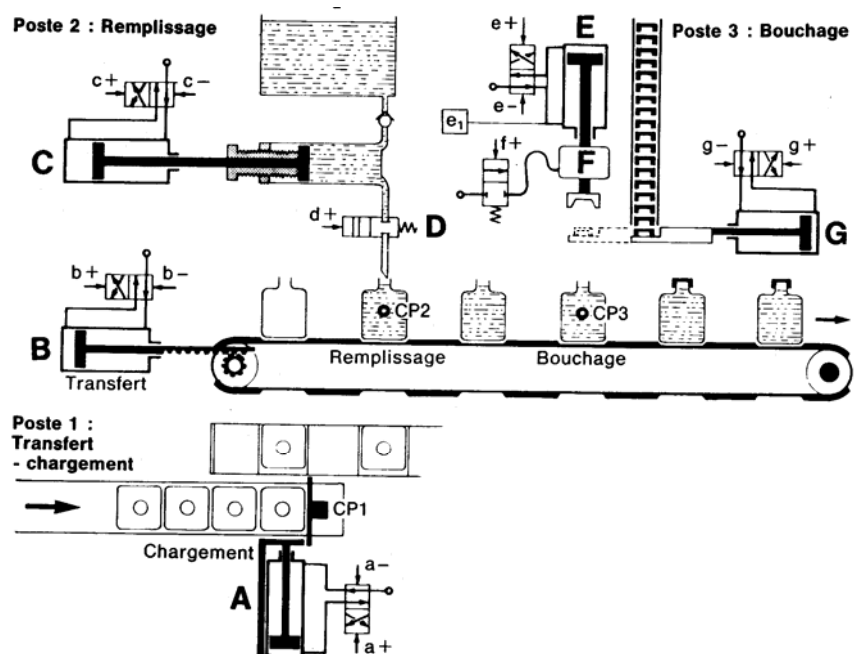
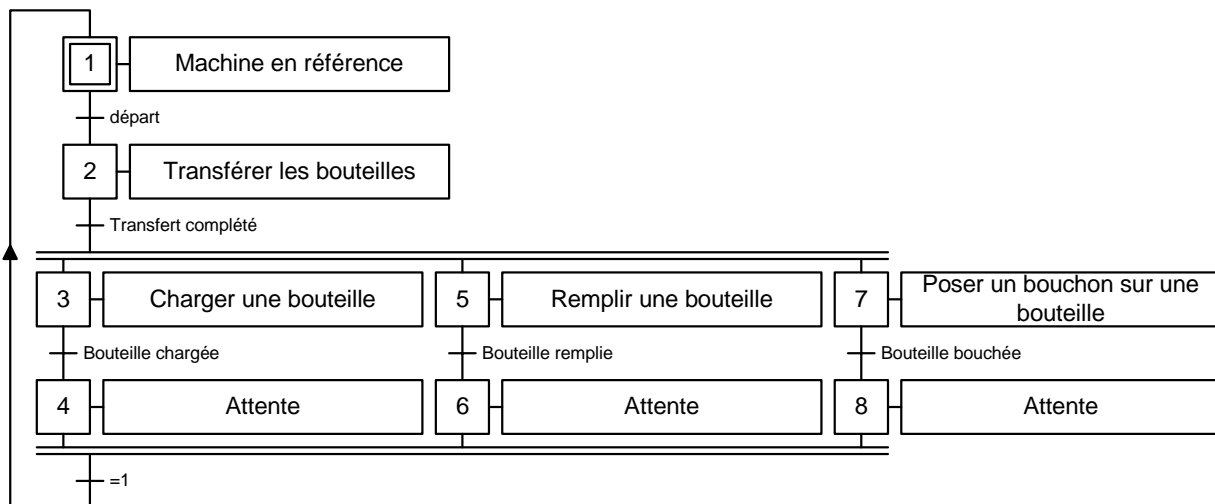


Figure 4-7 : Schéma de la machine à remplir et à boucher des bouteilles.

La machine à remplir et à boucher est composée de trois postes pouvant travailler simultanément :

- Le poste 1 sert au transfert et au chargement. Dans un premier temps, on sort le vérin de transfert B pour à décaler le convoyeur d'une position vers la droite. Ensuite, le vérin A sert au chargement d'une nouvelle bouteille vide.
- Le poste 2 sert au remplissage des bouteilles. On sort le vérin C, tout en ouvrant la vanne D, pour vider le contenu du cylindre doseur dans la bouteille. Ensuite, on ferme la vanne D et on rétracte le vérin C pour remplir à nouveau le cylindre doseur.
- Le poste 3 est le poste de bouchage. L'extension du vérin G présente un nouveau bouchon sous le dispositif de vissage composé du vérin E et du moteur F. Le vérin E est alors sorti pour saisir le bouchon. Ensuite, le vérin E doit être rentré de même que le vérin G pour retirer le dispositif présentant le bouchon. Enfin, le vérin E est mis en extension en même temps que le moteur pneumatique F tourne, pour permettre le vissage du bouchon sur la bouteille.

Pour analyser cette machine au niveau fonctionnel, il faut établir le GRAFCET de niveau 1. Le GRAFCET de niveau 1 est le suivant :



Ce GRAFCET sert à identifier les fonctions nécessaires au bon fonctionnement de l'automatisme. De ce GRAFCET, certains choix technologiques sont faits pour réaliser les fonctions. Dans le présent exemple, la technologie choisie pour les actionneurs est le pneumatique.

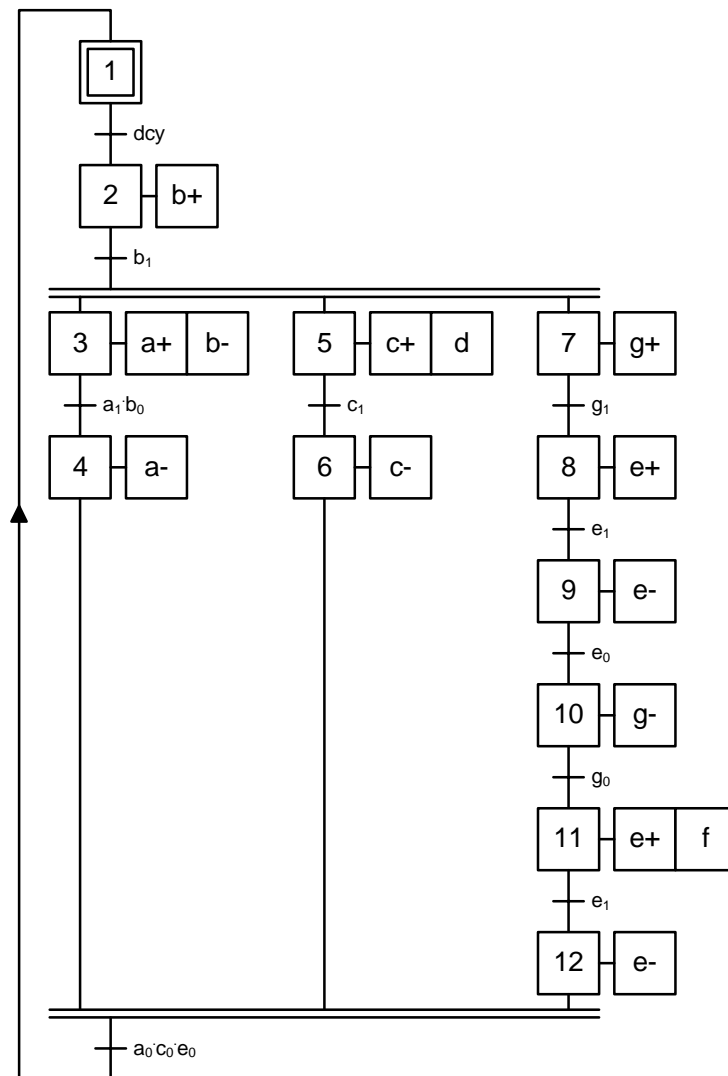
Voici la liste des capteurs :

- a0 : vérin A complètement rentré ;
- a1 : vérin A complètement sorti (bouteille sur le convoyeur) ;
- b0 : vérin B complètement rentré ;
- b1 : vérin B complètement sorti (convoyeur avancé d'un pas) ;
- c0 : vérin C complètement rentré (cylindre doseur rempli) ;
- c1 : vérin C complètement sorti (cylindre doseur vidé dans la bouteille) ;
- e0 : vérin E complètement rentré (moteur de vissage en haut) ;
- e1 : vérin E complètement sorti (moteur de vissage en bas ou en butée) ;
- g0 : vérin G complètement rentré ;
- g1 : vérin G complètement sorti (bouchon présenté sous le dispositif de vissage) ;
- dcy : départ de cycle.

Voici la liste des actionneurs :

- a- : rentrer le vérin A ;
- a+ : sortir le vérin A (pousser la bouteille sur le convoyeur) ;
- b- : rentrer le vérin B ;
- b+ : sortir le vérin B (avancer le convoyeur) ;
- c- : rentrer le vérin C (remplir le cylindre doseur) ;
- c+ : sortir le vérin C (remplir la bouteille) ;
- d : ouvrir la vanne de remplissage (simple action) ;
- e- : rentrer le vérin E (monter le dispositif de vissage) ;
- e+ : sortir le vérin E (descendre le dispositif de vissage) ;
- f : faire tourner le moteur de vissage ;
- g- : rentrer le vérin G ;
- g+ : sortir le vérin G (présenter le nouveau bouchon).

Une fois ces choix technologiques faits, il faut alors établir le GRAFCET de niveau 2. Une première version de GRAFCET de niveau 2 serait :



Les équations correspondant à ce GRAFCET seraient :

- Transitions :

$$Y_1 = X_1 \text{ dcy}$$

$$Y_2 = X_2 \text{ b}_1$$

$$Y_3 = X_3 \text{ a}_1 \text{ b}_0$$

$$Y_{4-6-12} = X_4 \text{ X}_6 \text{ X}_{12} \text{ a}_0 \text{ c}_0 \text{ e}_0$$

$$Y_5 = X_5 \text{ c}_1$$

$$Y_7 = X_7 \text{ g}_1$$

$$Y_8 = X_8 \text{ e}_1$$

$$Y_9 = X_9 \text{ e}_0$$

$$Y_{10} = X_{10} \text{ g}_0$$

$$Y_{11} = X_{11} \text{ e}_1$$

- Étapes :

$$X_1 = (Y_{4-6-12} + X_1 \overline{Y_1}) + \text{Init}$$

$$X_2 = Y_1 + X_2 \overline{Y_2}$$

$$X_3 = Y_2 + X_3 \overline{Y_3}$$

$$X_4 = Y_3 + X_4 \overline{Y_{4-6-12}}$$

$$X_5 = Y_2 + X_5 \overline{Y_5}$$

$$X_6 = Y_5 + X_6 \overline{Y_{4-6-12}}$$

$$X_7 = Y_2 + X_7 \overline{Y_7}$$

$$X_8 = Y_7 + X_8 \overline{Y_8}$$

$$X_9 = Y_8 + X_9 \overline{Y_9}$$

$$X_{10} = Y_9 + X_{10} \overline{Y_{10}}$$

$$X_{11} = Y_{10} + X_{11} \overline{Y_{11}}$$

$$X_{12} = Y_{11} + X_{12} \overline{Y_{4-6-12}}$$

- Les actions :

$$\text{a} += X_3, \quad \text{a} -= X_4$$

$$\text{b} += X_2, \quad \text{b} -= X_3$$

$$\text{c} += X_5, \quad \text{c} -= X_6$$

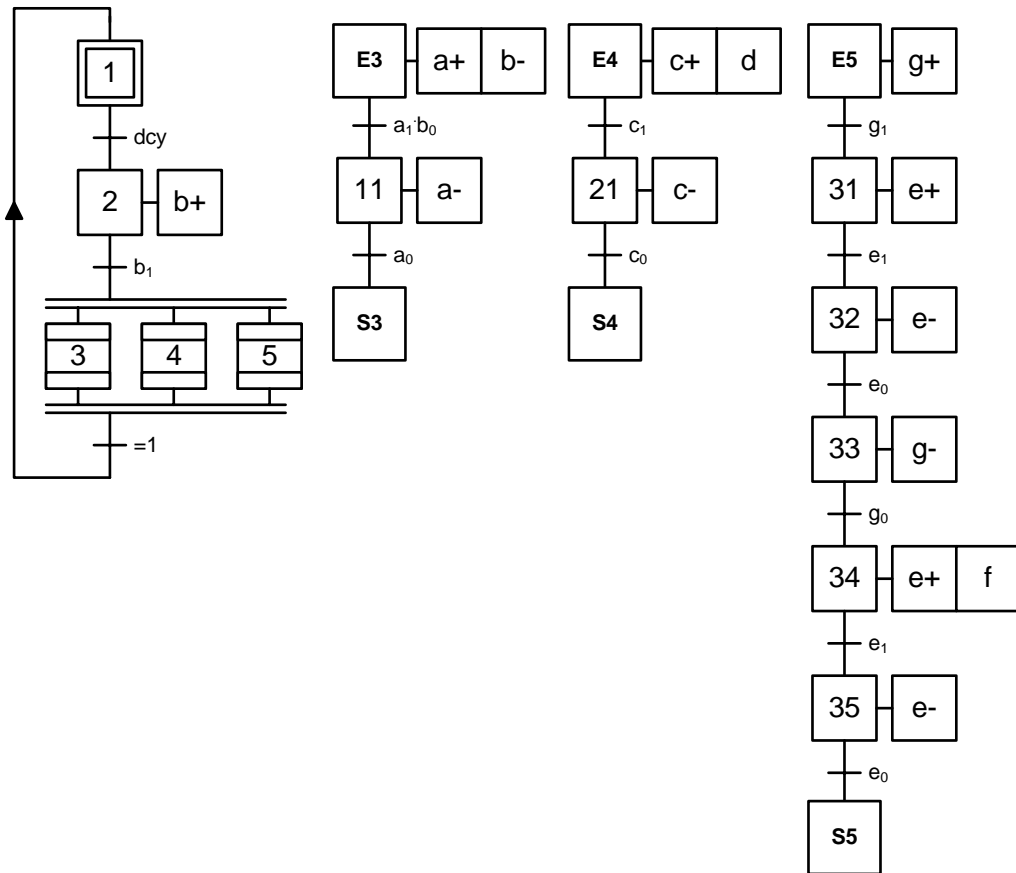
$$\text{d} += X_5$$

$$\text{e} += X_8 + X_{11}, \quad \text{e} -= X_9 + X_{12}$$

$$\text{f} += X_{11}$$

$$\text{g} += X_7, \quad \text{g} -= X_{10}$$

Une seconde variante de GRAFCET de niveau 2 utilisant les macro-étapes serait :



Les équations correspondant à ce GRAFCET seraient :

- Transitions :

$$Y_{3-4-5} = X_3 X_4 X_5 X_{s3} X_{s4} X_{s5}$$

$$Y_1 = X_1 dcy$$

$$Y_2 = X_2 b_1$$

$$Y_{E3} = X_{E3} a_1 b_0$$

$$Y_{11} = X_{11} a_0$$

$$Y_{E4} = X_{E4} c_1$$

$$Y_{21} = X_{21} c_0$$

$$Y_{E5} = X_{E5} g_1$$

$$Y_{31} = X_{31} e_1$$

$$Y_{32} = X_{32} e_0$$

$$Y_{33} = X_{33} g_0$$

$$Y_{34} = X_{34} e_1$$

$$Y_{35} = X_{35} e_0$$

- Étapes :

$$X_1 = (Y_{3-4-5} + X_1 \overline{Y_1}) + \text{Init}$$

$$X_2 = Y_1 + X_2 \overline{Y_2}$$

$$X_3 = Y_2 + X_3 \overline{Y_{3-4-5}}$$

$$X_4 = Y_2 + X_4 \overline{Y_{3-4-5}}$$

$$X_5 = Y_2 + X_5 \overline{Y_{3-4-5}}$$

$$X_{E3} = \uparrow X_3 + X_{E3} \overline{Y_{E3}}$$

$$X_{11} = Y_{E3} + X_{11} \overline{Y_{11}}$$

$$X_{S3} = Y_{11} + X_{S3} \overline{Y_{3-4-5}}$$

$$X_{E4} = \uparrow X_4 + X_{E4} \overline{Y_{E4}}$$

$$X_{21} = Y_{E4} + X_{21} \overline{Y_{21}}$$

$$X_{S4} = Y_{21} + X_{S4} \overline{Y_{3-4-5}}$$

$$X_{E5} = \uparrow X_5 + X_{E5} \overline{Y_{E5}}$$

$$X_{31} = Y_{E5} + X_{31} \overline{Y_{31}}$$

$$X_{32} = Y_{31} + X_{32} \overline{Y_{32}}$$

$$X_{33} = Y_{32} + X_{33} \overline{Y_{33}}$$

$$X_{34} = Y_{33} + X_{34} \overline{Y_{34}}$$

$$X_{35} = Y_{34} + X_{35} \overline{Y_{35}}$$

$$X_{S5} = Y_{35} + X_{S5} \overline{Y_{3-4-5}}$$

- Actions :

$$a+ = X_{E3}, \quad a- = X_{11}$$

$$b+ = X_2, \quad b- = X_{E3}$$

$$c+ = X_{E4}, \quad c- = X_{21}$$

$$d+ = X_{E4}$$

$$e+ = X_{31} + X_{34}, \quad e- = X_{32} + X_{35}$$

$$f+ = X_{34}$$

$$g+ = X_{E5}, \quad g- = X_{33}$$

Vous voyez donc que le GRAFCET principal ressemble plus au GRAFCET de niveau 1, puisque chaque macro-étape correspond à une fonction identifiée au GRAFCET de niveau 1.