

1. Introduction

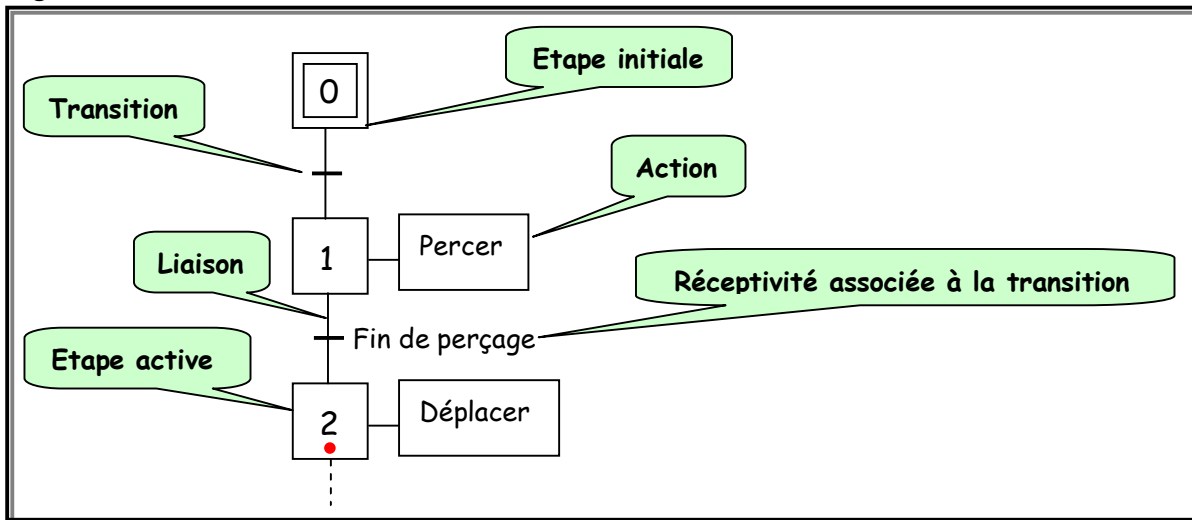
Le **GRAF CET** (**GRA**phe **FO**ncionnel de **CO**mmande des **ET**apes et **TR**ansitions) est un outil de représentation graphique de tout système automatisé.

C'est un langage clair qui permet de décrire sans ambiguïté le fonctionnement séquentiel du système. Il accompagne le système pendant toute sa durée de vie, de l'élaboration du cahier des charges à sa maintenance en passant par sa conception, sa mise au point et son exploitation.

2. Eléments graphiques de base

Considérons le Grafcet de la figure 1.

Figure 1



21. Etape

Une étape représente une situation ayant un comportement stable : pendant une étape les organes de commande et les capteurs ne changent pas d'état.

L'étape se représente par un carré qui porte un numéro d'ordre. Elle est soit active soit inactive.

On peut associer à chaque étape i une variable X_i image de son activité.

Exemple : Etape 2 active $\Rightarrow X_2 = 1$, Etape 2 inactive $\Rightarrow X_2 = 0$.

Remarque : L'étape initiale s'active au début du fonctionnement et se représente par un double carré.

22. Transition

Une transition indique une possibilité d'évolution d'activité entre deux ou plusieurs étapes. Cette évolution s'accomplit par le franchissement de la transition.

23. Liaisons orientées

Les liaisons orientées relient les étapes aux transitions et les transitions aux étapes. Le sens général d'évolution est du haut vers le bas. Dans le cas contraire, des flèches doivent être employées.

24. Réceptivité

Une réceptivité associée à une transition se traduit par une ou plusieurs conditions logiques.

25. Action

Une action indique, dans un rectangle, comment agir sur la variable de sortie. Elle n'est effective que lorsque l'étape associée est active.

3. Structures de base

31. Séquence linéaire

Une séquence linéaire ou unique est composée d'un ensemble d'étapes successives où chaque étape est suivie d'une seule transition et chaque transition n'est validée que par une seule étape.

32. Séquences simultanées : Figure 2a

321. Divergence en ET

Le système se dirige vers l'étape 2 ET l'étape 12 si a est franchie. Quand a est franchie, les étapes 2 ET 12 sont actives $\Rightarrow X2=X12=X1.a$

322. Convergence en ET

La transition d sera validée quand les étapes 3 ET 13 seront actives. Si la réceptivité associée à cette transition est vraie alors celle-ci est franchie $\Rightarrow X14=X3.X13.d$

33. Sélection de séquences : Figure 2b

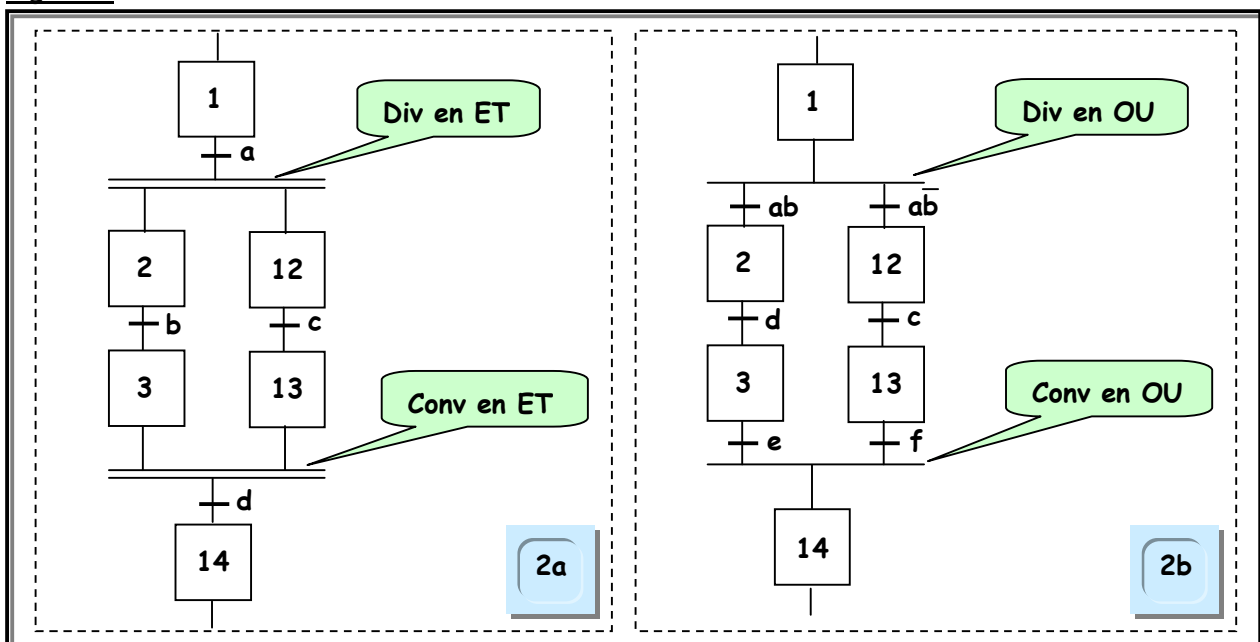
331. Divergence en OU

Le système se dirige vers l'étape 2 si (a.b) est vraie OU vers l'étape 12 si (a. \bar{b}) est vraie.
 $\Rightarrow X2=X1.(a.b)$ OU $X12=X1.(a.\bar{b})$

332. Convergence en OU

Le système converge vers l'étape 14 si e est vraie OU si f est vraie $\Rightarrow X14=(X3.e)+(X13.f)$

Figure 2



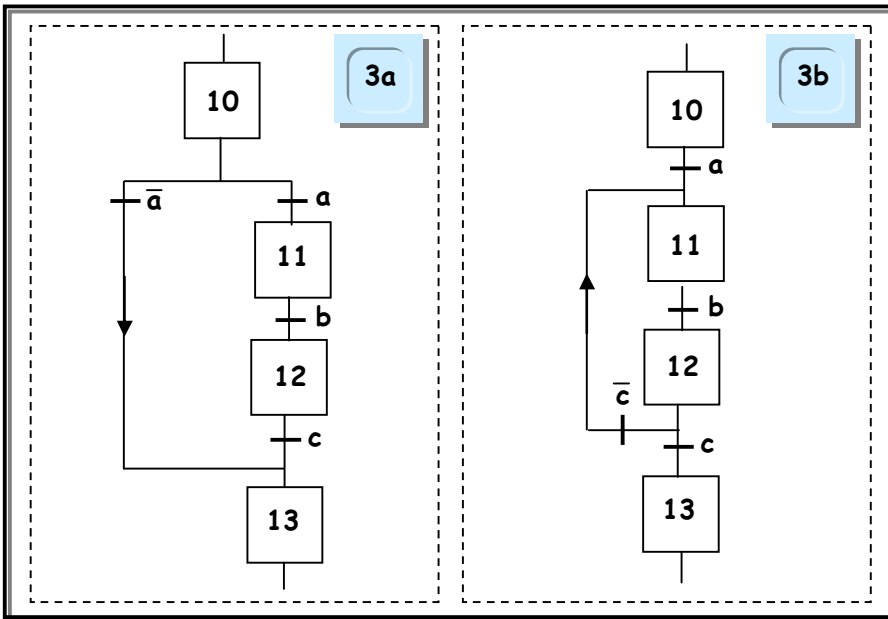
34. Saut d'étapes et reprise de séquence

341. Saut d'étapes : Figure 3a

Le saut d'étapes permet de sauter une ou plusieurs étapes lorsque les actions associées à ces étapes deviennent inutiles.

342. Reprise de séquence : Figure 3b

La reprise de séquence permet de recommencer plusieurs fois la même séquence tant qu'une condition n'est pas obtenue.

Figure 3**4. Règles d'évolution****41. Règle 1 : Situation initiale**

L'initialisation précise l'étape ou les étapes actives au début du fonctionnement. Les étapes initiales sont activées inconditionnellement en début de cycle.

42. Règles 2 : Franchissement d'une transition

Une transition est soit validée soit non validée, elle est validée lorsque toutes les étapes immédiatement précédentes sont actives, elle ne peut être franchie que :

- Si elle est validée.
- Et si la réceptivité associée à la transition est vraie.

43. Règles 3 : Evolution des étapes actives

Le franchissement d'une transition provoque simultanément :

- L'activation de toutes les étapes immédiatement suivantes.
- La désactivation de toutes les étapes immédiatement précédentes.

44. Règle 4 : Evolutions simultanées

Plusieurs transitions simultanément franchissables sont simultanément franchies.

45. Règle 5 : Activation et désactivation simultanée d'une même étape

Si au cours d'une évolution, une même étape se trouve être à la fois activée et désactivée, elle reste active.

5. Règle de syntaxe

L'alternance étape/transition et transition/étape doit être respectée quelle que soit la séquence parcourue.

6. Réceptivités particulières

61. Réceptivité dépendante du temps : Figure 4a

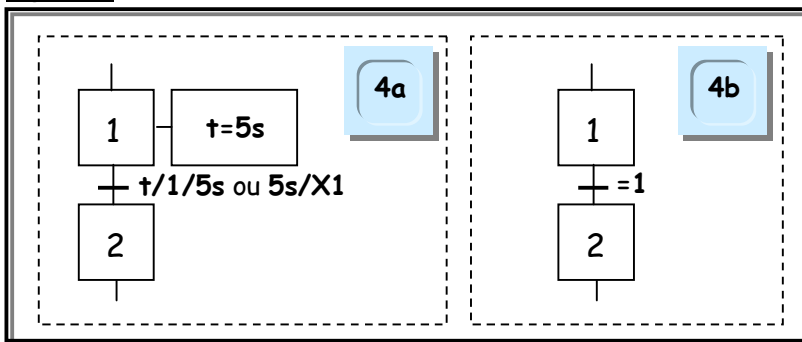
Pour faire intervenir le temps dans une réceptivité, il suffit d'indiquer après le repère "t" son origine et sa durée. L'origine sera l'instant du début de l'activation de l'étape déclenchant la temporisation.

La notation $t/1/5s$ signifie que la réceptivité sera vraie 5 secondes après l'activation de l'étape repérée 1. La notation normalisée s'écrit $5s/X1$.

62. Réceptivité toujours vraie : Figure 4b

Une telle réceptivité s'écrit "=1". Le franchissement de cette transition se fera dès que les étapes immédiatement antérieures seront actives sans autre condition. Le passage de l'étape 1 à l'étape 2 est toujours vrai.

Figure 4



7. Actions associées

71. Action continue : Figure 5a

L'action est exécutée pendant le temps durant lequel l'étape reste active \Rightarrow Action $A=X1$.

72. Action conditionnelle : Figure 5b

Une proposition logique, appelée condition d'assignation, qui peut être vraie ou fausse, conditionne l'action continue \Rightarrow Action $A=X1.c$

73. Action temporisée ou retardée : Figure 5c

L'action retardée est une action continue dont la condition d'assignation n'est vraie qu'après une durée t spécifiée depuis l'activation de l'étape associée. L'action A sera exécutée 5s après l'activation de l'étape 1.

74. Action limitée dans le temps : Figure 5d

L'action limitée dans le temps est une action continue dont la condition d'assignation n'est vraie que pendant une durée t spécifiée depuis l'activation de l'étape à laquelle elle est associée.

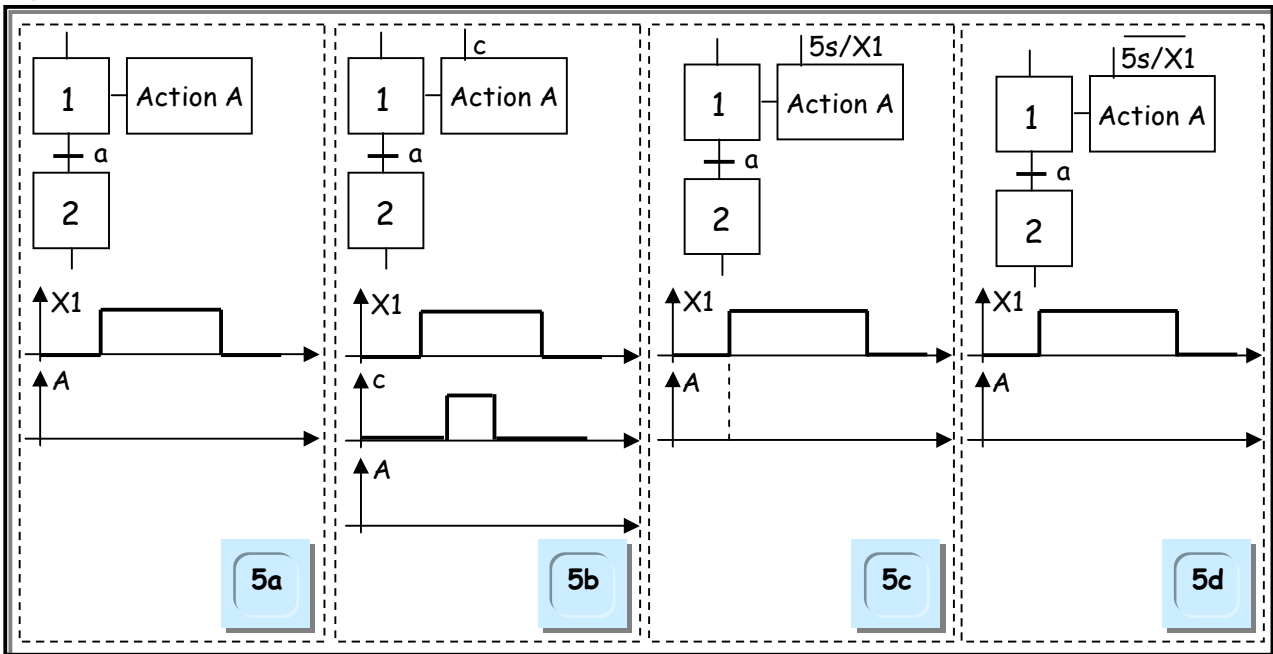
L'action A sera exécutée 5s depuis l'activation de l'étape 1.

75. Action maintenue ou mémorisée

Pour qu'une action reste maintenue lorsque l'étape qui l'a commandée vient d'être désactivée, il faut utiliser une action mémorisée. L'effet maintenu est réalisé soit :

- Par répétition de l'action ou ordre à toutes les étapes.
- Par mise en mémoire à l'étape de début (S pour Set), et mise hors mémoire à la dernière étape (R pour Reset).

Figure 5



8. Macro-étape

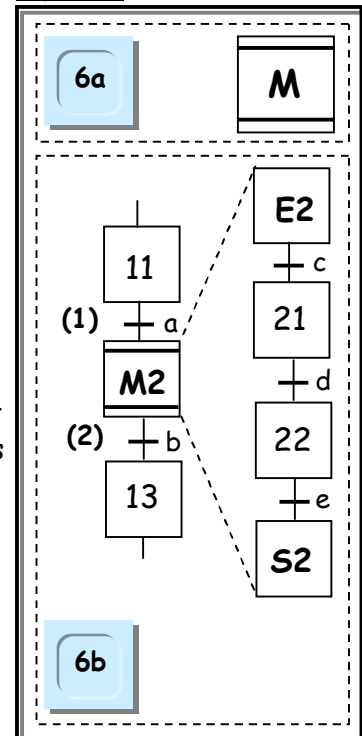
Une macro-étape **M** est la représentation unique d'un ensemble d'étapes et de transitions nommé "Expansion d'étapes" :

- ☑ La macro-étape se substitue à une étape du Grafcet.
- ☑ L'expansion de la macro-étape comporte une étape d'entrée repérée **E** et une étape de sortie repérée **S**.
- ☑ Tout franchissement de la transition amont de la macro-étape active l'étape **E** d'entrée de son Expansion.
- ☑ L'étape de sortie participe à la validation des transitions en aval de la macro-étape.
- ☑ La transition suivant la macro-étape n'est validée que lorsque la dernière étape de la macro-étape est active.
- ☑ L'utilisation de la macro-étape dans un Grafcet permet non seulement de simplifier la représentation mais aussi d'éviter les séquences linéaires trop longues.
- ☑ Une macro-étape peut elle-même comporter d'autres macro-étapes.

La figure 6a donne le symbole de la macro-étape alors que la figure 6b montre un exemple d'une macro-étape représentée avec son expansion :

- ☑ Le franchissement de la transition (1) active l'étape **E2**.
- ☑ La transition (2) ne sera validée que lorsque l'étape **S2** sera active.
- ☑ Le franchissement de la transition (2) désactive l'étape **S2**.

Figure 6



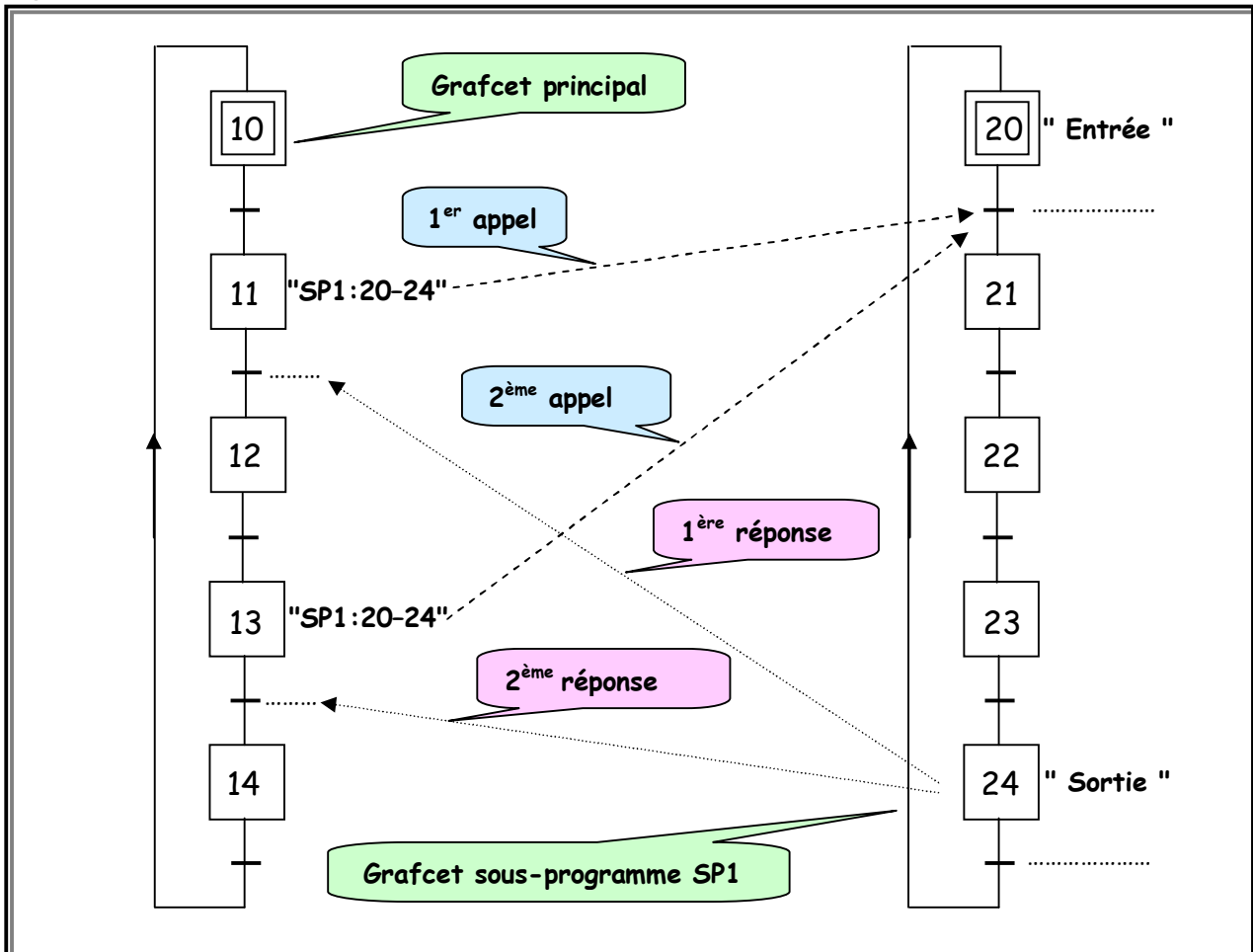
9. Tâche

Une **tâche** est un **sous-programme** qui peut être appelé à plusieurs reprises dans un même **Grafcet** dit **Grafcet principal**.

Le sous-programme est un **Grafcet** indépendant dont l'exécution et le déroulement sont synchronisés au **Grafcet principal**. Lorsque la tâche est terminée, une information d'état (fin de tâche) est retournée au **Grafcet principal** qui peut alors évoluer. La tâche est appelée en indiquant entre les guillemets le nom du sous-programme appelé.

La figure 7 donne un exemple de tâche.

Figure 7



10. Grafcet synchronisé

Le **Grafcet synchronisé** est généralement composé de deux Grafcets :

Le premier, le **Grafcet de coordination**, gère la synchronisation des différentes fonctions du système automatisé. Chaque fonction est décrite par un Grafcet, sous-programme du premier. Il faut donc une synchronisation entre le 1^{er} Grafcet et les seconds.

C'est une étape du 1^{er} Grafcet qui permettra de lancer un second Grafcet dédié à une fonction. Une étape de fin signalera l'exécution complète de ce dernier et permettra de faire évoluer le 1^{er} Grafcet vers une autre étape.

La structure d'un Grafcet synchronisé est basée sur la notion de tâche et apporte un échange d'informations entre le Grafcet appelant ou **Grafcet maître** et le Grafcet appelé ou **Grafcet esclave**. Ce dernier est un Grafcet de tâche qui peut être sollicité par le Grafcet maître, appelé aussi **Grafcet de conduite**, en plusieurs reprises.

La figure 8 donne un exemple de synchronisation de Grafcets.

Figure 8

