

MANIP 11 : Mode semi-automatique avec macro-étape et simulation

Introduction

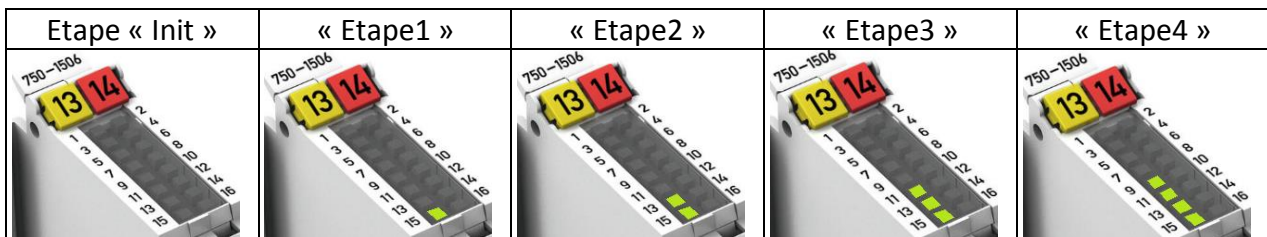
Lorsqu'on implémente un programme destiné à un API, il est possible d'effectuer certains tests sans connecter l'automate au PC ; on peut travailler en **mode simulation**. Le mode simulation ne remplace pas les tests réels mais apporte une aide utile.

Un autre outil intéressant est l'**écran d'exploitation** qui permet d'utiliser le PC pour interagir sur le processus. On peut définir une zone au sein de eCOCKPIT de manière à y placer des éléments (boutons, voyants, ...) auxquels on associera du code gérant les actions associées. L'écran d'exploitation reste cependant un outil d'aide à l'implémentation et aux tests et non à la mise en service !

Nous allons implémenter un mode **semi-automatique**. L'opérateur donne la commande pour démarrer un cycle puis celui-ci s'effectuera tout seul de manière autonome. Le cycle que nous utiliserons sera une séquence de bargraphe qui symbolise la réalisation d'une ou de plusieurs tâches. Nous utiliserons enfin une **macro-étape** pour contenir les différentes étapes du cycle.

A. Simulation d'un bargraphe

Nous allons reproduire la gestion du bargraphe (de la partie gauche uniquement) de la manip 6. Les sorties doivent évoluer comme ci-dessous :



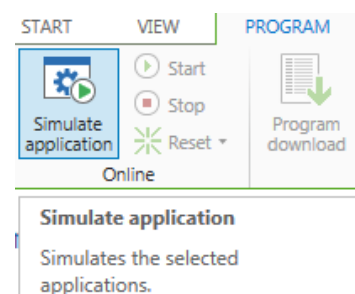
Le programme sera réalisé au sein d'un SFC comprenant 5 étapes.

Nommez les étapes telles qu'indiquées ci-dessous et arrangez vous pour que votre séquence active successivement ces étapes dans l'ordre avec une durée de 1 seconde chacune.

Par souci de clarté, nommez les sorties 15, 13, 11 et 9 par les identifiants respectifs led1, led2, led3 et led4.

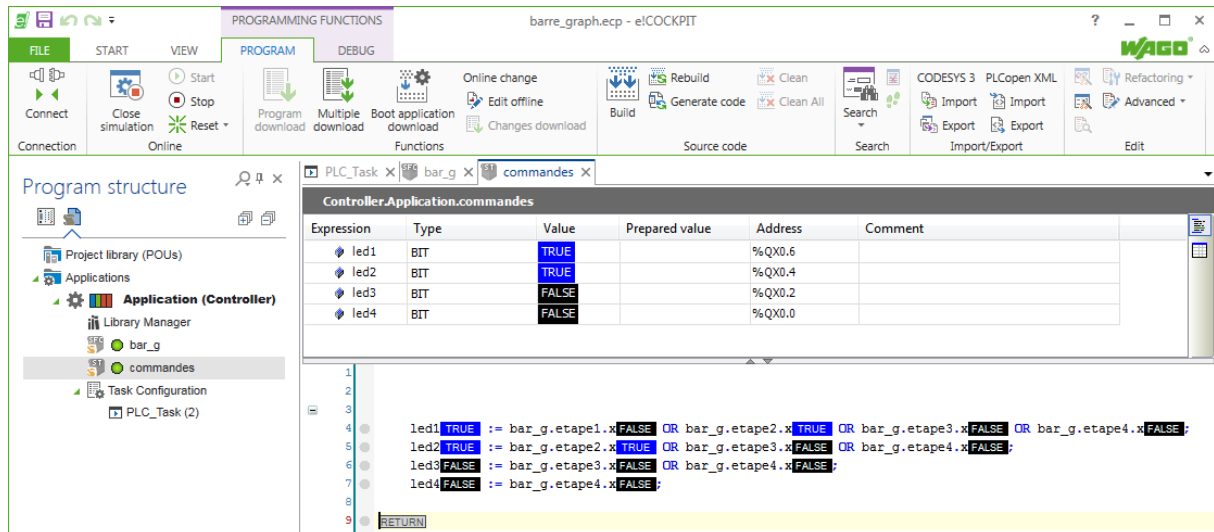
Nous allons maintenant tester le bon déroulement de la séquence sans connecter l'automate ... avec l'outil simulation !

L'outil est disponible en cliquant sur « Simulate application » au sein de l'onglet « PROGRAM » comme indiqué ci-contre. Vous verrez apparaître une fenêtre proposant de créer un élément 'Controller' ; vous acceptez pour avoir accès aux boutons.



MANIP 11 : Mode semi-automatique avec macro-étape et simulation

Une fois que vous cliquez sur « Start », vous pourrez voir s'exécuter la séquence en mode simulation comme représenté ci-dessous :

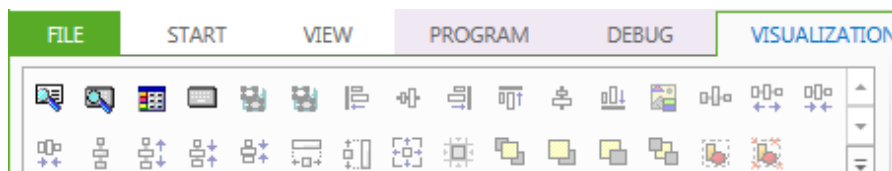


B. Écran d'exploitation – Visualisation des sorties

En faisant un click droit sur « Application (Controller) » comme ci-contre, on peut choisir « visualization ».



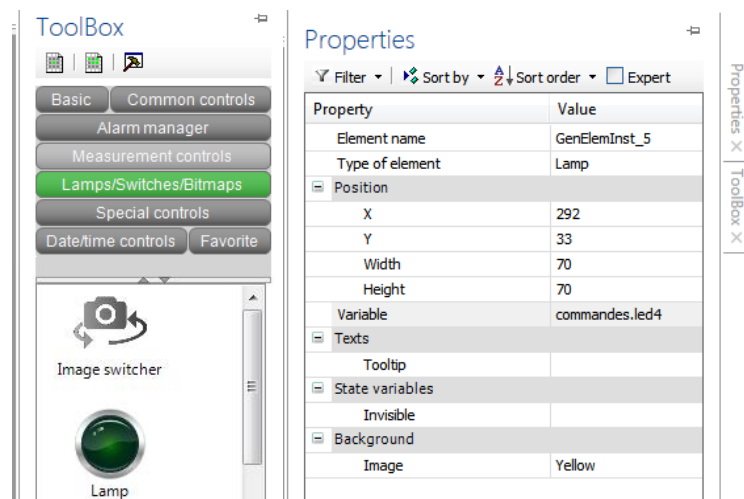
On peut renommer la zone de visualisation et voir apparaître des outils de l'onglet « VISUALIZATION » repris ci-dessous.



À droite de l'écran, on peut voir apparaître les onglets « Toolbox » et « Properties ».

On choisira « Lamp » de « Toolbox » pour les leds du bargraphe.

Chaque élément « Lamp » peut se voir assigner des propriétés propres et surtout une Variable.



MANIP 11 : Mode semi-automatique avec macro-étape et simulation

C. Écran d'exploitation – Bouton Start avec accroche

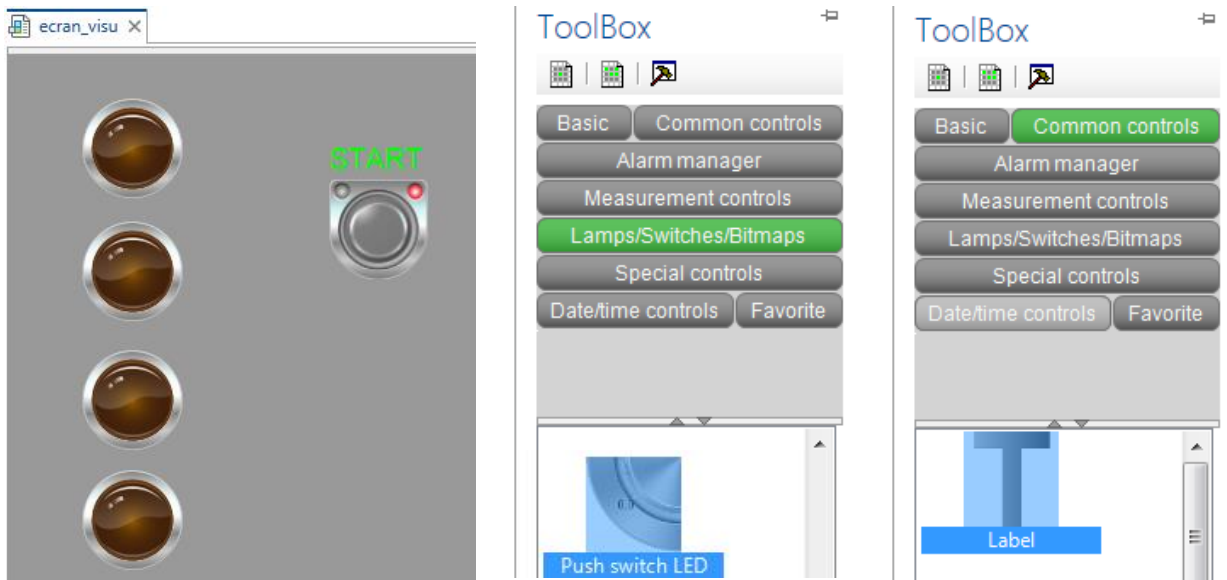
Après avoir testé le bon fonctionnement de la visualisation du bargraphe par simulation, on ajoutera un bouton « Start ».

On commencera par déclarer la variable qui contiendra l'information de l'état d'enfoncement du bouton. On la déclarera au sein de notre SFC de bargraphe pour pouvoir l'utiliser au sein des conditions-transition.

```

SFC bar_g x
1 PROGRAM bar_g
2 VAR
3     b_start: BOOL ;
4 END_VAR
    
```

On créera ensuite un écran d'exploitation comme ci-dessous par exemple. Le bouton utilisé est un « Push switch LED », un Label « START » décrit son rôle.



On pourra définir les propriétés du bouton et choisir un bouton avec accroche (Image toggler). Pour un bouton sans accroche, on choisira un « Image tapper ».

On associera également notre bouton à la variable préalablement déclarée au sein de notre SFC de bargraphe.

Properties

Filter | Sort by | Sort order | Expert

Property	Value
Element name	GenElemInst_11
Type of element	Push switch LED
Position	
X	222
Y	86
Width	70
Height	70
Variable	bar_g,b_start
Element behavior	Image toggler
Texts	
Image toggler	Image toggler
Image tapper	Image tapper
Tooltip	

MANIP 11 : Mode semi-automatique avec macro-étape et simulation

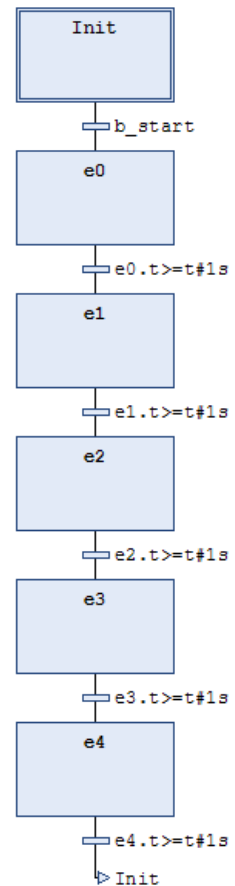
On adaptera finalement notre SFC de bargraphe en faisant apparaître une condition-transition tenant compte du bouton.

On ajoutera l'étape 0 pour s'assurer de voir toutes les leds éteintes pendant 1 seconde lors du déroulement de la séquence.

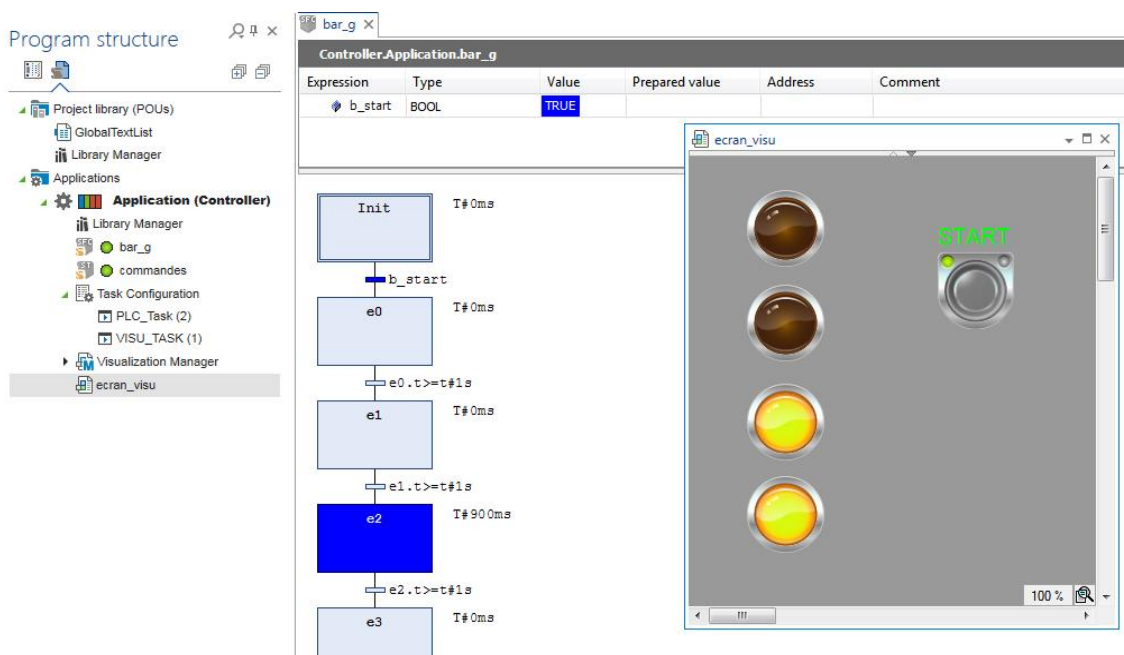
De plus, on a maintenant notre étape initiale « Init » qui représente davantage une "situation stable".

En testant le programme en mode simulation, on devrait voir apparaître un déroulement du bargraphe en boucle tant que le bouton sera maintenu enfoncé. Dès que le bouton se trouve en mode relâche, le cycle en cours se termine avant de revenir à notre étape initiale « Init ».

On a ici une sorte de mode semi-automatique, cependant on préférera s'assurer que l'opérateur commande le redémarrage d'un cycle à l'aide d'un front montant sur l'état du bouton (voir plus tard).



Pour une meilleure visualisation pendant la simulation, on peut sélectionner « Undock » de l'écran d'exploitation et avoir un aperçu tel que ci-dessous :



MANIP 11 : Mode semi-automatique avec macro-étape et simulation

D. Utilisation d'une macro-étape

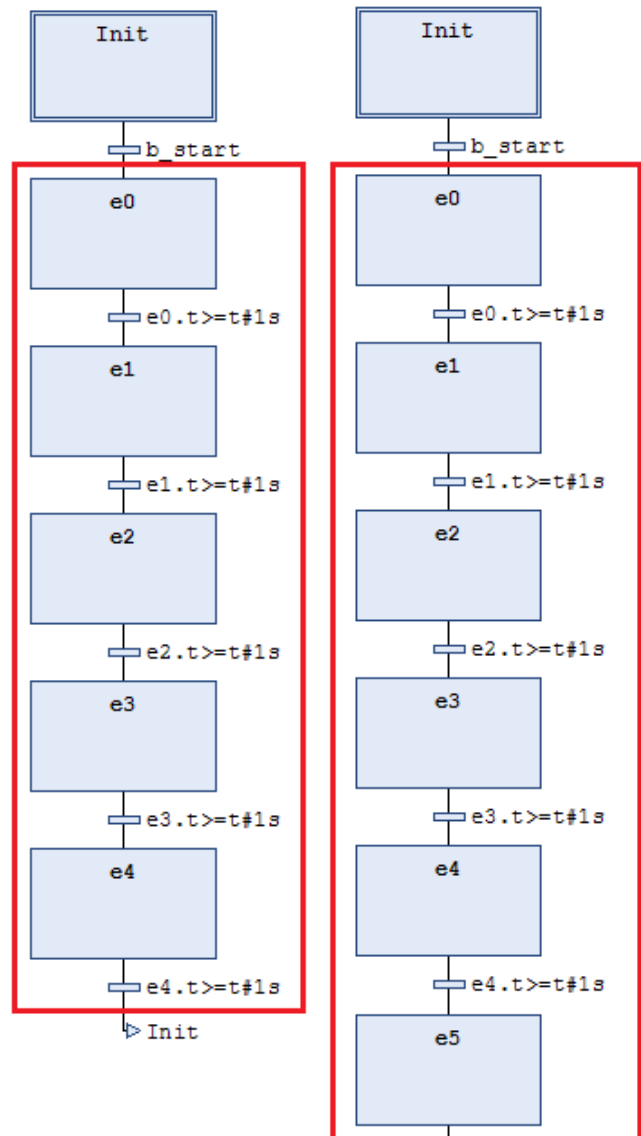
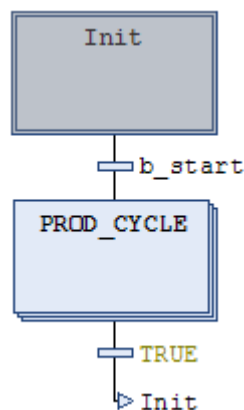
Pour donner une première illustration des modes de marche et d'arrêt, nous allons définir une macro-étape que nous appellerons PROD_CYCLE.

Nous allons placer au sein d'une macro-étape tout ce qui concerne un cycle de production. Ci-contre, ça représente les éléments depuis e0 à la condition-transition qui suit e4.

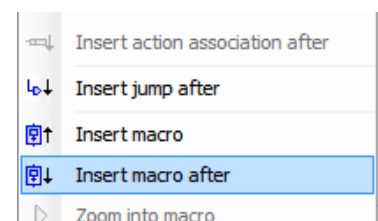
Par contre, on doit s'assurer que le contenu d'une macro-étape commence par une étape et se termine par une étape. Ce qui nous amène à ajouter une étape e5 comme illustré ici tout à droite.

La raison de cette règle est qu'on peut ainsi manipuler des macro-étapes de la même façon qu'une étape d'un SFC en conservant l'alternance étape ↔ transition !

Après l'insertion de la macro-étape, le SFC se réduit comme ci-dessous :



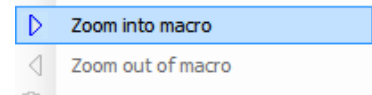
Sur eCOCKPIT, l'insertion d'une macro peut se faire par exemple en sélectionnant une condition-transition et en faisant un clic droit. On peut alors choisir, comme représenté ci-contre, d'insérer une macro avant ou après la condition-transition.



MANIP 11 : Mode semi-automatique avec macro-étape et simulation

On peut ensuite entrer dans le contenu de la macro à l'aide de l'option « Zoom into macro » (ou du double clic).

On sortira de la macro à l'aide de « Zoom out of macro ».



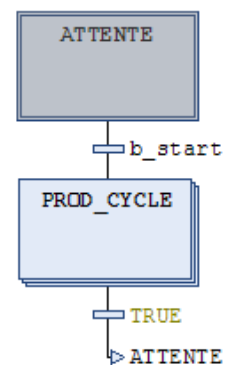
On peut renommer notre macro en « PROD_CYCLE » par exemple et y placer les étapes e0 à e5 ; le code de notre fichier « commandes » pourra rester inchangé.

E. Mode semi-automatique sur front montant de « Start »

La séquence ci-contre illustre une production en mode semi-automatique.

L'étape « Init » a été renommée en « ATTENTE » pour mettre en évidence qu'il s'agit d'une phase d'attente du signal de démarrage de cycle.

L'étape « PROD_CYCLE » correspond à une phase d'exécution d'un cycle de production.



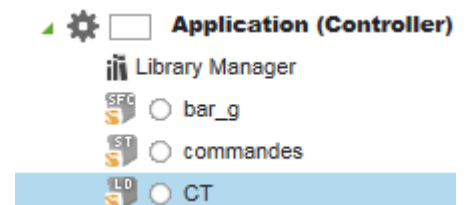
Nous modifierons la condition-transition de démarrage de cycle en un front montant sur « b_start » pour forcer l'opérateur à relâcher le bouton.

La détection du front montant de « b_start » **nécessite** une mémorisation de l'état précédent et **un morceau de code** pour vérifier qu'il s'agit d'un front montant.

Ce code peut être écrit dans plusieurs langages en utilisant le bloc R_TRIG (R pour Rising) ou en utilisant le contact type P (Positive Edge) en Ladder, ce que nous ferons.

À l'aide du clic droit sur Application (Controller), on créera un nouveau POU en LADDER (nommé ici CT pour condition-transition).

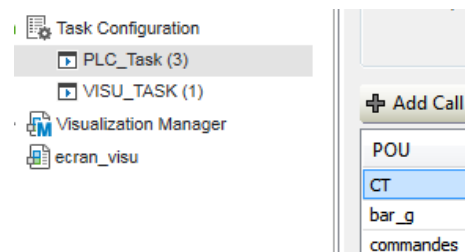
Remarque : Les POU's se placent par ordre alphabétique.



Au sein de l'API, l'exécution du programme se fait de haut en bas et de droite à gauche. Du coup, l'ordre alphabétique de l'arborescence du projet peut s'avérer trompeuse.

Le paramétrage de l'ordre d'exécution des POU's se fait au sein du « PLC_Task ».

Nous choisirons l'ordre logique CT → bar_g → commandes.

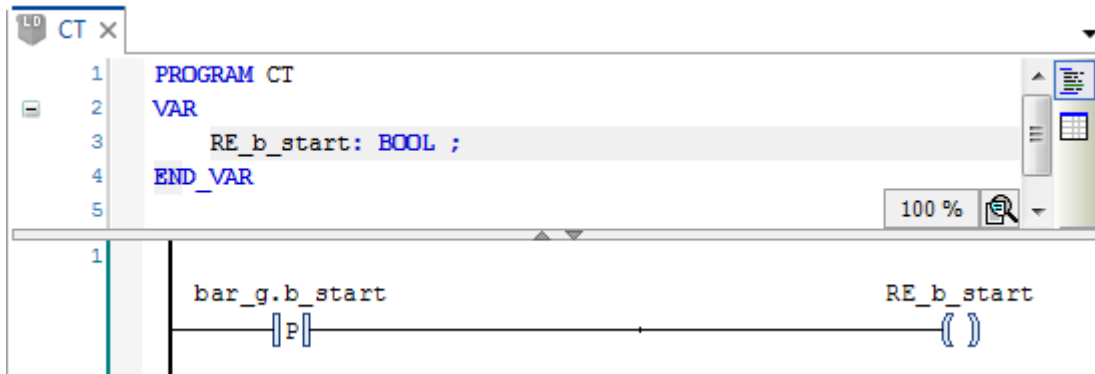


Complétons maintenant le contenu du POU « CT » à implémenter en Ladder.

MANIP 11 : Mode semi-automatique avec macro-étape et simulation

Comme illustré ci-dessous, on déclare une variable « RE_b_start » (RE pour Rising Edge) que l'on commande au travers d'une bobine (sortie) en Ladder.

Le contact marqué d'un « P » sera fermé lorsqu'on aura un front montant sur la variable « b_start » du SFC « bar_g ».



On peut maintenant modifier comme ci-contre notre condition-transition de démarrage de cycle.

En écrivant « CT.RE_b_start », on fait apparaître le nom de la variable et le nom du « POU » auquel elle est associée.

On prendra soin de bien vérifier le fonctionnement du programme, en particulier le cas où l'opérateur tenterait de maintenir le bouton « Start » enfoncé (avec un « Toggler » par exemple).

Il est à noter que, pour certains systèmes, la détection des fronts en simulation ne fonctionne pas toujours.

