

TP 1: Initiation à la programmation Automate M45 Automatismes Industriels

Vincent Laurain

Introduction

Félicitations! Votre entreprise vient de se porter acquéreur d'un automate M340 Schneider flambant neuf! Il vient tout juste d'être câblé par l'électronicien, et votre tâche, si vous voulez être embauché à la fin de votre période d'essai, est tout d'abord de le faire fonctionner. Il vous faudra ensuite réaliser un programme très simple demandé par un de vos supérieurs pour l'anniversaire de sa fille et quelques autres tâches.

Ici, il vous sera demandé un rapport. Ce rapport est important. Il doit être rendu au format PDF sur Arche. Vos rapports seront également les seuls autorisés durant un examen de TP.

Programmation de l'adresse réseau de l'automate

Etant donné que vous n'avez encore jamais connecté un automate, celui ci n'a pas encore d'adresse IP. Vous ne pourrez donc pas vous y connecter en utilisant le réseau informatique. La seule solution est de s'y connecter en utilisant un câble USB. Cependant, la connexion USB impose d'avoir un ordinateur à proximité pour programmer l'automate et ce n'est pas exactement une solution pratique, vous en conviendrez. Ainsi voici les étapes à suivre pour pouvoir programmer l'adresse IP dans l'automate.

Démarrage de Unity pro XL

Certains d'entre vous auront probablement déjà utilisé PL7 Pro. Cependant, les nouveaux automates de Schneider utilisent le logiciel *Unity Pro XL*. Lancez ce logiciel en effectuant une simple recherche dans les programmes de l'ordinateur. Vous allez alors créer un nouveau fichier. Lorsque vous effectuez cette action, le logiciel vous demande de choisir un modèle d'automate : **toutes les informations nécessaires sont inscrites sur l'automate. Soyez donc observateur.** Une fois le modèle choisi, le projet s'ouvre

Renseignement des cartes

En cliquant sur l'onglet Configuration , vous allez pouvoir renseigner le type de carte dont cet automate est pourvu (ou au moins celles que vous allez utiliser). De même que précédemment, tout est inscrit sur les cartes de l'automate.

Communication avec l'automate

Il est possible de se connecter à un automate de plusieurs manières, les plus courantes étant par USB ou par ETHERNET. En l'état, votre automate ne dispose pas encore d'adresse IP, donc vous devrez vous y connecter dans un premier temps, par USB. La manière dont vous vous connecterez à l'automate peut être modifiée dans l'onglet Automate-;Définir l'adresse.

Par exemple si vous souhaitez vous connecter par USB, le type de connexion sera USB, et l'adresse SYS. Si vous souhaitez vous connecter par Ethernet, le type de connexion sera TCP/IP, et l'adresse correspondra à l'adresse IP de l'Automate.

Vous pouvez dans ce même menu tester la connexion avec l'automate. L'adresse de connexion à l'automate est indiquée en Vert, dans le bas de la fenêtre de Unity.

Mais avant de vous connecter à l'automate, nous allons le configurer de sorte à pouvoir, par la suite, nous y connecter par Ethernet, ce qui sera dans notre cas, plus pratique.

Distribution de l'adresse IP

Afin de permettre à l'automate de se connecter par Ethernet, nous devons lui charger une configuration Réseau. Cela se fait en deux étapes :

- 1)Créer une configuration réseau
- 2)L'associer à la carte réseau de l'automate

En conditions réelles, une recherche de "Modicon M340 for Ethernet - Schneider Electric" sur un moteur de recherche quelconque vous mènerait au document de 382 pages contenant votre réponse. Comme je ne suis pas un monstre, je vais ici résumer les étapes se trouvant (à partir de la page 125) :

Etape 1 :

- 1) Sur la gauche, dans l'onglet **Communication/Réseaux** effectuer un clic droit et créer un nouveau réseau. Le type de réseau que l'on veut créer est un réseau Ethernet. Le nom importe peu, pour l'exemple, appelons le "Res1".
- 2) Il faudra ensuite définir une famille de carte de communications, qui correspondra à celle que vous avez entré tout au début BMX P4... (ici c'est une version 2.0)
- 3) Configuration de l'adresse IP. L'ingénieur Réseaux nous a donné les informations suivantes :
 - automate 1 : (adresse mac 00 :80 :f4 :12 :22 :61) - 193.54.28.48
 - automate 2 : (adresse mac 00 :80 :f4 :12 :22 :32) - 193.54.28.49
 - automate 3 : (adresse mac 00 :80 :f4 :12 :22 :63) - 193.54.28.50
 - Masque de sous réseau 255.255.255.0
 - Passerelle 193.54.28.254
- 4) Dans l'onglet sécurité activez les deux options (la version utilisée pour la carte est 2.5).

Etape 2 : Maintenant que la configuration réseau "Res1" est correctement définie, il s'agit de l'associer à la sortie Ethernet de la carte réseau. Pour cela , sur la gauche dans l'onglet configuration, déroulez le menu jusqu'à atteindre la broche Ethernet du bloc 0. Ensuite, dans la voie 3, changez sa fonction en "TCP/IP" et associez le lien réseau à la configuration "Res1".

Attention, si vous ne voyez pas "Res1" apparaître, c'est peut être que vous n'avez pas fait correctement l'étape 1.

Compilation du projet et transfert

Maintenant, nous allons charger cette configuration Réseau dans l'automate :

- Etape 1 : Compilation du projet pour vérifier qu'il ne contient pas d'erreur
- Etape 2 : charger le programme de configuration dans l'automate
- Etape 3 : Vérifier que la configuration réseau fonctionne

Etape 1 : Dans la barre d'icônes supérieure, il y a l'icône "générer le projet" (regénérer tout le projet fonctionne également). Cliquez dessus. Si tout a été fait correctement, alors, le glorieux message "Processus réussi : 0 erreurs, 0 avertissements" s'affichera fièrement sur votre écran (les avertissements n'empêchent pas le transfert ; mais les erreurs oui). Si il y a des erreurs, revérifiez chacune de étapes précédentes.

Etape 2 : Puisque l'automate n'a pour l'instant aucune configuration réseau, vous devez vous y connecter par câble USB. Pour que cela fonctionne, en bas de la fenêtre, en vert, doit être indiqué USB:SYS (Si ce n'est pas le cas, allez dans Automate→définir l'adresse). Allez maintenant dans "Automate → Connexion", puis "Transférer le projet vers l'automate".

Etape 3 : Si le transfert a fonctionné, alors vous allez vérifier que la configuration est bonne. Pour cela :

- Déconnectez vous de l'automate
- Débrancher le câble USB
- Modifiez dans Automate→définir l'adresse pour tenter de vous connecter à l'automate par Ethernet
- Automate → Connexion

Si votre configuration était bonne, alors vous pouvez vous connecter à l'automate.

Félicitations !

Si vous avez correctement effectué toutes les étapes précédentes, alors vous avez bien configuré votre projet. A partir de maintenant la méthodologie sera toujours la même :

- Programmer un schéma bloc
- Générer
- Se connecter
- Transférer
- Tester

Ici, lorsque vous avez testé la connexion et que tout marche, il est fortement conseillé de sauvegarder le projet et nommez le "ProjetVierge". Ainsi pour les prochains TPs, vous pourrez vous en servir comme d'une base pour ne pas avoir à reprogrammer tout l'aspect communication.

Si ça marche, vous pouvez passer à la suite.

Programmation en logique combinatoire

Votre supérieur despotique a vu la chose à la télé qui ferait plaisir à sa fille pour son anniversaire. Vous pourrez visionner cette vidéo à cette adresse : <https://www.youtube.com/watch?v=0mA19KVDC5Q>. Et bien sûr il aimerait se servir de vos talents pour que vous puissiez lui faire une démonstration. Visionnez la vidéo et programmez l'automate de sorte à obtenir le même fonctionnement que la boîte à ampoules. Bon courage !

Le salut est dans le cours de combinatoire

Constatant votre panique évidente, votre professeur d'informatique industrielle décide de vous aider. Nous allons simplement appliquer le cours de logique :

- Trouver les entrées/sorties
- Faire une table de vérité (Soit en comprenant, soit en observant)
- Extraire les équations depuis la table de vérité (soit canonique, soit Karnaugh)
- Les transformer en schéma bloc
- Les programmer sur Unity

La programmation de la boîte à ampoules

Une fois les équations de sorties établies, nous allons passer à la programmation en Langage LADDER, qui correspond en fait à du schéma bloc. Pour cela, à gauche, dans "Programme → MAST → Sections" faites un clic droit et créez une nouvelle section. Nommez la BoiteAAmpoule puis définissez le langage de programmation en LADDER (LD). Une page s'ouvre que vous allez pouvoir remplir d'icônes correspondant aux entrées et sorties.

Les entrées sorties

Comme vous aurez pu le remarquer dans l'onglet de configuration, chaque carte que vous avez ajouté possède un numéro. De plus, les cartes présentes sur les automates sont des cartes entrée/sorties tout ou rien disposant de 16 entrées et 16 sorties. Chacune d'elle à un nom bien spécifique : %I pour les entrées et %Q pour les sorties. Les entrées sont numérotées de 0 à 15 et les sorties de 16 à 31. De plus il est important de considérer sur quelle carte sont présentes ces entrées/ sorties. Par exemple, pour faire référence à l'entrée 6 de la carte 2 il faut la référencer ainsi : %I0.2.6. Le premier 0 correspond au numéro du

bloc CPU (certains automates peuvent en avoir plusieurs). Ainsi pour référence la sortie 20 de la carte 1, il faudra la référencer %Q0.1.20. Les différentes sorties de l'automate sont données en annexe. Pour les entrées, appuyez sur les boutons et vous verrez ce qui s'allume sur l'automate.

Vous allez vite comprendre qu'il est fastidieux de programmer correctement si vous devez référencer à chaque fois les entrées/sorties par leur nom réel. Ainsi, dans les variables et instances, vous pouvez entrer le nom de variables élémentaires, qui correspondront à vos sorties. En fait, on donnera ici un "alias". Pour les entrées/sorties. Attention, le type est EBOOL.

Il ne vous reste plus qu'à programmer. Les 4 boutons sont Gauche, Droite, Marche, Arrêt (Attention il y a un piège pour arrêt). Les 4 lampes de sorties sont les LEDS de marche. Réinitialisez le système grâce à l'arrêt d'urgence.

Annexe

Les sorties sont numérotées de 16 à 31. Carte 1 :

- Sortie 24 : Moteur Gauche
- Sortie 25 : Moteur Droite

Carte 2 :

- Sortie 16 à 19 : LED Marche
- Sortie 20 à 23 : LED Repos
- Sortie 24 à 27 : LED Default
- Sortie 28 : Voyant Marche
- Sortie 29 : Voyant Arrêt
- Sortie 30 : Voyant Acquit
- Sortie 31 : Alimentation du Variateur (**nécessaire pour faire tourner le moteur**)

Pour pousser plus loin : l'Utilisation des objets usuels sous Unity

Les objets habituels à manipuler lorsqu'on programme un automate sont :

- Les temporisations. Une temporisation peut fonctionner en mode TON ou TOFF ;
- Les compteurs : qui comptent ;
- Les blocs operate pour effectuer des opérations simples.

Il existe encore bien d'autres objets : les piles, les registres....

De manière générale, toute sorte de fonctions (compteurs, lecture dans un autre automate, tempo...) peuvent être intégrées à un schéma en LADDER. Pour insérer un bloc fonctionnel dans un schéma ladder, vous pouvez faire un simple clic droit "assistant de saisie FBB". Alors s'ouvre une fenêtre vous demandant un nom de bloc à insérer. Vous avez également à coté un symbole "..." vous permettant de chercher dans une liste. En cliquant sur ces points s'ouvrent tous les objets disponibles dans la bibliothèque "V8.0". En général, pour connaître le nom d'un bloc donné, une recherche sur votre moteur internet préféré fera l'affaire. Ici, nous aurons principalement besoin de temporisations (qui s'appellent TON ou TOF) et de compteurs (qui s'appellent CTU).

Réalisation d'une minuterie

En vous servant de la documentation fournie dans l'aide, réalisez le fonctionnement d'une minuterie de cage d'escalier :

- Lorsque l'on appuie sur le bouton, la lumière s'allume durant une période de 5 secondes
- Tout appui réinitialise la minuterie (quelqu'un pressant le bouton a bout de 4 secondes doit disposer de 5 secondes supplémentaires). Choisissez la bonne minuterie en fonction de l'aide et cablez la.

Utilisation des bits systèmes et des compteurs

Un automate dispose d'une mémoire propre pouvant stocker différents types de données (entiers, réels, doubles, booléens...)

- En utilisant un compteur, comptez le nombre d'appui sur le bouton marche. Ce compteur doit être réinitialisé par l'appui sur le bouton arrêt. Stockez la valeur de ce compteur dans un mot qui est un entier (vous pouvez déclarer une variable entière dans les variables, et même lui adjoindre un bloc mémoire %MWi(i pouvant aller de 1 à 100))
- En utilisant un simple compteur et le bit interne du système %S4 (plus précis que %S5 ou %S6 et regardez leur signification dans l'aide), mesurez le temps d'appui sur le bouton marche (ce compteur sera aussi réinitialisé par un appui sur arrêt). Stockez ce résultat dans un autre entier. Il existe beaucoup de bits systèmes indiquant un démarrage à froid, à chaud, ...

Utilisation des blocs operate et

Utilisez maintenant les blocs operate et compare réalisez la fonction suivante :

- Comptez le temps moyen d'appui sur le bouton marche : pour cela il faudra effectuer une simple division des résultats précédents. **ATTENTION CEPENDANT, TOUTE DIVISION PAR 0 METTRA L'AUTOMATE EN ERREUR. IL FAUDRA DONC TESTER D'ABORD LE DIVIDENDE en utilisant un bloc compare.** cette valeur doit être stockée dans une variable réelle flottante (REAL).

Pour programmer un bloc operate, il faut programmer en langage structuré : par exemple %MW2:= 2.*MW1. Il faudra faire attention cependant aux types de données. Un entier ne peut diviser qu'un entier, et le résultat sera aussi un entier. SI on veut un réel, il faudra utiliser une fonction INT_TO_FLOAT() qui transforme un type entier en réel (toutes les fonctions de conversion existent).

Amusez vous

Vous avez maintenant toutes les clefs en main pour réaliser le jeu suivant. A l'appui sur marche, 2 joueurs doivent appuyer le plus grand nombre de fois sur le bouton Bp Droite (respectivement Bp Gauche) dans un temps déterminé. Le score détermine le gagnant.