

Intégrer un automate M340 ou TSX Premium dans un système Schneider Electric (Unity Pro)

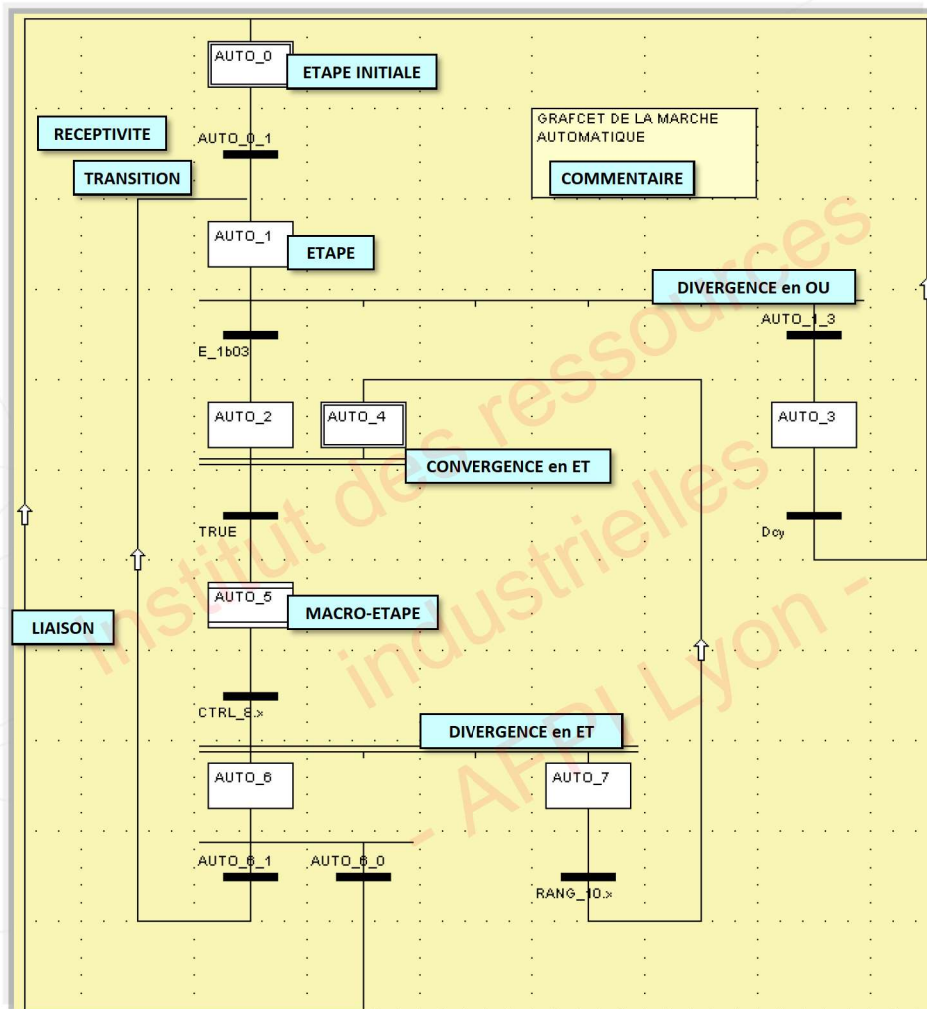
- 11 - Unity Pro: le langage de programmation grafcet « SFC »

1. INTRODUCTION	3
<input type="checkbox"/> REPRESENTATION D'UNE SECTION « SFC »	
2. CREATION D'UNE SECTION « SFC »	4
<input type="checkbox"/> CREATION	
<input type="checkbox"/> PARAMETRAGE DES OPTIONS	
3. COMPOSITION D'UNE SECTION « SFC »	5
<input type="checkbox"/> LE DIAGRAMME FONCTIONNEL	
4. LES ETAPES	6
<input type="checkbox"/> TYPES D'ETAPES	
<input type="checkbox"/> PROPRIETES	
<input type="checkbox"/> LES ACTIONS ASSOCIEES AUX ETAPES	
5. LES TRANSITIONS	12
<input type="checkbox"/> PRESENTATION	
<input type="checkbox"/> CONDITIONS DE TRANSITION	
<input type="checkbox"/> PROPRIETES	
6. LES MACRO-ETAPES	14
<input type="checkbox"/> PRESENTATION	
<input type="checkbox"/> MACRO-SECTION	
<input type="checkbox"/> ETAPE D'ENTREE – ETAPE DE SORTIE	
<input type="checkbox"/> ADRESSAGE	

Unity Pro: le langage de programmation grafcet « SFC »

1. INTRODUCTION

□ REPRESENTATION D'UNE SECTION « SFC »

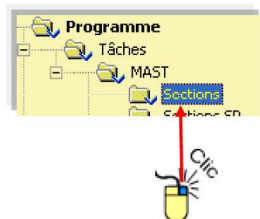


- ✓ 100 sections de 1024 étapes.
- ✓ Programmé uniquement dans la tâche maître (MAST).
- ✓ Les étapes, les transitions et les sauts occupent une cellule.
- ✓ 100 étapes activées simultanément par section.
- ✓ 20 actions par étape.

Unity Pro: le langage de programmation grafcet « SFC »

2. CREATION D'UNE SECTION « SFC »

❑ CREATION



Nouvelle section

Nouveau

Général Localisation Condition Commentaire

Nom : STATION_POMPAGE

Langage : SFC

Spécifique à SFC

Numéro zone : 0

Protection : Aucun

☐ Contrôle opérateur

- ✓ Nommer la section (ex: STATION_POMPAGE)
- ✓ Sélectionner le langage SFC
- ✓ Modifier éventuellement le numéro de la section
- ✓ Valider la section:

OK

❑ PARAMETRAGE DES OPTIONS

Outils

Options du projet...

- ✓ Paramétrage du langage SFC:

Options du projet

Options du projet

Général

- Gestion des messages de génération
- Paramètres de génération
- Sauvegarde automatique lors du transfert
- Données intégrées de l'automate
- Diagnostics automate

Variables

Programme

- Langages
 - Commun
 - FBD
 - LD
 - Vue mixte
 - SFC**

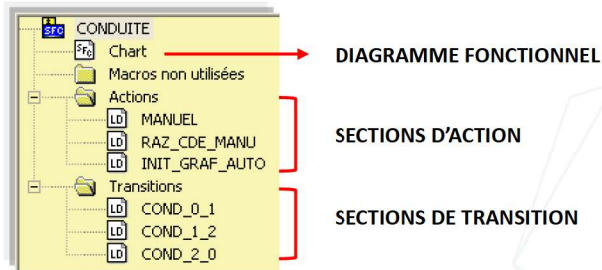
Propriété

Valeur

Autoriser les sections de macros ☒

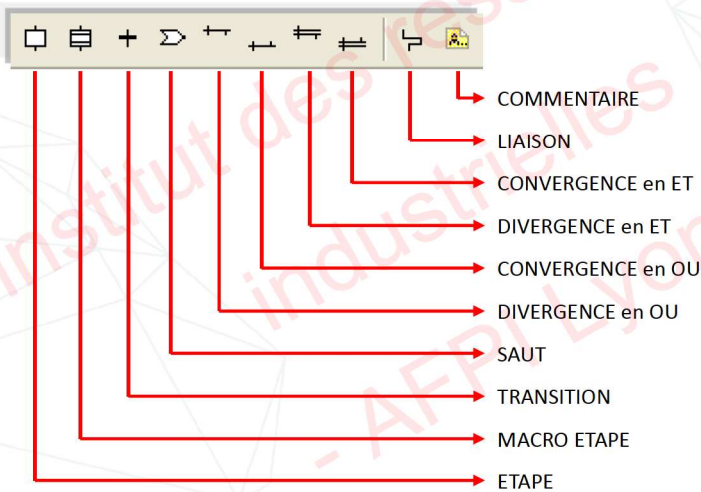
Autoriser plusieurs jetons ☒

3. COMPOSITION D'UNE SECTION « SFC »



❑ LE DIAGRAMME FONCTIONNEL

- ✓ Il contient les éléments de programmation graphique.
- ✓ Les éléments de base:



- ✓ Les pavés de séquence:



4. LES ETAPES

□ TYPES D'ETAPES

✓ ETAPE INITIALE:



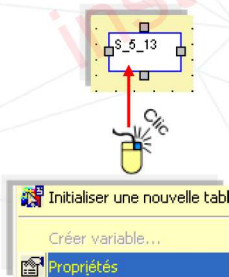
- Les étapes initiales sont activées lors de l'initialisation de la séquence ou du projet.
- Pour les jetons uniques, seule une étape initiale est admise par séquence.
- Pour les jetons multiples un nombre d'étapes initiales pouvant être défini (de 0 à 100) est possible.
- Généralement, aucune action n'est affectée aux étapes initiales.

✓ ETAPE « NORMALE »:



- Une étape est activée lorsque l'étape qui la précède est active et que la réceptivité est vérifiée (transition franchie).
- Chaque étape compte zéro ou plusieurs actions.

□ PROPRIETES



Propriétés Etape : S_5_13

Général | Actions | Commentaire

Nom d'étape : S_5_13 ☒ Etape initiale

Temps de contrôle et retard

☐ Variable 'SFCSTEP_TIMES' ☒ Valeur littérale

Maximum

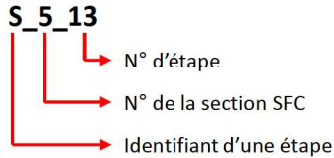
Minimum

Retard

4. LES ETAPES

❑ PROPRIETES

✓ NOM d'ETAPE:



- Il est conseillé de renommer les étapes (ex: AUTO_1).

✓ TEMPS de CONTRÔLE et RETARD:

Temps de contrôle et retard

☒ Variable 'SFCSTEP_TIMES' ☐ Valeur littérale

☒ Maximum ☐ Minimum ☐ Retard

- Durée maximum** pendant laquelle l'étape doit être **active**
- Durée minimum** pendant laquelle l'étape doit être **active**
- Forçage du temps d'activation** de l'étape

- Si les temps maximum et minimum ne sont pas respectés (temps enveloppe), l'étape change de couleur et un message est envoyé au viewer de diagnostic.
- Les temps d'étape peuvent être définis via une variable globale ('SFCSTEP_TIMES').

4. LES ETAPES

□ LES ACTIONS ASSOCIEES AUX ETAPES

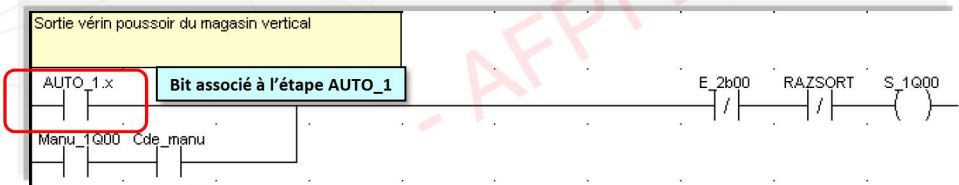
- ✓ **DEUX POSSIBILITÉS** pour programmer les actions:
 - Définition des **ACTIONS** dans une **SECTION INDEPENDANTE**,
 - Définition des **ACTIONS** dans les **ETAPES**.

- ✓ **DEFINITION des ACTIONS dans une SECTION INDEPENDANTE:**
 - Cette méthode permet de résoudre les problèmes liés à la non scrutation et de limiter le nombre d'affectation.
 - A chaque étape est affectée de façon implicite une variable de type de données SFCSTEP_STATE. Cette variable porte le nom de l'étape à laquelle elle est associée.
 - **Exemple:** étape « AUTO_1



AUTO_1.t	Temps d'activation actuel de l'étape
AUTO_1.x	Etat 1 si l'étape est active « état de l'étape »
AUTO_1.tminErr	Etat 1 si le temps d'activité inférieur au temps minimum
AUTO_1.tmaxEr	Etat 1 si le temps d'activité supérieur au temps maximum

- On utilise le bit « état de l'étape » pour commander les actions dans la section indépendante.
- **Exemple:**

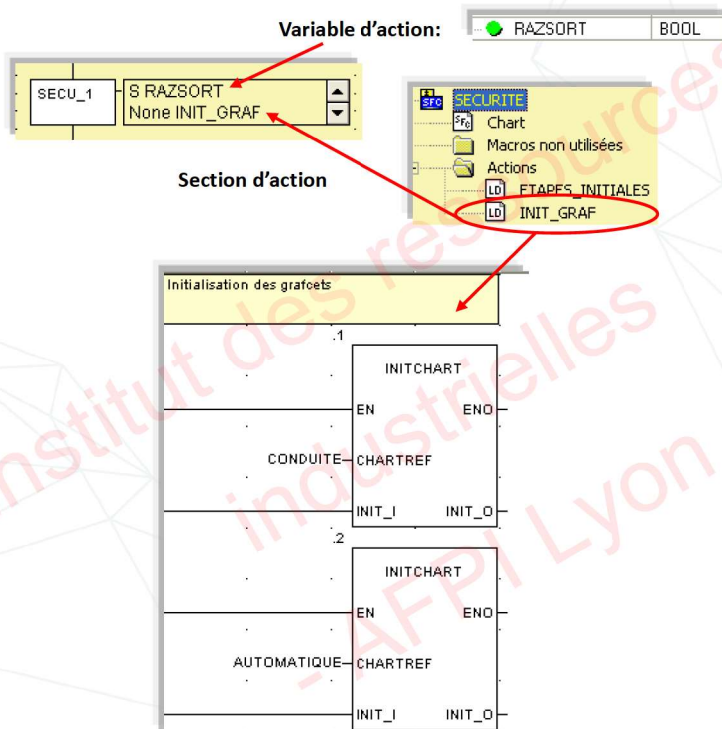


4. LES ETAPES

❑ LES ACTIONS ASSOCIEES AUX ETAPES

✓ DÉFINITION des ACTIONS dans les ETAPES:

- 20 ACTIONS maximum peuvent être affectées à chaque étape.
- Une ACTION peut être:
 - ❖ une VARIABLE d’ACTION (variable booléenne),
 - ❖ une SECTION d’ACTION



- Si plusieurs actions sont affectées à une même étape, elles sont traitées dans l'ordre de leur occurrence.

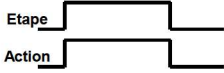
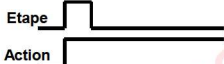





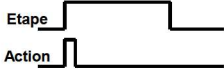
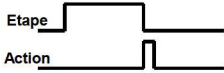
Unity Pro: le langage de programmation grafcet « SFC »

4. LES ETAPES

□ LES ACTIONS ASSOCIEES AUX ETAPES

✓ LES QUALIFICATIFS AFFECTES AUX ACTIONS:

- Chaque action associée à une étape doit avoir un identificateur définissant la commande de l'action.

QUALIFICATIF	SIGNIFICATION	DESCRIPTION	
N / NONE	Non mémorisé		Lorsque l'étape devient active, l'action passe à 1. Lorsque l'étape devient inactive, l'action passe à 0.
S	Activation mémorisée		Lorsque l'étape devient active, l'action passe à 1. Lorsque l'étape devient inactive, l'action reste à 1.
R	Remise à zéro prioritaire		Lorsque l'étape devient active, l'action passe à 0. Lorsque l'étape devient inactive, l'action reste à 0.
L	Limitée dans le temps		Lorsque l'étape devient active, l'action passe à 1 pendant la durée programmée. Lorsque l'étape devient inactive, l'action passe à 0.
D	Retardée		Lorsque l'étape devient active, l'action passe à 1 après l'écoulement du retard défini.
P	Impulsion		Lorsque l'étape devient active, l'action passe à 1 et le reste pendant un cycle programme.
DS	Retardée et mémorisée		Lorsque l'étape devient active, l'action passe à 1 avec un retard et reste à 1 lorsque l'étape est désactivée.
P1	Impulsion (front montant)		Lorsque l'étape devient active, l'action passe à 1 et le reste pendant un cycle de programme.
P0	Impulsion (front descendant)		Lorsque l'étape se désactive, l'action passe à 1 et le reste pendant un cycle de programme.

- **Remarque:** Indépendamment de leur position dans la zone de liste des actions, les premier et les actions avec l'identificateur P1 sont toujours exécutées en les actions avec l'identificateur P0 en dernier.

4. LES ETAPES

❑ LES ACTIONS ASSOCIEES AUX ETAPES

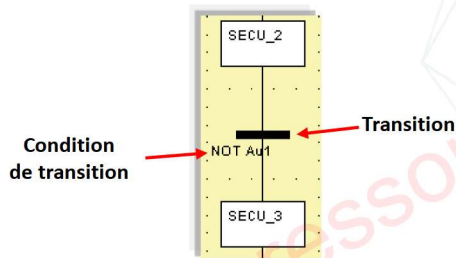
✓ LES QUALIFICATIFS AFFECTES AUX ACTIONS:

- 20 ACTIONS maximum peuvent être affectées à chaque étape.
- La variable affectée à une action peut également être utilisée dans des actions d'autres étapes.
- Seules les variables/adresses booléennes ou les éléments booléens de variables peuvent être utilisés comme variable d'action.
- Les actions ont un nom unique. Le nom de l'action est soit le nom de la variable d'action, soit le nom de la section d'action.
- Il est possible d'utiliser la variable d'action en lecture et en écriture dans plusieurs sections du projet (affectation multiple).
- Une étape à laquelle aucune action n'est affectée a une fonction d'attente; elle attend que la transition affectée soit vérifiée.
- Si plusieurs actions sont affectées à une même étape, elles sont traitées dans l'ordre de leur occurrence dans la zone de liste action.
- **Exception:** Indépendamment de leur position dans la zone de liste action, les actions comportant l'identificateur P1 sont toujours traitées en premier et les actions comportant l'identificateur P0 sont toujours traitées en dernier.
- **Les actions auxquelles un identificateur temporisé a été affecté ne peuvent être actives qu'une seule fois.**

5. LES TRANSITIONS

□ PRESENTATION

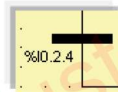
- ✓ Une transition indique les conditions d'évolution du grafcet. Elle est associée à une condition de transition (réceptivité).
- ✓ Elle est validée par lorsque les étapes qui la précèdent directement sont actives.
- ✓ Le franchissement d'une transition s'effectue lorsque la transition est validée et que la condition de transition est vérifiée.



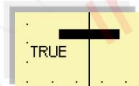
□ CONDITIONS DE TRANSITION

- ✓ A chaque transition est affectée une condition de transition qui peut être:

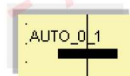
- Une **adresse** ou une **variable**:



- Un **libellé**:



- Une **section de transition**:

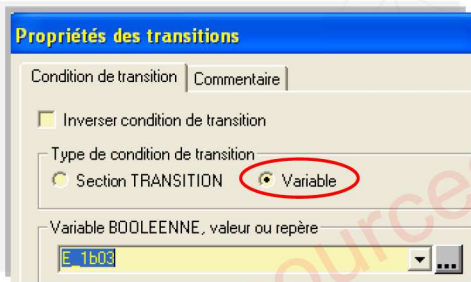


- ✓ L'adresse, la variable ou le libellé sont représentés en dessous du trait et la section de transition au-dessus du trait.

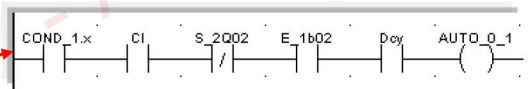
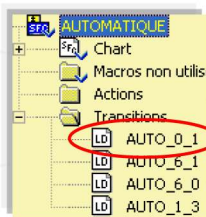
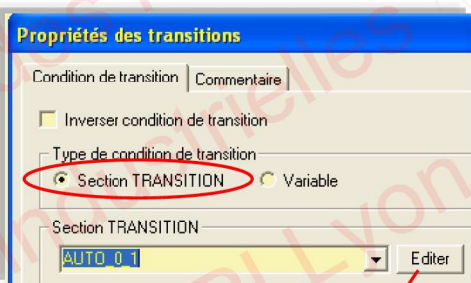
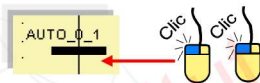
5. LES TRANSITIONS

❑ PROPRIETES

✓ Variable affectée à la transition:

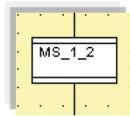


✓ Section de transition affectée à la transition:

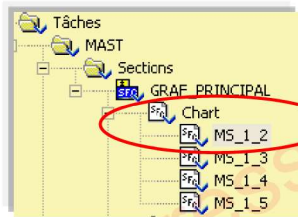


6. LES MACRO-ETAPES

❑ PRESENTATION



- ✓ Les macro-étapes servent à appeler une macro-section et ainsi à établir une structure hiérarchique des commandes d'enchaînement.
- ✓ elles se positionnent en sections de « commande de séquence » :



- ✓ Le nombre de macro-étapes n'est pas limité.
- ✓ **8 niveaux d'imbrication.**
- ✓ Chaque macro-étape est affectée de manière implicite une variable du type de données SFCSTEP_STATE:

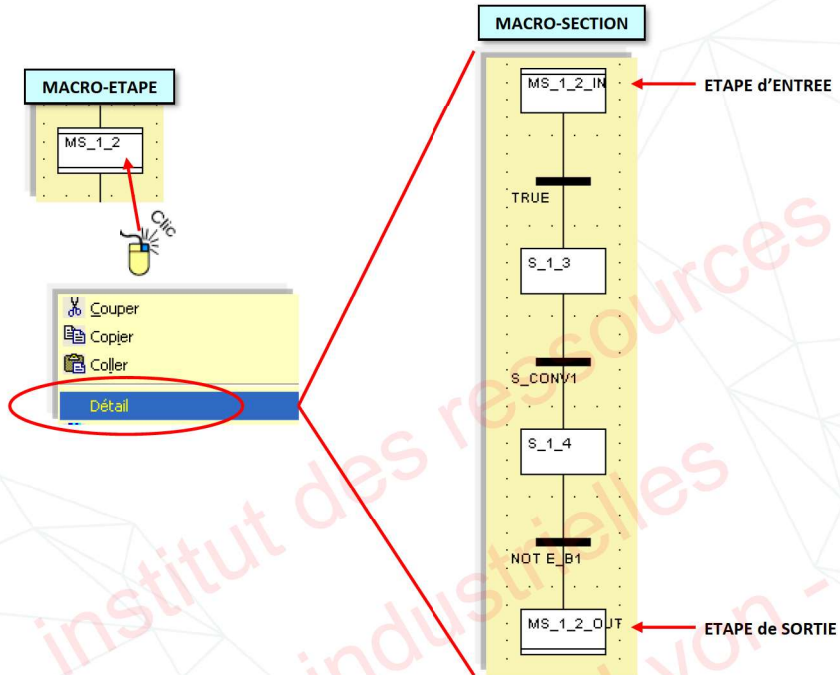
	MS_1_2	SFCSTEP_STATE
t		TIME
x		BOOL
tminErr		BOOL
tmaxErr		BOOL

- ✓ A chaque macro-étape peut être affectée un temps de contrôle et un temps de retard:

- ✓ **AUCUNE ACTION** ne peut être affectée aux macro-étapes.

6. LES MACRO-ETAPES

□ MACRO-SECTION



- ✓ Une **macro-section** se compose d'une **séquence unique**, laquelle dispose des mêmes éléments que les sections de "commande de séquence" (étapes, étape(s) initiale(s), macro-étapes, transitions, divergences, convergences....).
- ✓ Chaque macro-section comporte en début une **étape d'entrée** et en fin une **étape de sortie**.
- ✓ Chaque macro-étape peut être remplacée par la séquence contenue dans la macro-section correspondante.

6. LES MACRO-ETAPES

□ ETAPE D'ENTREE – ETAPE DE SORTIE



- ✓ Les étapes d'entrée et de sortie sont automatiquement placées dans des sections de macro-étape par l'éditeur SFC,.
- ✓ Une seule étape d'entrée et une seule étape de sortie est placée par macro-section.
- ✓ Les étapes d'entrée et de sortie ne peuvent pas être supprimées, copiées ou insérées manuellement.
- ✓ A chaque étape d'entrée et de sortie est affectée de manière implicite une variable du type de données SFCSTEP_STATE:

MS_1_5_IN	SFCSTEP_STATE
t	TIME
x	BOOL
tminErr	BOOL
tmaxErr	BOOL
MS_1_5_OUT	SFCSTEP_STATE
t	TIME
x	BOOL
tminErr	BOOL
tmaxErr	BOOL

- ✓ A chaque étape d'entrée peuvent être affectés les temps de contrôle et le temps de retard:

Temps de contrôle et retard

☐ Variable 'SFCSTEP_TIMES' ☒ Valeur littérale

☐ Maximum

☐ Minimum

Retard

- ✓ A chaque étape de sortie, ne peut être affecté qu'un temps de retard:

Temps de contrôle et retard

☐ Variable 'SFCSTEP_TIMES' ☒ Valeur littérale

☐ Maximum

☐ Minimum

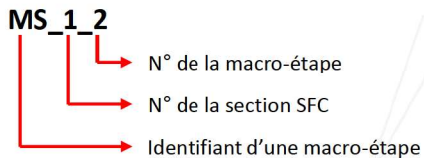
Retard

- ✓ Des actions peuvent être affectées à l'étape d'entrée mais pas à l'étape de sortie.

6. LES MACRO-ETAPES

□ ADRESSAGE

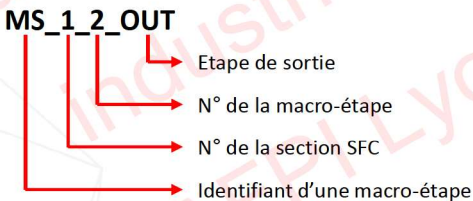
✓ MACRO-ETAPE:



✓ ETAPE d'ENTREE:



✓ ETAPE de SORTIE:



✓ ETAPE NORMALE:



✓ Il est possible de renommer les macro-étapes.