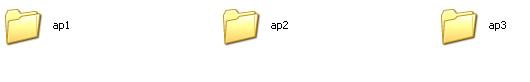
PL7

Création d’une application en ST

Etape 1

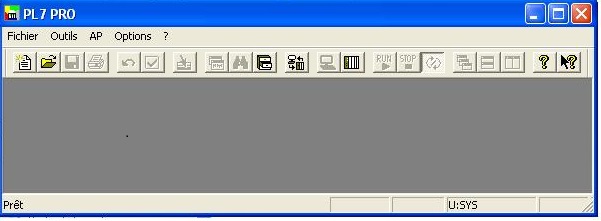
Copier un exemplaire du dossier « Fichier Configuration » dans lequel se trouve trois dossier contenant des fichier exploitables par PL7 et préconfigurés pour les automates 1,2 et 3.



Etape 2

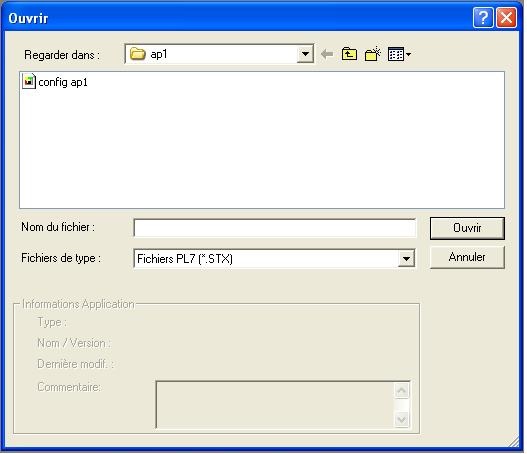
Ouvrir PL7 :

Démarrer->Tous les programmes->Modicon Telemecanique->PL7 v4.4



Etape 3

Ouvrir le fichier de configuration correspond à l’automate que vous voulez programmer

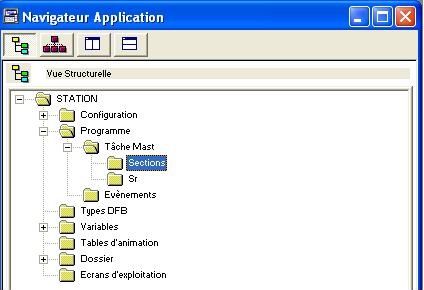


Etape 4

Une fois le fichier ouvert, le navigateur Application doit s’ouvrir. Il faut à présent créer un fichier sur lequel nous allons pouvoir développer le programme.

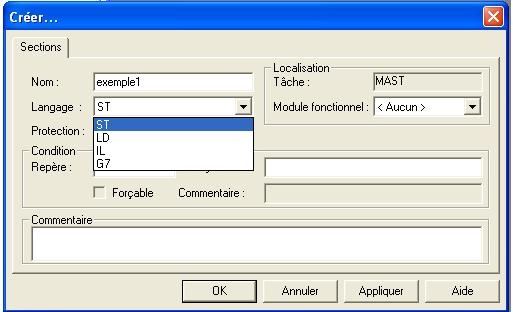
Pour cela Allez dans l’onglet :

Programme->Tâche Mast, puis clic droit sur « Sections » et sélectionner « Créer »

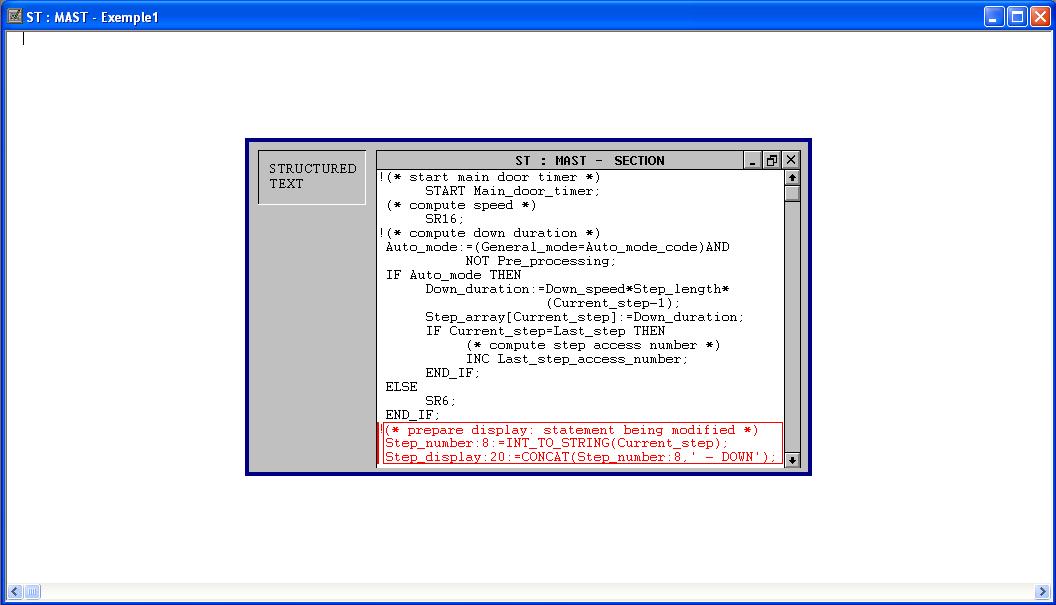


Etape 5

Saisir le nom du programme et sélectionner « ST » dans la fenêtre « langage »



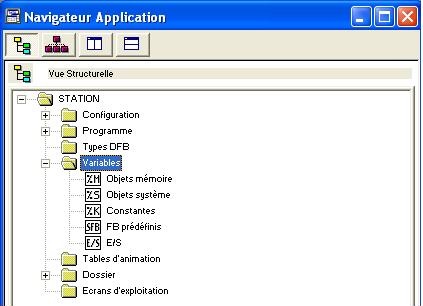
La fenêtre d’édition ci-dessous devrait apparaître et permettre de développer un programme en langage ST. Pour la syntaxe, ne pas hésiter à consulter l’aide de PL7.



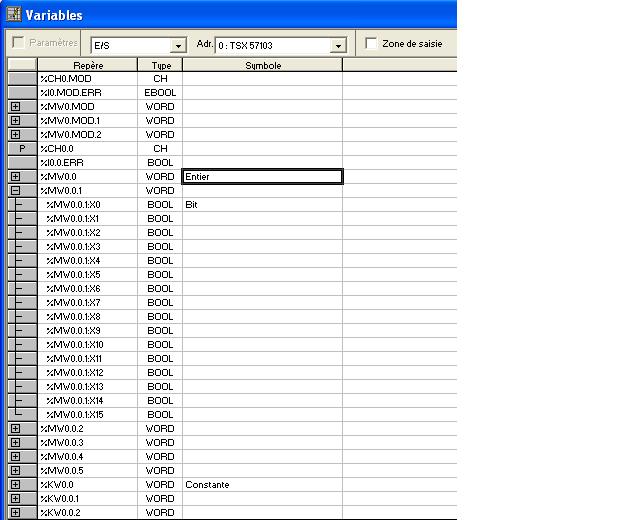
Etape 6 (facultative)

Si le programme nécessite la création de variables. Il faut les déclarer dans le fichier Entrées/Sorties de l’Automate situé dans le navigateur Application.

Station->Variables->E/S



Dans la fenêtre « Variables » il est possible de définir des variables entières, de bits ou alors des constantes. Les variables potentielles peuvent être reconnues avec leur préfixe %MWx.y avec x et y des entiers quelconques et il suffit d’écrire le nom de la variable dans la case « symbole ».

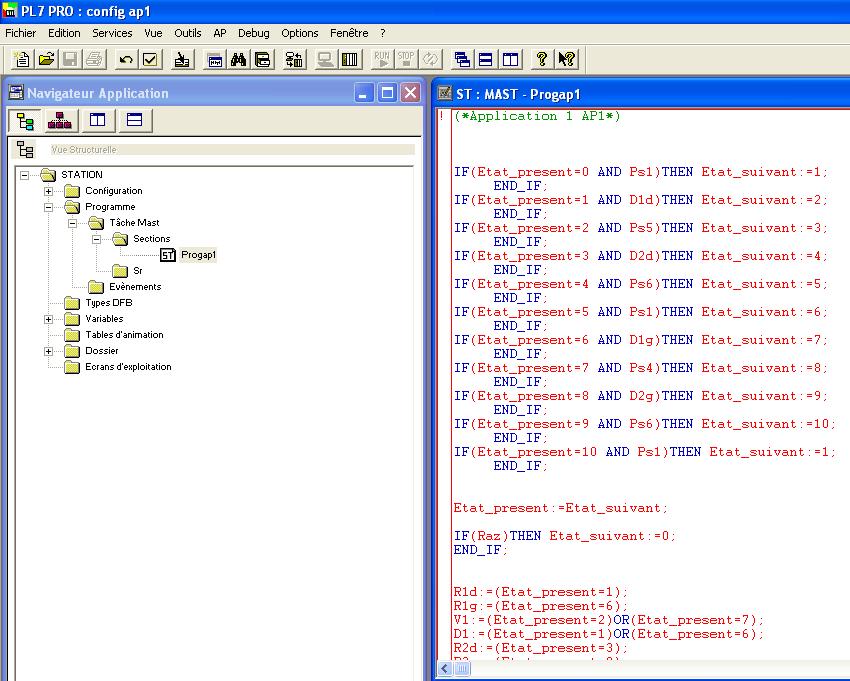


Etape 7

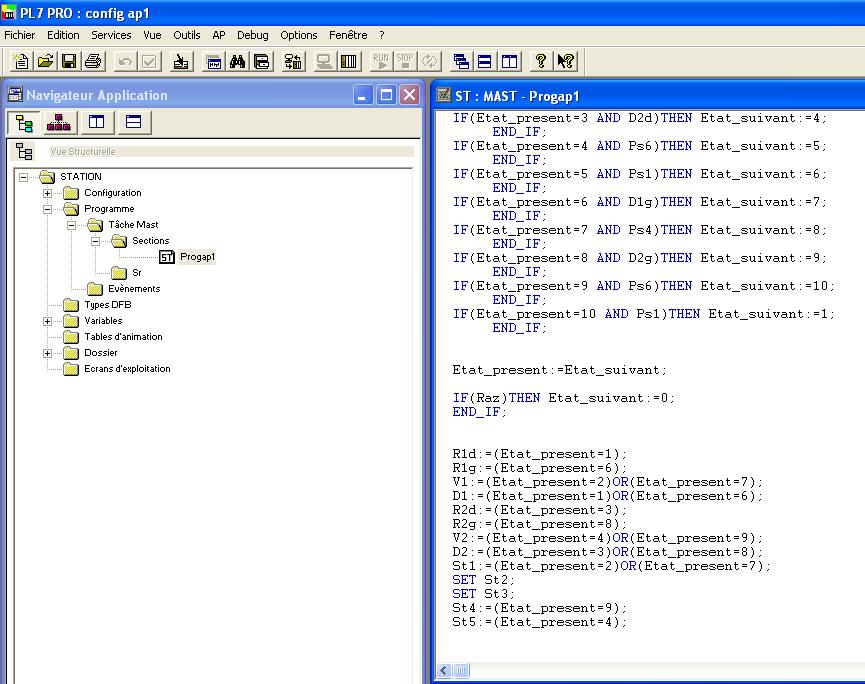
Une fois le programme édité et les variables définies, il faut le compiler avant de le transmettre vers l’automate en appuyant sur le bouton « valider ». Tant que le programme n’est pas bon, le texte apparaitra en rouge tandis qu’une compilation réussie fera apparaitre le texte en noir.



Bouton « Valider »



Échec de la compilation, texte rouge

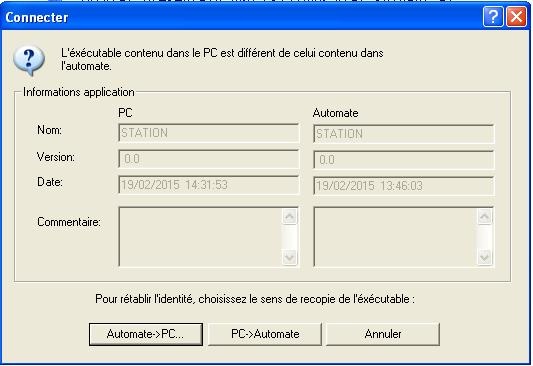


Compilation réussie

Texte noir

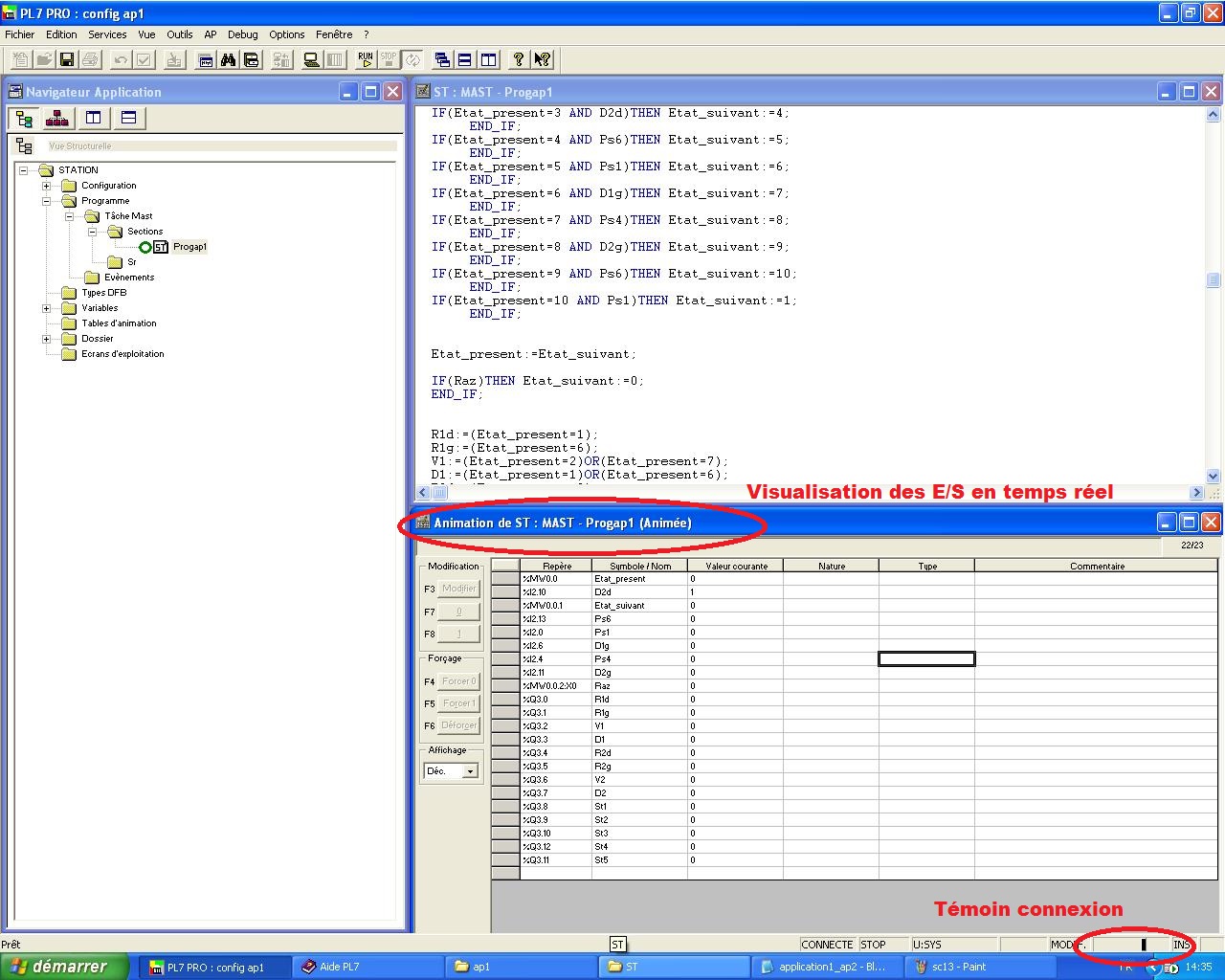
Etape 8

Il faut maintenant transférer le programme vers l’automate. Pour cela, cliquer sur l’icône « connecter ». Puis dans la fenêtre, sélectionner « PC->Automate ».



Si jamais une fenêtre indiquant que la version du système est en 5.1 et celle de l’automate en 5.2 ou alors que l’automate doit être en mode STOP, cliquer sur OK dans les deux cas.

Une fois l’automate connecté, cliquer sur « RUN » pour exécuter le programme. Cela va faire apparaitre une fenêtre de visualisation des variables et entrées/sorties en temps réel.



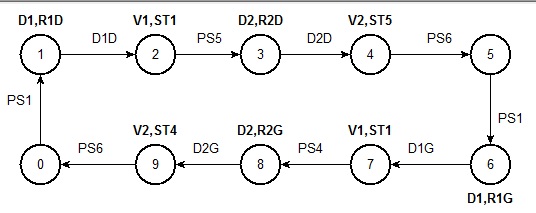
Il est possible de modifier les variables et les sorties à partir de cette fenêtre en sélectionnant une case « Valeur courante » associée à un paramètre pour ensuite modifier sa valeur à l’aide des icônes « Modification ».

Exemple d’implémentation avec des Machines à Etats Finis

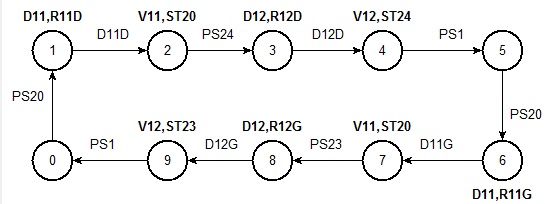
Fonctionnement désiré :

Une fois la navette positionnée sur les rails de l’automate 3 assurant la liaison entre l’automate 1 et 2. Celle-ci doit effectuer une alternance d’un tour s’effectuant par l’extérieur suivi d’un tour s’effectuant par l’intérieur comme résumé par le schéma ci-dessous.

Pour cela, il est indispensable de programmer les deux automates avec deux MAE différentes. On propose les deux machines à état suivantes.



Automate 1



Automate 2

Implémentation des MAE en ST

(\*AP1\*)

IF(Etat\_present=0 AND Ps1)THEN Etat\_suivant:=1;

END\_IF;

IF(Etat\_present=1 AND D1d)THEN Etat\_suivant:=2;

END\_IF;

IF(Etat\_present=2 AND Ps5)THEN Etat\_suivant:=3;

END\_IF;

IF(Etat\_present=3 AND D2d)THEN Etat\_suivant:=4;

END\_IF;

IF(Etat\_present=4 AND Ps6)THEN Etat\_suivant:=5;

END\_IF;

IF(Etat\_present=5 AND Ps1)THEN Etat\_suivant:=6;

END\_IF;

IF(Etat\_present=6 AND D1g)THEN Etat\_suivant:=7;

END\_IF;

IF(Etat\_present=7 AND Ps4)THEN Etat\_suivant:=8;

END\_IF;

IF(Etat\_present=8 AND D2g)THEN Etat\_suivant:=9;

END\_IF;

IF(Etat\_present=9 AND Ps6)THEN Etat\_suivant:=0;

END\_IF;

Etat\_present:=Etat\_suivant;

IF(Raz)THEN Etat\_suivant:=0;

END\_IF;

R1d:=(Etat\_present=1);

R1g:=(Etat\_present=6);

V1:=(Etat\_present=2)OR(Etat\_present=7);

D1:=(Etat\_present=1)OR(Etat\_present=6);

R2d:=(Etat\_present=3);

R2g:=(Etat\_present=8);

V2:=(Etat\_present=4)OR(Etat\_present=9);

D2:=(Etat\_present=3)OR(Etat\_present=8);

St1:=(Etat\_present=2)OR(Etat\_present=7);

SET St2 ;

SET St3 ;

St4:=(Etat\_present=9);

St5:=(Etat\_present=4);

(\*AP2\*)

if( Etat\_present=0 AND ps20 ) then etat\_suivant:=1;

end\_if;

if( Etat\_present=1 AND d11d ) then etat\_suivant:=2;

end\_if;

if( Etat\_present=2 AND ps24 ) then etat\_suivant:=3;

end\_if;

if( Etat\_present=3 AND d12d ) then etat\_suivant:=4;

end\_if;

if( Etat\_present=4 AND ps1 ) then etat\_suivant:=5;

end\_if;

if( Etat\_present=5 AND ps20) then etat\_suivant:=6;

end\_if;

if( Etat\_present=6 AND d11g ) then etat\_suivant:=7;

end\_if;

if( Etat\_present=7 AND ps23 ) then etat\_suivant:=8;

end\_if;

if( Etat\_present=8 AND d12g ) then etat\_suivant:=9;

end\_if;

if( Etat\_present=9 AND ps1 ) then etat\_suivant:=0;

end\_if;

etat\_present := etat\_suivant;

IF(Raz)THEN Etat\_suivant:=0;

END\_IF;

R11D:=(etat\_present=1);

R11G:=(etat\_present=6);

V11 :=(etat\_present=2) or (etat\_present=7);

D11 :=(etat\_present=1) or (etat\_present=6);

R12D:=(etat\_present=3);

R12G:=(etat\_present=8);

V12 :=(etat\_present=4) or (etat\_present=9);

D12 :=(etat\_present=3) or (etat\_present=8);

ST20:=(etat\_present=2) or (etat\_present=7);

SET ST21 ;

SET St22 ;

ST23:=(etat\_present=9);

ST24:=(etat\_present=4);