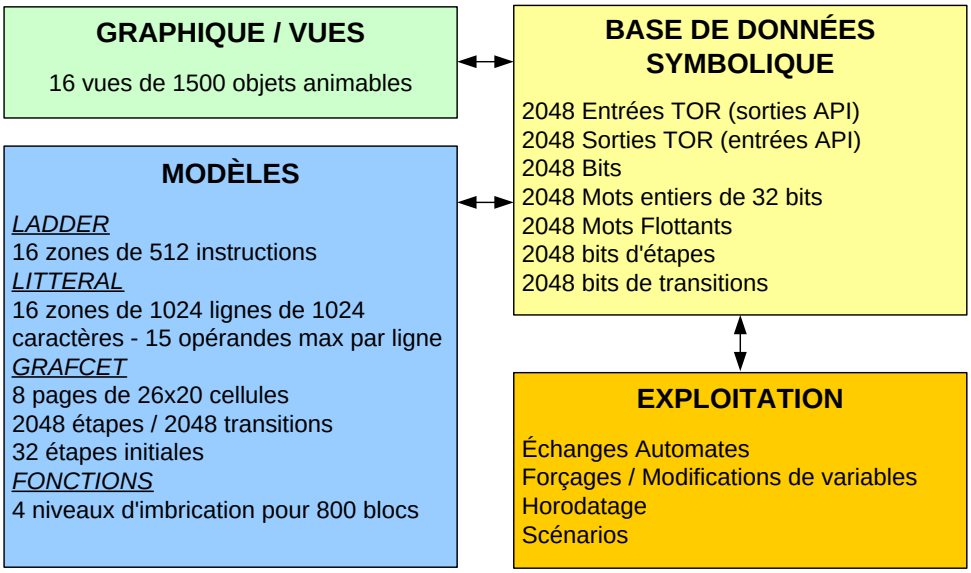
SOMMAIRE PROSIMUL 32 BITS

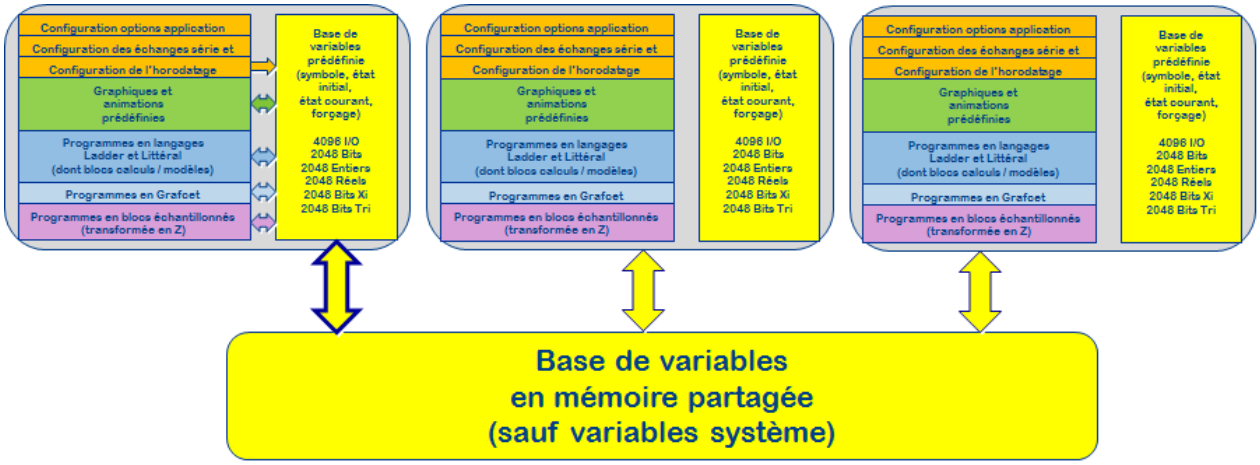
1. GENERALITES
2. INSTALLATION
3. MODES OPERATOIRES
4. VARIABLES
5. EDITEUR GRAPHIQUE
6. ANIMATIONS
7. EDITEUR LADDER
8. EDITEUR LITTERAL
9. EDITEUR GRAFCET
10. EDITEUR FONCTIONS
11. CONFIGURATION DES ECHANGES
12. EXPLOITATION
13. HORODATAGE
14. SCENARIO
15. IMPORT / EXPORT
16. CONVERTISSEUR
17. DOSSIER
18. ÉVOLUTIONS

GÉNÉRALITÉS

Structure du Projet (.SIM)



Multi applications ou applications Maître-Esclave



Import / Export

Un format unique (.BIM) pour les graphiques et modèles.
Un format texte (.TXT) pour les échanges de symboles d'entrées / sorties.

INSTALLATION

NOTE : l'installation en environnement virtuel est déconseillée pour des raisons de sous-performances des PC et de risques de « découpage » temporel des processus de simulation.

1 - Lancer l'installateur SetupPro32.
2 – Sélectionner le pilote des clés USB ou parallèle. Sous Windows 7/8/10/11 (32-64bits) il faut posséder les droits d'administration. Cette opération nécessite un redémarrage de la machine.
3 - Démarrer Prosimul avec ou sans fichier *.SIM en paramètre.
L'association des fichiers .SIM à Prosimul peut se défaire via la configuration Windows (Configuration->Programme->Définir les associations).
Un 2ème paramètre peut être donné :
- RUNFORM : démarrage en mode formation
- RUNWIN : démarrage en fenêtre (fenêtrage automatique sur plus grand synoptique)
- M1..M5 : démarrage sur le moniteur 1 à 5 si présent.
ex : « PRO32fr.EXE TEST.SIM RUNFORM M2 » lance l'application TEST en mode formation sur l'écran n°2.

Clés de protection

- Clé sur port **USB**. Clé de développement (toutes options du navigateur) ou clé d'exploitation (Lancement des applications, modifications possibles dans Configuration et Outils).

Installation du pilote : exécutée en automatique depuis l'installateur (détection de l'OS automatique). Pour **Windows 7/8/10**, les droits d'administration du PC sont nécessaires. Un redémarrage est nécessaire.

Lancement du Logiciel / Nettoyage PC

Lors de l'exécution de l'application, décharger le PC de tout programme pouvant ralentir inutilement la simulation.
- Si pas de connexion réseau, désinstaller l'antivirus (attention toutefois aux clés USB).
- Désinstaller les drivers et services utilisant Ethernet (ROCKWELL/SIEMENS/SCHNEIDER/…).
- Désactiver les services « Windows Search », « Windows Defender », « Windows Update ».
- Désactiver les services « Client DNS » et « Client DHCP » si PC en IP fixe et sans accès Internet.
- Désactiver les services « Bureau à distance » et « Gestionnaire de connexions d'accès distant ».
- Si des bases SQL sont installées, désactiver les services.
- Désactiver les services « Fichiers hors connexion » et « Explorateur d'ordinateurs ».
- Désactiver les services « Google Update » (ou autre si autres navigateurs), « Adobe Acrobat Update Service » et tout Updater non utile.
- Vérifier le planificateur de tâches (Gestion ordinateur) et désactiver les tâches inutiles.

Des priorités de process peuvent être réglées en « Configuration » « Application ».

MODES OPÉRATOIRES

Menu Initial (pas de projet ouvert)

Nouveau : création d'un projet vierge initialement (saisir un nom).
Ouvrir : ouverture d'un projet .SIM existant (choisir un nom). A l'ouverture du projet APPLI.SIM si le fichier batch APPLI.BAT existe sur le même répertoire, ce dernier est exécuté.
Convertir : convertisseur d'application DOS (sélectionner un fichier DOS .ACT).
Aide -> Sommaire PDF : accès au sommaire de l'aide.
Aide -> A propos : informations sur la version, la mémoire disponible et la taille maximum du projet sur disque.

Navigateur Projet

Configuration : accès aux informations de configuration de l'application et des échanges avec l'automate (par défaut : aucun échange).

Outils : accès aux outils de gestion du dossier, des informations d'horodatage et des fichiers scénarios.

Graphiques : accès aux pages graphiques 0 à 15. Des **noms de page** peuvent être fournis (simple clic sur le nom d'une page sélectionnée au préalable). Seules les pages utilisées sont visibles. **Un clic droit sur la racine des graphiques permet d'ajouter des pages vierges**.

Variables : accès aux symboles, états initiaux, valeurs courantes et état de forçage des variables de la base. En version standard 2048 Entrées TOR, 2048 Sorties TOR, 2048 mots entiers 32 bits, 2048 réels 32 bits, 2048 bits d'étapes et 2048 bits de transition Grafcet.

Ladder / **Littéral** : accès aux 16 zones de programmes en LADDER ou en LITTERAL , Main, sous-routines SR1 à SR15 et zone associé au Grafcet (exécution après cycle Grafcet). Des **noms de zone** peuvent être fournis (simple clic ou F2 sur le nom d'une zone sélectionnée au préalable). Le choix Ladder ou Litteral est effectué à la première sélection du programme ou via un **clic droit** si la zone programme est vide. Un clic droit sur la racine des programmes permet d'ajouter des pages. **Seul le programme nommé « Grafcet » sera exécuté après le cycle Grafcet**.

Grafcet : accès aux 8 pages Grafcet disponibles. Au total 2048 étapes utilisables dont 32 actives en simultané. Des **noms de page** peuvent être fournis (simple clic sur le nom d'une page sélectionnée au préalable). Un clic droit sur la racine des Grafcets permet d'ajouter des pages.

Fonction : accès à la page d'entrée des blocs fonction. Des blocs sous-pages permettent de structurer l'application en 4 niveaux. Au total 800 blocs utilisables.

Messages : accès au fichier d'enregistrement de messages PROSIMUL.LOG.

Préférences : accès aux préférences communes aux projets (stockées dans PROSIMUL.INI). Les préférences sont chargées à l'ouverture des applications ou lors de leur modification. Il contient :
Les paramètres pour les éditeurs Ladder et Littéral, l'adresse+port du serveur SMTP, les adresses locales pour la communication Ethernet, la palette de couleur « user », l'inversion **click souris** et la surveillance des connexions **Veyon**.

Menu Projet (projet ouvert – mode STOP)

Exporter : sauvegarde de symboles (.TXT) ou images+modèles (.BIM) pour utilisation dans un autre projet (saisir un nom). Cette fonction nécessite des sélections préalables (voir Import/Export).
Importer : lecture de symboles (.TXT) ou images+modèles (.BIM) sauvegardés au préalable (voir Import/Export).

Dictionnaire : cette fonction opère avec des fichiers texte Unicode tabulés .TXT. Les sous-menus disponibles sont :
- Exporter : permet d'exporter en format Unicode le dictionnaire déjà en mémoire complété des libellés d'objets graphiques nouveaux. Sélectionner le fichier destination (.TXT) et les 2 options suivantes :
    - l'export ou non des mots inutilisés et présents dans le dictionnaire.
    - l'export uniquement des textes issus des objets TEXTE.

Si plusieurs langues sont présentes dans le dictionnaire, la langue de création doit être indiquée au préalable (Configuration Application). Le fichier est ouvert avec l'éditeur Unicode sélectionné dans la Configuration Application.
- Importer : permet d'importer dans l'application un dictionnaire multilingue au format Unicode tabulé. Une initialisation du dictionnaire en mémoire est effectuée au préalable. La sélection de la langue de Création et de Runtime doit être réalisée a posteriori via la Configuration Application. Les mots importés sont stockés dans le fichier PROSIMUL.LOG. Les correspondances non trouvées sont marquées "------Not Found-----"
- [XX] (choix des langues) : permet de basculer les libellés d'objets graphiques selon les langues présentes dans le dictionnaire. La langue source est modifiée selon les choix effectués via ce menu. Lorsqu'une langue est sélectionnée, les modifications de textes sont aussi apportées au dictionnaire.
- Initialiser : permet de faire une RAZ complète du dictionnaire (en cas de problème seulement).

Enregistrer : sauvegarde immédiate de la totalité du projet sur disque dur (.SIM). Une copie de l'ancienne version est faite (fichier .BAK).
Enregistrer Sous : sauvegarde immédiate de la totalité du projet sur disque dur avec choix du nom/chemin d'enregistrement (.SIM).

Run : passage en mode exploitation de l'application selon les modes d'échanges API et de configuration application retenus. Une sauvegarde de l'application est réalisée au lancement.

Reset Messages : permet d'effacer directement le fichier PROSIMUL.LOG.

Window : permet un basculement entre applications PROSIMUL ouvertes (jusqu'à 3 applications Maître / Esclaves en simultané). **Sélection écran** déplace la fenêtre principale sur l'écran suivant si disponible.

Zones de Messages

Trois zones de messages existent dans la barre d'état (en bas d'écran).

- gauche : nom de la fenêtre active
- centre : en **rouge** - dernier message d'erreur (reporté dans le fichier « **PROSIMUL.LOG** »)
- droite : position du curseur pour les éditeurs graphiques

VARIABLES (BOITE DE DIALOGUE)

Dans la version standard, 2048 variables de chaque type sont disponibles (Entrées TOR, Sorties TOR, Bits, Mots entiers 32 bits, Flottants 32 bits, Bits d'étapes Grafcet et de transitions.
Cette boîte de dialogue permet la saisie des symboles/valeurs et forçages des variables. Elle sera aussi utilisée pour la sélection de variables pour les animations et programmes.

Numérotation (V12.02)

Par défaut les variables sont numérotées de 0 à 2047. **En réglant** le paramètre des **préférences** « **DataBase.IO_HexaMode » à la valeur 1**, la numérotation des I/O sera faite en X.Y (X=0..127 / Y=0..15).

Variables Systèmes

Plusieurs variables systèmes sont créés automatiquement.
Il est **impératif de ne pas modifier** l'emplacement de ces variables.

BIT 0 : SYS_TOP_DIXIEME → signal carré périodique (100ms – si période cycle inférieure).
BIT 1 : SYS_TOP_CENTIEME → signal carré périodique (10ms – si période cycle inférieure).
BIT 2 : SYS_INIT_GRAFCET → le passage à 1 (SET) provoque l'initialisation des Grafcets.
BIT 3 : SYS_INIT_APPLI → le passage à 1 (SET) provoque la réinitialisation de l'application.
BIT 4 : SYS_RAZ_FORCAGE → le passage à 1 (SET) provoque l'annulation des forçages.

MOT 0 : SYS_HORLOGE_DIXIEME → mot incrémenté toutes les 100ms.
MOT 1 : SYS_CYCLE_PROSIMUL → mot contenant le temps de cycle effectif en ms (-1 sur passage en STOP).
MOT 2 : SYS_PAGE_PROSIMUL → en Run il permet l'affichage des pages graphiques (valeurs 0 à 15), Grafcet (valeurs 20 à 27), Blocs Fonctions (valeur 30), Horodatage (valeur 31).
MOT 3 : SYS_COEFF_TEMPS → mot permettant la modulation du temps réel s'écoulant - valeur exprimée en % (100% initialement - un coeff à 0% fige l'application).
MOT 4 : SYS_COMPTEUR_ECHANGE → mot contenant le nombre d'échanges total avec l'API en liaison série (-1 si défaut de communication).
MOT5 à 20 : SYS_COLOR_BLACK…WHITE → contiennent les valeurs des couleurs de 0 à 15.

Symboles

- 28 caractères au maximum.
- pas d'espace en début ou fin de chaîne (un point d'interrogation les remplace le cas échéant).
- caractères interdits : « **(** » « **)** » « **{** » « **}** » « **,** » « **.** » « **;** » « **:** » « **<** » « **>** » « **=** » « **+** » « **-** » « ***** » « **/** » « **%** » « **#** », mots-clés du Littéral (If, While, Set,…) et des opérateurs de calculs (Pow, Abs,…)
- un contrôle de double existence de symbole est effectué à chaque nouvelle saisie.
- une chaîne de caractères peut être mémorisée ([CTRL][INS]) en cours de saisie pour utilisation ultérieure ([SHIFT][INS]).
- un **clic droit** sur une zone de saisie provoque l'apparition d'un menu flottant proposant les 10 dernières saisies.

Chaînes dédiées aux touches fonctions

- Associés aux variables binaires les chaînes **{FnnK}** inclues dans le symbole permettent la modification de l'état de la variable en mode pousser/pousser (application en RUN). La variable reste modifiable par programme.
- **nn** varie de **01** à **12** (12 touches fonction traitées).
- **K** prend les valeurs : **N** (touche seule), **S** (avec Shift enfoncée), **C** (avec Control) et **A** (avec Alt).
- Exemple : " BPM **{F03S}** ". La variable sera changée d'état à chaque appui combiné sur [SHIFT] et [F3] lors de l'exploitation de l'application.

État Initial

Un état initial peut être saisi **dans la base retenue**. Cet état sera transféré au lancement de l'application si l'option initialisation est retenue dans l'écran de configuration application. Cet état peut être modifié en RUN avec l'instruction « WRITEINI ».

Valeur Courante

Cette zone permet de modifier les variables en dynamiques (forcées ou non). Elle permet aussi de tester des animations lors de la phase de construction. Cette valeur doit être saisie dans la base retenue.

Bases

Pour les mots entiers la base d'affichage et de saisie des valeurs peut être choisie en bas de page à gauche parmi les 3 suivantes :

binaire :	2#
décimal :	par défaut
hexadécimal :	16#

Sélectionner la (les) variable(s) au préalable (voir ci-dessous).

Forçages

Une case à cocher est réservée pour le forçage directe des variables. En RUN, les programmes ne peuvent plus écrire l'état courant des variables forcées : les variables peuvent être forcées ou déforcées via ces cases.

Sélection de Variables(s)

- Les **boutons numérotés** à gauche des symboles permettent la sélection d'une variable (simple clic) ou de plusieurs variables (clic avec SHIFT enfoncée sur la dernière variable). Les variables sélectionnées passent en visualisation bleue. Les variables sélectionnées sont obligatoirement consécutives.

- Différents outils opèrent sur les variables sélectionnées :

. les changements de base (pour mots entiers).
. les fonctions copier/coller/couper/supprimer/insérer/import/export.

*Un **double clic** sur un bouton numéroté provoque la **sortie de la boîte de dialogue** et mémorise la sélection (idem sortie avec sélection unique). À utiliser pour les sélections depuis les renseignements de variables d'animations et de programmes.*

Outil Copier / [Ctrl][C] / [Ctrl][Insert]

En STOP, les variables sélectionnées sont stockées en mémoire tampon (symboles, valeurs initiales et courantes). Le collage est possible dans une autre application.
En STOP on peut aussi utiliser la fonction « Export » pour créer un fichier texte formaté (SYMBOL tab VALEUR) contenant les variables sélectionnées.

En RUN, le symbole est copié dans la mémoire de l'outil « Rechercher » **et** dans le presse-papier.

Outil Couper / [Ctrl][X] / [Shift][Suppr]

Les variables sélectionnées sont stockées en mémoire tampon puis supprimées de la base. Cette fonction peut être utilisée pour **déplacer** des variables dans la base (voir collage).

Outil Coller / [Ctrl][V] / [Shift][Insert]

Les variables en mémoire tampon sont stockées sur les variables sélectionnées en destination (libres normalement). Un contrôle de **double existence** de symbole est effectué.
Les bases sont adaptées si nécessaire.
Modifier les valeurs initiales et courantes si nécessaires.
Un **remplacement des variables** coupées initialement est effectué dans les animations et modèles Ladder, Grafcet et Fonctions. Ceci peut notamment permettre de changer des variables d'emplacement dans la base.
Important : Si d'autres opérations sont effectuées en les fonctions « Couper » et « Coller » (type supprimer/insérer/...), elles doivent porter sur l'ensemble de la zone coupée. Prêter attention lors des couper/coller entre types de variables différents car les animations et programmes (Ladder, Grafcet et Fonctions) les utilisant attendent parfois des types de variables fixés par défaut.

Outil Incrémenter

La variable sélectionnée (unique) est dupliquée sur la variable suivante avec incrément d'un indice situé en fin de chaîne (exemple : BOX_01 est dupliquée en BOX_02).

Outil Insérer

Cette fonction permet d'insérer N emplacements libres dans la base pour augmenter la taille d'un tableau par exemple. Les variables sélectionnées (visualisation bleue) sont décalées vers le bas.

Outil Supprimer

Cette fonction permet de libérer N emplacements libres dans la base. Les variables sélectionnées (visualisation bleue) sont décalées vers le haut. Il est intéressant de libérer des emplacements libres pour **limiter le temps de cycle** en exploitation.

Outils Chercher / Suivant

Cette fonction permet la recherche d'une chaîne de caractère ou de forçage dans la base de variables.

. Pour la recherche de texte saisir la chaîne de caractères désirée.
. Pour la recherche de **forçages** saisir la chaîne **F#** .

La première variable trouvée est sélectionnée. La fonction " Suivant " permet la poursuite de la même recherche.

Outil Imprimer

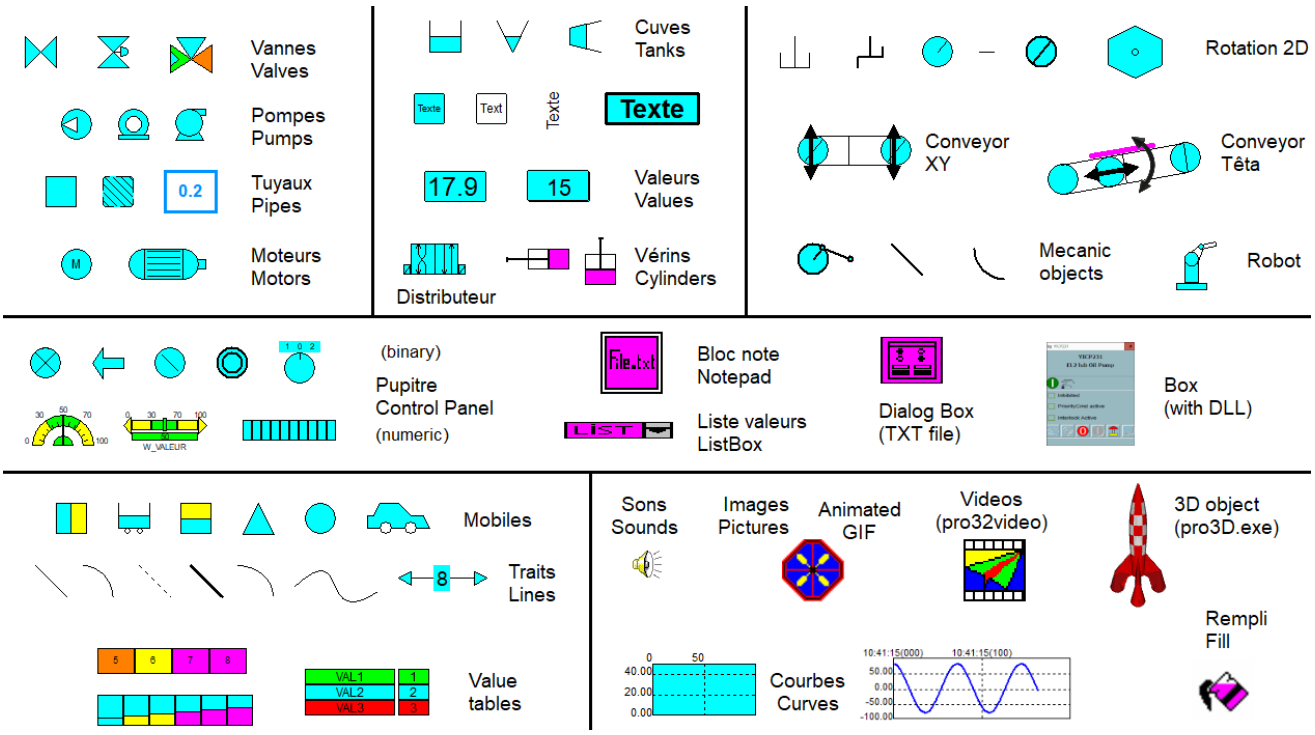
Les variables sélectionnées sont imprimées sur l'imprimante choisie.

Raccourcis lors de la sélection des variables dans les éditeurs

- Associées à la touche **[SHIFT]**, les touches [/], [P], [N], [F] apportent les fonctions complément, front montant, front descendant et test du forçage à la variable utilisée.
- Associées à la touche **[SHIFT]**, les touches [R], [S], [U] apportent les fonctions Set/Reset/Déforçage à la variable utilisée.
- Ces raccourcis sont disponibles via le **clic droit souris**.

EDITEUR GRAPHIQUE

Cet éditeur permet d'exploiter une librairie d'objets qui disposent d'animations par défaut (couleur, positions, ...), activables ou non. Il est préférable de les connaître au préalable avant de choisir un objet.



Menu Graphique

Graphique → Créer : accès à la palette d'objet disponible. Un ***double-clic*** sur une ***zone vide*** de la page Graphique provoque la même action. Saisir le libellé Unicode de l'objet.
Graphique → Modifier Type : accès à la palette d'objet pour modification du type des objets sélectionnés au préalable (permet aussi une nouvelle sélection de nom de fichier pour les objets Son, Image, Vidéo, Notepad et ListBox).
Graphique → Propriétés : accès à l'écran de configuration des propriétés d'objets (position, taille, animations,...). Un double-clic sur des objets sélectionnés provoque la même action.
Graphique → Redimensionner : permet le redimensionnement de tous les objets selon un ratio à saisir. Si le magnétisme est actif les objets sont repositionnés sur la grille, les objets proches doivent conserver leurs points de contact.
Graphique → Capturer : réalise une copie de l'image dans le presse-papier.
Graphique → Nettoyer : permet de corriger les applications présentant un problème d'affichage (type ou positions d'objets, variables d'animation inadéquates).
Zoom[+]/[-] : en STOP, permet un affichage des synoptiques avec un ratio de 50 à 200% par pas de ±10%. La **molette** de la souris associée à la touche [CTRL] réalise la même fonction.

Edition → Défaire : provoque l'annulation de la dernière opération effectuée (sur 5 niveaux).

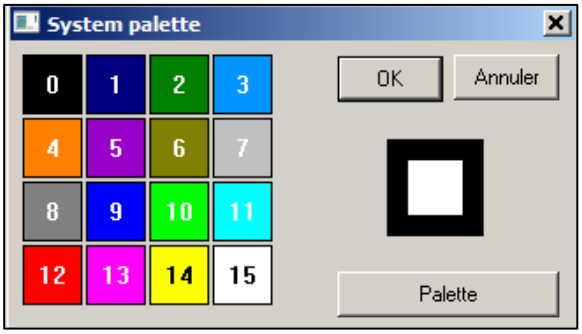
*NOTE : certaines fonctions sont accessibles via le **clic droit** avec ou sans sélection d'objets.*

Sans sélection
Edition → Coller : provoque la recopie des objets mémorisés à la position d'origine (décalés de 5 pixels en X et Y sauf si la touche [SHIFT] est activée).
Edition → Coller objet seul : provoque la recopie sans animation des objets mémorisés.

Avec sélection
Edition → Copier : provoque la mise en tampon des objets sélectionnés. Les objets copiés pourront être exportés.
Edition → Couper : provoque la copie puis l'effacement des objets sélectionnés.
Edition → Dupliquer : provoque la duplication des objets sélectionnés.
Edition → Dupliquer+Incrément : provoque la duplication des objets sélectionnés en incrémentant les variables d'animation ayant un indice en fin de chaîne (exemple : FDCS_VERIN_01 devient FDCS_VERIN_02 si le symbole existe consécutivement dans la base).
Edition → Dupliquer+Suivant : provoque la duplication des objets sélectionnés en utilisant les variables d'animation immédiatement suivantes dans la base.

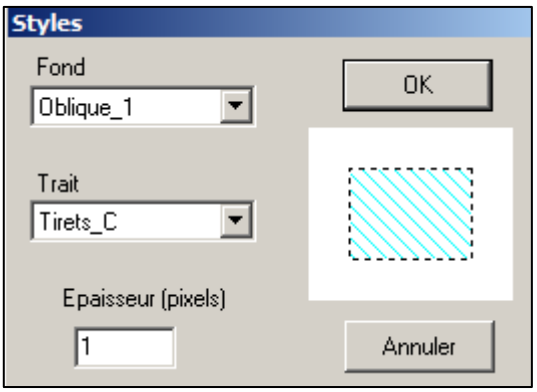
Edition → Sélectionner tout : provoque la sélection de l'ensemble des objets de la page courante.
Edition → Sélectionner remplis / textes / valeurs : provoque la sélection de l'ensemble des objets remplis, textes ou valeurs de la page courante.

Ordre → Grouper : provoque la création d'un objet groupant l'ensemble de la sélection. Des opérations de re-dimensionnement, déplacement avec magnétisme, peuvent être effectuée sur l'objet. Chaque objet ainsi groupé garde ses propres caractéristiques d'animation.
Un groupe peut être regroupé avec d'autres objets. L'ensemble se transforme en un seul et unique groupe (un seul niveau de groupement). Deux animations sont possibles : « Cacher » les objets et/ou déplacer en x, y les objets groupés.
Ordre → Dissocier : provoque la dissociation d'objets groupés.
Ordre → Premier Plan : provoque le passage au 1er plan des objets sélectionnés.
Ordre → Arrière-Plan : provoque le passage en arrière-plan des objets sélectionnés.



Couleur -> Objets : accès à la boîte de sélections des couleurs (System : numéro 0 à 15 User : 16 à 31).
- Sélection du **contour : clic gauche** sur couleur
- Sélection du **fond** : **clic droit** sur couleur
- Double clic gauche : provoque la validation (OK).

Couleur -> Fond Page : sélection de la couleur de fond de page.



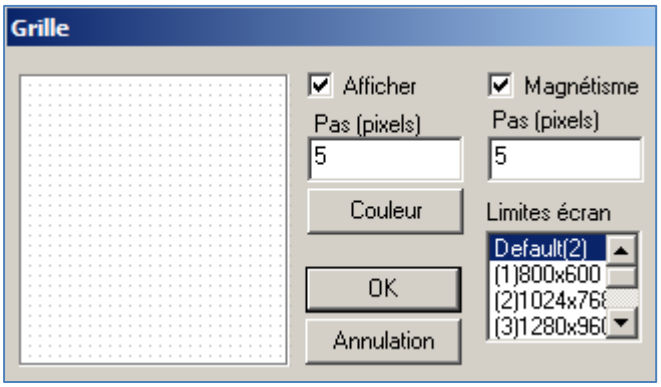
- Le bouton « **Palette** » permet de permuter les palettes « system » et « user ». La palette « user » se règle via les préférences (navigateur) et est stockée dans PROSIMUL.INI (version < 11.06) ou dans le fichier .SIM (version >= 11.06).

Style : accès à la boîte de sélection des épaisseurs trait, styles de bordure et fond. Une épaisseur de trait supérieur à 1 rend le texte **gras** pour les objets Texte et Valeur.

Police : accès à la boîte de sélection de la police de caractère et du cadrage des textes.

NOTE : Si aucun objet n'est sélectionné, les choix de couleur, styles et police effectués (ci-dessus) deviennent les options par défaut. Dans le cas contraire seuls les objets sélectionnés sont affectés par ces choix.

Grille : accès à la boîte de sélection relative à la grille et au magnétisme. Possibilité de sélectionner le pas de la grille et du magnétisme, sa couleur, son affichage ou non et le magnétisme actif ou non.



*MAGNETISME : s'il est actif les opérations de déplacements / dimensionnement (**clavier ou souris**) opèrent sur le pas de la grille pour le point 1 uniquement. Dans le cas de déplacement **d'objets groupés** les objets gardent les mêmes distances relatives entre eux mais se déplacent du pas de la grille. La **touche [M]** active ou désactive la fonction magnétisme.*

Une taille limite de synoptique (Limites écran) peut être indiquée et visualisée en Stop. En Run, combinée avec le paramètre « Animations » positionné à « Page », la taille des graphiques s'adaptera automatiquement à la taille de l'écran utilisé.

Chercher : permet la recherche d'une chaîne de caractère dans les libellés ou symboles des objets graphiques. Saisir la chaîne de caractère. Le 1er objet contenant la chaîne est sélectionné (recherche dans toutes les pages).

Suivant : poursuite de la même recherche.

Palette d'objets / Création d'objets

Le menu **Graphique->Créer** ou le double-clic sur une zone vide provoque l'apparition de la palette d'objets.

- Sélectionner le type d'objet à créer dans le navigateur.
- Saisir l'intitulé de l'objet si nécessaire.
- Cas particuliers d'intitulés :
  . Texte : *29 caractères* **Unicode** maximum (pour rappel la police et le cadrage se règlent indépendamment - Graphique -> Police). Le **caractère " \ "** permet de créer des textes sur plusieurs lignes (ex : Une ligne\Deux lignes).
  . Valeur : Indiquer le format d'affichage **XXX.YY** souhaité (format valable pour les valeurs entières et réelles). Les valeurs peuvent être affichées en binaire (syntaxe **2#**XXXX...) ou en hexadécimal (format **16#**XXXX...). Hors animation, une valeur par défaut est affichée (50).
  . Inter_3P : Le texte sera affiché au-dessus de l'interrupteur (ex : MAR…0…MAV).
  . Bitmap / Image / GIF animé / Vidéo / Notepad / ListBox : Sélectionner le nom de fichier sur le ***répertoire de l'application*** .SIM obligatoirement. L'image s'affiche dans sa taille d'origine.
  . Son : Sélectionner le nom de fichier .WAV (sur ***répertoire de l'application*** .SIM obligatoirement). Un icône s'affiche. Seul l'animation de l'objet rendra le son audible.

- L'objet est créé en haut et à gauche de la page graphique avec les propriétés par défaut de couleurs, styles et police. Pour l'objet **« Fill / Rempli »**, la couleur de contour désigne la couleur d'arrêt de remplissage ou la couleur de remplissage selon le mode (objet animé).
- L'objet peut être déplacé et dimensionné (voir ci-dessous).
- Régler ses propriétés d'animations si nécessaire.

*OBJETS : les objets positionnés sur les pages ont deux points de fixation (**1** & **2**) définissant leurs position et taille. Voir les opérations de Sélection/Déplacements/Dimensionnement ci-dessous. L'accès aux propriétés de l'objet (double-clic sur objets ou menu Graphique->Propriétés) donnent l'accès aux réglages manuels des coordonnées des 2 points de fixation*

Sélection d'objets

Un objet sélectionné est représenté avec deux " poignées " carrées.

- Lorsque la souris est déplacée au-dessus d'un objet, le curseur " flèche " est changé en curseur " croix ". L'objet est alors sélectionnable d'un simple clic.

- Pour sélectionner plusieurs objets, répéter la même opération en maintenant la touche **[SHIFT]** du clavier (sélections multiples).

- Avec la touche **[ALT]**, les objets superposés sont successivement sélectionnés.

- Un objet ou un ensemble d'objets peut être sélectionné par une **zone rectangulaire**. Ceci peut être utile notamment pour des objets se recouvrant. Pour cela :
  . Faire un simple clic sur un point exempt d'objet ou en appuyant sur la touche **[CTRL]**.
  . Maintenir l'appui de la souris et agrandir la zone pour englober les objets totalement (pour le **vérin** la zone doit couvrir la tige sortie).
  . Relâcher l'appui souris. Les objets sélectionnés apparaissent avec " poignées ".
- Le menu **Edition->Sélectionner tout** permet de sélectionner l'ensemble des objets d'une même page.

Déplacement d'objets

- Des objets sélectionnés peuvent être déplacés avec les flèches de direction du clavier.
- Le maintien du bouton gauche de la souris sur des objets sélectionnés provoque l'apparition d'un curseur " 4 directions ". Les mouvements de la souris affectent alors sa position.
- Dans les 2 cas si le magnétisme est actif le pas de déplacement sera celui de la grille.

Dimensionnement d'objets

- Le passage du curseur sur l'une des 2 " poignées " modifie son aspect (" croix_1 " ou " croix_2 "). Un appui maintenu sur ces points entraîne un déplacement des " poignées " selon les mouvements de la souris.

- Avec le **clavier** si les objets sont sélectionnés :
  . la touche **[SHIFT]** associée aux flèches de direction modifie le Point_1.
  . la touche **[CTRL]** associée aux flèches de direction modifie le Point_2.

- Avec la **souris** :
  . la touche **[SHIFT]** associée au mouvements de souris provoque un mouvement des objets sélectionnés proportionnel au ratio Y/X de l'écran.
  . si les objets sont tous groupés, le dimensionnement du groupe provoque le dimensionnement homothétique des objets groupés.

Déplacement de fenêtre

La fenêtre graphique peut être déplacée :

- Avec la **molette de la souris** :
  . par défaut les mouvements sont gauche ou droite.
  . la touche **[SHIFT]** en plus permet le mouvement haut ou bas.

- Avec le **clavier** si les objets ne pas sont sélectionnés :
  . la touche **[SHIFT]** accélère le mouvement des touches de direction.
  . la touche **[HOME]** ramène l'origine à (0,0)
  . la touche **[END]** amène sur l'objet le plus éloigné

Clic droit de souris

Si des objets sont sélectionnés le clic droit provoque l'apparition du menu contextuel d'Edition et permet en plus des commandes d'édition :
. **[Pos]** : la modification des positions X ou Y (point 1) de tous les objets sélectionnés.
. **[Size]** : la modification de la taille X ou Y ou les 2 (Z) de tous les objets sélectionnés.
. **[Format Value]** : la modification du format d'affichage des valeurs numériques ou réelles
. **[Modify Value]** : la modification de valeurs
. **[Rotate Group]** : rotation de l'ensemble des objets d'un **groupe** selon le point central du groupe
. **[Dev.Only/Training]** : la modification (complément) de l'option « Dev.Only ».

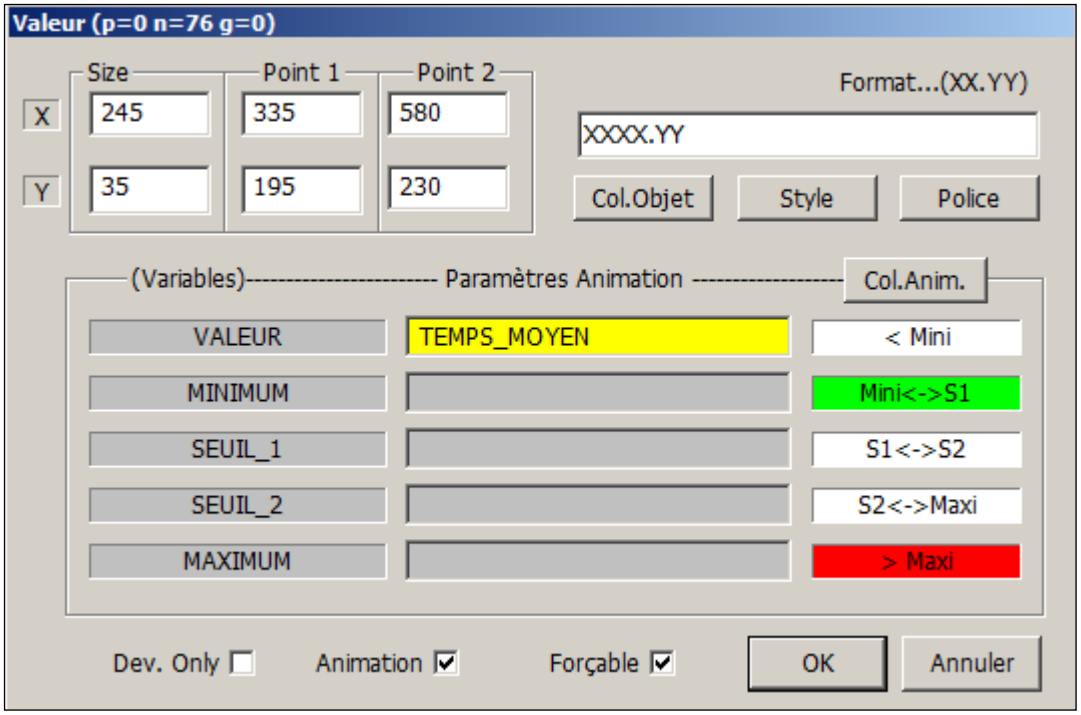
Raccourcis clavier

. **[HOME]** : déplacement de l'écran de visualisation à l'origine.
. **[END]** : déplacement de l'écran de visualisation sur l'objet le plus éloigné.
. **[ESC]** : annule toutes les sélections d'objets.
. **[CTRL][F]** : recherche de texte dans les intitulés et symboles liés aux objets graphiques.
. **[F3]** : recherche de texte suivante.
. **[CTRL][INS] ou [CTRL][C]** : fonction copier dans le presse-papier.
. **[SHIFT][INS] ou [CTRL][V]** : fonction coller depuis le presse-papier.
. **[CTRL][W]** : fonction coller sans animation.
. **[SUPPR] ou [CTRL][X]** : copie et suppression des objets sélectionnés.
. **[CTRL][D]** : fonction dupliquer.
. **[CTRL][I]** : fonction dupliquer avec incrément des variables (derniers caractères).
. **[CTRL][Z]** : fonction « Undo » à 5 niveaux.
. **[<-][->][^][v]** : hors sélection, déplacement de l'écran de visualisation. Avec sélection(s), déplacement/dimensionnement des objets selon magnétisme et selon touche [SHIFT] (point 1) ou touche [CTRL] (point 2).
. **[M]** : active ou désactive le magnétisme.
. **[C]** : appelle la palette de réglage « Couleur objets ».
. **[S]** : appelle la palette de réglage « Styles ».
. **[P] ou [F]** : appelle la palette de réglage « Polices ».
. **[ESPACE] ou [A]** : modifie l'alignement des objets sélectionnés.
. **[O]** : bascule les objets sélectionnés en premier plan.
. **[B]** : bascule les objets sélectionnés en dernier plan.
. **[X]/[Y]/[Z]** : modification de la taille X ou Y des objets sélectionnés ou les 2 avec Z.
. **[V]** : modification du format d'affichage des valeurs sélectionnées.
. **[R]** : rotation d'un groupe d'objet selon le point central.
. **[0, 1..5]** : permet de modifier les couleurs d'animation 1 à 5 (0 pour toutes).

ANIMATIONS

Généralités

Chaque objet possède des caractéristiques d'animation propres. Celles-ci sont accédées par le double-clic sur objet, ou via le clic droit->Propriétés ou via le menu **Graphique->Propriétés** (ou F2 après sélection).



La boîte de dialogue " Propriétés Objets " donne accès à :

. l'intitulé de l'objet au format Unicode (voir Menu Principal Dictionnaire).
. ses 2 positions de fixation et/ou sa taille X, Y.
. ses attributs de couleur [C], styles [S] et police [P].
. ses couleurs et variables d'animation associées selon des règles prédéfinies (voir tableaux ci-après).
. son mode d'affichage en mode développement seulement (dev.only - voir configuration Application).
. la validation de l'animation (case à cocher).
. la validation du forçage (case à cocher) de la 1ere variable d'animation. Ce forçage sera obtenu cliquant sur l'objet en RUN (voir exploitation).

Dès lors qu'un objet est déclaré animé (case à cocher) et ses variables correctement renseignées, il sera dessiné à l'écran selon ses règles d'animation. On parlera alors d'objet " dynamique " en opposition à des objets " statiques ".

Sélection des couleurs

Un simple clic sur les rectangles affectés au couleur provoque l'apparition de la palette de couleurs.
Un double-clic sur l'une d'elles la sélectionne.
Un clic sur le bouton « Col.Anim » permet de remplacer l'ensemble des couleurs d'animation.
Dans l'éditeur graphique, les touches [1] à [5] permettent de sélectionner directement les couleurs 1 à 5 pour les objets sélectionnés. La touche [0] modifie toutes les couleurs.

Saisie des variables d'animation

- Les zones de saisie de variables peuvent :

. être laissées vierges (animation non retenue - **fond gris**).
. contenir des valeurs immédiates (0,1, ou valeurs numériques/réelles - **fond blanc**).
. recevoir des symboles issus de la base (**fond jaune**). Le symbole peut être **saisi directement ou** accédé dans la base par un simple clic sur son texte d'information (à gauche). Un **clic droit** sur la zone d'édition permet un choix parmi les 10 dernières saisies (menu flottant).

- Les types de variables indiqués ci-dessous doivent être respectés.
- Pour les **variables binaires** les opérations de complément, front montant ou descendant peuvent être utilisées :

. syntaxe pour le complément :	/#VARIABLE,
. syntaxe pour le front montant :	**P#**VARIABLE,
. syntaxe pour le front descendant :	**N#**VARIABLE,
. syntaxe du test de l'état 1 du bit de forçage :	**F#**VARIABLE

NOTE : les fronts montant et descendant sont vrais pendant seulement 1 tour de cycle.

- En exploitation ces variables seront animées.

Raccourcis lors de la sélection des variables dans les éditeurs

- Associées à la touche **[SHIFT]**, les touches [/], [P], [N], [F] apportent les fonctions complément, front montant, front descendant et test du forçage à la variable utilisée.
- Associées à la touche **[SHIFT]**, les touches [R], [S], [U] apportent les fonctions Set/Reset/Déforçage à la variable utilisée.
- Ces raccourcis sont disponibles via le **clic droit souris**.

Table des animations

[CF] = changement couleur de fond,
[CC] = changement couleur contour,
[PR3] = proportion selon règle de 3,
[POSXY] = mouvement en X et Y (**X+ à droite, Y+ en haut**),
[LGXY] = variation longueur en X et Y,
[R2D] = effet de rotation 2D,
[VAL] = affichage de valeurs (alpha ou courbe),
[IMG][WAV][VID] = spécifique aux images, sons, vidéos.
[SYM] = affichage du symbole de la variable
[DIS] = affiche ou cache les objets
[TXT] = changement / saisie de texte
[DIA] = Dialog Box

Cuve - Pompe-Moteur - Vanne – Vérin - Convoyeur - Mobile - Rotation 2D - Robot - Distributeur
Tuyau-Pupitre TOR – Valeur-Pupitre NUM - Mécanique - Surface
Texte – Notepad – ListBox – Dialog Box – Trait - Bézier– Courbe
Rempli – Son - Image – GIF animé – Vidéo – Objet 3D - Table - Groupe

Vannes / Valves

2 voies : 5 variables binaires + 4 couleurs - **3 voies** : 4 variables entières (idem Vérin)

[CF]	FDCF(NC)	FDCO(NC)	CTRF	CTRO	CLIGNO.	État
	0	1	-	-	-	Fermée
	1	0	-	-	-	Ouverte
	1	1	1	0	0 1	Manœuvre (fermeture)
	1	1	0	1	0 1	Manœuvre (ouverture)
	1	1	0	0	-	Manœuvre
	Autres cas...					Défaut

Les couleurs d'animation de la **vanne 3 voies** sont indiquées selon le % d'ouverture.

Pompes / Pumps — Moteurs / Motors

2 variables binaires + 2 variables entières + 3 couleurs

	MARCHE	INCIDENT (NO)	État
[CF]	0	0	Couleur Arrêt
[POSXY]	1	0	Couleur Marche
	-	1	Couleur Défaut

HORIZONTAL : déplacement de X pixels de l'objet
VERTICAL : déplacement de Y pixels de l'objet

Tuyaux / Pipes — Text

Pupitre / Control Panel (Binary)

3 variables binaires + 2 variables entières + 4 couleurs

	VISU 1 / FOND 1	VISU 2 / FOND 2	État
[CF]	0	0	Couleur Etat 0
[POSXY]	1	0	Couleur Etat 1
	0	1	Couleur Etat 2
	1	1	Couleur Etat 3

La **couleur SECONDAIRE** est utilisée pour les lampes et poussoirs

HORIZONTAL : déplacement en X de l'objet
VERTICAL : déplacement en Y de l'objet
CMD_MODE :
0 → l'objet est forcé en Pousser/Pousser
1 → l'objet est forcé par action continue avec la souris.
21 → l'objet est forcé à 1.
20 → l'objet est forcé à 0.
>=100 → l'objet est forcé par une impulsion de durée définie par la valeur en ms.
Notes :
- l'élément INTER_2P peut servir d'interrupteur 2 positions (avec trait oblique selon l'état de VISU 1).
- l'élément INTER_3P peut servir d'interrupteur 3 positions. La position du trait varie en fonction de l'état des variables : gauche si VISU1=1, droite si VISU2=1 ou centrale dans les autres cas.

Surface (tuyau)

0.3

Ratio Pixel (réel) : les dimensions X et/ou Y sont multipliées entre elles et par ce ratio pour produire une unité de surface affichée selon le format indiqué dans le titre (ex : 0.0).
Font Size (entier) **:** indique la taille de police pour l'affichage de la surface

Robot

4 variables entières

[POSX]
[R2D]

Affichage de la rotation 2D des axes selon les valeurs angulaires **AXE_1** et **AXE_2** exprimées en degrés (+ dans sens anti-horaire).
HORIZONTAL : déplacement de X pixels de l'objet
VERTICAL : déplacement de Y pixels de l'objet

Texte	Texte	Texte	Texte	Texte

	3 variables binaires + 2 variables entières + 4 couleurs
[CF][CC]	T_VISU_FOND : Affichage du texte selon la variable **VISU** (0/1) et selon le **FOND** (0/1)
[POSXY]	T_VISU : Affichage du texte selon la combinaison de variables **VISU1 et VISU2**
[TXT]	T_FOND : Affichage du fond selon la combinaison de variables **FOND1 et FOND2**
	CMD_MODE : **0** → l'objet est forcé en Pousser/Pousser **1** → l'objet est forcé par action continue avec la souris. **10 → le texte est modifié en cliquant directement sur l'objet**. **21** → l'objet est toujours forcé à 1. **20** → l'objet est toujours forcé à 0. **30** →affichage de l'heure (format du titre ci-dessous – ex : **HH:mm:ss**→12 :27 :45) **31** → affichage de la date (format du titre ci-dessous – ex : **dd-MMM-yy**→06-Oct.17) **>=100** → l'objet est forcé par une impulsion de durée définie par la valeur en ms. **HORIZONTAL** : déplacement de X pixels de l'objet **VERTICAL** : déplacement de Y pixels de l'objet

FORMAT TIME

h	Hours with no leading zero for single-digit hours; 12-hour clock
hh	Hours with leading zero for single-digit hours; 12-hour clock
H	Hours with no leading zero for single-digit hours; 24-hour clock
HH	Hours with leading zero for single-digit hours; 24-hour clock
m	Minutes with no leading zero for single-digit minutes
mm	Minutes with leading zero for single-digit minutes
s	Seconds with no leading zero for single-digit seconds
ss	Seconds with leading zero for single-digit seconds
t	One character time marker string, such as A or P
tt	Multicharacter time marker string, such as AM or PM

FORMAT DATE

d	Day of month as digits with no leading zero for single-digit days.
dd	Day of month as digits with leading zero for single-digit days.
ddd	Day of week as a three-letter abbreviation.
dddd	Day of week as its full name.
M	Month as digits with no leading zero for single-digit months.
MM	Month as digits with leading zero for single-digit months.
MMM	Month as a three-letter abbreviation.
MMMM	Month as its full name.
y	Year as last two digits, but with no leading zero for years less than 10.
yy	Year as last two digits, but with leading zero for years less than 10.
yyyy	Year represented by full four digits.
gg	Period/era string.

Notepad

	2 variables binaires + 3 variables entières
[DIS][CF]	**VISIBLE** : Affichage ou non du texte (fin de texte = caractère ¤)
[POSXY]	**COULEUR FOND** : Couleur (0..31) selon les palettes PROSIMUL
[TXT]	**HORIZONTAL/VERTICAL** : Déplacement sur l'écran de X/Y pixels

ListBox

	NOMBRE LIGNES (Int) :
[CF]	0 = pas d'affichage de texte (affichage du fond), pas de saisie possible
[POSXY]	1 = 1 ligne affichée, pas de saisie possible
[TXT]	N(>=2) = 1 ligne affichée, saisie possible avec N items visibles si la case forçage est active
	LIGNE AFFICHÉE (Bool / Int) : n° de la ligne qui est affichée depuis le fichier texte (<0 : pas d'affichage) indiqué par le titre.
	FOND ou BORD/TEXTE(Bool / Int) : selection de la couleur de fond 0 à 4
	HORIZONTAL/VERTICAL (Int) : Déplacement sur l'écran de X/Y pixels

17.9 Valeurs / Values

Pupitre / Control Panel (Numeric)

	5 variables (tous types) + 5 couleurs
	La **VALEUR** est affichée selon le format retenu (ex : 0.00)
[CF][VAL]	Formats spéciaux : binaire : 2#xxxxxxx, hexa :16#xxxx, Time : T#format
[SYM]	Pour le format T# la valeur de temps est exprimée en millisecondes à partir de 0h.
	La couleur du fond d'affichage est fonction des **MINI, MAXI et SEUILS**
	La valeur de l'objet « **VALEUR_T2 »** peut être saisie directement en cliquant sur l'objet et les limites MINI et MAXI des valeurs saisies sont testées.
	La valeur de l'objet « **CURSEUR »** peut être pilotée par les flèches gauche et droite ou le mouvement du curseur ou le clic sur la valeur.
	Pour les objets « AnalogTxt » et « Curseur » le symbole peut être retiré de l'affichage en introduisant le carcatère « - » en fin de fromat.

Dialog Box

[DIA]	5 variables tous types + 5 couleurs Possibilité de créer une boîte de dialogue personnalisée via un fichier texte .TXT. Format du fichier TXT : voir ci-dessous (utilitaire Excel disponible sur demande)

FORMAT du fichier texte (.TXT) pour l'objet CUSTOM Dialog Box
Une ligne en tête pour le descripteur de box.
Code Box : 33150/33151 = séparateur '**,**' 32150/32151 = séparateur Tabulation
33150/32150 : positions des contrôles indiqués dans la fenêtre
3315**1**/3215**1** : positions des contrôles indiqués dans le cadre de la fenêtre, taille horizontale automatique (pleine largeur) si <=0.
PosX / PosY = position sur écran. -1=positionnement automatique à côté du point de clic.

Ligne 1	Caractéristiques générales de la boîte								
	Code Box	PosX (écran)	PosY (écran)	Largeur (écran)	Hauteur (écran)	Titre			
Ligne N	Définition des objets (3 lignes par objet)								
	Type contrôle	PosX (box)	PosY (box)	Largeur (box)	Hauteur (box)	Text0	Text1	Text2	Text3
Ligne N+1	Attributs d'affichage					Couleurs de fond			
	Font Size	**Fore** Color (0..31)	**Border** (0 / 1)	**Align** |0_1_2|	**Bold** / Gras (0 / 1)	**C0** (0..4)	**C1** (0..4)	**C2** (0..4)	**C3** (0..4)
Ligne N+2	**Visible**		**Enable**		**State**		**Valeurs**		
	VarV (-1,0..4)	Default (0 / 1)	**VarE** (-1,0..4)	Default (0 / 1)	**Var1** (-1,0..4)	**Var2** (-1,0..4)	**Val1** Min	**Val2** Max	

Type contrôle : 50=Texte, 51=Valeur sans saisie, 60/62=Bouton Texte sans/avec couleur, 61/63=Bouton Valeur sans/avec couleur, 64=Bouton avec icône, 70=Bouton Radio, 72=Case à cocher, 81=Zone de saisie numérique, 90=Boîte de listes de texte, 101=Bargraphe, 103=Échelle avec curseur.

Text0…3 : textes ou noms des fichiers .bmp (code 64) affichés selon l'état de State.
Var V/E/S : n° de variable d'animation (0 à 4).
Visible : si VarV vraie (1) le contrôle est visible. Si -1 prise en compte de **Default**.
Enable : si VarE vraie (1) le contrôle est modifiable. Si -1 prise en compte de **Default**.
State : si utilisées, l'**état** est (Var 1 + Var 2) pour les variables de type numérique ou (Var 1 + 2* Var 2) pour les variables de type booléen. Si -1 prise en compte de **Val 1/Min**.
Pour les contrôles texte (codes 50, 60, 90) **Var1/2** pilote et/ou reçoit le n° de texte affiché.
Pour les contrôles valeur (codes 51, 61, 81, 101, 103) l'état est affiché ou dessiné ou transmis (81).

Valeurs :
- pilotages (codes 70, 72) : valeurs Val1 et Val2 affectées à Var1 et Var2 si utilisées
- saisies numériques (code 81) : mini/maxi des saisies
- bargraphes (101) et échelles (103) : mini/maxi des affichages

Cn : couleurs de fond sélectionnées (0..4) parmi les couleurs configurées (animation).
Pour les contrôles de type numériques (51, 61, 81, 101, 103) :
- État < Min →**C0**, Min <= État <= Max → **C1**, État > Max → **C2**, Enable=false → **C3**.

Pour les contrôles de type texte, la couleur sélectionnée Ci dépend de l'**état** obtenu avec Var1 et Var2 (0..3 - modulo). Si **Enable** est à faux c'est la couleur de « Window » (4) qui est affichée.

Cuves
Tanks

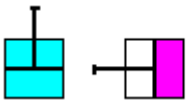
[CF][PR3]	5 variables (tous types) + 5 couleurs La **HAUTEUR** est affichée en proportion du **MAXIMUM** indiqué. La couleur d'affichage est fonction des **SEUILS** retenus ou non. **Si le commentaire commence par le caractère « | » la cuve est fermée.**

Cuves
Tanks (rotative)

[CF][PR3] **[R2D]**	2 variables tous types + 3 variables entières + 5 couleurs Le **VOLUME** est affiché en proportion du **VOLUME MAX** indiqué. La couleur d'affichage est fonction des **%** du volume p/r au maxi. L'angle de cuve est fixé par **FORM ANGLE EXT** (Int). L'angle de rotation de la cuve est fixé par **FIX ANGLE** (Int). Le point de rotation est calculé ainsi (en valeurs entières) : PR.X = MilieuObjetX + OFFSET_XY/10000 – 1000 PR.Y = MilieuObjetY + OFFSET_XY%10000 – 1000 Ex: OFFSET_XY = 10001000 ➔ point de rotation au centre du convoyeur Ex: OFFSET_XY = 09001000 ➔ point de rotation décalé de 100 pixel à gauche

Pupitre / Control Panel (Numeric)

[CF][PR3]	4 variables entières ou réelles Dessin du bargraphe selon la **VALEUR** proportionnellement aux **MINIMUM** / **MAXIMUM** et selon le nombre de quartiers **N-PART** choisi. La valeur de l'objet BARGRAPHE peut être pilotée sur un de ses quartiers. La raz est obtenu en cliquant sur le dernier quartier supérieur actif.



Vérins / Cylinders

[CF][PR3] **[POSXY]**	4 variables entières + 2 couleurs Dessin de la tige en fonction de la valeur **TEMPO** et proportionnellement au **MAXIMUM**. La couleur entrée ou sortie s'affiche selon le sens de manœuvre. **HORIZONTAL / VERTICAL** : déplacement de X / Y pixels de l'objet (Droite+ ; Haut+)

Mobiles

[CF][PR3] **[POSXY]**	5 variables entières + 3 couleurs La **HAUTEUR** est affichée en proportion du **MAXIMUM** indiqué. La couleur d'affichage est fonction du **SEUIL** retenu ou non. Pour les mobiles RX et RY la couleur SECONDAIRE est utilisée pour remplir de la HAUTEUR courante jusqu'au MAXIMUM **HORIZONTAL / VERTICAL** : déplacement de X / Y pixels de l'objet (Droite+ ; Haut+)

Distributeur

2 variables binaires + 2 variables entières

[POSX] Selon les commandes **GAUCHE** (tous types) et **DROITE** (bi-stables) le distributeur se décale à droite (tous types) ou à gauche.
HORIZONTAL : déplacement de X pixels de l'objet
VERTICAL : déplacement de Y pixels de l'objet

Traits / Lines

5 variables entières

[CF][CC]
[POSXY]
[LGXY]

HORIZONTAL et **VERTICAL** contiennent un nombre de pixels (X+ à droite et Y+ en haut) pour faire varier la position du trait selon le *type d'animation* retenu.
COLOR AFFICHE contient la couleur d'affichage du trait (0...15). -1 pour non affichage.
COLOR EFFACE contient la couleur d'effacement à l'ancienne position (0...15 - pour permettre des effets de mouvement par exemple). -1 pour non affichage.

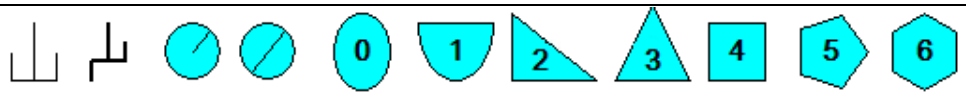
TYPE ANIMATION	Point 1		Point 2	
	HORIZONTAL	**VERTICAL**	**HORIZONTAL**	**VERTICAL**
0	+	+	+	+
1 (rotation)	Fixe	Fixe	+	+
2	-	-	+	+
3 (pont-roulant)	+	Fixe	+	+
4	Fixe	+	+	+
5	+	+	-	+
6	+	+	+	-

Cote (trait)

Ratio Pixel (réel) : les dimensions X et/ou Y sont multipliées entre elles et par ce ratio pour produire une unité de surface affichée selon le format indiqué dans le titre (ex : 0.0).
Delta X1 Delta Y1 (entier) : offset du point 1.
Delta X2 Delta Y2 (entier) : offset du point 2.
Note : par défaut les cotes sont affichées avec des pointes aux extrémités. Elles peuvent être retirées en mettant un « - » en début et/ou en fin de format dans le titre (ex : -0.0-).

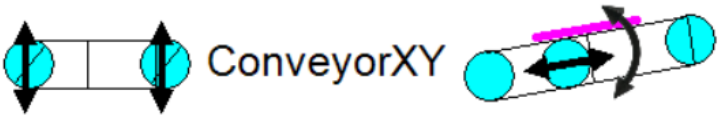
Bezier (trait)

Affiche une courbe de Bézier (2 points de liaison + 2 points de contrôle).
Pour régler les points de contrôle sélectionner l'objet, appuyer sur espace pour les faire apparaître ou disparaître et bouger les points avec les flèches combinées avec [SHIFT](p1) ou [CTRL](p2).
Offset AX1 / AY1 / BX1 / BY1 (entier) : offset des points de contrôle 1 et 2.



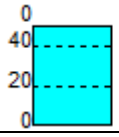
Rotation 2D

Rotation2D	4 variables entières
[R2D]	**ANGLE ROTATION** contient la valeur angulaire en degré pour l'effet de rotation.
[POSXY]	**HORIZONTAL** : déplacement de X pixels de l'objet **VERTICAL** : déplacement de Y pixels de l'objet **MODE CONTOUR** : 0=contour de cercle affiché, 1=contour non affiché Ou **TYPE_ANIM** : défini la forme géométrique de l'objet (voir ci-dessus)



ConveyorXY ConveyorTêta

	5 variables entières
[R2D]	**ANGLE ROTATION** contient la valeur angulaire en degré pour l'effet de mouvement
[POSXY]	**Type XY** : les points P1 et P2 sont déplacés des valeurs **Delta_X1/2** et **Delta_Y1/2**.
[R2D] **X 2**	**Type Têta** **FIX POINT(%)** : point de rotation du convoyeur exprimé en % de sa longueur **FIX ANGLE(Deg)** : angle de rotation du convoyeur p/r au point fixe **PRODUCT START/END(%)** : produit sur le convoyeur exprimé en % de la longueur **Product Color** + **Layer2..5** : si différent de la couleur de page, l'épaisseur du produit varie de 1 à 5 multiplié par l'épaisseur de l'objet



Courbes / Curves

	5 variables (tous types) + 3 couleurs
[VAL]	**ABSCISSE** contient la position d'affichage en pixels des points par rapport à l'origine (modulo la longueur de l'axe à l'écran). Lorsque la limite est atteinte les courbes sont décalées et les nouvelles graduations d'abscisse sont affichées. Pour le type « COURBE_AUTO » cette abscisse est convertie en heure selon la base de temps **BT (ms)** fournie. **VALEUR 1/2/3** contiennent les coordonnées des points à afficher en fonction du **MAXIMUM** indiqué (règle de 3 rapportée à la taille des ordonnées à l'écran - uni ou bipolaire). Chaque valeur dispose de sa propre couleur d'affichage. Lors d'activation de page, les 1000 dernières valeurs disponibles sont affichées. Pour le type « COURBE_AUTO » les valeurs Maxi et Mini et le nombre d'ordonnées (10 par défaut) sont indiqués dans le libellé sous la forme « **Maxi/Mini/NbOrdonnées** » (ex : 100.0/-100.0/5). Pour le type « COURBE_PC » les valeurs sont passées en 0..100% et les ordonnées ne sont pas affichées. Le libellé (ex : 2/1) indique le nombre de traits d'ordonnées (2) affichés et si les abscisses sont affichées (1) ou non (0) en bas de courbe.

Sons / Sounds

Son **[WAV]**	<u>1 variable binaire</u> **LANCEMENT** : sur passage à 1 (front montant) de la valeur, le fichier .WAV associé est joué en arrière-plan.

Images / Pictures

[IMG] **[POSXY]** **[LGXY]** **[R2D]**	<u>5 variables numériques ou réelles</u> **HORIZONTAL** : déplacement de X pixels de l'objet **VERTICAL** : déplacement de Y pixels de l'objet **RATIO_X.Y** : contient les facteurs d'agrandissement/réduction de l'image en horizontal et vertical (4 digits par axe – Ratio X sur poids forts). **Saisir la valeur 0100 0100** par défaut pour afficher l'image à 100%. **POINT ROTATION** : valeur de 0 (vide) à 8 (voir schéma ci-dessous) **ANGLE ROTATION** : indique l'angle de rotation (+ = sens anti horaire) en degré de l'image (selon le point de rotation indiqué). **Note** : les fichiers BMP/GIF/JPG/PNG/TIF/EMF sont acceptés.

GIF animés

[IMG]	<u>2 variables binaires + 1 variable réelle + 2 variables réelles/entières</u> **MARCHE** (binaire) : variable de pilotage sans effet sur l'animation. **ETAT** (binaire) : si à 1, le GIF s'anime au rythme de l'intervalle de temps indiqué. **INTERVALLE** (réel) : indiqué en seconde, il rythme l'affiche des images **IMAGE NUM** : si ETAT = 0, il permet de piloter les images affichées. La valeur -1 efface le GIF. Si ETAT =1, il reçoit la valeur de l'image en cours d'affichage (optionnel). **CMD MODE** : indique si la commande MARCHE agit en pousser/pousser (0) ou en appui maintenu (1).

VIDEO

[VID]

5 variables entières + nom de la vidéo.AVI ou .WMV

ID : n° 0 à 9 du lecteur vidéo utilisé. Un même lecteur peut être utilisé pour plusieurs vidéos (voir la fonction VideoLoad).

ETAT : variable de pilotage / retour du lecteur

b0 : initialise le lecteur « Pro32Video.EXE » sur front montant ou change de fichier
b1 : état Play (1) / Stop (0) du lecteur
b2 : 1=redémarrage automatique au début en fin de vidéo
b3 : 1=commande de rembobinage manuelle au début
b4 : 1=modifie la position temporelle du lecteur (avec paramètre TIME)
b5 : 1=modifie la position / le ratio d'affichage (avec paramètres SCREEN et RATIO)
b6 : 1= fermeture du lecteur
b7 : bit de cadencement (facultatif). Utilisé pour recevoir le temps et la position du lecteur
b8 : 1=retour lecteur initialisé
b9 : retour état Play (1) / Stop (0)
b10 : 1=fin de lecture de la vidéo
b12 : 1=retour temps lecteur disponible
b13 : 1=retour position lecteur disponible. Utiliser les fonctions GetSX, GetSY, …

SCREEN*10 + POS : numéro d'écran du lecteur * 10 + position de la fenêtre

0 : position indiqué par l'objet PROSIMUL
1 : coin haut gauche de l'écran — 2 : coin haut droit
3 : coin bas gauche — 4 : coin bas droit
5 : centré sur l'écran — 6 : plein écran
+100 : affichage avec bordures — +200 : avec bordures et barre de lecture
Ex : W_VIDEOPOS=125 → vidéo avec bordures sur écran 2 et centré

RATIO : indique le facteur de réduction/agrandissement en **pourcentage**

TIME : indique ou reçoit la position temporelle du lecteur exprimé en **1/10ème seconde**

Exemple: 1 lecteur pour 2 vidéos

```
If Fe B_VIDEO2 Then : W_EXEC = VideoLoad ("PageVideos" ; "Wildlife" ) : End_If
If Re B_VIDEO2 Then : W_EXEC = VideoLoad ("PageVideos" ; "Test" ) : End_If
W_VIDEO_STATE . 0 = Re B_VIDEO_PLAY
W_VIDEO_STATE . 1 = B_VIDEO_PLAY
W_VIDEO_STATE . 2 = 1
W_VIDEO_STATE . 3 = Re B_VIDEO_REWIND
If Fst W_VIDEO_TIME_1\10S Then : W_VIDEO_STATE . 4 = 1 : UnF W_VIDEO_TIME_1\10S : End_If
If Fst W_VIDEO_RATIOPC Then : W_VIDEO_STATE . 5 = 1 : UnF W_VIDEO_RATIOPC : End_If
W_VIDEO_STATE . 6 = B_VIDEO_EXIT
W_VIDEO_STATE . 7 = SYS_TOP_DIXIEME And B_VIDEO_PLAY
If W_VIDEO_STATE . 13 Then : W_VIDEO_WIDTH = GetSX ("PageVideos" ; "Wildlife" ) : End_If
```

NOTE: le lecteur Pro32Video.EXE peut être utilisé de manière autonome avec les paramètres suivants :

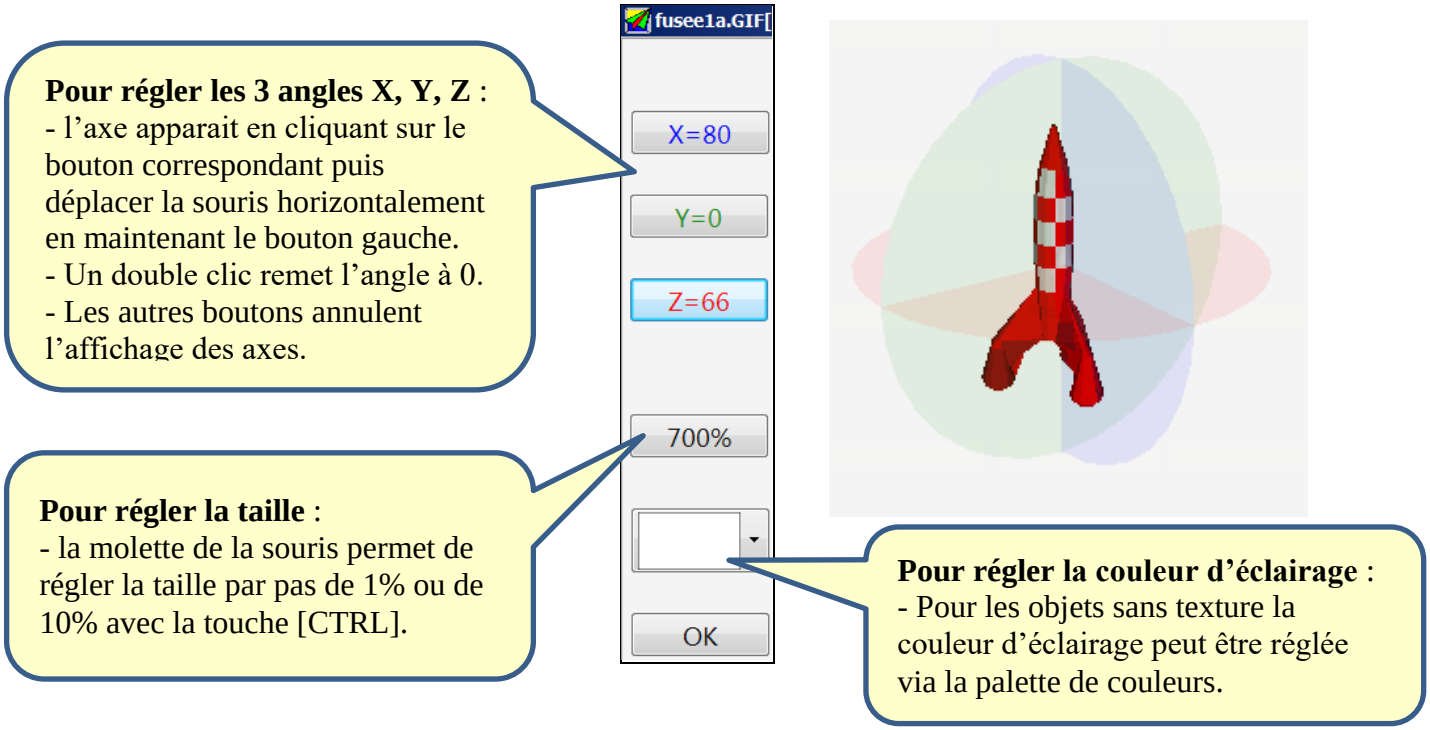
1er paramètre obligatoire : nom du fichier vidéo (.AVI ou .WMV) lancé.
ID0…ID9 : numéro de l'ID si le lecteur est utilisé par PROSIMUL
M1 M5 : numéro d'écran — P0 P6 : position sur l'écran (selon définitions de SCREEN-POS)
%vvv, Xvvv, Yvvv : ratio d'affichage et position X,Y sur l'écran

❑ OBJET 3D

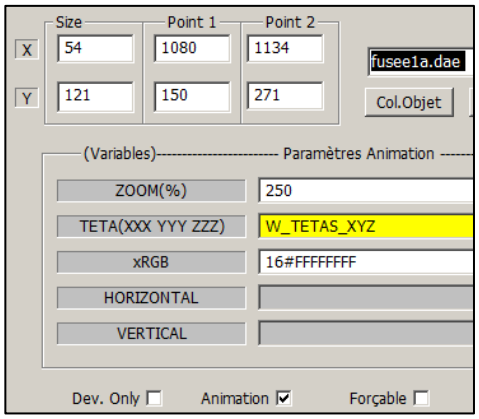
Indiquer le nom du fichier sélectionné (sur le répertoire de l'application).

- ⇨ Formats acceptés : .ASE, .DAE (Collada) et .OBJ.
- ⇨ Dimensions maximum : 1024 pixels X 768 pixels
- ⇨ Couleurs : 32 bits.

L'incorporation d'un objet 3D provoque le lancement d'une instance du viewer Pro3D.exe. En STOP, la taille et les angles (X,Y,Z) initiaux de l'objet sont réglés dans le viewer et échangés avec PROSIMUL. Utiliser des fichiers d'objets simples sous peine de temps d'affichage importants.

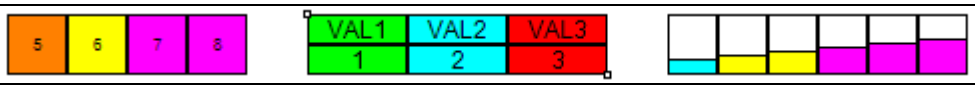


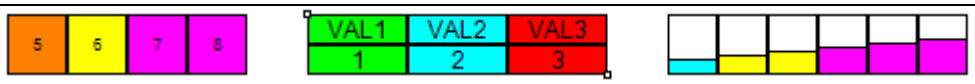
En RUN, l'animation du mobile est réglée via 5 variables :
- **ZOOM(%)** : récupère ou transmets la valeur de grossissement de l'objet en % (100%=taille de chargement)
- **TETA (XXX YYY ZZZ)** : les 3 angles sont stockés dans un seul mot (3 digits par angle – Ex : 90 000 000 → X=90° - Y et Z = 0°)
- **xRGB** : contient la couleur d'éclairage sous forme de 4 octets [Transparence][Red][Green][Blue] (ex : 16#FFFFFFFF = blanc)
- **HORIZONTAL** et **VERTICAL** permettent le déplacement de l'objet sur l'écran (en pixels).




```
If SYS_INIT_APPLI Then
    W_ANGLE_X = W_TETAS / 1000000 :
    W_ANGLE_Y = ( W_TETAS % 1000000 ) / 1000
    W_ANGLE_Z = W_TETAS % 1000
End_If
If Re SYS_TOP_DIXIEME Then :  W_TETAS = W_ANGLE_X * 1000000 + W_ANGLE_Y * 1000 + W_ANGLE_Z : End_If
```

NOTE: le « viewer » Pro3D.exe peut être utilisé de manière autonome avec les paramètres suivants : 1er paramètre obligatoire : nom du fichier 3D (.ASE, .DAE ou .OBJ). Cela permet de vérifier la capacité d'affichage dans PROSIMUL.

FILL / VALUE		
[CF][VAL]	<u>1 variable Bool/Int + 1 variable Int/Float + 3 variables Int</u> **REFRESH** : sa variation permet le rafraîchissement du tableau. **TABLE** (LEN=First Element) : position du 1er élément de la table. Ce **1er élément** doit contenir la **longueur** de la table qui le suit (tables de type INT ou FLOAT). Les éléments de la table sont affichés selon les couleurs choisies et les paramètres **MINIMUM / SEUIL / MAXIMUM** (valeurs affichées pour table « Value »).	

SYMB BAR TETA		
[CF][VAL] [R2D]	<u>1 variable Bool/Int + 1 variable Bool/Int/Float + 3 variables Int</u> **REFRESH** : sa variation permet le rafraîchissement du tableau. **TABLE START** : 1er élément de la table. **TABLE LENGHT** : nombre d'éléments de la table. Les données de la table sont affichés selon les couleurs choisies et les paramètres **SEUIL1 / SEUIL2** pour les tables **SYMB** et **BAR** et par pas de 20% dans les couleurs choisies pour la table **TETA**. Pour les tables **BARH** / **BARV** / **TETA**, l'affichage dépend des valeurs **Maxi** et **Mini** saisies dans le titre. Ex1 : 100.0/0.0 Ex2 : 0.5/0.0 Ex3 : 100.0/-100.0 Les tables **TETA** admettent un point de rotation **OFFSET XY** et un angle de rotation **ANGLE(°)**. Le point de rotation est calculé ainsi (en valeurs entières) : PR.X = MilieuObjetX + OFFSET_XY/10000 – 1000 PR.Y = MilieuObjetY + OFFSET_XY%10000 – 1000 Ex: OFFSET_XY = 10001000 ➔ point de rotation au centre du convoyeur Ex: OFFSET_XY = 09001000 ➔ point de rotation décalé de 100 pixel à gauche	

	Rempli / Fill
[CF]	<u>5 variables entières</u> **COULEUR FOND** contient la valeur de la couleur (0 à 15) d'affichage du remplissage dans le style retenu. La couleur de contour est la couleur d'arrêt du remplissage. **CADRE** : **0 ou non renseigné** : Le remplissage s'arrête sur détection de bordure. **1** : Une bordure rectangulaire est dessinée et remplie. **2** : Une bordure circulaire est dessinée et remplie. **3** : Une bordure étoilée est dessinée et remplie. **10** : Le remplissage suit la couleur sur le centre du remplissage (surface). **OFFSET COULEUR** : (a) si renseignée valeur retranchée à COULEUR FOND. **MODULO COULEUR** : modulo effectué après opération (a). **MINI COULEUR** : valeur additionnée en final. → couleur finale = MINI + mod((COULEUR FOND – OFFSET) , MODULO)

BIELLE	Mécanique
[R2D] **[POSXY]**	<u>1 variable numérique + 2 variables numériques en sortie</u> **ANGLE ROTATION** : angle de l'arbre entraînant. **X1**(Output) / **X2**(Output) / **Y1**(Ouput) : retour des valeurs de déplacement circulaire (X1,Y1) et de translation (X2). La course totale est obtenue par X1+X2.

LIAISON	Mécanique
[R2D] **[POSXY]**	<u>4 variables numériques + 1 variable numérique en sortie</u> **Delta X1 Delta Y1** : offset du point 1 (X+ = droite écran, Y+ = haut écran). **Delta X2 Delta Y2** : offset du point 2. **LEN**(Output) : valeur en retour de la longueur de la liaison.

CHARNIERE	Mécanique
[R2D] **[POSXY]**	<u>4 variables numériques + 1 variable numérique en (entrée/sortie)</u> **Delta X1 Delta Y1** : offset du point 1 (X+ = droite écran, Y+ = haut écran). **Delta X2 Delta Y2** : offset du point 2. La variable LEN (In/out) contient initialement la longueur totale de la charnière. Le signe de la longueur détermine l'angle de la charnière. Elle sera écrite par PROSIMUL en cas de défaut de longueur de charnière lors de son déplacement.

	GROUPE
[DIS] **[POSXY]**	<u>1 variables binaire + 2 variables entières</u> **CACHE** (binaire) : variable de pilotage pour cacher les objets groupés. **HORIZONTAL** : déplacement de X pixels des objets. **VERTICAL** : déplacement de Y pixels des objets.

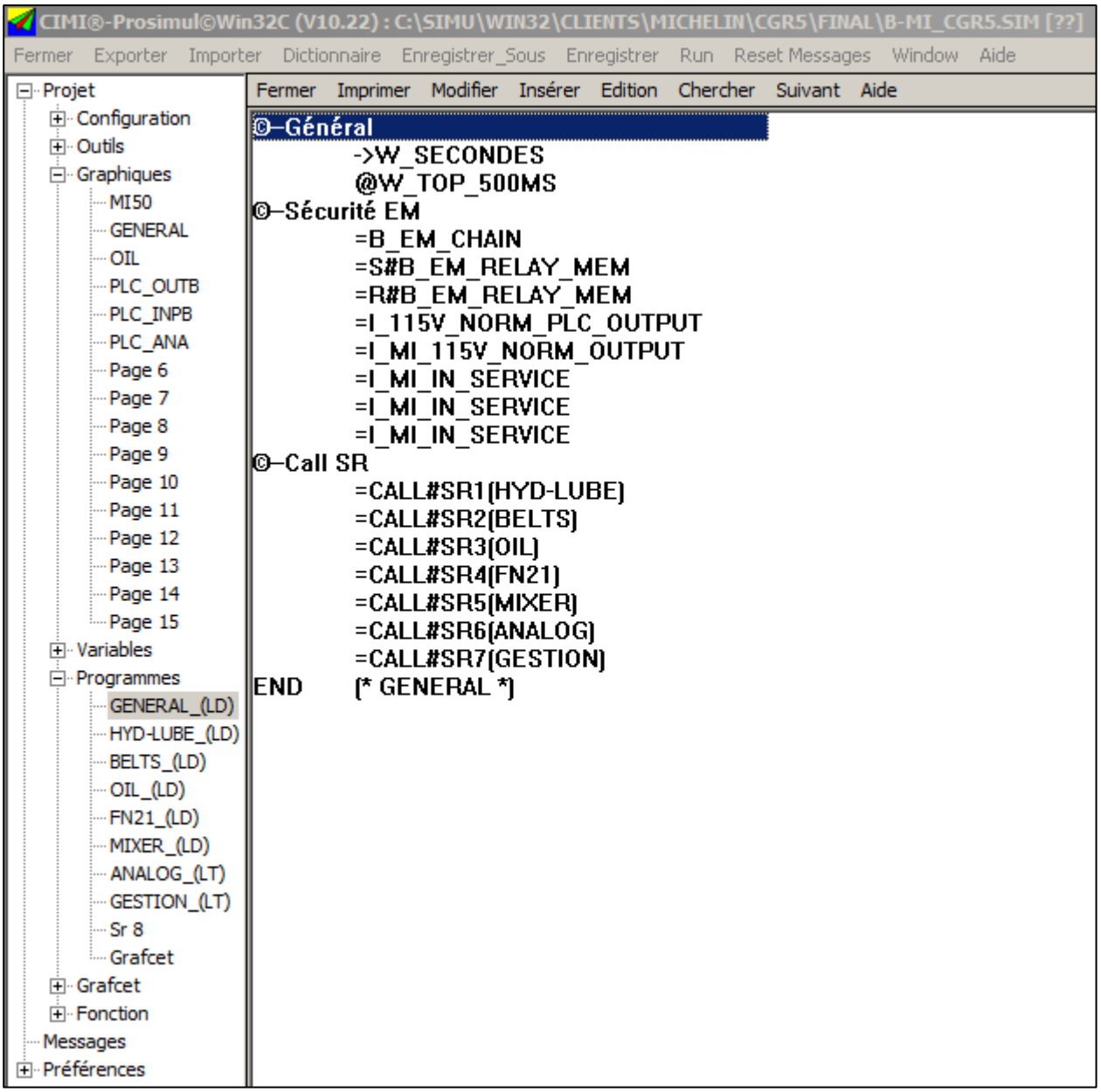
EDITEUR LADDER

Cet éditeur met à disposition un jeu d'instructions simples (temporisation, compteur, équation logique, bloc calcul) qui peuvent être saisies à travers 1 programme Main, 15 sous-routines et 1 programme spécifique au traitement des transitions/actions liées au Grafcet (si nommé « Grafcet »).

Les programmes sont visualisés par défaut en mode liste ou en instructions détaillées.

Chaque programme peut recevoir un intitulé depuis le navigateur application.

Au maximum **512 instructions** par programme.



LADDER : MODE LISTE

Menu

Imprimer : permet d'imprimer sous forme graphique l'ensemble des instructions **sélectionnées**.

Modifier : entrée dans l'écran de modification de l'instruction sélectionnée dans la liste (mode affichage détaillé). Un **double-clic** sur une instruction provoque le même effet.

Insérer : permet l'insertion d'une instruction vierge devant l'instruction sélectionnée.

Edition -> Couper : provoque la copie puis l'effacement des instructions sélectionnées.
Edition -> Copier : provoque la mise en tampon des instructions sélectionnées. Les instructions copiées pourront être exportées .
Edition -> Coller : provoque la recopie des instructions mémorisées devant la 1ere position sélectionnée.
Edition -> Dupliquer+ Incrément : provoque la duplication des instructions sélectionnées avec incrément des variables (si possible). Par exemple : FDCS_VERIN_01 devient FDCS_VERIN_02 si le symbole existe consécutivement dans la base.
Edition -> Dupliquer + Suivant : provoque la duplication des instructions sélectionnés en utilisant les variables immédiatement suivantes dans la base.

Les fonctions Copier/Coller/Couper sont accessibles via le clic droit (menu contextuel).

Chercher : permet la recherche d'une chaîne de caractère dans les commentaires ou symboles des différentes instructions. Saisir la chaîne de caractère. La première instruction contenant la chaîne est sélectionnée (recherche dans tous les programmes Main et SR).

Suivant : poursuite de la même recherche.

Format d'Affichage des instructions

©-- « Texte du Commentaire »
LABEL **[T]** VARIABLE ou ACTION_RESULTANTE

[T] représente le type d'instruction

@	Temporisation
±	Compteur
=	Équation logique (3x3)
->	Bloc calcul
©	Commentaire seul

La dernière instruction END (*Fin de programme*) existe par défaut.

Les couleurs des instructions et la taille de police peuvent être réglés via les préférences du navigateur.

Sélection d'instructions

- Un simple clic sur une instruction provoque sa sélection. Les flèches de direction clavier et la molette de la souris permettent de se déplacer dans la liste. La molette associée à la touche [CTRL] est équivalent aux flèches gauche et droite.

- Un simple clic avec **[SHIFT]** sur une instruction provoque la sélection étend la sélection jusqu'à l'instruction choisie.

- Un simple clic avec **[CTRL]** permet une sélection multiple d'instructions non consécutives.

LADDER : SAISIE DES INSTRUCTIONS

Commentaire

Chaque instruction peut recevoir un commentaire de 28 caractères maximum. Ce commentaire est facultatif.

Label

Chaque instruction peut recevoir un de 4 caractères maximum utilisé pour les **JUMP**. Ce label est facultatif.

Saisie des variables

- Les zones de saisie de variables peuvent :

. être laissées vierges (état 0 - **fond gris**).
. contenir des valeurs immédiates (0,1, ou valeurs numériques/réelles - **fond blanc**).
. recevoir des symboles issus de la base (**fond jaune**). Le symbole peut être **saisi directement ou** accédé dans la base par un simple clic sur son texte d'information (au-dessus). Un **clic droit** sur la zone d'édition permet un choix parmi les 10 dernières saisies (menu flottant).
. recevoir des opérateurs, actions de saut ou d'appel de programme (**fond bleu**).

- Les types de variables indiqués pour les différentes instructions doivent être respectés.
- Pour les **variables binaires lues** les opérations de complément, front montant ou descendant peuvent être utilisées :

. syntaxe pour le complément :	**/#**VARIABLE,
. syntaxe pour le front montant :	**P#**VARIABLE,
. syntaxe pour le front descendant :	**N#**VARIABLE,
. syntaxe du test de l'état 1 du bit de forçage :	**F#**VARIABLE

NOTE : les fronts montant et descendant sont vrais pendant seulement 1 tour de cycle.

- Pour les **variables binaires écrites (résultats)** les opérations de Set/Reset de variables, RAZ du forçage, saut et appel de programme peuvent être utilisées :

. syntaxe Set :	**S#**VARIABLE,
. syntaxe Reset :	**R#**VARIABLE,
. syntaxe pour la RAZ du forçage :	**U#**VARIABLE,
. syntaxe saut de programme (*sur équation et bloc calcul*) :	**JUMP#LABEL**,
. syntaxe appel sous-programme (*sur équation et bloc calcul*) :	**CALL#SR1..15** **CALL#NOMPROG**.

- En exploitation ces variables seront animées.

Raccourcis lors de la sélection des variables dans les éditeurs

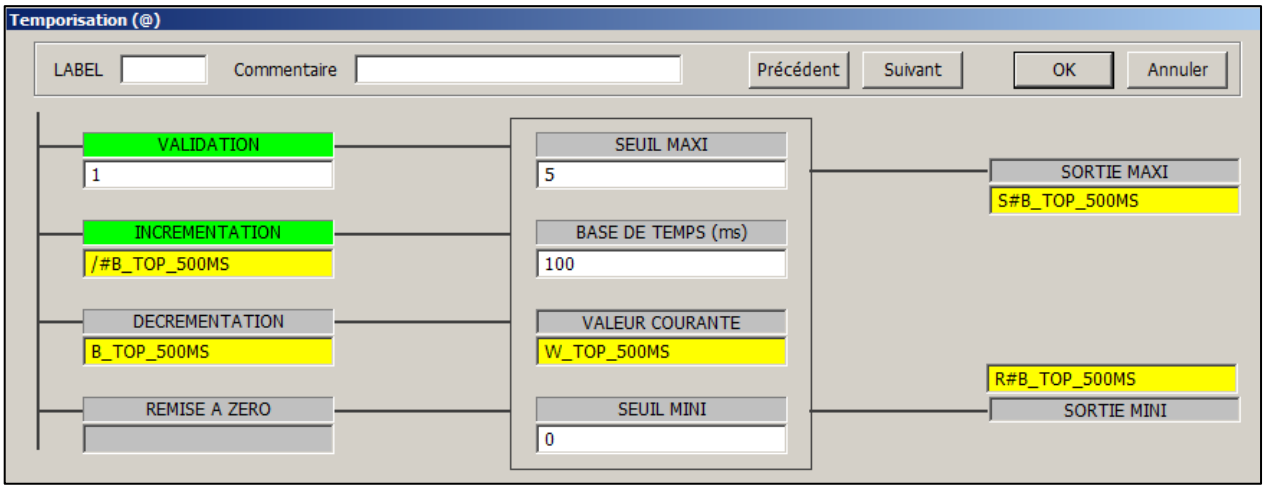
- Associées à la touche **[SHIFT]**, les touches [/], [P], [N], [F] apportent les fonctions complément, front montant, front descendant et test du forçage à la variable utilisée.
- Associées à la touche **[SHIFT]**, les touches [R], [S], [U] apportent les fonctions Set/Reset/Déforçage à la variable utilisée.
- Ces raccourcis sont disponibles via le **clic droit souris**.

Validation

- Les boutons [OK], [Suivant] et [Précédent] valident l'instruction en cours si tous les champs de variables sont correctement renseignés.

- Les boutons **[Suivant]** et **[Précédent]** provoquent de plus le passage aux instructions suivantes ou précédentes du mode liste.

TEMPORISATION



VALEUR COURANTE : variable entière écrite contenant la valeur de la temporisation.

BASE DE TEMPS : valeur de la base de temps en milliseconde.

SEUIL MAXI : valeur du seuil maximum de la temporisation (blocage au delà).

SEUIL MINI : valeur du seuil minimum de la temporisation (blocage en deçà).

VALIDATION : condition binaire autorisant le fonctionnement de la tempo ainsi que l'activation des sorties maxi et mini.

INCREMENTATION : condition binaire permettant l'écoulement positif du temporisateur.

DECREMENTATION : condition binaire permettant l'écoulement négatif du temporisateur.

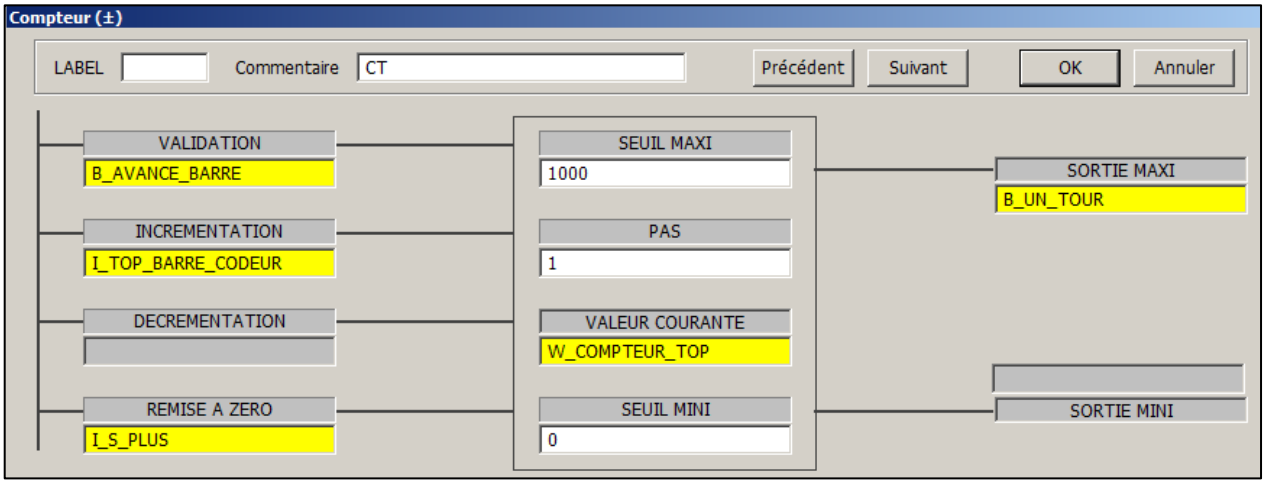
REMISE A ZERO : condition binaire provoquant une remise au seuil minimum de la valeur courante (prioritaire).

SORTIE MAXI : action effectuée ou variable binaire écrite à 1 si *valeur courante >= seuil maximum* et si *validation = 1*.

SORTIE MINI : action effectuée ou variable binaire écrite à 1 si *valeur courante <= seuil minimum* et si *validation = 1*.

NOTE : une temporisation peut compter plusieurs bases de temps entre 2 cycles PROSIMUL selon le temps de cycle en RUN (voir variables système*)*

COMPTEUR



VALEUR COURANTE : variable entière écrite contenant la valeur du compteur.

PAS : valeur d'incrément / décrément du compteur.

SEUIL MAXI : valeur du seuil maximum du compteur (blocage au delà).

SEUIL MINI : valeur du seuil minimum du compteur (blocage en deçà).

VALIDATION : condition binaire autorisant le fonctionnement du compteur ainsi que l'activation des sorties maxi et mini.

INCREMENTATION : condition binaire permettant l'incrément du compteur (le comptage est effectué *sur* ***front montant*** de la condition).

DECREMENTATION : condition binaire permettant le décrément du compteur (le décomptage est effectué *sur* ***front montant*** de la condition).

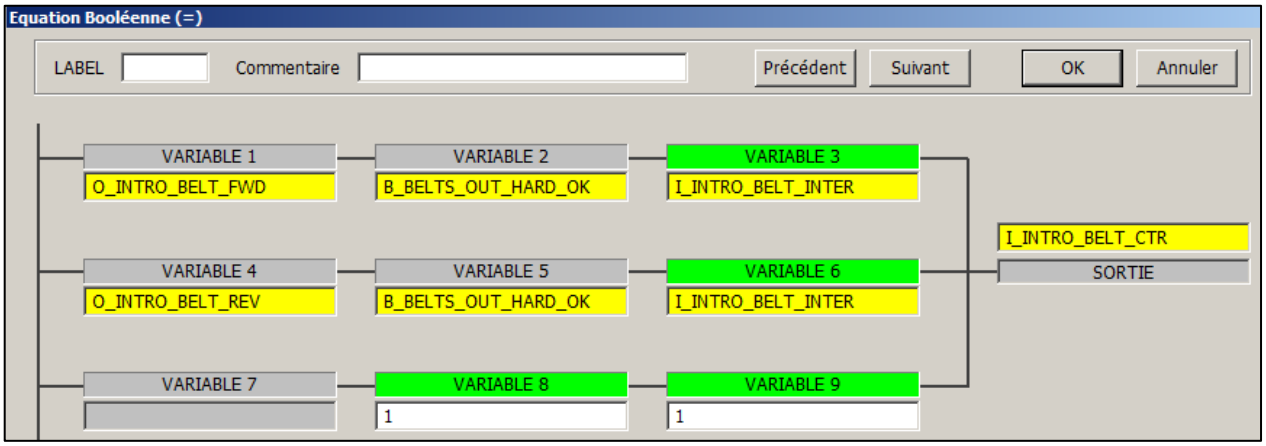
REMISE A ZERO : condition binaire provoquant une remise au seuil minimum de la valeur courante (prioritaire).

SORTIE MAXI : action effectuée ou variable binaire écrite à 1 si *valeur courante >= seuil maximum* et si *validation = 1*.

SORTIE MINI : action effectuée ou variable binaire écrite à 1 si *valeur courante <= seuil minimum* et si *validation = 1*.

EQUATION LOGIQUE

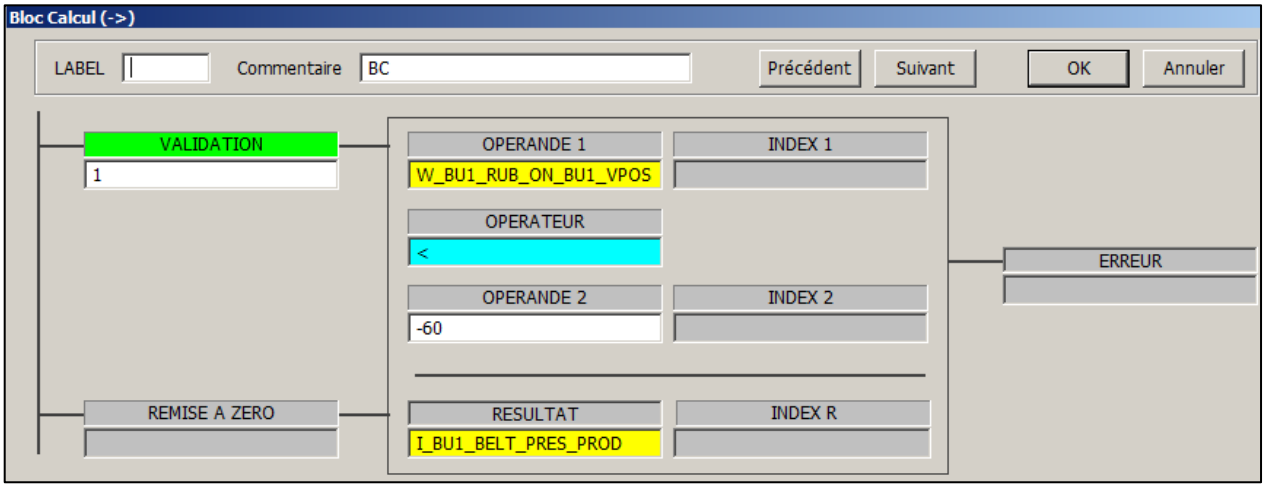
Equation booléenne de structure fixe formée de 6 fonctions “ ET ” et 3 fonctions “ OU ”.



VARIABLE 1 à VARIABLE 9 : conditions binaires. Pour les **conditions toujours vraies** saisir la valeur 1.

SORTIE : action effectuée ou variable écrite à 1 si l’équation logique est vraie.

BLOCS CALCUL



VALIDATION : condition binaire autorisant le fonctionnement du bloc calcul.

REMISE A ZERO : condition binaire provoquant le passage à zéro de la variable résultat.

ERREUR : action effectuée ou variable écrite à 1 en cas d'erreur de traitement.

INDEX 1, INDEX 2, INDEX R : si utilisées, ces variables contiendront les index respectifs des Opérande 1, Opérande 2 et Résultat. Les index sont *obligatoirement de* type entier.

NOTE : la valeur courante d'un ***index*** *agit en tant que déplacement d'adresse (+N position dans la base - normalement positif) par rapport à sa variable associée. Un contrôle de dépassement du nombre de variables accessibles (2048) est effectué. En cas d'usage (sauf cas particuliers indiqués) les positions dans la base des opérandes 1 ou 2 ou du résultat seront décalées des valeurs d'index*

OPERANDE 1, OPERANDE 2, RESULTAT : variables de différents types selon l'opérateur utilisé (Arithmétiques , Logiques , Mathématique , Débit-Air Humide , Comparaison , Tableaux , Conversion , Bit<->Mot , Informations).

Opérateurs et types des variables

[B] : valeur binaire
[M] : valeur entière ou réelle (mot)
[Me] : valeur entière 32 bits ou valeur réelle convertie en entier implicitement.
[Mr] : valeur réelle ou valeur entière convertie en réel implicitement.
[BM] : variables binaires ou mots (concordance de types à respecter normalement)

OPÉRATEURS ARITHMÉTIQUES	OP1, OP2, R	Opérateur LADDER/Littéral	Résultat
Addition	[M][M]→[M]	+	OP1 + OP2
Soustraction	[M][M]→[M]	-	OP1 - OP2
Multiplication	[M][M]→[M]	*	OP1 * OP2
Division	[M][M]→[M]	/	OP1 / OP2
Valeur absolue	[M][M]→[M]	ABS	ABS(OP1)*OP2
Maximum	[M][M]→[M]	MAX	Maximum de OP1 et OP2.
Minimum	[M][M]→[M]	MIN	Minimum de OP1 et OP2.
Modulo	[M][M]→[M]	MOD ou %	Reste de OP1 divisé par OP2.
Valeur aléatoire	[Me][Me]→[M]	RND	OP1 <= Résultat <= OP1+ OP2.
Test p/r à zéro	[M][M]→[M]	DELTAZERO	OP2 si OP1<OP2 ou –OP2 si OP1>OP2
Moyenne	[M][M]→[M]	MEAN	(OP1+OP2) / 2

OPÉRATEURS LOGIQUES	OP1, OP2, R	Opérateur LADDER/Littéral	Résultat
ET	[Me][Me]→[M]	AND	OP1 AND OP2
OU	[Me][Me]→[M]	OR	OP1 OR OP2
OU Exclusif	[Me][Me]→[M]	XOR	OP1 XOR OP2
Décalage Gauche	[Me][Me]→[M]	SHL	OP1 décalé à gauche de OP2 bits (0 introduits).
Décalage Droite	[Me][Me]→[M]	SHR	OP1 décalé à droite de OP2 bits (0 introduits).
Décalage Circulaire Gauche	[Me][Me]→[M]	ROL	OP1 décalé à gauche de OP2 bits (bits sortis réintroduits à droite).
Décalage Circulaire Droite	[Me][Me]→[M]	ROR	OP1 décalé à droite de OP2 bits (bits sortis réintroduits à gauche).

OPÉRATEURS MATHÉMATIQUES	OP1, OP2, R	Opérateur LADDER/Littéral	Résultat
Cosinus	[Mr][Mr]->[M]	COS	cosinus(OP1)*OP2 *(OP1 en degré)*
Exponentielle base e	[Mr][Mr]->[M]	EXPN	OP2 * e puissance(OP1/OP2)
Exponentielle base 10	[Mr][Mr]->[M]	EXP10	OP2 * 10 puissance(OP1/OP2)
Logarithme népérien	[Mr][Mr]->[M]	LOGN	OP2 * ln(OP1/OP2)
Logarithme base 10	[Mr][Mr]->[M]	LOG10	OP2 * log10(OP1/OP2)
Puissance	[Mr][Mr]->[M]	POW	OP1 à la puissance OP2
Sinus	[Mr][Mr]->[M]	SIN	sinus(OP1)*OP2 *(OP1 en degré)*
Racine Carrée	[Mr][Mr]->[M]	SQRT	OP2 * sqrt(OP1)
Hypoténuse	[Mr][Mr]->[M]	HYPO	Sqrt(Pow(OP1,2.0)+Pow(OP2,2.0))
Tangente	[Mr][Mr]->[M]	TAN	tan(OP1)*OP2 *(OP1 en degré)*
Arc Tangente PI/2	[Mr][Mr]->[M]	ATAN	atan(OP1)*OP2 $-90° \leq RES \leq +90°$
Arc Tangente PI	[Mr][Mr]->[M]	ATGPI	atan(OP1)*OP2 $-180° \leq RES \leq +180°$

OPÉRATEURS Air Humide	OP1, OP2, R	Opérateur LADDER/Littéral	Résultat
Pression vapeur (Pa)	[Mr][Mr]->[M]	PVAPEUR_PATM_RS PVapPatmRs	Pression vapeur pour une pression atmosphérique OP1 (Pa) et une humidité absolue OP2 (kg/kgas)
Pression vapeur saturée (Pa)	[Mr][Mr]→[M]	PSVAPEUR_TAH PSVapeurTah	Pression vapeur saturée pour une température d'air OP1 (°C). Formula OP2 : 0: Pow(10 ;2.7877+(7.625*OP1)/(241.6+OP1) 1: 6,11*Pow(10;7.5*OP1/(237.3+OP1)) 2: Exp(46.784-6435.0/(273.15+OP1) - 3.868*Ln(OP1+273.15))/750*1000
Pression vapeur (Pa)	[Mr][Mr]→[M]	PVAPEUR_TAH_HR PVapeurTahHr	Pression vapeur pour une température d'air OP1 (°C) et une humidité relative OP2 (%) (PSat avec formule 2)
Densité air sec	[Mr][Mr]→[M]	ROAIR_PAS_TAH RoAirPasTah	Densité de l'air pour une pression OP1 (Pa) et une température OP2 (°C)
Densité vapeur	[Mr][Mr]→[M]	ROVAPEUR_PV_TAH RoVapeurPvTah	Densité vapeur pour une pression vapeur OP1 (Pa) et une température OP2 (°C)
Humidité absolue (kg/kgas)	[Mr][Mr]→[M]	RS_PV_PAS RsPvPas	Humidité absolue pour une pression vapeur OP1 (Pa) et une pression d'air OP2 (Pa)
Enthalpie air humide (kJ/kg)	[Mr][Mr]→[M]	HS_TAH_RS HsTahRs	Enthalpie pour une température OP1 (°C) et une humidité absolue OP2 (kg/kgas)
Variation d'enthalpie (kJ/kg)	[Mr][Mr]→[M]	HK_DT_RS HkDtRs	Variation d'enthalpie pour OP1 (°C) degré d'écart et une humidité absolue OP2 (kg/kgas)
Température d'air (°C)	[Mr][Mr]→[M]	TAH_HS_RS TahHsRs	Température d'air pour une enthalpie OP1 (kJ/kg) et une humidité absolue OP2 (kg/kgas)
Température de rosée (°C)	[Mr][Mr]→[M]	TR_PV TrPv	Température de rosée pour une pression vapeur OP1 (Pa)
Delta pression(K,Q)	[Mr][Mr]→[M]	DELTAP	=OP1 * Pow(OP2;2) * sign(OP2)
Débit(K, DP)	[Mr][Mr]→[M]	FLOW	=OP1 * Sqrt(abs(OP2)) * sign(OP2)

OPÉRATEURS de COMPARAISON	OP1, OP2, R	Opérateur LADDER/Littéral	Résultat
Supérieur	[M][M]→[B]	>	Variable binaire vraie si OP1 > OP2
Égalité	[M][M]→[B]	=	Variable binaire vraie si OP1 = OP2
Inférieur	[M][M]→[B]	<	Variable binaire vraie si OP1 < OP2
Supérieur ou Egal	[M][M]→[B]	>=	Variable binaire vraie si OP1 >= OP2
Inférieur ou Egal	[M][M]→[B]	<=	Variable binaire vraie si OP1 <= OP2
Différent de	[M][M]→[B]	<>	Variable binaire vraie si OP1 <> OP2
Intervalle 1..2..5..10 (i)	[M][M]→[B]	?1 ?2 ?5 ?10	Variable binaire vraie si : OP2-i <= OP1 <= OP2+i
Proximité d'une valeur	[M][M]→[B]	><	Variable binaire vraie si : Abs(OP1)<=OP2

OPÉRATEURS sur TABLEAUX	OP1, OP2, R	Opérateur LADDER/Littéral	Résultat
Copie d'une valeur dans N variables	[BM][Me]→BM]	NCOP (Ladder seul)	OP1 -> RESULTAT[OP2]
Copie d'un tableau dans un tableau	[BM][Me]→[BM]	TBCOP (Ladder seul)	OP1[OP2] -> RESULTAT[OP2]
Somme des éléments d'un tableau	[M][Me]→[M]	TBSUM	Somme(OP1[OP2]) -> RESULTAT
Recherche d'une valeur dans 1 tableau (égalité) INDEX 1 doit indiquer la longueur du tableau à partir d'OP1	[M][M]→[M] Note INDEX 1 et INDEX R de type [Me]	TBEQ	Recherche dans le tableau OP1[INDEX 1] de la 1ère valeur égale à OP2. Si la valeur est trouvée RESULTAT la reçoit et INDEX R reçoit l'offset par rapport à OP1 (-1 sinon).
Recherche d'une valeur dans 1 tableau (inférieur à) INDEX 1 doit indiquer la longueur du tableau à partir d'OP1	[M][M]→[M] Note INDEX 1 et INDEX R de type [Me]	TBLT	Recherche dans le tableau OP1[INDEX 1] de la 1ère valeur inférieure à OP2. Si la valeur est trouvée RESULTAT la reçoit et INDEX R reçoit l'offset par rapport à OP1 (-1 sinon).
Recherche d'une valeur dans 1 tableau (supérieur à) INDEX 1 doit indiquer la longueur du tableau à partir d'OP1	[M][M]→[M] Note INDEX 1 et INDEX R de type [Me]	TBGT	Recherche dans le tableau OP1[IND1] de la 1ère valeur supérieure à OP2. Si la valeur est trouvée RESULTAT la reçoit et INDEX R reçoit l'offset par rapport à OP1 (-1 sinon).

OPÉRATEURS CONVERSION	OP1, OP2, R	Opérateur LADDER/Littéral	Résultat
Binaire vers Gray	[Me][Me]→[Me]	BTG	Valeur en code Gray de OP1 codée sur OP2 bits.
Gray vers Binaire	[Me][Me]→[Me]	GTB	Valeur en binaire de OP1 codée sur OP2 bits.
Binaire vers BCD	[Me][Me]→[Me]	BCD	Valeur BCD (base 16) de OP1. Un complément est réalisé si OP2 négatif.
BCD vers Binaire	[Me][Me]→[Me]	BIN	Valeur Binaire (base 10) de OP1. Un complément est réalisé si OP2 négatif.
Entier vers Réel	[Me][Me]→[Mr]	FLOAT	Valeur réelle de OP1 / OP2.
Réel vers Entier	[Mr][Mr]→[Me]	INT	Valeur entière de OP1 / OP2.

OPÉRATEURS Bit ←→ Mot	OP1, OP2, R	Opérateur LADDER	Résultat
Éclatement de mot	[Me][Me]→[B]	BIT	OP2 variables binaires sont chargés vers RESULTAT et suivants avec les bits d'OP1 (poids faibles en tête).
Regroupement de bits	[B][Me]→[Me]	MOT	OP2 variables binaires depuis OP1 sont regroupés dans RESULTAT (poids forts mis à zéro si nécessaire).
Ecriture d'un bit de mot	[B][Me]→[Me]	EBM	Ecriture du bit de rang OP2 de la variable entière RESULTAT avec la variable binaire OP1.
Lecture d'un bit de mot	[Me][Me]→[B]	LBM	Ecriture de la variable binaire RESULTAT avec le bit de rang OP2 extrait de la variable entière OP1.

OPÉRATEURS ACCES DATA	OP1, OP2, R	Opérateur LADDER/Littéral	Résultat
Valeur initiale	[M][M]→[M]	WriteIni	Ecriture de OP2 dans OP1 et sa valeur initiale. R reçoit l'ancienne valeur initiale de OP1.

OPÉRATEURS Débit / Pression / Temp	OP1, OP2, R	Opérateur LADDER/Littéral	Résultat
Delta pression(K,Q)	[Mr][Mr]→[M]	DELTAP	=OP1 * Pow(OP2;2) * sign(OP2)
Débit(K, DP)	[Mr][Mr]→[M]	FLOW	=OP1 * Sqrt(abs(OP2)) * sign(OP2)
Celsius → Fahrenheit	[Mr][Mr]→[Mr]	Fahrenheit(OP1;OP2)	OP1*1.8 + 32.0 + OP2 (offset)
Fahrenheit → Celsius	[Mr][Mr]→[Mr]	Celsius (OP1;OP2)	(OP1 – 32.0) / 1.8 + OP2 (offset)

OPÉRATEURS INFORMATION	OP1, OP2, R	Opérateur LADDER/Littéral	Résultat
Taille d'un objet	[Me][Me]→[Me]	GetSX GetSY	Recherche de la taille dynamique en X ou Y d'un objet sur la page OP1 et de numéro OP2.
Position d'un objet	[Me][Me]→[Me]	GetLeftX GetRightX GetTopY GetBotY	Recherche des positions dynamiques du cadre d'un objet sur la page OP1 et de numéro OP2.
Coordonnées d'un objet	[Me][Me]→[Me]	GetX1 GetY1 GetX2 GetY2	Recherche des coordonnées des points dynamiques 1 et 2 d'un objet sur la page OP1 et de numéro OP2.

Note : attention aux modifications de pages (ex : 1er plan) qui entraîne la modification du numéro d'ordre des objets antérieurs.
Pour éviter ce problème, OP2 peut être indiqué sous forme du libellé de l'objet entouré de guillemets par exemple :: GetSX("Synoptique";"Tapis1"). Pour les images et les vidéos seule la racine du fichier est indiquée (sans l'extension).

OPÉRATEURS FRIGO (*)	OP1, OP2, R	Opérateur LADDER/Littéral	Résultat
Densité gaz	[Mr][Mr]→[M]	FRIGO_ROGAZ_T FrigoRoGazT	Densité du gaz OP1 à la température OP2 (°C).
Densité liquide	[Mr][Mr]→[M]	FRIGO_ROLIQ_T FrigoRoLiqT	Densité du liquide OP1 à la température OP2 (°C).
Pression de saturation (bar)	[Mr][Mr]→[M]	FRIGO_PSAT_T FrigoPsatT	Pression de saturation du fluide OP1 pour une température OP2 (°C) donnée
Température de saturation (°C)	[Mr][Mr]→[M]	FRIGO_PSAT_P FrigoPsatP	Température de saturation du fluide OP1 pour une pression OP2 (bar) donnée
Enthalpie gaz (kJ/kg)	[Mr][Mr]→[M]	FRIGO_HTGAZ_T FrigoHtGazT	Enthalpie de saturation du gaz OP1 pour une température OP2 (°C) donnée
Enthalpie liquide (kJ/kg)	[Mr][Mr]→[M]	FRIGO_HTLIQ_T FrigoHtLiqT	Enthalpie de saturation du liquide OP1 pour une température OP2 (°C) donnée
Enthalpie latente (kJ/kg)	[Mr][Mr]→[M]	FRIGO_HLAT_T FrogoHlatT	Enthalpie latente du fluide OP1 pour une température OP2 (°C) donnée

(*) Gaz possibles : R134a : OP1= 134.0. Pour d'autres gaz nous consulter.

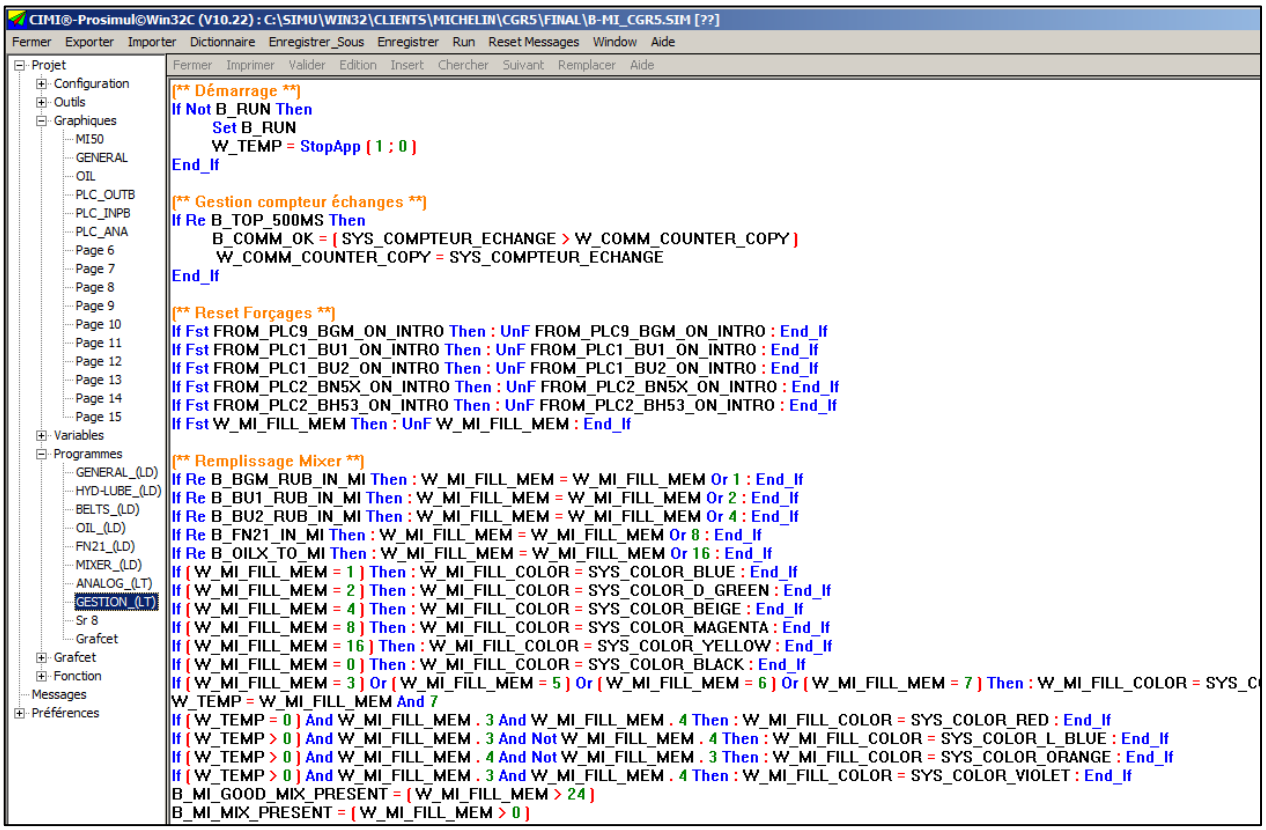
ÉDITEUR LITTERAL

Cet éditeur met à disposition un jeu d'instructions (structure IF , structure WHILE , structure FOR , affectations logiques, fonctions calculs et instructions sur bits) qui peuvent être saisies dans le programme principal, 15 sous-routines et 1 programme spécifique au traitement des transitions/actions liées au Grafcet (si nommé « Grafcet »).

Les instructions sur bits et sur mots peuvent être utilisées.

De plus, le langage littéral permet d'utiliser l'adressage indexé, les bits extraits de mots et les tableaux de bits ou de mots

Au maximum **1024 lignes de 1024 caractères contenant 15 opérateurs** par programme.
Le nombre total de caractères par programme est de 55.000 et le nombre maxi de mots est de 8192.



Menu Littéral

Valider : lance une compilation du programme édité. En cas d'échec un message est affiché dans la barre d'état et est enregistré dans le fichier Errors.TXT

Insérer → Mots-Clés : provoque l'insertion automatique des mots-clés de structure
IF…THEN…ELSE…END_IF , WHILE…DO…END_WHILE , FOR…TO…STEP…NEXT.
Insérer → Fonctions : librairies des fonctions
Insérer → Commentaire : insère les balises (*…*).

Edition → Copier : provoque la mise en tampon du texte sélectionné.
Edition → Coller : provoque la recopie du texte en mémoire.
Edition → Couper : provoque la copie puis l'effacement du texte sélectionné.
Edition → Annuler : annule la dernière opération effectuée.

Chercher : permet la recherche d'une chaîne de caractère dans le texte à partir de la position courante du curseur.
Remplacer : fonction Chercher avec remplacement d'une chaîne de caractère dans le texte.
Remplacer tous : idem Remplacer mais pour toutes les occurrences dans tous les programmes en Littéral.
Suivant : poursuite de la même recherche ou du remplacement.

Commentaire

Un commentaire débute sur une ligne contenant **(*** et termine après la ligne contenant les caractères ***)**.
Les touches F1 & F2 sont des raccourcis pour introduire ou retirer les débuts ou fins de commentaires.
Le **double tiret** « -- » **en fin de ligne** est aussi interprété comme un commentaire (-- *Comment*).

Séparateur d'instruction

Les instructions sont normalement séparées par des sauts de ligne ou le caractère « **:** »

Adressage indexé

Les variables peuvent être utilisées sous forme d'adressage indexé par la syntaxe :
VARIABLE , MOT_INDEX
Le mot index est obligatoirement une variable entière.
Exemple : Set MOTEUR_01 , M10 (Si M10 vaut 3, met à 1 la 3ème variable suivant MOTEUR_01).

Bit extrait de mot

Les bits de variables numériques entières peuvent être utilisés individuellement via la syntaxe
VARIABLE_INT_NUM . RANG
Exemple : BIT08 = TAPIS . 8 : transfert le bit de rang 8 du mot tapis sur la variable BIT08.

Tableaux

Les variables peuvent être utilisées sous forme de tableau via la syntaxe
VARIABLE [LONGUEUR]
Les tableaux de variables ne peuvent être utilisés que dans les instructions de transferts simples.
Exemple 1 : TAPIS = CASE_01 [8] : transfert les 8 bits à partir de CASE_01 et suivants dans le mot TAPIS en commençant par les bits de poids faible.
Exemple 2 : TABLEAU_1[10] = TABLEAU_2[10] (équivalent à TBCOP en LADDER)
Exemple 3 : TABLEAU_1[10] = VAR (équivalent à NCOP en LADDER)
L'adressage indexé et les bits extraits peuvent être utilisés en simultané avec les tableaux.
Exemple : MOT . 4 [4] = 5 (met à1 les bits 4 et 6 de la variable MOT et met à zéro les bits 5 et 7)

Structure IF <CONDITION> THEN <OPE1> ELSE <OPE2> END_IF

La structure est une structure de condition.
Si la Condition est vérifiée les opérations composant OPE1 sont exécutées mais pas celles d'OPE2.
Si la Condition n'est pas vérifiée les opérations composant OPE2 sont exécutées mais pas celles d'OPE1.
La structure peut être utilisée simplement : IF <CONDITION> THEN <OPE1> END_IF.

Structure WHILE <CONDITION> DO <OPE> END_WHILE

La structure est une structure de boucle avec condition. La boucle exécute les opérations de OPE tant que la CONDITION est vérifiée.

Structure FOR <VAR=VALINI> TO <VALFIN> STEP <N> <OPE> NEXT

La structure est une structure d'incrémentation ou de décrémentation selon un pas fixé. La boucle exécute les opérations de OPE de l'état de la variable initiale VALINI jusqu'à la valeur finale VALFIN inclue.

Exemple – Somme d'un mot sur 2 dans un tableau de 20 :

```
SOMME = 0
FOR INDEX = 0 TO 20 STEP 2
        SOMME = SOMME + VALMOT , INDEX
NEXT
```

Appel de sous-programme CALL

L'appel de sous-programme peut être réalisé par n° de Sr ou par le titre du programme indiqué entre simples guillemets.

Ex : Call Sr5 or Call 'MonProgramme'

Transferts simples

Les transferts booléens et numériques sont effectués via l'opérateur **=** .
Les transferts simples ne permettent pas d'usage d'opérateurs. Ils peuvent être notamment utilisés pour les transferts de tableaux et les affectations de variables de types différents.
VAR_INT = VAR_FLOAT

Opérateurs booléens

Chaque opérateur doit être entouré d'un espace. Certains espaces sont introduits automatiquement.
Les opérateurs booléens utilisables pour les tests sont par **ordre de priorité** :
Not : Complément du bit (inutilisable sur expression logique)
Re : Syntaxe pour le front montant de bit
Fe : Syntaxe pour le front descendant de bit
Fst : test de l'état de forçage de variable
< , > , = , >= , <= , <> : Comparaison de valeurs
?1 , ?2 , ?5 , ?10 : Test d'intervalles spécifiques. Exemple : VAR ?5 100 sera vrai pour VAR comprise entre 95 et 105.
>< : test d'intervalles quelconques. Exemple : B_INTERVAL = (W_VALUE – 100) >< 7 : est vraie si W_VALUE a une valeur située entre 93 et 107.
And : ET
Or : OU
Xor : OU Exclusif

Les opérations sur mots +, -, *, /, % (modulo) peuvent être utilisés dans les équations logiques.

Les **parenthèses** doivent être utilisées pour élever des priorités.

Opérations sur bits

Les opérateurs booléens utilisables pour les écritures sont :
Set : Mise à 1 du bit (sur variables booléennes uniquement)
Reset : Mise à 0 du bit (sur variables booléennes uniquement)
Unf : RAZ du bit de forçage de la variable.

Exemple : remise à zéro d'un bit et de son forçage

```
IF BIT_10 THEN
        Unf BIT_10 : Reset BIT10
END_IF
```

Opérateurs numériques

Chaque opérateur doit être entouré d'un espace. Certains espaces sont introduits automatiquement.
Les opérateurs numériques utilisables sont par ordre de priorité :

Priorité 1 : **Fonctions** (voir ci-dessous)
Priorité 2 : ***** , / : traités de gauche à droite
Priorité 3 : + , - : traités de gauche à droite
Priorité 4 : **AND** , **OR** , **XOR**.

Les **parenthèses** peuvent être utilisées pour élever des priorités.

Fonctions

Les fonctions utilisables sont identiques à celles de l'éditeur LADDER et sont traitées dans l'ordre suivant :
ABS, MAX, MIN, MOD, RND,
SHL, SHR, ROL, ROR,
COS, SIN, TAN, ACOS, ASIN, ATAN, ATGPI,
EXPN, EXP10, LOGN, LOG10, POW, SQRT,
TBSUM,
BTG, GTB, BIN, BCD, INT, FLOAT
WRITEINI, GETSX, GETSY, GetLeftX, GetRightX, GetTopY,GetBotY, GetX1, GetY1, GetX2, GetY2

Pour utiliser ces fonctions, il faut utiliser la syntaxe suivante :
FONCTION (OPERATEUR1 ;OPERATEUR2)
Exemple : RES = SHL(MOT1 ; 8) . RES reçoit MOT1 décalé à droite de 8 bits (0 introduits).

Les opérateurs et fonctions sont **imbricables** sur OP1 et OP2.

Les fonctions suivantes n'existent qu'en LADDER :
- EBM, LBM : en Littéral utiliser les bits extraits de mots.
- BIT, MOT : en Littéral utiliser les tableaux de bits.
- TBGT, TBLT, TBEQ : en Littéral utiliser une recherche indexée sur tableau de variables.

Exemples

Exemple 1 : recherche indexée

```
INDEX = 0 : FIND = -1
WHILE (INDEX < 10) And (FIND = -1) DO
        IF TAB , INDEX = SEARCHVALUE THEN
                FIND = INDEX
        END_IF
END_WHILE
(* Traitement de la valeur trouvée *)
IF (FIND > -1) THEN ….
```

Exemple 2 : création d'un bit de cadence

```
TEMPO_TOP = TEMPO_TOP  + SYS_CYCLE_PROSIMUL
TOP_CADENCE = (TEMPO_TOP >= TIME_MS)
IF  TOP_CADENCE  THEN : TEMPO_TOP = TEMPO_TOP - TIME_MS : END_IF
(* Usage du Top *)
If TOP_CADENCE And ….
```

Exemple 3 : création d'une base de temps « réelle » pour une courbe type « Auto »

W_ELAPSED_MS = GetDateTime (10 ; W_NB_MS)

W_NB_MS : contient la période de la base de temps en millisecondes.

W_ELAPSED_MS: reçoit en retour le nombre de périodes écoulées depuis le début de la journée.

Fonction ConnectToTcp(LocalIP, CommTable[3])

Cette fonction permet d'ouvrir une connexion TCP-IP.
Paramètre LocalIP : selon les adresses configurées (configuration échanges) : par défaut (0) ou du fichier des préférences (1..3).
Paramètre CommTable : 1er mot de la table d'échange composée de 3 mots :
1er mot : adresse IP du serveur Modbus sous forme 16#DDCCBBAA
ex : 192.168.1.180 = 16#B401A8C0)
2ème mot : n° port (ex : modbus=502)
3ème mot : délai d'attente en ms

Valeur de retour : 0..7=OK : numéro de socket attribué (8 max)
-1=No Winsock / -2=NoSocket available /-3=Bad local IP / -4=Socket Error / -5=Bind Error / -6=Connect Error

Fonction Disconnect(NumSocket, AllSocket)

Cette fonction permet de fermer une ou plusieurs connexions TCP-IP.
Paramètre NumSocket : numéro de socket reçu lors de la connexion « ConnectToTcp »..
Paramètre AllSocket : si différent de 0, toutes les connexions sont fermées.

Valeur de retour : -1=OK / -2=erreur de numéro de connexion

Fonction ModbusReadRegister(NumSocket, CommTable[4])

Cette fonction permet d'exécuter une lecture de registres depuis une connexion Modbus.
Paramètre NumSocket : numéro reçu lors de la connexion « ConnectToTcp ».
Paramètre CommTable : 1er mot de la table d'échange composée de 4 mots.
1er mot : délai de réponse en ms + n° d'esclave RTU x 1000
ex : 1010_D = esclave 1 / timeout 10 ms
2ème mot : adresse du 1er registre Modbus lu.
3ème mot : nombre de registres 16 bits + code requête (1=bit / 3=Holding / 4=Input) x 1000
ex : 3002_D = code requête 3 / 2 registres 16 bits
4ème mot : adresse de rangement des registres + type de data (Bit=0000_H / Int16=1000_H / Int32=2000_H / Float32=4000_H)
ex : 16#4014 = position 20 dans les réels

Valeur de retour : 0=OK / -1=erreur de socket / -2=Write Error / -3=No Data / -4=Negative Answer

Fonction GetTcpPortState (Ip ; Port)

Paramètre IP : 0..3 : **0** adresse IP par défaut / **1..3** : une des 3 adresses configurée dans les préférences.
Paramètre Port : numéro du port TCP qui doit être testé
Valeur de retour : **5** = le port est actif / -1 : le port est fermé ou inexistant

Fonction SendMail (n°X ; Serveur)

Cette fonction utilise des fichiers *PRO32_Email_X.TXT*, situés sur le même répertoire que l'application .SIM et selon le format suivant :

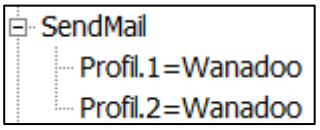
From: sender@cimi.fr
To: recept@cimi.fr
*Subject: Title of the email (**Current Date and Time** will automatically be added)*

Free body text (one blank line must exist like above) with dynamic variables..
HEURE=%MW28% ← one dynamic variable per line (rank is indicated between % characters)
MINUTE=%MW29%
DEBIT B2=%MF10% ← %MFXX% for real value, %MWXX% for integer value
The body must be terminated by a single point on the last line (see below)
Bye, bye.

.

La date et l'heure sont automatiquement rajoutées dans le sujet du message.
Paramètre X :est le n° de terminaison du fichier *PRO32_Email_X.TXT*

V12 : deux serveurs de messagerie peuvent être configurés (*) avec l'application SendMail.exe installée sur le répertoire PROSIMUL et leurs noms doivent être indiqués dans les **préférences** de PROSIMUL (fichier PROSIMUL.INI).
Paramètre Serveur : est le n° de serveur (**1 ou 2**) du fichier des préférences (Profil.1 / Profil.2). 0 est pris pour la valeur 1 (pour compatibilité V11).
(*) Configuration des serveurs (2 maxi) : lancer SendMail.exe, « Options » → « Configuration » → « Ajouter / Modifier ».



V11 : **paramètre Serveur** : 0=protocole SMTP / 1=protocole ESMTP

Valeur retournée : 1 si OK, 0 sinon (voir fichier LOG).

Fonctions DialogOpen / DialogClose ("Page";"Dialog")

Ces fonctions permettent d'activer / fermer par programme des boîtes de dialogue présentes sur une page graphique. Elles remplacent le clic sur l'objet pour le lancement ou la fermeture.
Si l'opération réussie le n° de l'objet est retourné ou -1 en cas d'échec.
Exemple :

```
If B_APPEL Then : W_APPEL = DialogOpen ( "Synoptique" ; "DIALOG_PID" ) : End_If
If B_CLOSE Then : W_APPEL = DialogClose ( "Synoptique" ; "DIALOG_PID" ) : End_If
```

Fonctions BOX externe (V12.05)

L'appel d'exécutables externes (12 au maximum en simultané) avec échange de variables (R/W) est possible via une zone de mémoire partagée accessible via la DLL Pro32BoxDLL.dll qui inclue 4 fonctions.

❑ Structure de la zone de mémoire partagée

struct **pro32_box** {
boxzone BoxData[12]; };
 	
union **var32** {
__int32 i;
float f; };
	struct **boxzone** {
	HWND boxH; *// Box Handle stocké par l'exe*
	__int32 boxID; *// Box ID stocké par PROSIMUL*
	__int32 state; *// 0 initialement - Gestion libre*
	__int32 monitor; *// 0 initialement - Gestion libre*
	__int32 left; *// Gestion libre*
	__int32 top; *// Gestion libre*
	var32 var[10]; }; *// Gestion libre en int ou float*

❑ PROSIMUL : instructions utilisables en Littéral

W_ID = **BoxInit**(X ; BoxID) -- Appel de l'exécutable **Pro32BoxX.exe** avec l'identifiant **BoxID** utilisateur.
Valeur de retour dans W_ID. L'exécutable est lancé avec l'attribut « Hidden ».

- ✅ Appel réussi : **valeur n** de 0 à 11 de la Box activée.
- ⚠ Si BoxID déjà utilisé : valeur de la Box déjà active + 20 (20 à 31)
- ⚠ Lancement impossible : -2

Si l'appel réussi, l'exécutable est lancé avec **BoxData[n].boxId** =**BoxID** et **BoxData[n].state** = **0**.

W_SEL = **BoxSelect**(W_ID ; -1) – Présélection des données de la zone W_ID (0..11)
W_SEL = **BoxSelect**(-1 ; BoxID) – Présélection des données de la zone associée au BoxID (> 0).
W_SEL reçoit la valeur 0..11 de la zone de données présélectionnée ou -1 si la sélection est impossible.
Si la demande réussi **Select** = W_SEL (**Select** interne à chaque application).

W_VAR = **BoxSelRead**(NUM ; 0) -- Lecture de la variable NUM (selon le sélecteur de zone **Select**).
Pour 10 à 19, si NUM est un entier le retour sera un entier, si NUM est un réel le retour sera un réel.

NUM=0	BoxData[Select].boxId
NUM=1	BoxData[Select].state
NUM=2	BoxData[Select].monitor
NUM=3	BoxData[Select].left
NUM=4	BoxData[Select].top
NUM=10…19 ou 10.0 …19.0	BoxData[Select].var[NUM-10]

W_VAR = **BoxSelWrite**(NUM ; VAL) -- Écriture de la variable NUM (voir BoxSelRead sauf 0) avec la valeur VAL selon le sélecteur de zone **Select**.
Si NUM est un entier, VAL est un entier, si NUM est un réel, VAL un réel.

W_SEL = **BoxClose**(W_ID ; -1) – Arrêt de la Box de la zone W_ID (0..11)
W_SEL = **BoxClose**(-1 ; BoxID) – Arrêt de la Box de la zone associée au BoxID (> 0).
W_SEL reçoit la valeur -1 si l'arrêt est possible ou W_ID ou BoxID si la Box n'est pas trouvée.
Si la demande réussi **BoxData[n].boxId** =**-1** et **BoxData[n].state** = **0** puis un message WM_CLOSE est transmis à la Box selon son Handle puis **BoxData[n].boxH** est remis à 0.

La gestion de la fermeture de la Box ou sa visibilité peuvent être gérés directement par l'applicatif via **BoxData[n].state**.

❑ Exemple de code PROSIMUL

```
(*** Lancement BOX ***)
If Re B_BOX_VISIBLE Then
    W_BOX99 = BoxInit ( 99 ; 100 ) : W_TEMP = BoxSelect ( W_BOX99 ; -1 )
    W_TEMP = BoxSelWrite ( 10 ; Int ( B_START_STOP ; 1 ) ) -- écriture du sélecteur binaire dans la 1ère variable User
    -- La position de l'objet appelant peut être retrouvé avec les fonctions GetLeftX et GetTopY ( avec leurs noms )
    -- La Box peut alors être positionnée à l'endroit voulu
    W_TEMP = BoxSelWrite ( BOX_MONITOR ; 1 )
End_If

(* Gestion courante *)
If Re SYS_TOP_DIXIEME And ( W_BOX99 > -1 ) Then
     W_TEMP = BoxSelRead ( BOX_STATUS ; 0 )
    If ( W_TEMP = WANT_CLOSE ) Then
        UnF B_BOX_VISIBLE : B_BOX_VISIBLE = 0
         W_BOX99 = BoxClose ( W_BOX99 ; -1 )
    Else
        W_TEMP = BoxSelect ( W_BOX99 ; -1 )
         W_TEMP = BoxSelRead ( 10 ; 0 ) -- lecture sélecteur StartStop
        B_START_STOP = W_TEMP . 0
        If B_START_STOP Then : W_TEMP = BoxSelWrite ( 11 ; SYS_HORLOGE_DIXIEME ) : End_If-- écriture de
l'horloge dans la 2ème variable User
         R_INPUT = BoxSelRead ( 12.0 ; 0.0 ) -- lecture SetPoint en réel
    End_If
End_If

(* Fermeture BOX *)
If ( SYS_CYCLE_PROSIMUL = -1 ) Or Fe B_BOX_VISIBLE Then
     If ( W_BOX99 <> -1 ) Then : W_BOX99 = BoxClose ( W_BOX99 ; -1 ) : End_If
End_If
```

❑ Application Cliente

L'application appelée reçoit en paramètre le n° de zone exploitée (ex. appel : **Pro32Box0.exe ID5**).
La librairie Pro32Box (.LIB + .DLL) doit être liée statiquement au projet client.
4 fonctions permettent de gérer la zone mémoire partagée.

```
extern "C" __declspec(dllimport) bool _Pro32BoxInit(int Num, HWND hbox, int State);  //true if OK
extern "C" __declspec(dllimport) var32 _Pro32BoxGetData(int Id, int var);    //-9999 if error
extern "C" __declspec(dllimport) int _Pro32BoxSetData(int Id, int var, var32 val); //-9999 if error
extern "C" __declspec(dllimport) bool _Pro32BoxClose(int Id); //true if OK
```

❑ Exemple de Box construite en C++ (C++ Builder)

```
//---------------------------------------------------------------------------

#include <vcl.h>
#pragma hdrstop

#include "main.h"
//---------------------------------------------------------------------------
#pragma package(smart_init)
#pragma resource "*.dfm"
#pragma comment(lib, "Pro32BoxDLL");
```

```
#define MAXBOX     12
#define BOXID                    0
#define STATE                    1
#define MONITOR    2
#define LEFT                     3
#define TOP                                  4
#define VAL10I                   10
#define VAL11I    11
#define VAL12R    12.0
#define TOCLOSE   0
#define UNVISIBLE 1
#define VISIBLE       2
#define WANTCLOSE                3

union var32 {__int32 i; float      f; };
struct infoecran {__int32 width; __int32 height; __int32 left; __int32 top; };
infoecran tabmon[5];
TForm1 *Form1;
HMONITOR hmonitor[5];
int Ecran=0,Px=0,Py=0,NbEcrans,boxId=-1,State=0,ValI=0,Id=-1;
float ValR=0.0;
bool IsPro32=false;
var32 ret;

extern "C" __declspec(dllimport) bool _Pro32BoxInit(int Num,HWND hbox,int State);  //true is OK
extern "C" __declspec(dllimport) var32 _Pro32BoxGetData(int Id,int var);    //-9999 if error
extern "C" __declspec(dllimport) int _Pro32BoxSetData(int Id,int var,var32 val); //-9999 if error
extern "C" __declspec(dllimport) bool _Pro32BoxClose(int Id); //true if OK

//---------------------------------------------------------------------------
__fastcall TForm1::TForm1(TComponent* Owner) : TForm(Owner)
{
Int NbParam=ParamCount();
NbEcrans=InfoEcrans();
Caption="Pro32Box99";
State=UNVISIBLE;
if (NbParam > 0) {
      int sp=1;
      AnsiString pn;
      while (sp <= NbParam) {
            pn=ParamStr(sp);
            if (LeftStr(pn,2) == "ID") Id=StrToInt(RightStr(pn,pn.Length()-2));
            sp++;
            }
      if (Id >=0 && Id < MAXBOX) {
            IsPro32=_Pro32BoxInit(Id,Form1->Handle,State);
            if (IsPro32) {
                  boxId=_Pro32BoxGetData(Id,BOXID).i; //read only - for use is necessary is the BoxN is is used several times with different BoxID
                  }
            else Caption=Caption + " Mapping_error";
            }
      else Caption=Caption + " Id error";
      }
Else {
      Caption=Caption + " No_param";
      Show();
      }
}
//---------------------------------------------------------------------------//-----------------------------------------------------------------------------
void __fastcall TForm1::CalcPos(void)
{
if (IsPro32) {
      ret.i=Px; _Pro32BoxSetData(Id,LEFT,ret);
      ret.i=Py; _Pro32BoxSetData(Id,TOP,ret);
      }
Ecran=Min(Ecran,NbEcrans);
Px=Px+tabmon[Ecran-1].left; Form1->Left=Px;
Py=Py+tabmon[Ecran-1].top; Form1->Top=Py;
}
//---------------------------------------------------------------------------//-----------------------------------------------------------------------------
void __fastcall TForm1::FormClose(TObject *Sender, TCloseAction &Action)
{
if (IsPro32) {
      State=_Pro32BoxGetData(Id,STATE).i; //from Pro32
      if (State > TOCLOSE) {
            State=WANTCLOSE; //dmde Close user
            ret.i=State; _Pro32BoxSetData(Id,STATE,ret);
```

```
            }
        }
else State=TOCLOSE;
if (State == TOCLOSE) {
        if (IsPro32) _Pro32BoxClose(Id);
        Action=TCloseAction::caFree;
        }
else Action=TCloseAction::caNone;
}
//---------------------------------------------------------------------------
void __fastcall TForm1::Timer1Timer(TObject *Sender)
{
if (IsPro32) {
        if (State == UNVISIBLE) {
                Ecran=_Pro32BoxGetData(Id,MONITOR).i;
                if (Ecran > 0) {
                        ValI=_Pro32BoxGetData(Id,VAL10I).i;
                        if (ValI) B_Stop->Enabled=true;  else B_Start->Enabled=true;
                        Px=_Pro32BoxGetData(Id,LEFT).i; Py=_Pro32BoxGetData(Id,TOP).i;
                        CalcPos();
                        State=VISIBLE;
                        ret.i=State; _Pro32BoxSetData(Id,STATE,ret);
                        Show();
                        }
                }
        Else {
                ret=_Pro32BoxGetData(Id,VAL11I);  // récupération de l'horloge
                Horloge->Caption=IntToStr(ret.i);
                }
        }
}
//---------------------------------------------------------------------------
BOOL CALLBACK MonitorEnumProc(HMONITOR hmonitor,HDC,LPRECT r,LPARAM)
{
static int i=0;
MONITORINFO mi;
mi.cbSize=sizeof(MONITORINFO);
GetMonitorInfo(hmonitor,&mi);
if (mi.dwFlags == 1) i=0;
else i++;
tabmon[i].left=mi.rcWork.left;  tabmon[i].top=mi.rcWork.top;
tabmon[i].width=mi.rcWork.right-mi.rcWork.left; tabmon[i].height=mi.rcWork.bottom-mi.rcWork.top;
return(true);
}
//---------------------------------------------------------------------------
int __fastcall TForm1::InfoEcrans()
{
int i,nb;
nb=GetSystemMetrics(80);
for (i=0;i < nb;i++) {
        hmonitor[i]=NULL;
        tabmon[i].left=tabmon[i].top=0;
        tabmon[i].width=1280; tabmon[i].height=1024;
        }
EnumDisplayMonitors(NULL,NULL,MonitorEnumProc,NULL);
return(nb);
};
//---------------------------------------------------------------------------
void __fastcall TForm1::B_StartClick(TObject *Sender)
{
ret.i=1; _Pro32BoxSetData(Id,VAL10I,ret);
B_Start->Enabled=false; B_Stop->Enabled=true;
}
//---------------------------------------------------------------------------
void __fastcall TForm1::B_StopClick(TObject *Sender)
{
ret.i=0; _Pro32BoxSetData(Id,VAL10I,ret);
B_Start->Enabled=true; B_Stop->Enabled=false;
}
//---------------------------------------------------------------------------
void __fastcall TForm1::E_spKeyDown(TObject *Sender, WORD &Key, TShiftState Shift)
{
if (Key == VK_RETURN) {
        ret.f=_wtof(E_sp->Text.c_str()); _Pro32BoxSetData(Id,VAL12R,ret); // écriture 3ème paramètre en réel
        }
}
```

Fonction WinExec (n°X ; Affichage)

Cette fonction utilise des fichiers *PRO32_WinExec_X.TXT*, situés sur le même répertoire que l'application .SIM et contenant une ligne de commande unique formatée :
CHEMIN\APPLI.EXE CHEMIN\PARAMETRE
Le répertoire PROSIMUL peut être indiqué par <REPPRO> (sans \) et le répertoire de l'application (si identique) par <REPAPPLI>.
Le chemin + nom exécutable PROSIMUL est indiqué par <APPPRO>
Exemple 1 : C:\Program Files\Internet Explorer\iexplore.exe http://www.cimi.fr
Exemple 2 : C:\SIMU\PRO32FR.EXE C:\SIMU\GTC\SUPERVISEUR.SIM
Exemple 3 : <REPPRO>PRO32FR.EXE <REPAPPLI>SUPERVISEUR.SIM
Exemple 4 : <APPPRO> <REPAPPLI>SUPERVISEUR.SIM

Paramètre X : est le n° du fichier *PRO32_WinExec_X.TXT*
Paramètre Affichage : est le mode d'affichage de l'application
0 SW_HIDE
1 SW_SHOWNORMAL
2 SW_SHOWMINIMIZED
3 SW_SHOWMAXIMIZED
4 SW_SHOWNOACTIVATE
5 SW_SHOW
6 SW_MINIMIZE
7 SW_SHOWMINNOACTIVE
8 SW_SHOWNA
9 SW_RESTORE
10 SW_SHOWDEFAULT
11 SW_FORCEMINIMIZE
Valeur retournée : > 30 si OK, sinon voir fichier LOG.

Fonction MouseCtrl (param1 ; param2)

Param1 : spécifie le type de fonction exécutée
0 Retour de la coordonnée X courante de la souris
1 Retour de la coordonnée Y courante de la souris
2 Limitation max de la souris en X selon la valeur **param2** et retour de la nouvelle coordonnée
3 Limitation max de la souris en Y selon la valeur **param2** et retour de la nouvelle coordonnée
4 Limitation min de la souris en X selon la valeur **param2** et retour de la nouvelle coordonnée
5 Limitation min de la souris en Y selon la valeur **param2** et retour de la nouvelle coordonnée
10 Retourne le n° d'écran sur lequel se trouve la souris

Fonction GetDateTime (param1 ; param2)

Param1 : spécifie le type de valeur retournée
0 Année
1 Mois
2 Jour du mois
3 Jour de la semaine
4 Heures
5 Minutes
6 Secondes
7 Millisecondes courantes
8 Secondes depuis le début de semaine (depuis lundi 00:00)
10 Millisecondes depuis le début de la journée (depuis 00:00)
Param2 : si différent de 0, il est soustrait de la valeur de retour (pout Param1 de 0 à 7) ou divise la valeur de retour (Param1=10).

Fonction MsgLog (param1 ; param2)

Fonction utilisée pour écrire un message utilisateur dans le fichier PROSIMUL.LOG.
Si OK, retourne Param1.
Param1 : n° de ligne (0..n) du message dans le fichier MSGLOG.TXT (sur le répertoire du .SIM)
Param2 : couleur d'affichage dans la barre d'état.
Une variable peut être intégrée dans le message sous la forme **%MWx%** (entier) ou **%MFx%** (réel).
Ex : La température=%MW125%

Fonction StopApp (param1 ; param2)

Fonction utilisée pour lire l'état de verrouillage ou afficher / stopper / redémarrer l'application.
Param1 : =0 : la fonction retourne seulement l'état de verrouillage de l'application
=1 : pour (dé)verrouiller l'application (**Param2** : 0=déverrouille ou 1=verrouille)
=2 : pour arrêter et quitter l'application (**Param2** = délai en ms avant l'opération)
=3 : pour arrêter et relancer l'application (**Param2** = délai en ms avant l'opération)
=4 : met en avant l'application de n° **Param2** si existante.
=5 : stoppe l'application indiquée par **Param2** (0..n=1 application, 99=toutes applications, -99= annule tous les arrêts)
=6 : minimise l'application de n° **Param2** si existante.
=7 : cache l'application de n° **Param2** si existante.
La fonction retourne soit l'état du verrou (P1=0 à 3) ou le n° de l'application (-1 en cas d'échec).

Fonction InfoApp (param1 ; param2)

Fonction utilisée pour lire l'état des applications.
Param1 : =0 : la fonction retourne le numéro de l'application
=1 : la fonction retourne le nombre d'application en mémoire
=2 : la fonction retourne le nombre d'application en Run
=3 : la fonction retourne l'état Run(1)/Stop(0) de l'application indiquée par **Param2**
=4 : la fonction retourne l'état de la fenêtre application (voir WinExec)
=10 / 11 : la fonction retourne l'origine X / Y du synoptique
si **Param2** =1 l'origine de l'écran est rajoutée ; si **Param2** =-1 l'origine de l'écran est retirée.

Fonction ScreenParam (param1 ; param2)

Fonction utilisée pour recevoir des informations sur la position des applications et des écrans et pour repositionner l'application.

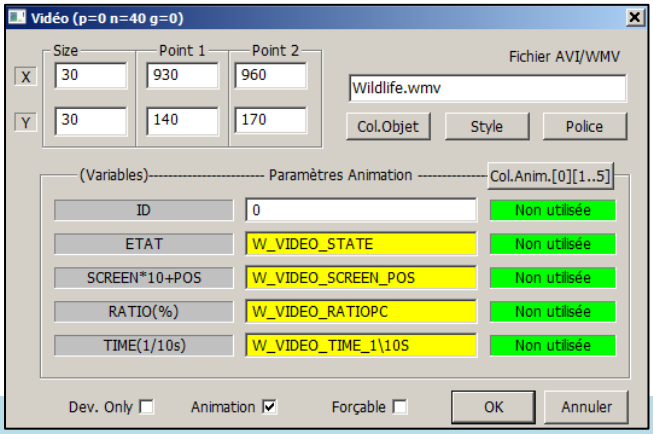
Param1	Param2	Retour	Action
0	0	Nombre d'écrans connectés	-
0	1	N° écran de l'application	-
1	0=appli	Coordonnée X gauche	-
2		Coordonnée Y haut	-
3	1..5= écran	Largeur L	-
4		Hauteur H	-
5	0	Ratio affichage (%)	-
10	1..5 (n° d'écran)	Écran obtenu après contrôle de l'opération	Application plein écran
11			Application au coin haut gauche
12			Application au coin haut droit
13			Application au coin bas gauche
14			Application au coin bas droit
15			Application centrée
21	0…n (X, Y, L, H)	X, Y, L, H obtenue après contrôle de validité	Positionne X gauche de l'application
22			Positionne Y haut de l'application
23			Change la Largeur de l'application
24			Change la Hauteur de l'application

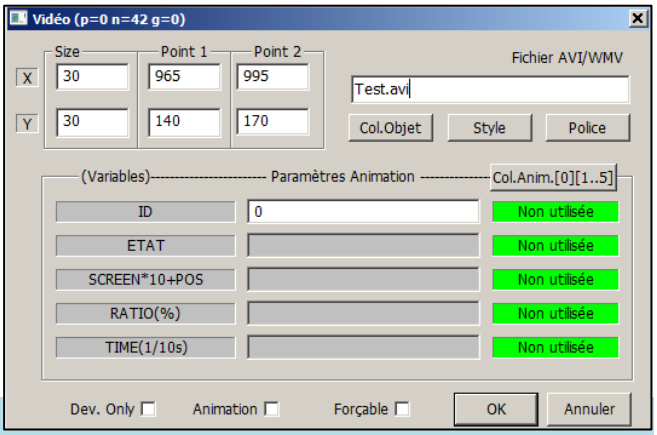
Fonction VideoLoad (param1 ; param2)

Fonction utilisée pour changer le nom de fichier d'un lecteur vidéo.

Param1 : =n ou "nompage": numéro ou nom de page de l'objet vidéo cherché
Param2 : = "nomfichier" (entre guillemet et **sans extension**) : nom du fichier de l'objet vidéo
Retour = ID de la vidéo trouvée (1er paramètre) ou -1 si objet non trouvé ou ID non configuré

Ex : W_VIDEO=**VideoLoad("Pvideos";"TheVideo")** : cherche sur la page « Pvideo » l'objet faisant référence au fichier « TheVideo.AVI » ou « TheVideo.WMV » et prépare le nom du fichier pour le lecteur vidéo 0 à 9 indiqué dans son premier paramètre. Pour la lecture voir l'objet « Vidéo ».
NOTE : si n objets « Vidéo » utilise le même lecteur (switch entre vidéos), un seul des lecteurs doit être animé (avec variables d'animation) et les autres ne doivent pas animés et seulement contenir le même ID (1er paramètre) de lecteur (voir ci-dessous).



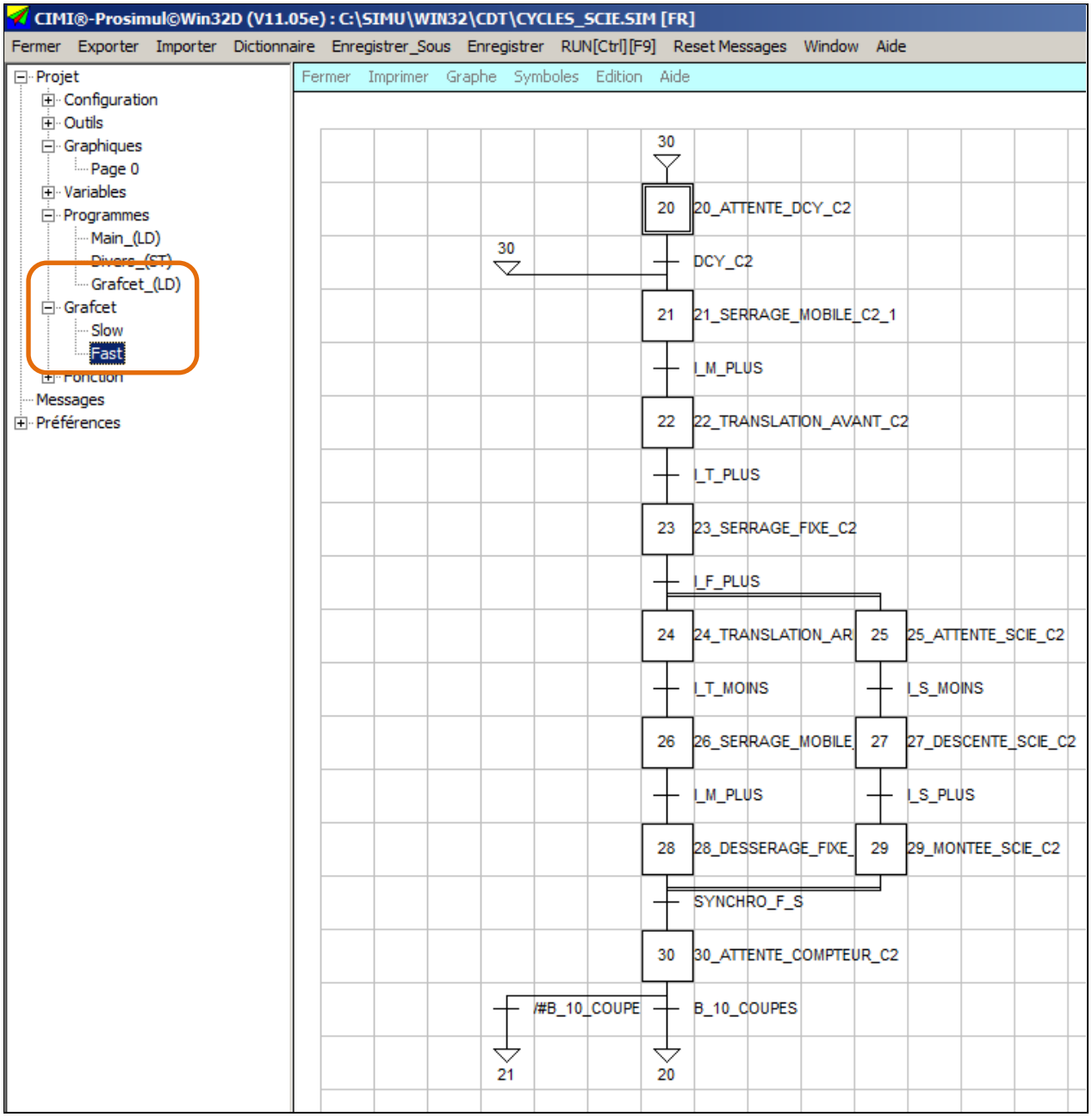


ÉDITEUR GRAFCET

Cet éditeur permet la construction de graphes sur 8 pages de 26 x 20 cellules recevant les symboles.

Au maximum, 2048 étapes dont 32 actives en simultané.

*NOTE : chaque étape gère un bit d'étape unique (pas d'autre action possible) et chaque transition teste une variable binaire unique. Une **zone d'action Ladder nommée " Grafcet ",** réservée au traitement des graphes, devra être utilisée pour provoquer les actions à partir des bits d'étape et pour traiter les transitions.*



Menu Grafcet

Graphe -> Vérifier : permet d'effectuer un contrôle de cohérence des graphes (structure, renvois,...). Suivre la zone message.
Graphe -> Renuméroter : permet d'effectuer une renumérotation (de haut vers bas et gauche vers droite) des étapes et renvois sélectionnés au préalable.
Graphe -> Propriétés : accès au renseignement des étapes / transitions sélectionnées. Le *double-clic* sur une cellule provoque le même effet.

Symboles -> Abandon : retour au curseur de sélection. Action identique obtenu avec le *clic droit* sur une cellule du graphe.
Symboles -> Etape Initiale : sélectionne le curseur pour déposer des étapes initiales.
Symboles -> Etape : sélectionne le curseur pour déposer des étapes simples.
Symboles -> Transition : sélectionne le curseur pour déposer des transitions.
Symboles -> Renvoi Destination : sélectionne le curseur pour déposer des renvois de destination.
Symboles -> Renvoi Origine : sélectionne le curseur pour déposer des renvois d'origine .
Symboles -> Fonction ET : sélectionne le curseur pour construire des divergences et convergences en ET (actions parallèles simultanées).
Symboles -> Fonction OU : sélectionne le curseur pour construire des divergences et convergences en OU (actions parallèles exclusives).
Symboles -> Lien bas : sélectionne le curseur pour déposer des liaisons basses .

NOTE : les étapes, transitions et renvois doivent être renseignés.

Edition -> Copier : provoque la mise en tampon des cellules sélectionnées. Les cellules copiées pourront être exportées .
Edition -> Coller : provoque la recopie des cellules mémorisées à la position sélectionnée (coin haut gauche).
Edition -> Couper : provoque la copie puis l'effacement des cellules sélectionnées.

*NOTE : toutes ces fonctions sont accessibles via le **clic droit** (menu contextuel). Les éléments peuvent être déplacés avec la souris après sélection.*

Renseignement des étapes / transitions / renvois

- Après dépose du symbole ou via Graphe->Propriétés ou via double-clic sur cellule, le renseignement de variable associée est proposé pour les étapes et transitions et renvois.
- Dans les 3 cas vous pouvez soit :

. Saisir son symbole associé (modifiable dans la base).
. Sélectionner son symbole dans la base via le simple clic sur le texte " symbole/numéro ".
. Un clic droit sur la zone de saisie donne le choix parmi les 10 dernières saisies.

- Spécifiquement **pour les étapes et renvois** :

. Les variables de type " Etapes " sont **obligatoires**.
. Les numéros d'étapes doivent être uniques.
. Vous pouvez saisir directement leurs **numéros** (0...2047) en lieu et place du symbole.

Fonction ET et OU

- Sélectionner le curseur correspondant.
- **Divergence en ET :** sélectionner la transition d'origine puis l'étape finale de raccordement.
- **Convergence ET :** sélectionner l'étape finale puis la transition de raccordement.
- **Divergence en OU :** sélectionner l'étape d'origine puis la transition de raccordement.
- **Convergence en OU :** sélectionner la transition d'origine puis l'étape de raccordement.

NOTE : dans tous les cas, les couples de raccordements étape/transition seront distants d'une ligne au maximum. Prévoir des renvois le cas échéant.

Renvois de destination et d'origine

- Les renvois servent aux liens entre étapes / transitions de pages différentes ou pour palier à l'absence de liaisons orientées. Ils fonctionnent par paires.
- Les renvois de **destination** sont positionnés derrière une transition. Indiquer le numéro d'étape de destination.
- Les renvois d'**origine** sont positionnés devant une étape. Indiquer le numéro d'étape d'origine (située directement au-dessus de la transition de départ).

Sélection d'un ensemble de symboles

- Cliquer sur le coin haut-gauche, maintenir l'appui et déplacer la souris vers le coin bas-droite.
- Au relâchement, l'ensemble des cellules inclues dans la zone sont marquées en rouge. Les opérations de couper/copier peuvent opérer dessus.

Visualisation dynamique

En exploitation les graphes sont visualisables et forçables en dynamique.

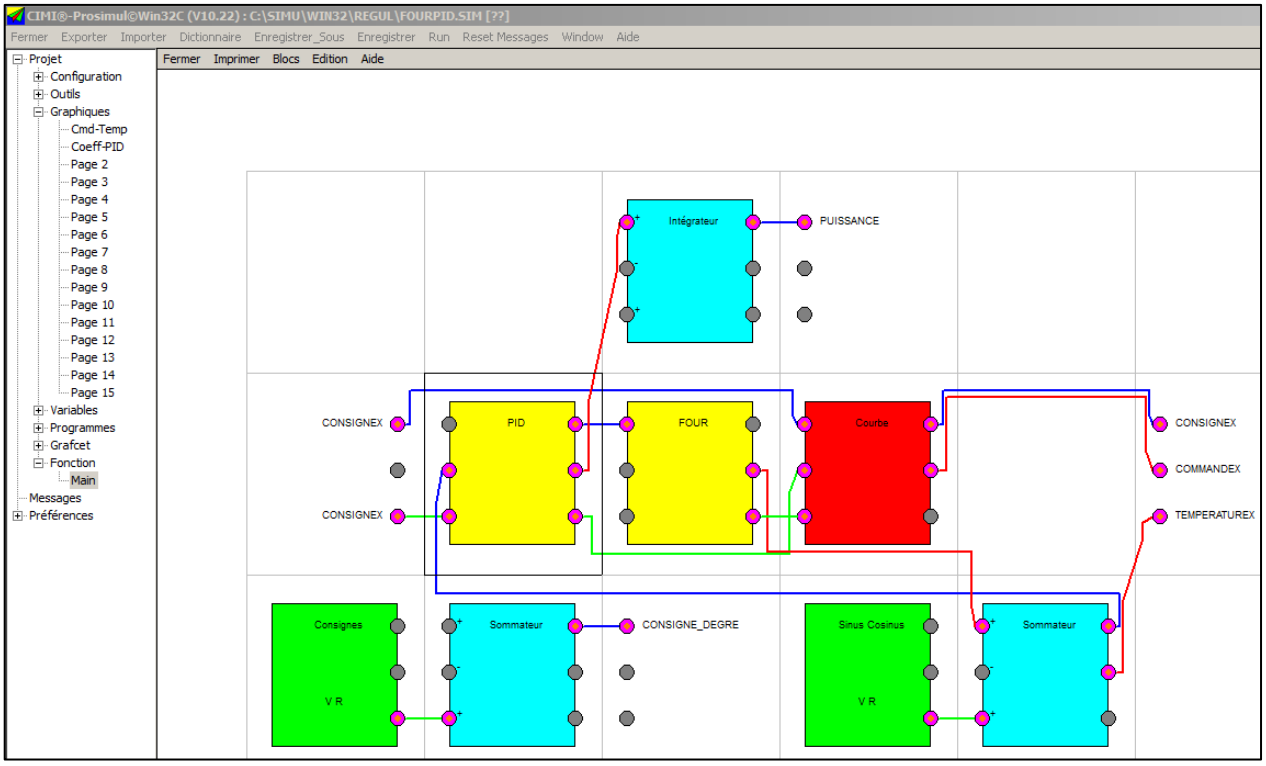
ÉDITEUR FONCTIONS

Cet éditeur permet la création de modèles de procédés de type « continus ». Il permet d'imbriquer des blocs fonctions issus de différentes familles entre eux (sources, circuits, systèmes, logiques, organisation, sorties) soumis à une **période d'exécution** (signaux discrétisés). Des paramètres permettent d'ajuster la fonction des blocs.

Deux blocs spécifiques " Pulsations " et " Bode " permettent le tracé de plans de Bode .

Les pages sont constituées de 6 x 4 cellules.

Les blocs « sous-page » permettent de structurer les applications sur 4 niveaux de profondeur. Ils disposent de 3 points d'entrées et 3 points de sorties de valeurs pour les liaisons entre niveaux.



NOTE : la ***scrutation des blocs*** *s'effectue de gauche à droite et de haut en bas. De façon récurrente, les blocs " sous-page " sont exécutés de la même manière dès qu'ils sont rencontrés dans les niveaux supérieurs. Les valeurs issues des blocs sont " tirées " par les blocs consommateurs et non " poussées " par les producteurs. Ceci entraîne un décalage d'un tour de " période " pour les valeurs tirées de droite vers gauche.*

Un total maximum de 800 blocs fonction est autorisé.

Menu Fonction

Bloc -> Créer/Renseigner : après sélection d'une cellule vide ce menu donne l'accès à la palette de blocs disponibles. Un **double-clic** sur cellule vide de la page Graphique provoque la même action. Si la cellule n'est pas vide la boîte de dialogue de paramétrage apparaît.
Bloc -> Zoom In : provoque l'accès à une " sous-page ".
Bloc -> Zoom Out : provoque la sortie d'une " sous-page ".
Bloc -> Période : permet de changer le paramètre " période " sur l'ensemble des blocs de la page courante. Les sous pages sont aussi affectées.
Bloc -> Saturation : permet de changer le paramètre " saturation/maximum " sur l'ensemble des blocs sélectionnés.

Edition -> Copier : provoque la mise en tampon des cellules sélectionnées. Les blocs copiés pourront être exportés .
Edition -> Coller : provoque la recopie des cellules mémorisées à la position sélectionnée (coin haut gauche). Les liens avec d'autres blocs sont reconstruits le cas échéant. Si l'outil d'identification polynomial du menu horodatage a été utilisé le polynôme résultant est collé).
Edition -> Couper : provoque la copie puis l'effacement des cellules sélectionnées.

*NOTE : toutes ces fonctions sont accessibles via le **clic droit** (menu contextuel). Les blocs peuvent être déplacés avec la souris après sélection.*

Sélection d'un ensemble de blocs

- Cliquer sur le coin haut-gauche, maintenir l'appui et déplacer la souris vers le coin bas-droite.
- Au relâchement, l'ensemble des cellules inclues dans la zone sont marquées en rouge. Les opérations de couper/copier peuvent opérer dessus.

Liens entre blocs fonctions

Les liens entre blocs se font exclusivement entre paires de points de sortie et d'entrée.

- **Création de liens** : au passage de la souris sur un point de connexion, le curseur se change en " croix 1 ". Cliquer sur le point et **maintenir la souris** tout en la déplaçant sur le second point. Le curseur se transforme en " croix 2 " au-dessous du 2ème point. Relâcher l'appui souris, le lien est créé.

- **Effacement de liens** : cliquer simplement sur l'un des 2 points de connexion. La suppression d'un bloc entraîne l'effacement de ses liens.

Paramétrage communs des blocs fonctions

NOTE : tous les paramètres de blocs peuvent recevoir soit des valeurs immédiates soit des variables réelles issues de la base (converties dans tous les cas en réels).

Commentaire : texte de 19 caractères maximum (affiché dans le bloc).

DT_Echantillon(sec) : période d'exécution requise exprimée en **seconde**. Dans le cas de dépassement de cette période en RUN un message l'indique. Ajuster le temps de cycle PROSIMUL.

Gain Bloc : gain multiplicateur appliqué en sortie de bloc.

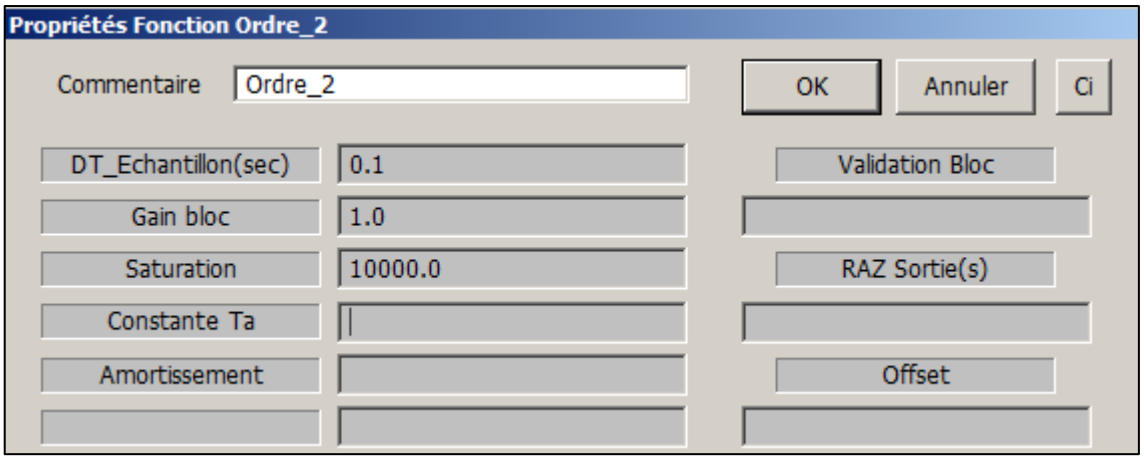
Saturation : valeur absolue maximum en sortie de blocs.

Validation Bloc : condition binaire permettant ou non l'exécution périodique du bloc (*par défaut un bloc est validé en permanence*).

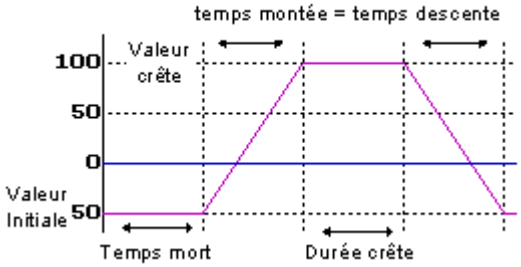
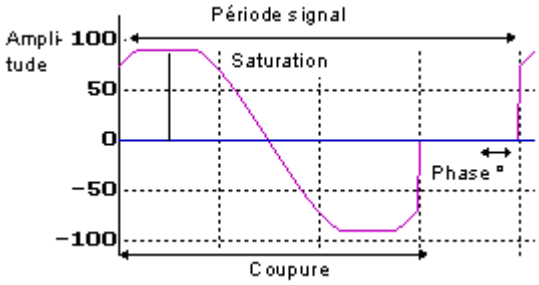
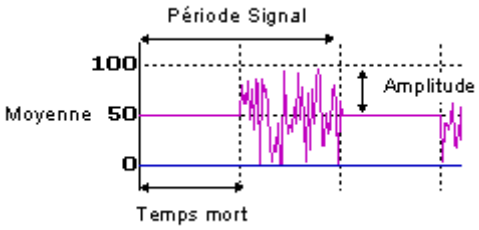
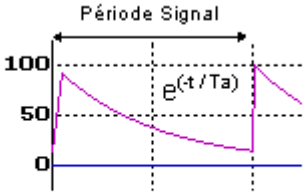
RAZ Sortie(s) : condition binaire permettant la remise à zéro forcée des sorties du bloc.

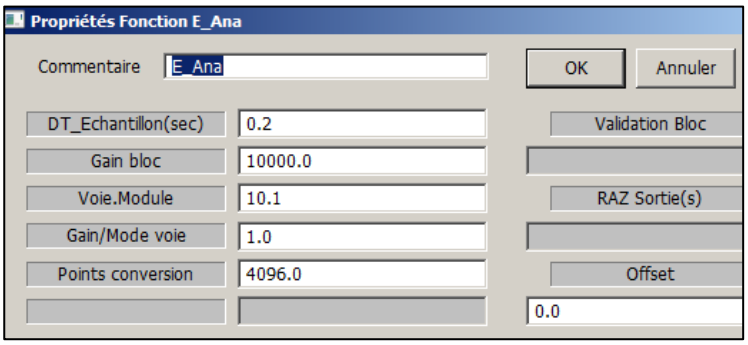
Offset (ou Coefficient) : si elle est renseignée et sauf cas particulier documenté, cette variable mot agit par défaut en offset sur la sortie du bloc.

Variable base 1/2/3 : pour les **blocs entrées et sorties base**, ces variables sont utilisées pour faire des échanges de valeurs internes aux blocs depuis / vers la base de données (normalement de type réelles).



Paramètres des blocs SOURCES

Blocs	Signal Paramètres	Entrées Sorties
Carré-Triangle		- Pas d'entrée - 3 sorties identiques
Sinus-Cosinus		- Pas d'entrée - 3 sorties identiques
Bruit blanc		- Pas d'entrée - 3 sorties identiques
Entrée Base	Variables base 1/2/3	- Pas d'entrée - Sorties = Variables
Exponentielle		- Pas d'entrée - 3 sorties identiques

E_Ana	**Selon cartes configurées** - Gain Bloc K_B : valeur pleine échelle - Voie . Module : n° de voie et n°de module EASYPORT=0..3 •1..4 SEALEVEL=1..16•1..4 PIO-DAxU=0..1•0..n - Gain/Mode voie : gain d'acquisition du signal LABPC+ : 1, 1.25, 2, 5, 10, 20, 50, 100 ACL8112 : 1, 2, 4, 8, 16 EASYPORT : mettre **1.0** (inutilisé) SEALEVEL : mettre **1.0** (inutilisé) PIO-DAxU= mettre **1.0** (inutilisé) - Points conversion (2048, 4096,..,32767) EASYPORT : 2048 (11 bits) SEALEVEL : 4096 = 12 bits unipolaire – en bipolaire valeur positives 0..2047 et négatives 4095..2048) - Offset: rajouté/soustrait optionnellement à la valeur.  	- Pas d'entrée - 3 sorties identiques =$EAna_{(GE)}$*K_B/Points
Consignes	Génération de consignes - Valeur Initiale des sorties - Période : relance du cycle global - 12 couples de consignes Temps, Valeur (accès par bouton [Ci]) - *Saisir le **dernier temps** à la valeur **-1.0***	- Pas d'entrée - 3 sorties identiques
Pulsation	Utilisé pour produire le plan de Bode - Nombre N d'échantillon par pulsation (360°) - Nombre de périodes i (pulsations) avant erreur - Omega Min (en rd/s) - Point par décade LOG10 - Omega Max (en rd/s) - Amplitude (/ 2 à chaque décade franchie)	- Pas d'entrée - Sortie Pulsation Pi = Amplitude * sin(ω.N.Te)

Paramètres blocs CIRCUITS

Blocs	Signal Paramètres	Entrées Sorties
Intégrateur	- Intégrale de 0 à n.T_e de la somme des entrées - Valeur initiale de l'intégrateur - Saturation - Gain K - Valeur initiale (peut être différente de 0) - Minimum (blocage en deça)	- Entrées +, -, + - 3 sorties identiques
Dérivateur	- Dérivée sur la période de la somme des entrées - Saturation - Gain K	- Entrée +, -, + - 3 sorties identiques
Sommateur	- Somme des entrées (0 si non raccordée(s)) - Saturation - Gain K - Valeur initiale (peut être différente de 0) - Minimum (blocage en deça)	- Entrée +, -, + - 3 sorties identiques
Multiplicateur	- Entrée1 / Entrée2 * Entréee3 (1 si non raccordée(s)) - Saturation - Gain K - Valeur initiale (peut être différente de 0) - Minimum (blocage en deça)	- Entrée *, /, * - 3 sorties identiques
Retard	- Retard des signaux d'entrées de N périodes d'échantillonnage (**1..28**) - Saturation - Gain K - Les valeurs peuvent être ressorties dans la table **Values**	- Une entrée - Sortie = Retard
Seuil	- Seuil bas : entre + ou – ce seuil la sortie est figée à 0. Sinon elle reçoit la somme des entrées (+, -, +) décalée du seuil. - Saturation - Gain K	- Entrée +, -, + - 3 sorties identiques

Blocs	Signal Paramètres	Entrées Sorties
Puissance	- Calcul de la puissance n de la somme des 3 valeurs d'entrée - Saturation - Gain K - Exposant n	- Entrée +, -, + - 3 sorties identiques
Minimum	- Calcul du minimum des 3 valeurs d'entrée (non raccordées = $\propto$) - Saturation - Gain K	- 3 sorties identiques
Maximum	- Calcul du maximum des 3 valeurs d'entrée (non raccordées = $-\propto$) - Saturation - Gain K	- 3 sorties identiques
Moyenne	- Calcul de la moyenne glissante de N (1..28) valeurs - Saturation - Gain K - État initial - Le paramètre « Start elements » indique le nombre de valeurs initiales (0 ..N) à prendre en compte au démarrage (0 par défaut).	- 1 sortie Moyenne

NOTE : les fonctions F(p) sont traitées en discret (opérateur z^{-1}) avec bloqueur d'ordre 0.

Paramètres des blocs SYSTEME

Blocs	Signal Paramètres	Entrées Sorties
Ordre 1	- Fonction F(p) = **K / (1+Ta . p)** - Gain K - Constante Temps Ta - Saturation - Valeur initiale (peut être différente de 0) - Minimum (blocage en deça)	- Entrées +, -, + - 3 sorties identiques
Ordre 2	- Fonction F(p) **. K / (Ta^2.p^2 + 2.m.p + 1)** si *0 < m < 1* **. K / (1 + Ta.p)^2** si *m = 1* **. K / (1 + (m+(m^2-1)$^{0.5}$).Ta.p)*** **(1 + (m-(m^2-1)$^{0.5}$).Ta.p)** si *m > 1* - Gain K - Constante Temps Ta - Amortissement m - Saturation	- Entrées +, -, + - 3 sorties identiques
Filtre Passe-Haut	- Fonction F(p) = **K * Ta.p / (1 + Ta.p)** - Gain K - Constante de temps Ta - Saturation	- Entrées +, -, + - 3 sorties identiques
Avance / Retard de phase	- Fonction F(p) = **K * (1 + Tb.p) / (1 + Ta.p)** - Gain K - Constantes de temps Ta, Tb - Saturation	- Entrées +, -, + - 3 sorties identiques
Régulateur PIDP (structure parallèle)	- Fonction F(p) = **K + 1/Ti.p + Td.p / (1 + Td/N.p)** - Gain K - Constantes de temps 1/Ti, Td - Filtre N de la dérivée - Saturation Min and Max : l'intégrateur se fige en-dessous et au-dessus de ces limites	- Entrées +, -, + - 3 sorties identiques - les valeurs P/I/D peuvent être récupérés ou écrites (*) → Vi
Régulateur PIDM (structure mixte)	- Fonction F(p) = **K (1 + 1/Ti.p + Td.p / (1 + Td/N.p)** - Gain K - Constantes de temps 1/Ti, Td - Filtre N de la dérivée - Saturation Min and Max : l'intégrateur se fige en-dessous et au-dessus de ces limites	- Entrées +, -, + - 3 sorties identiques - les valeurs P/I/D peuvent être récupérés ou écrites **(*)** → Vi

(*) Les valeurs PID sont écrites si les variables externes sont renseignées (valeur 0 sinon) et si la variable « Manual Vi | Raz Out » est à l'état vrai.

Polynôme	- Polynôme $P(X) = \Sigma C_i * X^{Ni} + Val_Initiale$ (12 termes maxi) - Indiquer **-1** pour le dernier Ni (accès par bouton [Ci]) - Valeur initiale (peut être différente de 0) - Minimum (blocage en deça)	- Entrées +, -, + - 3 sorties identiques
Déphasage non minimal	- Fonction F(p) = K * (1 - Tb.p) / (1 - Ta.p)2 - Gain K - Constantes de temps Ta, Tb - Saturation	- Entrées +, -, + - 3 sorties identiques
Convoyeur	- Retard des signaux d'entrées de **Nb périodes** d'échantillonnage (**1..24**) selon sens de marche (MAV/MAR) et avec gestion d'accumulation si vitesse nulle - La **Saturation** agit en écrêtage de sortie (perte matière). - Le **gain** est réparti sur les N éléments (racine nième du gain). - Variable **Coefficient** : contient le coefficient de vitesse du convoyeur en entier compris entre –24 et +24 (0 = arrêt, < 0 MAR, > 0 MAV). 1 = une avance par période. - Les valeurs internes peuvent être utilisées dans la table **Values (N Output)** **NOTE** : la **période** d'échantillonnage peut aussi être variable pour obtenir une avance d'une case (Coefficient=1) à vitesse variable.	- Entrées Q_eMAV inutilisée Q_eMAR - Sorties Q_sMAV Charge totale Q_sMAR
Screw	- Retard des signaux de **N périodes** (1 à 27). - La **Saturation** agit en filtre de sortie mais l'excédent E= (Nième valeur – Saturation) est réparti de façon décroissante sur les N valeurs (Vi) précédentes (reflux de matière). Ei= E / ((N+1)*N/2) * i. - Le **gain** est réparti sur les N éléments (racine nième du gain). - Les valeurs internes peuvent être utilisées dans la table **Values (N Output)** et peuvent prendre une **valeur initiale.** **NOTE** : la **période** d'échantillonnage peut aussi être variable pour obtenir une avance d'une case (Coefficient=1) à vitesse variable.	- Entrée Qentrée - Sorties Inutilisée Charge totale Qsortie

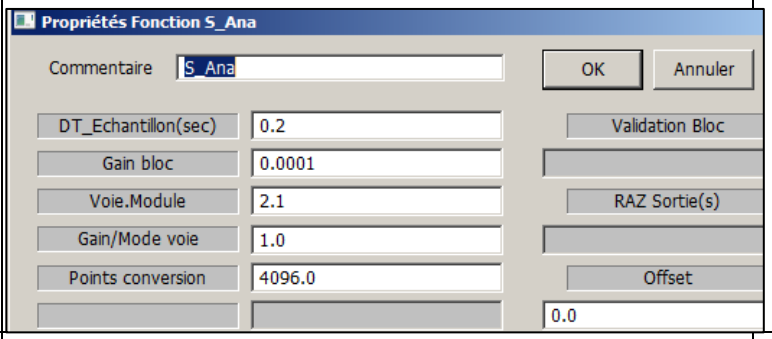
Paramètres blocs LOGIQUES

Blocs Logiques	Signal Paramètres	Entrées Sorties
OU	- Fonction OU entre entrées (0 si non raccordée(s)) - Gain K - Saturation	- 3 sorties identiques
ET	- Fonction ET entre entrées (1 si non raccordée(s)) - Gain K - Saturation	- 3 sorties identiques
NON	- Fonction NON pour chaque entrée - Gain K - Saturation	- 1 sortie par entrée

Paramètres blocs ORGANISATION

Blocs Organisation	Signal Paramètres	Entrées Sorties
Sous-Page	- Sous-page de fonction (accessible par Zoom In ou simple clic en RUN) - Gain K - Saturation	- entrées de “ sous-page ” - sorties de “ sous-page ”
Copie	- Copie d’une valeur en 3 sorties - Gain K - Saturation	- 1 entrée - 3 sorties identiques
Multiplexeur	- Multiplexage de 2 valeurs (entrée 2 ou 3) selon un seuil comparé à entrée 1 - Gain K - Saturation (Seuil)	- 3 sorties identiques (entrée 2 si e1 >= seuil ou entrée 3 si e1 < seuil)
Split-Range	- Gain K - Saturation - **D**irecte/**R**everse : S1/S3 type Rapide/Précis (mêmes sens) ou type Chaud/Froid (sens inverses) - Seuil bas (SB) + Seuil Haut (SH) → S2 < SB type D : S1=S2*Gain+[Offset] type R : S1=(SB – S2)*Gain+[Offset] S3=0 → SB < S2 < SH type D : S1=SB*Gain+[Offset], S3=0 type R : S1=S3=0 → S2 > SH S3=(S2 - SH)*Gain+[Offset] type R : S1=0 type D : S1=SB*Gain+[Offset]	- S1 : action Froid ou Lente - **S2** = somme des entrées (+, -, +) - S3 : action Chaud ou Rapide
Share	Permet de séparer une proportion des entrées (**Ratio Out (0..1)**) valeur en 2 parties selon un **Coeff Out(%)**. Qe = Min((E1-E2+E3)*Gain ; Saturation) Qr= Qe*(1- RatioOut*CoeffOut / 100) * ScaleRemain Qo= (Qe – Qr) * ScaleOut	- Entrées Ei +, -, + - Qr - Qo - Qo

Paramètres blocs SORTIES

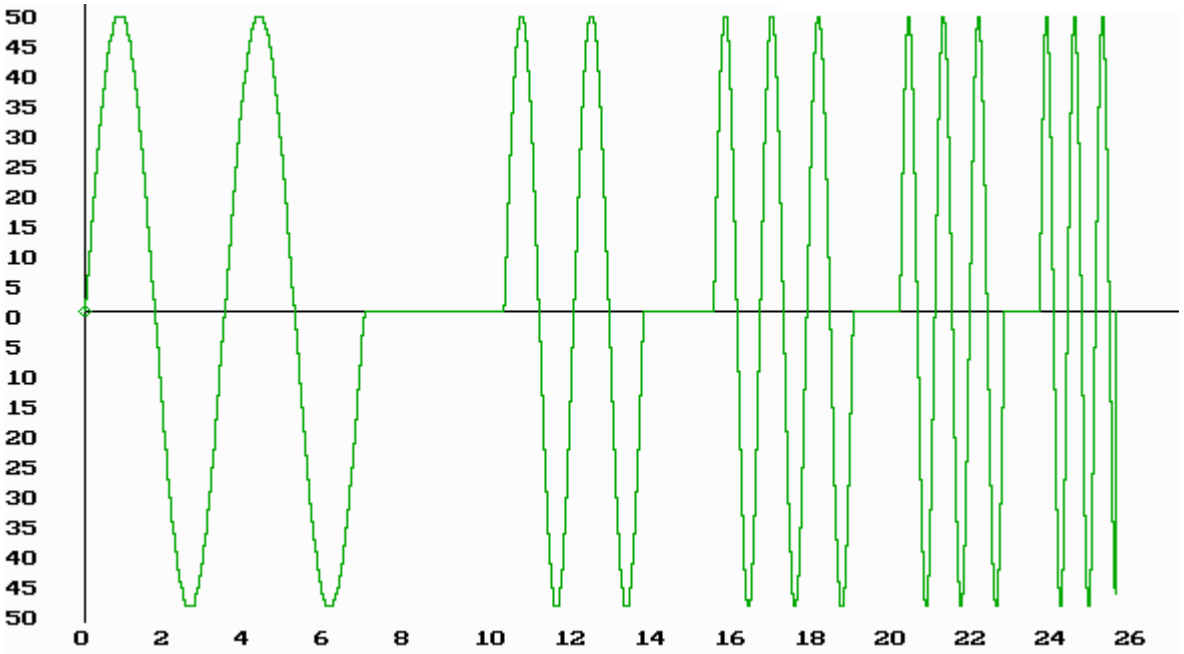
Blocs Sorties	Signal Paramètres	Entrées Sorties
Courbe	- Affichage des entrées*K en exploitation - Gain K - Saturation (maximum ordonnée)	- entrées isolées - sorties = entrées . K
Lampe	- Visualisation de 3 lampes (1 si entrée*K >= Saturation) en exploitation - Gain K - Saturation	- entrées isolées - sorties = état lampe (0 ou 1)
Sortie Base	- Chargement des variables base 1/2/3 avec les valeurs d'entrées*K - Gain K - Saturation	- entrées isolées - sorties = entrées . K
S_Ana	**Selon cartes configurées** - Gain KB : rapport tel que Entrée* KB =1 - Voie . Module : n° de voie et n°de module EASYPORT=0..1 •1..4 SEALEVEL=1..2•1..4 - Gain/Mode voie : 1.0 (inutilisé) - Points conversion (2048, 4096,..,32767) EASYPORT : 2048 (11 bits) SEALEVEL : 4096 = 12 bits unipolaire - Offset: rajouté/soustrait optionnellement à la valeur.  	- Une entrée - 3 sorties identiques = Entrée*Points* K_B
Bode	Utilisé pour produire le plan de Bode - Coefficient N du filtre (> 1 – Sans=100) - Erreur admissible d'écart intégrales - Échelle de tracé de la phase (en degré) et du gain (en dB) sous forme de mini/maxi. Le lancement du tracé s'effectue par la fonction " Zoom In " si le bloc est correctement raccordé.	- Entrée 1 : signal référence (issu de pulsation). - Entrée 2 : signal à analyser. - En sortie, le bloc transmet les valeurs du signal filtré, de sa dérivée et de son intégrale.

Plan de BODE - Méthode

Dans le cas de modèles simulés, l'envoi de signaux sinusoïdaux en entrée de ceux-ci ne pose aucun problème de comportement physique. De plus, puisque toutes les fonctions utilisées sont discrètes, la durée des essais peut être singulièrement raccourcie par l'accélération du temps interne auquel les traitements sont liés. Ainsi un signal *sin(0.001 t)* pourra être envoyé sous forme d'échantillons $sin(\omega.n.T_e)$ au même rythme qu'un signal *sin(1000 t)*, à condition que les systèmes discrets le recevant soient réglés sur la même période d'échantillonnage. Il est à constater que le temps fictif ainsi généré offrira une précision inégalable pour le traitement des blocs fonction.

La méthode retenue consiste donc, à entrer dans le système une suite de pulsations *sin(ω.N.Te)*, par l'intermédiaire d'une boîte source spécifique. La plage de pulsations, le nombre d'échantillons du signal ainsi que son gain pourront être ajustés selon les besoins.

Un nombre maximum d'envoi de pulsations era indiqué, afin de stopper l'application dès lors que le signal de sortie ne pourra être identifié (gain trop faible ou retard pur trop important).



Forme du signal source (Bloc Pulsation)

Les blocs fonctions traités dans l'application seront initialisés avant chaque nouvel envoi de pulsation, et leur période ainsi que les incréments de temps système seront forcés à la valeur Te correspondante à l'oméga courant. A noter, pour le bloc retard pur, qu'un décalage itératif des valeurs a été introduit, tant que ce temps fictif transmis n'est pas égal à la valeur de retard paramétrée.

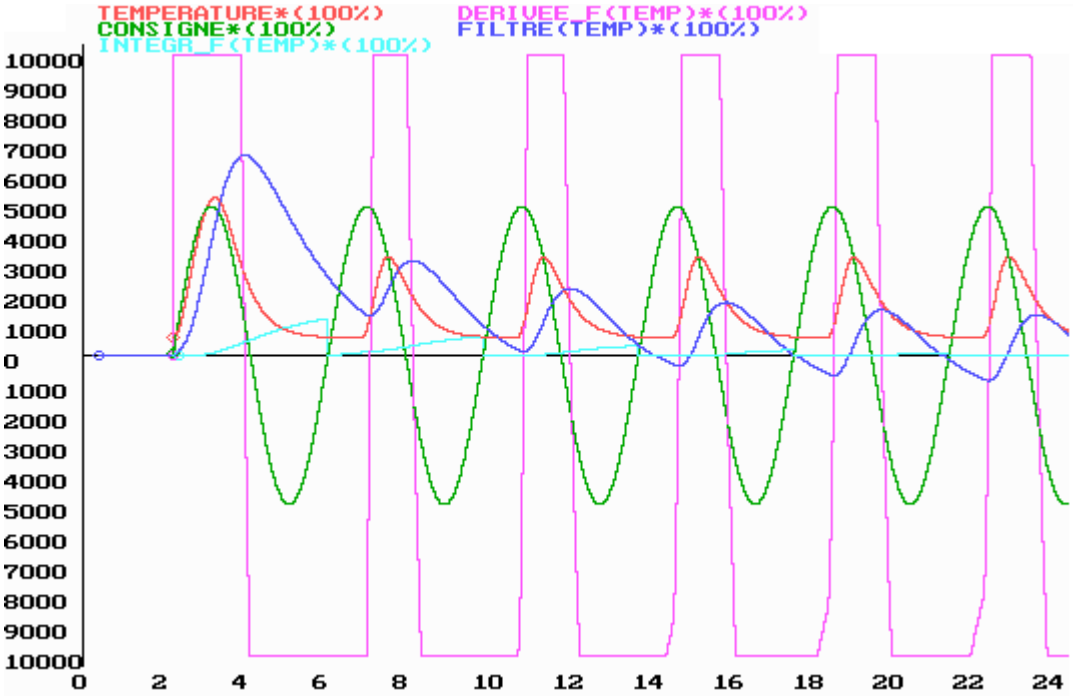
Une boîte “ Plan de Bode ” recueille la réponse issue du procédé ainsi que le signal d’origine, et tente d’identifier le gain et le déphasage introduits par le système sollicité. Plusieurs opérations sont effectuées sur le signal afin d’accélérer l’identification :

1. un filtrage passe-bande $F(p) = \dfrac{1}{1+\dfrac{T_c}{N}\cdot p} * \dfrac{N.T_c.p}{1+N.T_c.p}$ (N >= 1) afin d’éliminer les composantes continues ainsi que les hautes fréquences parasites,

2. une dérivation première afin de détecter les maximums du signal (mémorisation des valeurs maximales et des numéros d’échantillon pour le déphasage),

3. une dérivation seconde permettant le comptage des oscillations dans le cas d’avance de phase sur la première pulsation ou de composante continue non filtrée,

4. une intégrale du signal filtré calculée sur chaque période permettant de mesurer la stabilisation par comparaison entre 2 périodes.



Exemple de traitement de signal
(source=consigne, sortie=température)

Le coefficient N du filtre passe-bande est réglable par l’utilisateur (un N petit laisse passer une faible bande de fréquence) et l’erreur admissible sur l’intégrale peut être ajustée selon les besoins (compromis entre précision et vitesse de convergence). Dans le cas de signaux sans harmonique, ce filtre peut être inhibé par une valeur de N=100.

Lorsque l'erreur admise sur l'intégrale est atteinte et des maxima détectés, le gain et la phase sont calculés avec retrait de l'action du filtre basse-bande connu.

$$Gain_Système = 20 * \log_{10}\left[abs\left(\frac{\max_sortie - \min_sortie}{\max_source - \min_source} \right) * \frac{1}{K_{correcteur}} \right]$$

$$Phase_Système = \left(Num_echant(\max_sortie) - Num_echant(\max_source) \right) * \frac{360}{Max_echant} - \varphi_{correcteur}$$

or pour la pulsation spécifique le filtre devient :

$$F(j\omega_c) = \frac{j.N.T_c.\frac{2.\pi}{T_c}}{\left(1 + j.\frac{T_c}{N}.\frac{2.\pi}{T_c}\right) * \left(1 + j.N.T_c.\frac{2.\pi}{T_c}\right)} = \frac{j.N.2\pi}{\left(1 + j.\frac{2\pi}{N}\right) * \left(1 + j.N.2\pi\right)}$$

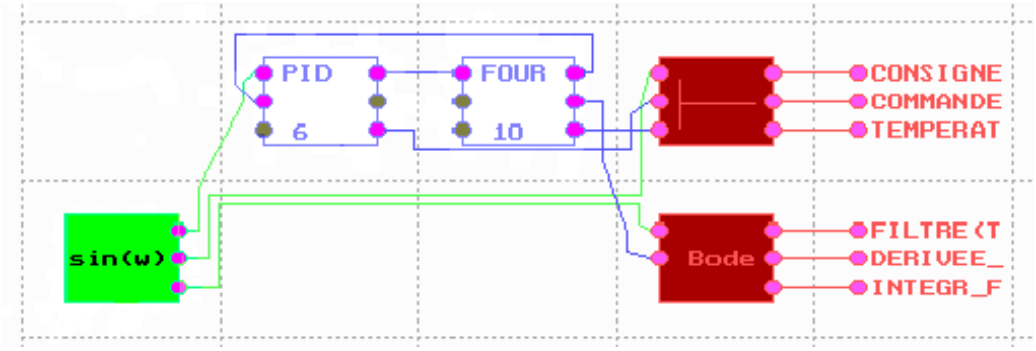
soit $\varphi_{Correcteur}(\omega_c) = 90 - \arctan\left(\frac{2\pi}{N}\right) - \arctan(2\pi.N)$ $(en\deg ré)$

$$K_{Correcteur}(\omega_c) = \frac{2\pi.N}{\sqrt{1 + (2\pi / N)^2} * \sqrt{1 + (2\pi.N)^2}}$$

Ces valeurs obtenues peuvent être tracées dans le plan de Bode, selon les échelles de gain et de phase retenues (deux échelles et couleurs différentes).

Plan de BODE - Mise en œuvre

L'utilisateur doit raccorder une boîte " Pulsation " en entrée du système à traiter. Une boîte " Bode " recueille d'une part cette entrée et d'autre part la sortie du procédé.



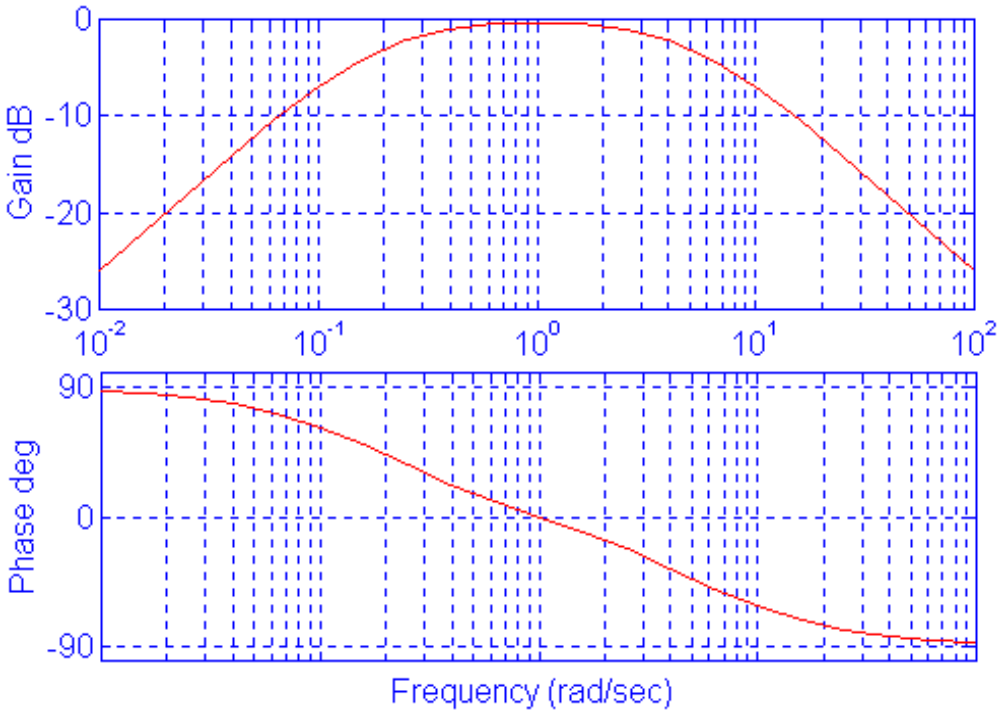
Exemple de raccordements des boîtes Pulsation et Bode

Dès lors, l'application peut être lancée et le tracé activé par la fonction " **Zoom In** " sur la boîte " Bode " en **STOP**. Les *saturations des différents blocs* peuvent être maintenues (cas réel) ou éliminées (cas théorique).

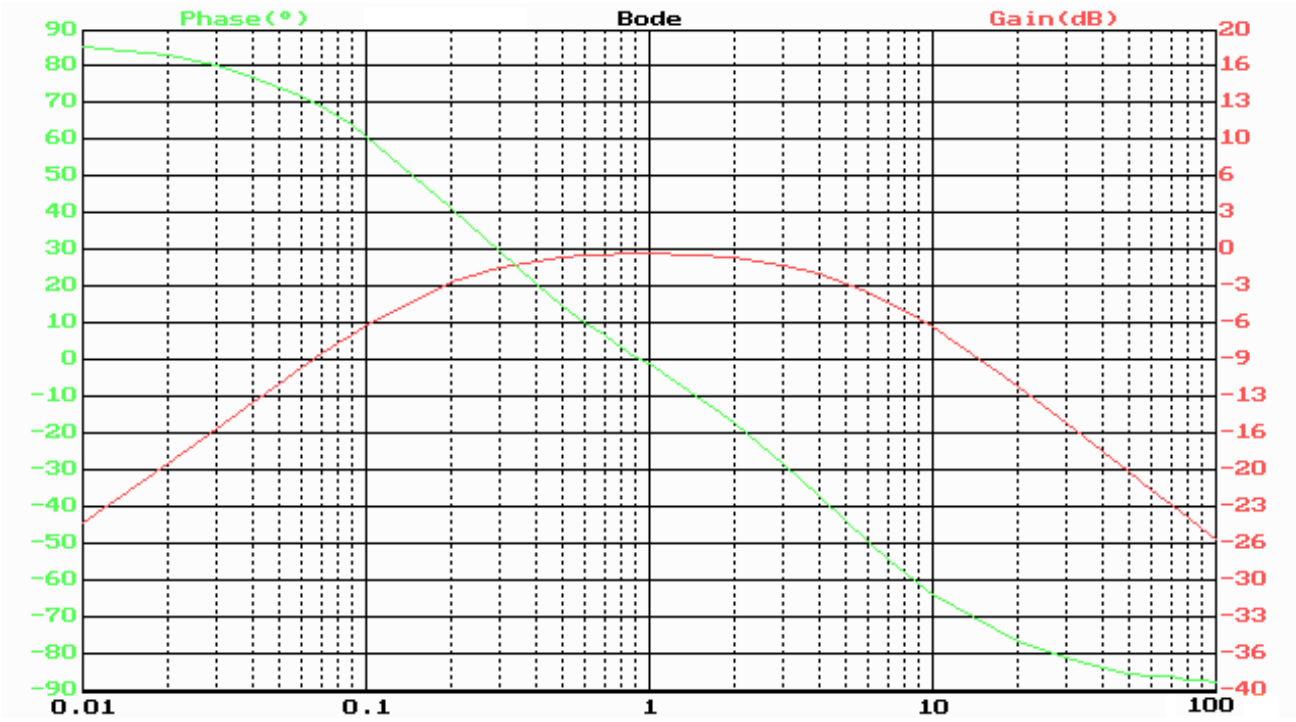
D'un point de vue méthodologique, les essais doivent débuter sans filtrage (N=100), et la vitesse de convergence vers zéro de l'intégrale (indiquée sur l'écran) ou la forme du signal de sortie enregistré permettront d'ajuster ce coefficient et l'erreur admissible. Une attention particulière doit être portée aux grandeurs constantes des systèmes qui, si elles sont conservées, vont engendrer un temps de filtrage important et masquer les signaux source d'amplitude trop faibles. Les procédés unipolaires présenteront, eux aussi, une composante continue, et leur gain sera atténué d'un coefficient 2. Les saturations de blocs, bien que réellement présentes, peuvent générer des oscillations parasites et diminuer les performances intrinsèques des systèmes.

Le nombre d'échantillons formant la source, doit au moins être égal à la précision souhaitée de tracé du déphasage. Ainsi pour 1 degré de précision, 360 échantillons seront nécessaires.

À titre d'exemple, les figures ci-dessous donnent le plan de Bode de $F(p) = \dfrac{5p}{(1+0.2p).(1+5p)}$ tracé d'une part avec Matlab, et d'autre part avec Prosimul (sans saturation des blocs).



Exemple de tracé de plan de Bode (Matlab)



Exemple de tracé de plan de Bode (Prosimul)

CONFIGURATION DES ÉCHANGES

PROSIMUL permet des échanges simultanés via liaison parallèle TOR et analogique et via liaisons séries (pour les grandes quantités d'échange de variables).

Les cartes sur port USB sont assimilées à des liaisons parallèles.

Selon le mode d'échange retenu, les temps d'échange avec les API peuvent varier. Une variable système enregistre le nombre total d'échange en liaison série (1 échange = lecture des sorties API et écriture des entrées API). En cas d'erreur de communication cette variable SYS_COMPTEUR_ECHANGE prend la valeur -1 et peut être testée.

L'accès à la configuration des échanges se fait à partir du navigateur de l'application « Configuration → Échanges ».

Configuration Échanges - Zone TOR

- Sélectionner si nécessaire le type de carte dans les types disponibles (nous contacter au besoin pour une intégration spécifique).
* Cartes EPC64100 et ACL7225 : avec pilote « PORT95NT » (compléments)
* Cartes PCI7432 : avec pilote « PCI-DASK » (adlink)
* Cartes PIO32IO[A] : avec pilote « DRIVERLINX »
* Cartes PISO-P32A32 : avec pilote « UNIDAQ » (ipcdas). **Sélectionner la carte PISO lors de l'installation du pilote**.
* Cartes FESTO EASYPORT sur port USB
* Cartes SEALEVEL 8227 sur port USB

- Indiquer l'adresse de la carte en décimal si nécessaire (cartes EPC64100). Pour les cartes pilotées avec DriverLinx, indiquer la configuration des cartes via " DriverLinx Control Panel ".

- Indiquer le nombre voies (modulo 16) lues et écrites via les cartes (selon raccordements). Les entrées cartes seront rangées dans les 1^{eres} entrées Prosimul et les sorties seront issues des 1ères sorties Prosimul.

Configuration Echanges - Zone Analogique

- Sélectionner si nécessaire le type de carte dans les types disponibles (nous contacter au besoin pour une intégration spécifique).

- Indiquer l'adresse de la carte en décimal.

- Sélectionner son mode fonctionnement (unipolaire, bipolaire, voies isolées,...) selon son type.

- L'exploitation des valeurs est effectuée via les **blocs fonctions** " Entrée Ana " et " Sortie Ana ".

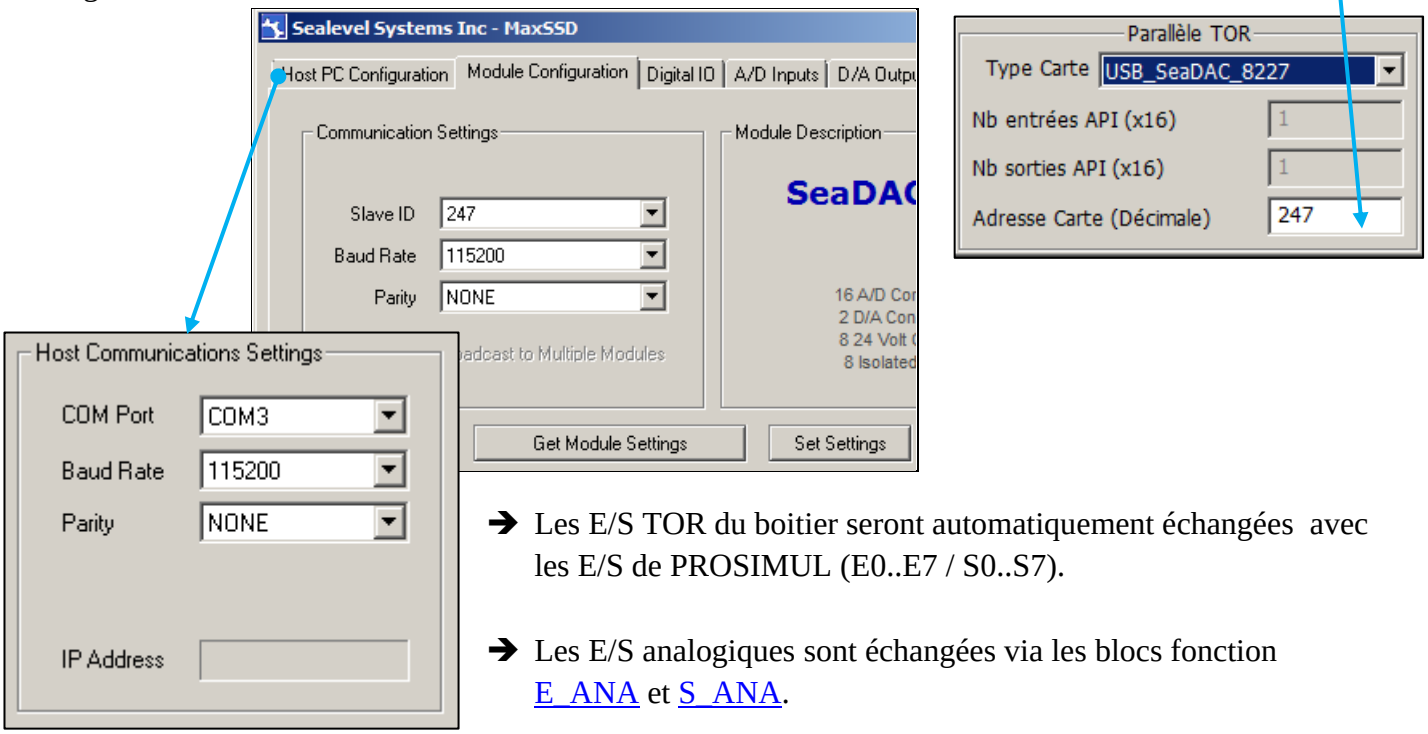
Cas des E/S sur port USB

- IMPORTANT : avant de connecter le matériel, **installer le pilote / driver** fourni par le fabricant.

(*) Configurer les options disponibles soit par switch (cnf doc fabricant) soit via le logiciel fourni.

(*) CARTE FESTO EASYPORT
Le boitier fonctionne par défaut à la vitesse de 115.200 bauds.
Le mode 0-10V ou +-10V se règle par switchs.
Le gain réglé sur switch devra être indiqué dans le bloc fonction E_ANA.

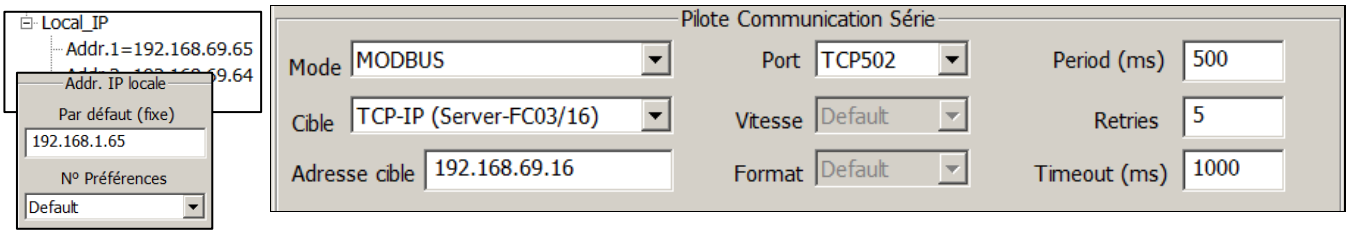
(*) CARTE SEALEVEL 8227 (voir pages 29 et suivantes du manuel SeaDAC User Manual)
- Régler éventuellement les switchs permettant de régler :
    * E1/E2 : sortie analogique 1 et 2 en 0-5V ou 0-10V (par défaut)
    * E3 : gain des entrées analogiques : 10 (signaux 0,5/1,0V) ou 1 (signaux 5/10V par défaut)
    * SW3/SW4 : entrées analogiques en boucle de courant ou en tension (par défaut)
- Puis lancer le logiciel MAXSSD.EXE (installé avec le pilote).
    * Sélectionner le port COMn qui est mappé sur le port USB
    * En 1er usage le boitier est configuré à une vitesse de 9.600 bauds. Pour améliorer les performances, celle-ci doit être modifiée à la valeur 115.200 bauds (laisser la parity à NONE).
    * Dans l'onglet « Module configuration » lancer la commande « Get Module Settings »
    * Modifier la vitesse à 115.200 baud et lancer la commande « Set Settings ». Après exécution, la vitesse du port USB sur PC doit aussi être modifiée (onglet Host PC Configuration)
    * Le slave ID peut être modifié. Il est renseigné comme « Adresse carte » dans PROSIMUL
    * Des tests de lecture / écritures d'E/S TOR et ANA peuvent être effectués via les autres onglets.



➔ Les E/S TOR du boitier seront automatiquement échangées avec les E/S de PROSIMUL (E0..E7 / S0..S7).

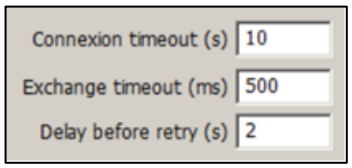
➔ Les E/S analogiques sont échangées via les blocs fonction E_ANA et S_ANA.

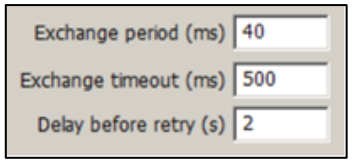
Paramètres de liaison série



- **Addr. IP locale** : « Défault » ou « Addr.1/2/3 ». Ce paramètre indique si l'adresse IP locale utilisée est une des 3 adresses sélectionnées dans le fichier des préférences ou l'adresse fixée par défaut à l'application. Si cette adresse est vide, ce sera la 1ère adresse physique de la carte qui sera prise.

- **Mode** : sélection du pilote d'échange série ou Ethernet ou spécifique constructeur.
  . Sans (pas d'échange série)
  . Modbus : RTU série ou Ethernet TCP/IP client**(C)** ou serveur**(S)**.
  . Unitelway : série
  . DF1-Direct : série
  . P3964-RK512 : série
  . Pilote Rockwell : spécifique via serveur RSLinx
  . Client OPC : spécifique via Serveur OPC
  . Pilote Applicom : spécifique via cartes et protocoles Applicom
  . S7-MPI : spécifique via S7Online sur bus MPI
  . EthernetIP client : client d'équipement Rockwell sur Ethernet **(C)**
  Adressage par fichier Nx (CIP-PCCC) ou par Tags (CIP-TAGS)
  . S7-ISO on TCP : client d'équipement Siemens sur Ethernet **(C)**
  . EthernetIP serveur : serveur pour module Ethernet Rockwell **(S)**





- **Cible** : sélection du type d'équipement distant pour le mode retenu.

- **Adresse cible** : adresse IP ou numéro ou nom de l'équipement distant.

- **Port** : ports TCP pour Ethernet(**C/S**) - COMx/USB avec/sans signaux de contrôle pour liaisons série.

- **Vitesse** : pour COM1 ou COM2 - 110 à 115200 bauds.
- **Format** : pour COM1 ou COM2 - 8 bits, 1 stop, parité paire/impaire/sans.

- **Exchange period (ms)** : pour liaisons série et Ethernet en mode « **client** »(**C**) : règle la périodicité des échanges (cycle lecture / écriture). Il est conseillé de régler la période « Process » sur la moitié de la période « Exchange ».

- **Connexion timeout (s)** : pour Ethernet en mode « **serveur** »(**S**) : temps d'attente en seconde de la connexion du client avant relance du pilote de communication.

- **Exchange timeout (ms)** : délai de détection d'un défaut de communication et relance du pilote.

- **Delay before retry (s)** : délai en seconde avant relance du pilote après timeout de connexion ou d'échange.

- **Retries** : pour liaisons série : nombre d'essai d'échange avant arrêt définitif du pilote.

- **Permute Octets** : permet la permutation des octets de poids forts et de poids faibles pour les E/S TOR et/ou numériques.

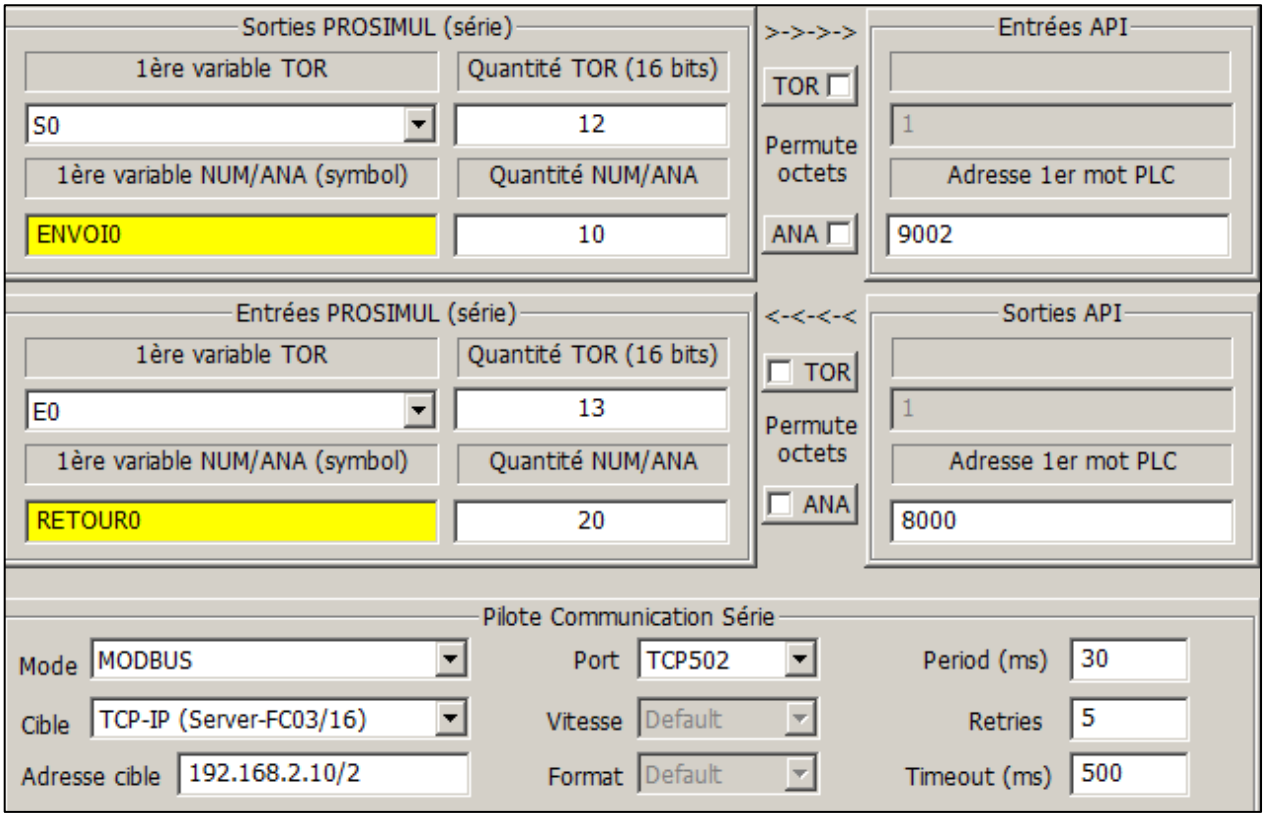
Protocole Modbus

- Liaison série RTU (vers cible Esclave)
Prosimul est **maître** des échanges. Indiquer le numéro d'esclave **S** accédé. Configurer les tables d'échanges avec les adresses Modbus de l'automate.

Note : pour les automates **APRIL série 1000** : le mot %IW0 est situé à un index négatif de -4353 par rapport à %MW0, %IW1 à -4354, %IW10 à -4355, etc. Utiliser une BFC Copy avec indexation pour charger les entrées automate. Préférer les échanges asynchrones pour la configuration du port Modbus APRIL (plus rapide).

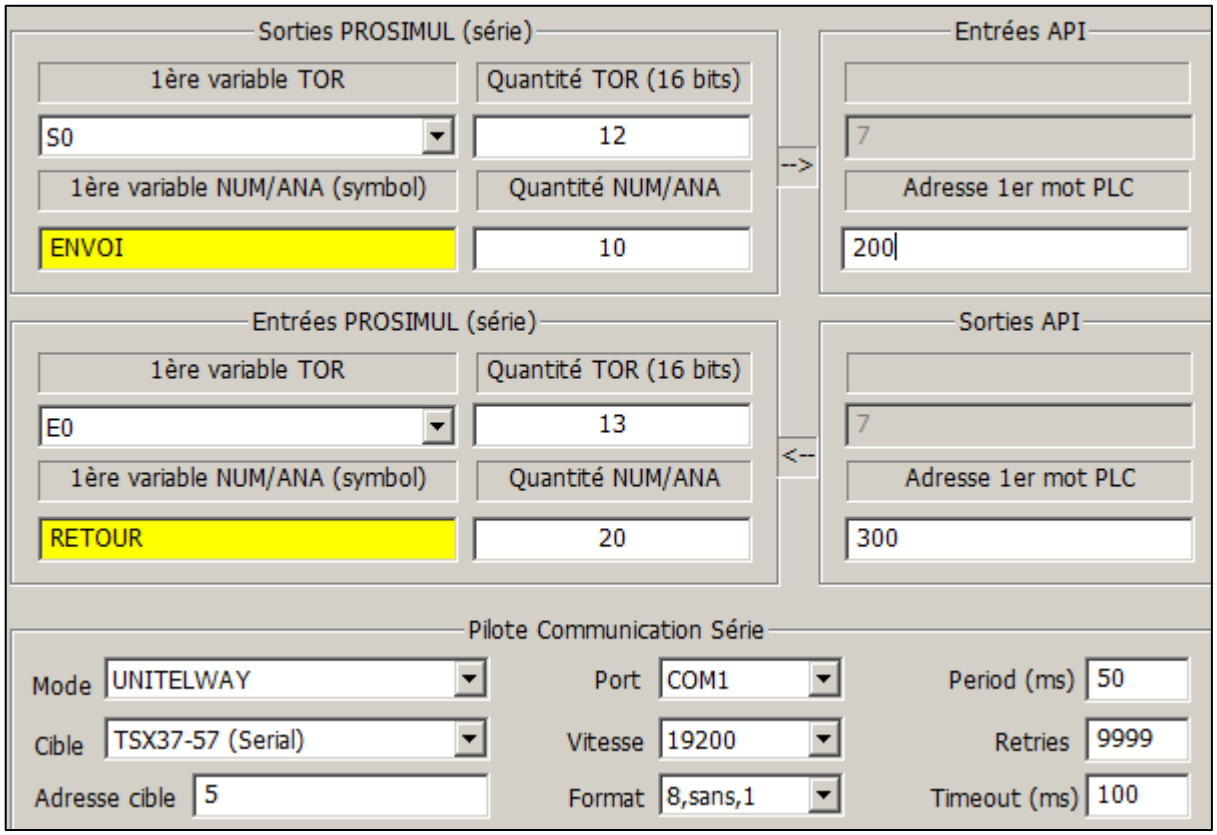
- Liaison TCP/IP Cible=Serveur (PROSIMUL est Client)
Sélectionner les codes requêtes utilisés FC03 (Read) / FC16 (Write) ou FC23 (Write + Read)
Indiquer l'adresse IP de l'équipement distant. Éventuellement indiquer derrière l'adresse IP (séparé par un caractère « / ») le numéro d'esclave (ex : 192.168.2.65/**2** pour esclave 2). Par défaut l'esclave 1 est choisi. Configurer les tables d'échanges avec les adresses Modbus de l'équipement cible. Régler le Timeout d'attente de réponse du serveur en millisecondes. L'adresse IP, les adresses lues/écrites et le nombre de mots doivent coïncider avec la cible.
L'ordre des octets peut être paramétré (0=poids faible en tête, 1=poids fort en tête).

- Liaison TCP/IP Cible=Client (PROSIMUL est Serveur)
Sélectionner les codes requêtes utilisés FC03/FC04 (Read) + FC16 (Write) ou FC23 (Write + Read)
Indiquer l'adresse IP de l'équipement distant client. Configurer les tables d'échanges avec les adresses Modbus ouvertes pour lecture/écriture. Régler le Timeout en millisecondes (délai d'attente des requêtes client). L'adresse IP, les adresses lues/écrites et le nombre de mots doivent coïncider avec la cible. La connexion initiale bénéficie d'un délai de 100 x Timeout après le passage en Run.



Protocole série Unitelway

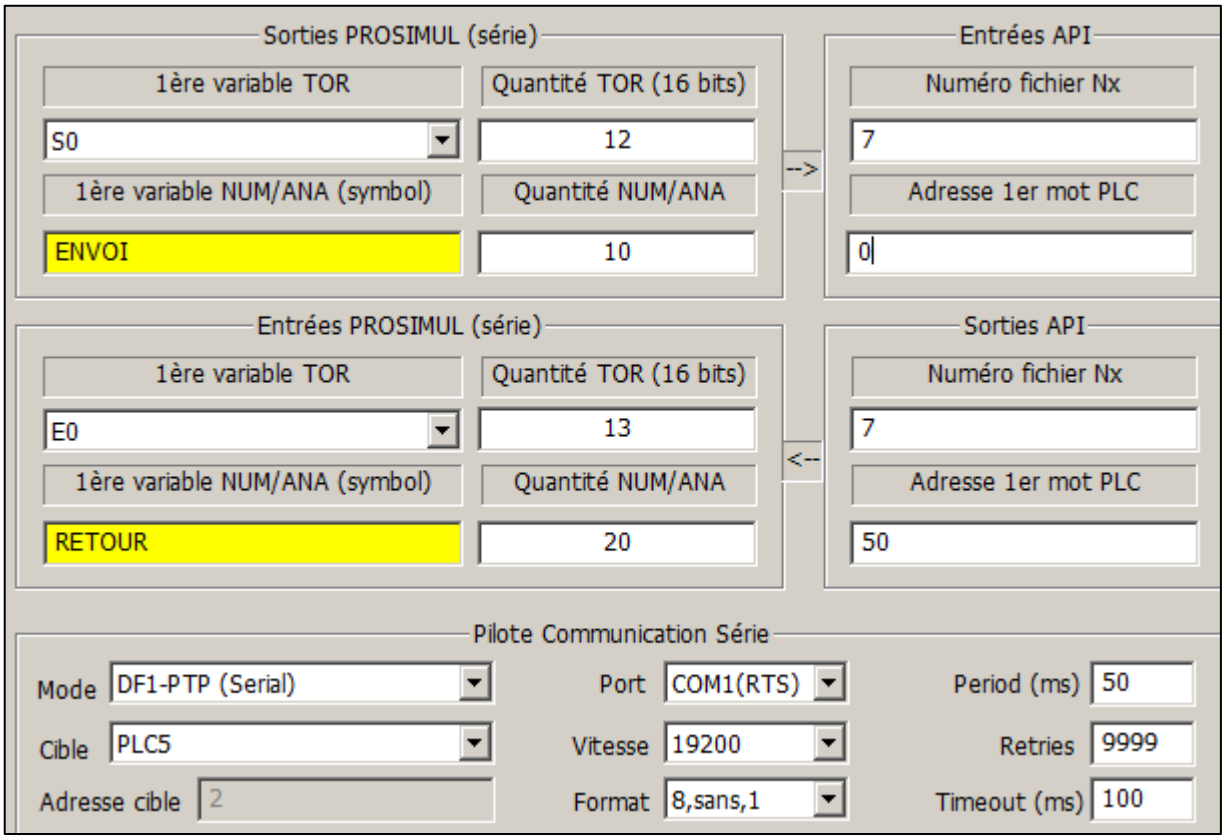
Prosimul est **maître** des échanges. Indiquer le type d'automate accédé TSX27-47 (Unitelway_1) ou TSX37-57 (Unitelway_2). Indiquer le numéro d'esclave **S** accédé. Configurer les tables d'échanges.



Note : pour configurer les automates TSX37-57 en esclave (Ad0=S) il est nécessaire, en plus de la configuration logicielle via PL7-Junior, d'utiliser un boitier TSXP ACC1.

Protocole série DF1 - Point à Point

Indiquer le type d'automate accédé (PLC2, PLC3, PLC5, PLC5XX, SLC5XX).
Configurer les tables d'échanges (voir spécificité automate ci-dessous).



Pour les plate-formes **ControlLogix** :

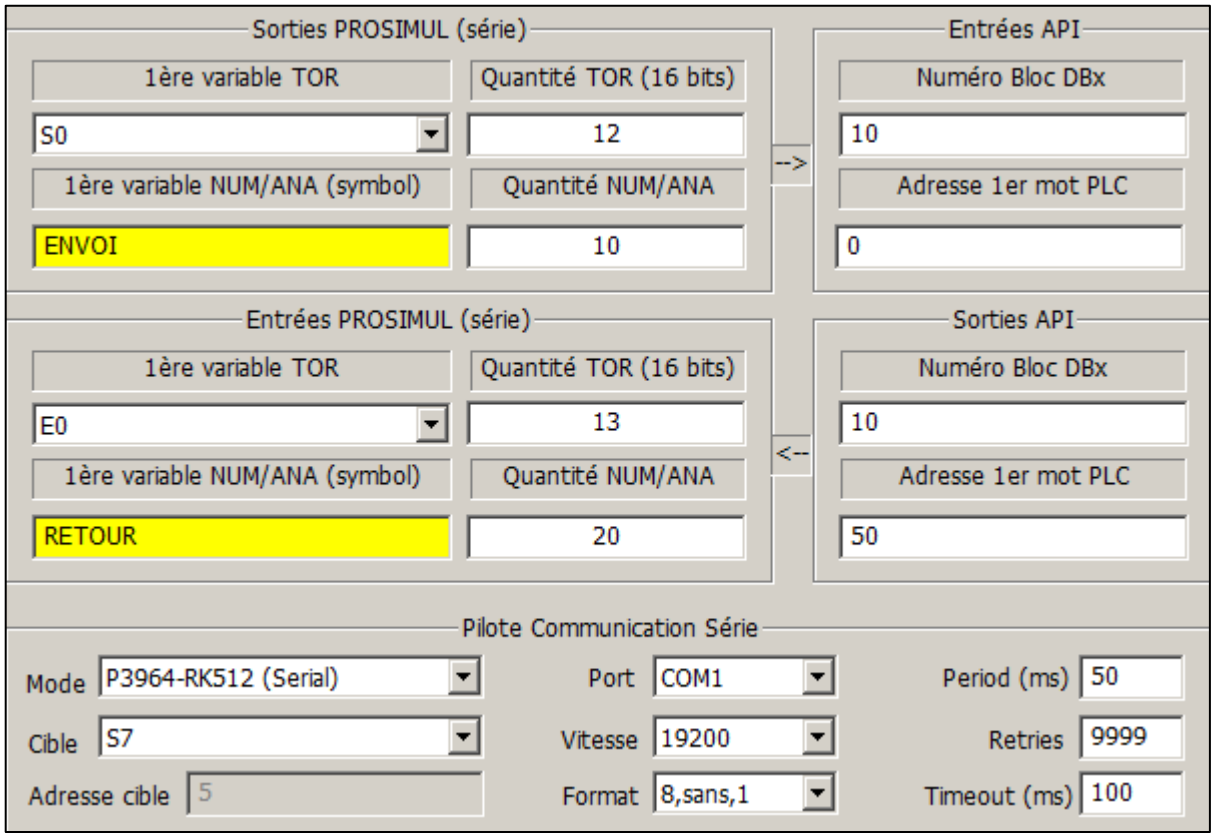
. Produire un " tag " de type tableau d'entier à la longueur souhaitée (ex : N7 de type INT[50]) via le logiciel RSLogix5000.
. Configurer l'option " **PLC2/3/5, SLC Mapping** " via le menu " Logic->Map PLC/SLC Messages ". Indiquer le tag produit dans les zones réservées au Mapping. Pour les automates PLC5/SLC le numéro de fichier accédé doit être de plus indiqué.

*Pour les automates **SLC5XX** le fichier d'échanges sera obligatoirement N9.*

*Note : configurer le port série côté automate avec détection d'erreur **BCC** et ne pas prendre en compte les numéros de message identiques (message 0 toujours utilisé).*

Protocole série P3964-RK512

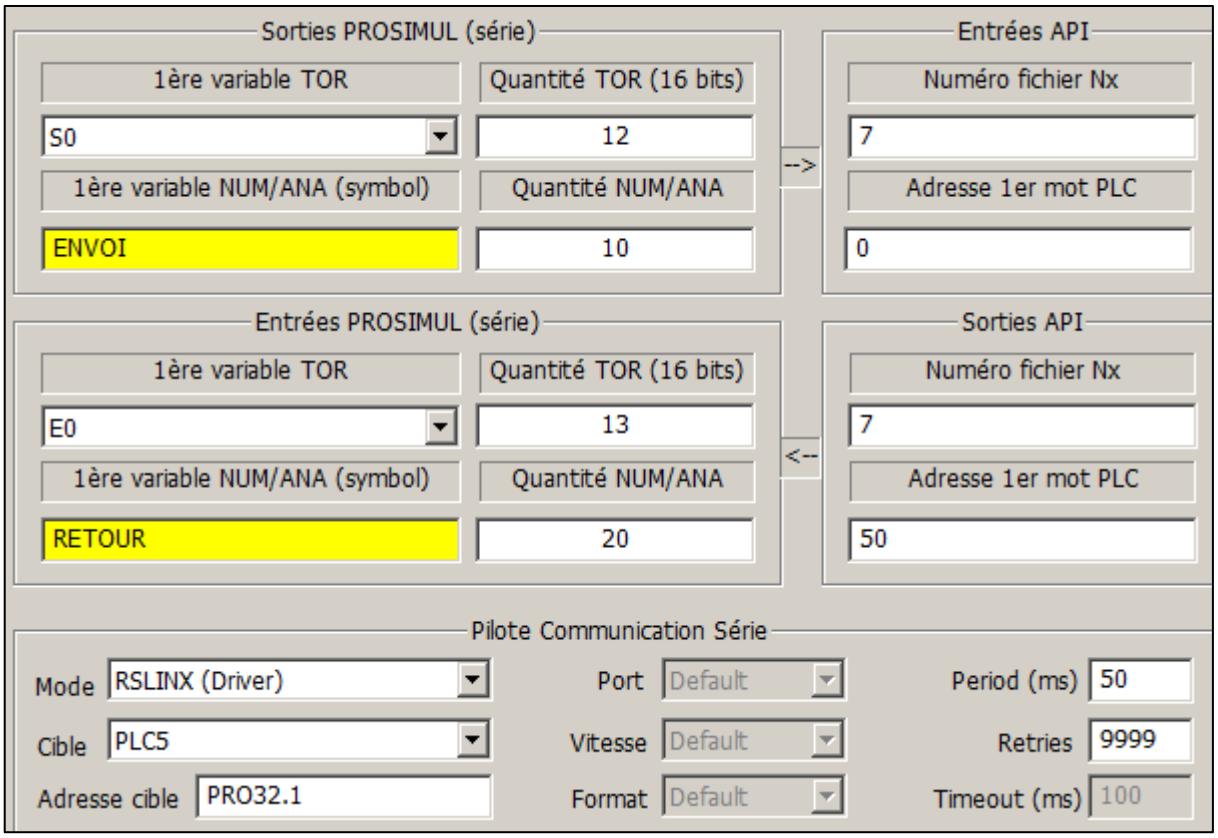
Indiquer le type d'automate accédé (Siemens S5 ou S7).
Configurer les tables d'échanges (voir spécificité automate ci-dessous).



*Note : pour simplifier la récupération des mots côté **automate S5** (dans bloc DBi), les mots numériques ont leurs octets Poids faible et Poids Fort inversés (arrivent dans les bon poids DR et DL du DB). Les E/S TOR arrivent à l'inverse et peuvent être envoyées ou issues directement des mots images EW et AW.*

Pilote Rockwell - RSLINX

Sur RSLINX, configurer un Driver (DF1, DH+,...) avec comme **Device Name** le nom de la cible choisie dans PROSIMUL.
Sur PROSIMUL, indiquer le type d'automate accédé (PLC2, PLC3, PLC5, PLC5XX, SLC5XX) et l'adresse de la cible pour ce Driver. L'adresse doit être désignée sous forme **NOM.X** (NOM du driver et X numéro de l'automate cible).
Configurer les tables d'échanges.



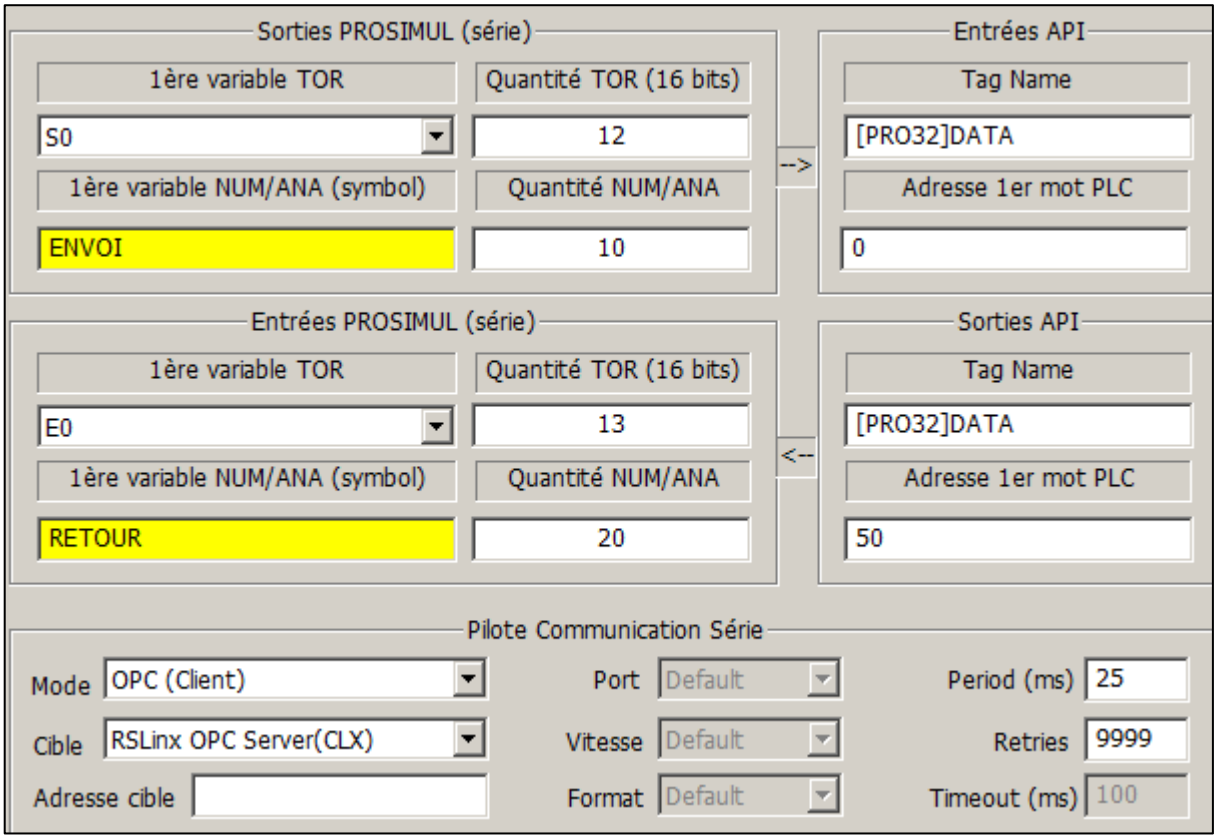
Pour les plate-formes **ControlLogix** :

. Produire un " tag " de type tableau d'entier à la longueur souhaitée (ex : N7 de type INT[50]) via le logiciel RSLogix5000.
. Configurer l'option " **PLC2/3/5, SLC Mapping** " via le menu " Logic->Map PLC/SLC Messages ". Indiquer le tag produit dans les zones réservées au Mapping. Pour les automates PLC5/SLC le numéro de fichier accédé doit être de plus indiqué.

Client OPC

1 – Dans PROSIMUL, sélectionner le serveur souhaité :

- **RSLinx OPC Server** : serveur OPC pour gammes Rockwell SLC/PLC ou CLX,
- **Schneider-Aut.OFS** : serveur OPC pour gammes Schneider (Modbus / XWAY)
- **OPC.SimaticNET** : serveur OPC pour gammes Siemens S5/S7.
- **APPLICOM.OPCServer** : serveur OPC (Applicom/Woodhead) multi-protocoles pour gammes Schneider/Siemens/Rockwell/…).
- **BR.OPC.Server_AR** : serveur OPC embarqué sur automate B&R.
- **Kepware.KEPServerEX.V5** : serveur OPC KEPWare.
- **Matrikon.OPC.Universal.1 / Matrikon.OPC.Siemens.1** : serveurs OPC MATRIKON.
- **CoDeSys.OPC.02** : serveur OPC pour logiciel CODESYS (émulateur PLC).



Indiquer éventuellement dans la zone « **Adresse cible** » l'adresse IP du serveur OPC distant ou laisser la zone de texte vide pour une adresse locale (127.0.0.1).

2 - Configurer le serveur OPC retenu (objectif : créer une cible nommée par exemple **PRO32**).

- **RSLINX** (Rockwell) : Menu DDE/OPC -> créer un “ Topic ” nommé **PRO32** et pointant sur l’automate désiré via le navigateur réseau. Régler le paramètre “ Polled Message ” (onglet Data Collection).

- **OFS** (Schneider) : utiliser l’ “ Outil de Configuration OFS ”. Créer un “ Alias ” nommé **PRO32** pointant sur l’adresse de la cible. Régler la période de groupe min (onglet Alias) et la vitesse d’échantillonnage en réception (onglet Comm).

- **Simatic NET** (Siemens) : avec STEP7 créer l’architecture de votre projet (réseau, sous-réseau, station S7, passerelle si nécessaire, et une seule et unique SIMATIC PC Station qui sera votre poste Prosimul). Configurer cette SIMATIC PC Station qui doit contenir au minimum un serveur OPC avec liaison S7 active (régler la période) et une carte de communication (IE général). Créer une liaison S7 nommée **PRO32** entre la station cible et la station OPC serveur. Localiser le fichier XDB résultant de la configuration de la SIMATIC PC Station (onglet Configuration des propriétés de l’objet). Uiliser l’outil “ Station Configurateur ” de SimaticNET pour importer le fichier de configuration XDB. Vérifiez que les services SIMATIC NET soit lancés.

- **APPLICOM.OPCServer** (multi-protocoles) : utiliser la console de configuration. Insérer la carte de communication adaptée à votre PLC (par exemple SW1000ETH pour une liaison Ethernet). Configurer le protocole (Modbus/S7/…) du canal utilisé (0 par défaut). Insérer un serveur (Topic) nommé **PRO32** pointant sur l’adresse de la cible. Régler le nombre de requêtes simultanées à 2 (Paramètres Messagerie) et la période de rafraichissement (Configuration Topics).

- **BR.OPC.Server_AR** : « à compléter »

- **Kepware.KEPServerEX.V5** : créer un « Channel » avec le « Device Driver » adapté et régler ses paramètres. Créer un « Device » nommé PRO32 par exemple et régler ses paramètres. Créer 2 « Tags » (1 pour les entrées et 1 pour les sorties) de type champ d’entier 16 bits et régler le « Scan Rate » en milliseconde.
Ex : Tag « Read » d’adresse « 400301[10] » (10 mots à partir de l’adresse 300 (301-1) dans les registres de sortie Modbus (valeur 400) de l’automate cible)

- **Matrikon.OPC.Universal.1 / Matrikon.OPC.Siemens.1** : créer un « Objet » avec le « Device Driver » adapté (ex : MODBUS) et régler ses paramètres. Créer une « Cible/Objet » nommé PRO32 par exemple et régler ses paramètres (adresse IP, port, timeout, …). Créer 2 « Alias » (1 pour les entrées et 1 pour les sorties) de type champ d’entier 16 bits et régler le « Update Rate » en milliseconde.
Ex : Alias « Read » « MODBUS.PRO32.**4** :301 :10 » (10 mots à partir de l’adresse 300 (301-1) dans les registres de sortie Modbus (valeur 4) de l’automate cible).

- **CoDeSys.OPC.02** :
Lancer le programme « PLCWinNT » (logiciel fourni avec Codesys).
Sur CODESYS : configurer la cible sur « PLCWinNT ». Créer les variables globales de type ARRAY [1..N] OF INT. Configurer un fichier de symbole (options) avec accès R/W. Régler la communication sur pilote « Tcp/Ip Level 2 Route » (gateway=local). Faire « En ligne », « Accéder au système » puis « Démarrer ».
Démarrer « Codesys OPC Configurator », régler l’Update rate du serveur, choisir le mode de connexion du PLCx à l’identique de celui de Codesys.

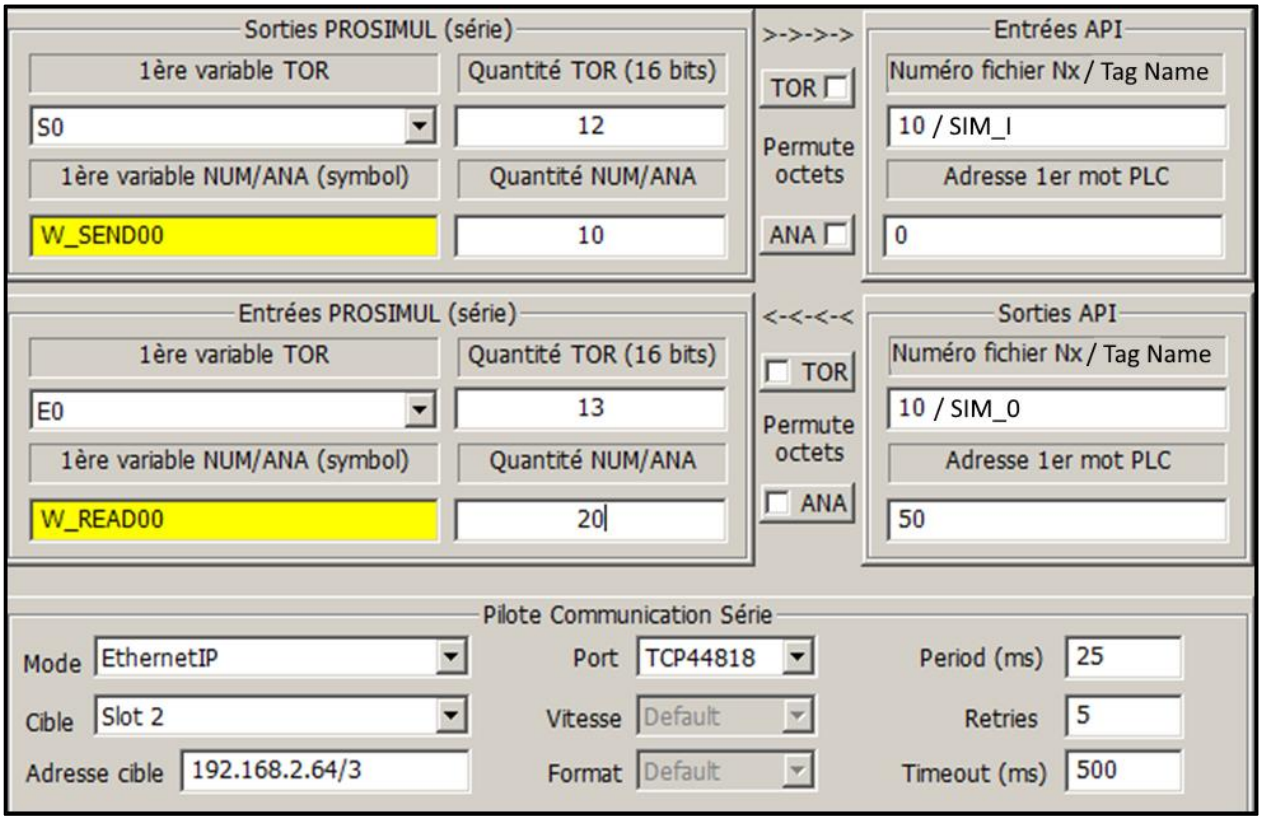
3 – Dans PROSIMUL, configurer les tables d'échanges. Dans la table d'échange les données cibles (champs DATA NAME – 15 caractères maximum) doivent être indiquées sous forme textuelle (le texte « PRO32 » ci-dessous est explicité dans le point 2) :

- pour RSLinx OPC Server : « **[PRO32]VarName** » par exemple (VarName est le nom des données accédées dans l'automate)
- pour Schneider-Aut.OFS : « **PRO32_!%MW** » par exemple
- pour OPC.SimaticNET : « **[PRO32]DB10.DBW** » par exemple
- pour APPLICOM OPCServer : « **PRO32.VarName** » (VarName est le nom des données accédées pour l'API concerné).
- pour BR.OPC.Server_AR : « **en attente** »
- pour Kepware.KEPServerEX.V5 : « **ChannelName.PRO32.TagName** »
- pour Matrikon.OPC.Universal.1 et Matrikon.OPC.Siemens.1 : « **Alias** »
- pour CoDeSys.OPC.02 : « **PLCx :.VarGlobaleName** ». Les longueurs de table (binaire+numérique) doivent correspondre aux longueurs des tableaux de variables OPC.

Indiquer si nécessaire en complément l'adresse de la première variable accédée (sauf KEP, B&R MATRIKON, CODESYS – cette adresse est incluse dans la configuration OPC).

Pilote EthernetIP client (CIP-PCCC / CIP-TAGS)

Choisir le type de cible (position de la CPU) :
- « Direct UC » pour les **modules ENI ou coupleurs intégrés sur SLC/PLC** (CIP-PCCC seulement).
- « Slot 0..19 » pour les ports sur **CLX ou modules déportés** (l'UC doit être dans le slot 0 à 19).



Indiquer l'adresse IP de la cible sur Ethernet. Optionnellement, indiquer derrière l'adresse IP (séparé par un caractère « / ») le numéro de « channel » de la connexion (ex : 192.168.2.65/**3** pour le channel 3). Par défaut le channel 1 est choisi. Au besoin configurer des routes statiques sur l'ordinateur (route add).

Configurer les tables d'échanges. Le nombre total de variables TOR+NUM est limité à 250.

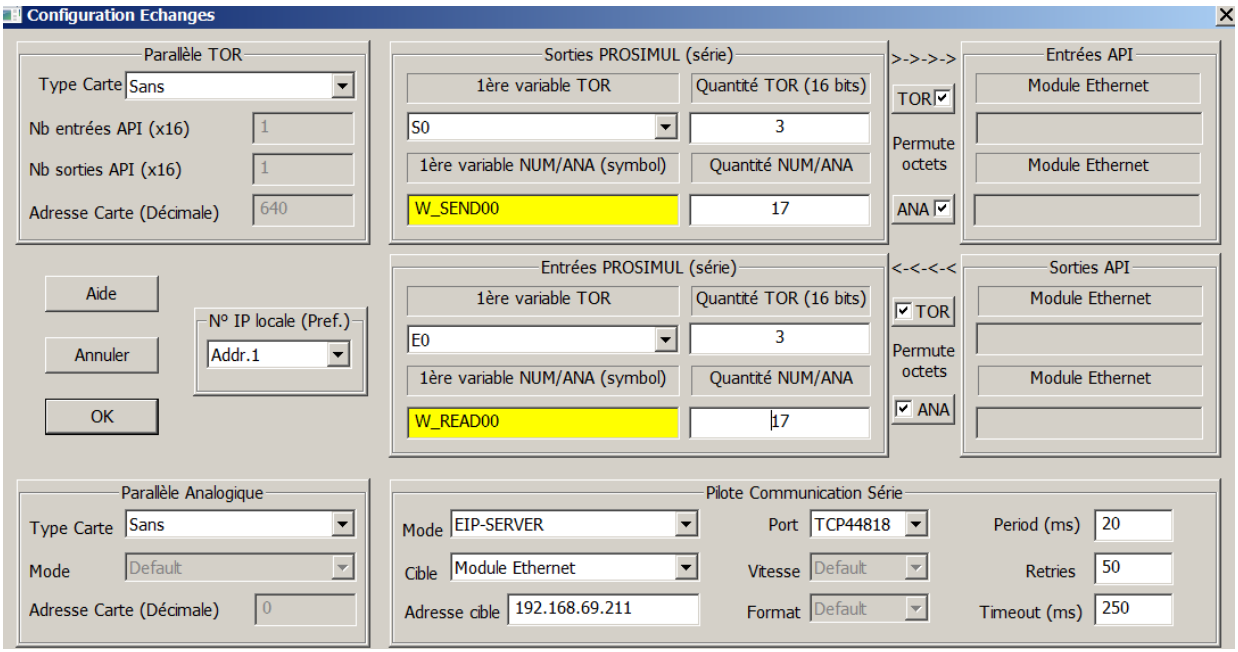
Pilote CIP-PCCC : seuls les fichiers Nx sont accessibles.
Pilote CIP-TAGS : seuls les Tags sont accessibles.

Pour les plate-formes **ControlLogix** en mode **CIP-PCCC** :

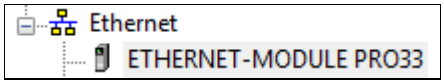
. Produire un " tag " de type tableau d'entier à la longueur souhaitée (ex : N7 de type INT[50]) via le logiciel RSLogix5000.
. Configurer l'option " **PLC2/3/5, SLC Mapping** " via le menu " Logic->Map PLC/SLC Messages ". Indiquer le tag produit dans les zones réservées au Mapping. Pour les automates PLC5/SLC le numéro de fichier accédé doit être indiqué.

Pilote EthernetIP serveur

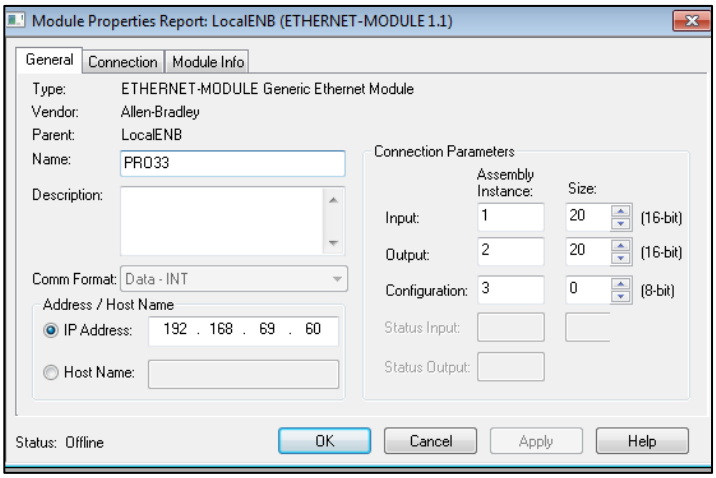
Ce pilote permet de répondre aux **modules Ethernet** configurés dans les automates CLX Rockwell.

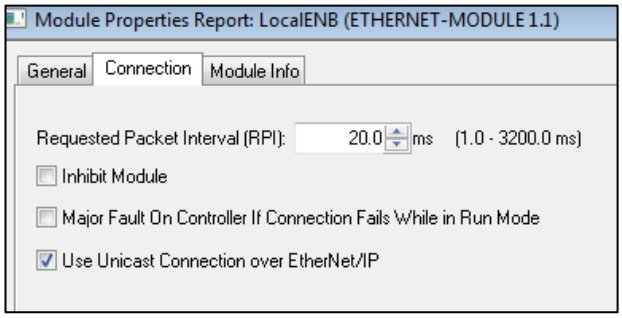


- **Dans PROSIMUL**, indiquer l'adresse IP de l'automate client sur Ethernet et le port TCP utilisé (44818 par défaut). Optionnellement, indiquer derrière l'adresse IP (séparé par un caractère « / ») le numéro de port UDP utilisé (2222 par défaut).
  Si des connexions sont effectuées avec **plusieurs applications** PROSIMUL hébergées sur le même PC, il faut configurer plusieurs adresses IP sur la carte Ethernet, indiquer dans les **préférences** les adresses disponibles (3 maxi), et sélectionner l'IP locale utilisée.
  Configurer les tables d'échanges. **Le nombre total de variables TOR+NUM est limité à 250.**
  Il est souhaitable que l'application soit configurée en priorité 6 (temps réel).



- **Dans le programme automate** configurer le module Ethernet qui séquencera les communications.
  Indiquer l'adresse IP de l'application PROSIMUL et configurer les longueurs de tables I/O (**16 bits**) à l'identique de PROSIMUL (ex : 17+3 = 20).





Dans l'onglet « Connection » vérifier que la case « **Use Unicast Connection** » soit bien cochée et indiquer la période de mise à jour des I/O.

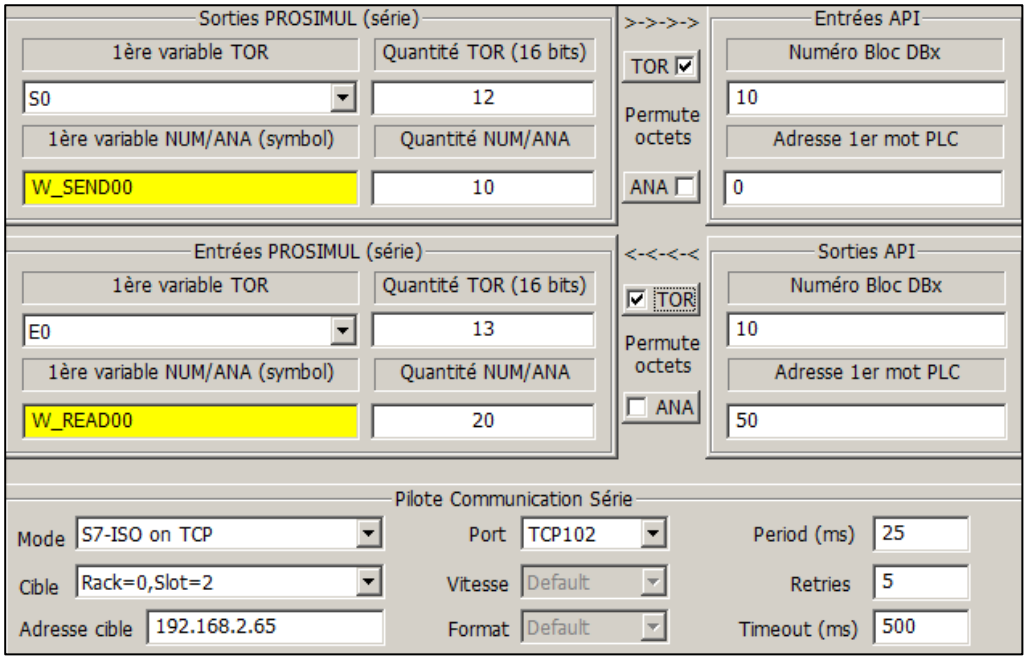
Pilote S7-ISO on TCP

Choisir le n° de slot de l'UC ciblée.

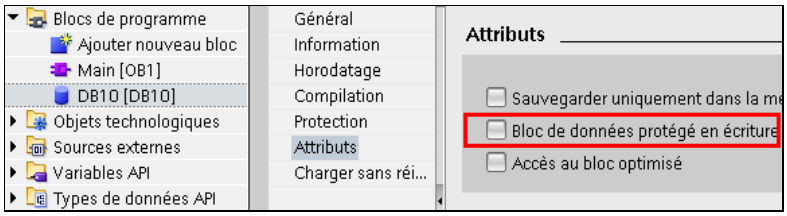
Sélectionner l'adresse IP de la voie Ethernet. Au besoin configurer des routes statiques sur l'ordinateur (route add).

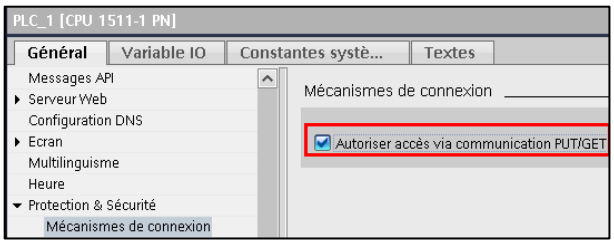
Configurer les tables d'échanges.

Seuls les fichiers **DBx** sont accessibles.



Pour les séries 1200-1500 l'attribut DBx « Accès au bloc optimisé » ne doit pas être sélectionné.
Il faut aussi sélectionner l'option « Autoriser accès GET/PUT » dans la rubrique « Protection & Sécurité » / « Mécanisme de connexion) de la CPU.





Utilisation de l'émulateur PLCSim

PROSIMUL peut aussi être utilisé avec l'émulateur PLCSim de SIEMENS.
1 – Avant tout lancement de logiciel SIEMENS, démarrer NetToPLCSim V1.2.4.
2 – Démarrer STEP7 ou TIA-PORTAL et charger l'application dans PLCSim (driver Ethernet).
3 – Configurer ou charger la configuration PLCSim dans NetToPLCSim
Ex : PC :192.168.2.212 → PLCSim :192.168.2.215 / Port 102
4 – Démarrer le serveur NetToPLCSim.
5 – Dans PROSIMUL, sélectionner le driver S7-ISO on TCP avec l'adresse configurée dans NetToPLCSim (ex: 192.168.2.212) ainsi que le port TCP102.

Pilote Applicom

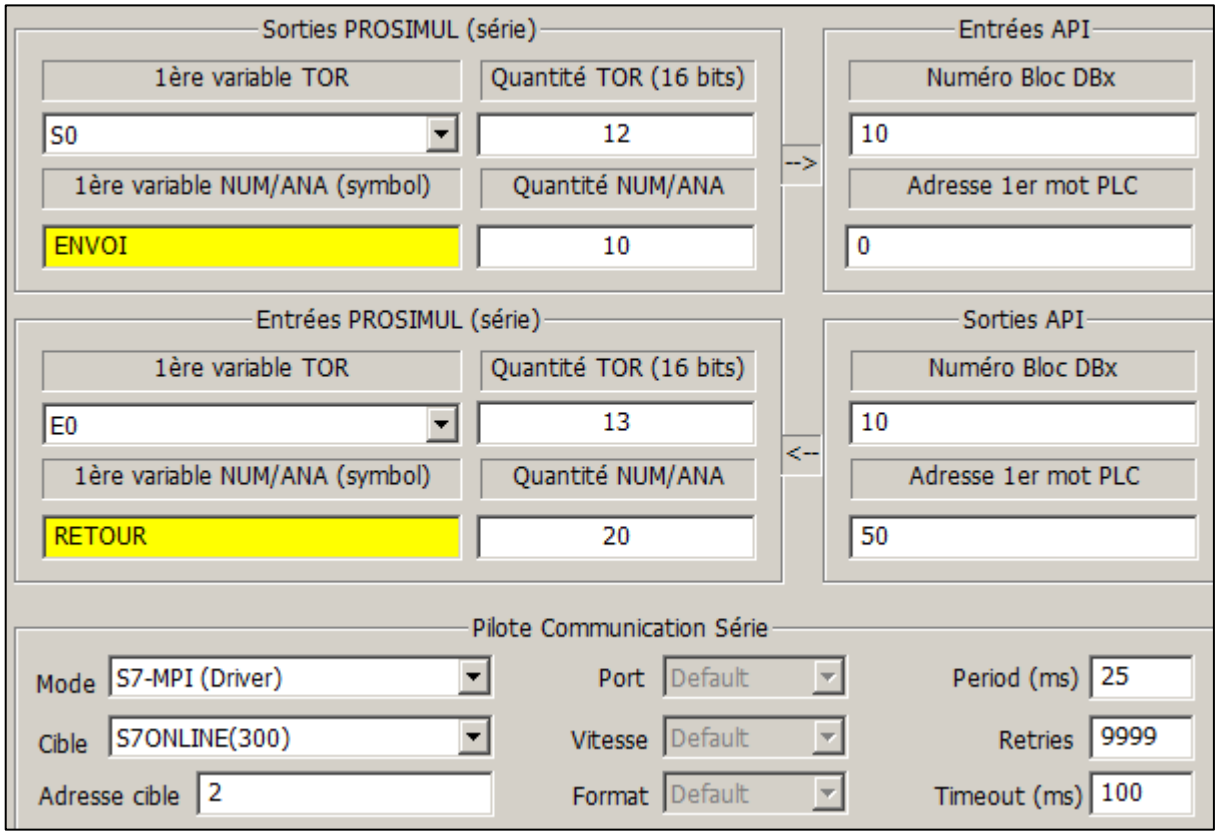
Les échanges seront effectuées avec la DATABASE Applicom (carte physique ou logiciel Direct-Link). Des échanges cycliques doivent être programmés avec l'outil **PCCYCL**. Les échanges sont cycliques lancés via la commande PCINIT.
Configurer les tables d'échanges en indiquant le numéro de carte accédée et les adresses de mots situés dans la DATABASE.

Pilote S7-MPI (via PG/PC interface)

Les échanges seront effectués via la liaison **S7ONLINE** configurée par le " Paramétrage interface PG/PC " du Panneau de configuration de STEP7.

Sélectionner le type d'automate S7-200 ou S7-300/400. Pour les **S7-200**, seul le **DB1** est accessible. Indiquer le numéro d'automate accédé (MPI ou PROFIBUS).

Configurer les tables d'échanges.

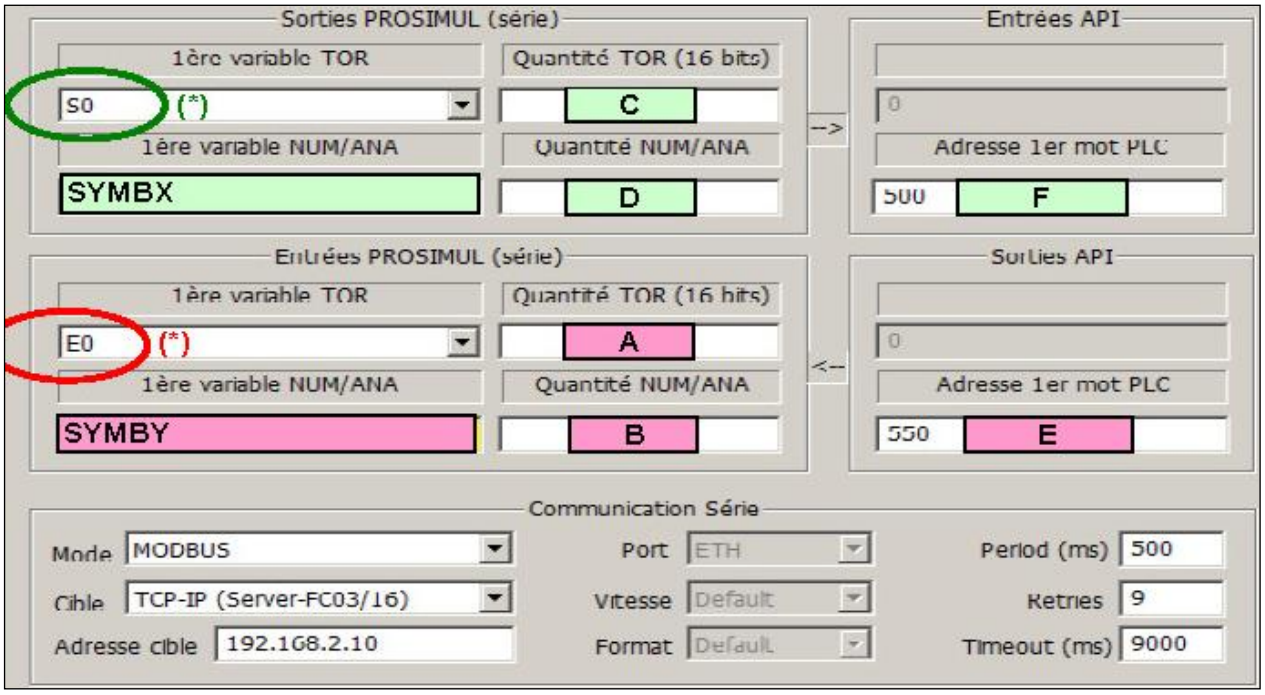


Principe des tables des échanges

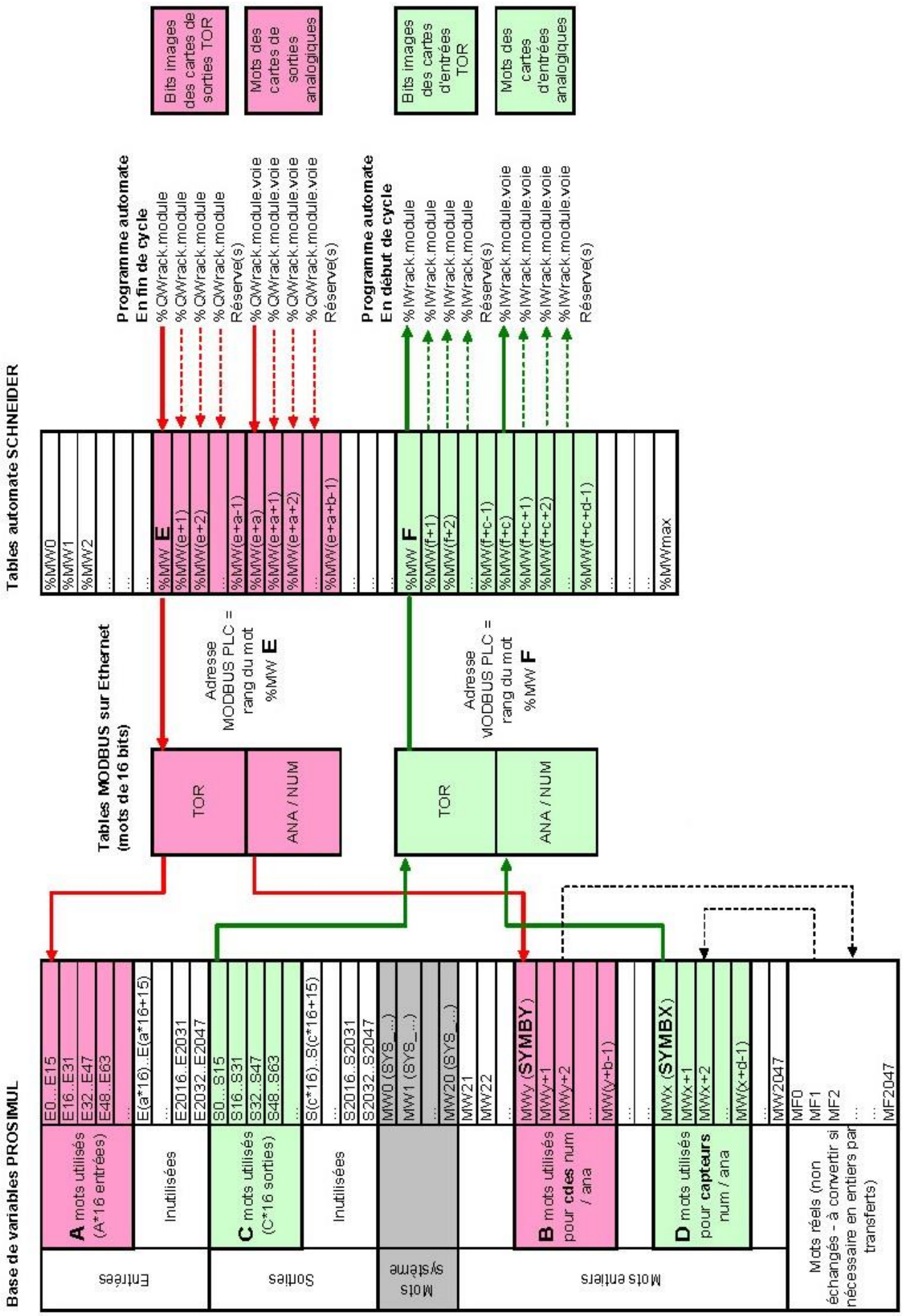
Les échanges seront effectués via des tables d'échanges devant être configurées.
Les **sorties API** seront lues dans l'automate depuis une table unique constituée des Sorties TOR puis des Sorties Numériques (16 bits).
Les **entrées API** seront écrites dans l'automate dans une table unique constituée des Entrées TOR puis des Entrées Numériques (16 bits)
Voir schéma page suivante.

Configuration des tables

Paramètre	**Zone entrées API**	**Zone Sorties API**
Nombre de mots TOR	Quantité **C**	Quantité **A**
Nombre de mots ANA	Quantité **D**	Quantité **B**
Table Prosimul ANA	Symbole associé au mot **Y**	Symbole associé au mot **X**
N° bloc de données DBx	Pour SIEMENS	Pour SIEMENS
N° fichier Nx	Pour ROCKWELL	Pour ROCKWELL
N° carte	Pour Applicom	Pour Applicom
Tag Name	Pour client OPC	Pour client OPC
Adresse mot PLC	Adresse **F**	Adresse **E**



La limite par défaut de chacune des 2 tables est de 127 mots (TOR+NUM/ANA).
Le driver EIP permet d'accéder jusqu'à 254 mots en 2 échanges synchronisés.


Bits images des cartes de sorties TOR
Mots des cartes de sorties analogiques
Bits images des cartes d'entrées TOR
Mots des cartes d'entrées analogiques
Tables automate SCHNEIDER
Programme automate En fin de cycle
Programme automate En début de cycle
%MW0
%MW1
%MW2
%MW E
%MW(e+1)
%MW(e+2)
%MW(e+a-1)
%MW(e+a)
%MW(e+a+1)
%MW(e+a+2)
%MW(e+a+b-1)
%MW F
%MW(f+1)
%MW(f+2)
%MW(f+c-1)
%MW(f+c)
%MW(f+c+1)
%MW(f+c+2)
%MW(f+c+d-1)
%MWmax
%QWrack.module
%QWrack.module.voie
%IWrack.module
%IWrack.module.voie
Réserve(s)
Tables MODBUS sur Ethernet (mots de 16 bits)
Adresse MODBUS PLC = rang du mot %MW E
Adresse MODBUS PLC = rang du mot %MW F
TOR
ANA / NUM
Base de variables PROSIMUL
Entrées
Sorties
Mots système
Mots entiers
A mots utilisés (A*16 entrées)
Inutilisées
C mots utilisés (C*16 sorties)
B mots utilisés pour cdes num / ana
D mots utilisés pour capteurs num / ana
Mots réels (non échangés - à convertir si nécessaire en entiers par transferts)
E0...E15
E16..E31
E32..E47
E48..E63
E(a*16)..E(a*16+15)
E2016..E2031
E2032..E2047
S0...S15
S16..S31
S32..S47
S48..S63
S(c*16)..S(c*16+15)
S2016..S2031
S2032..S2047
MW0 (SYS_...)
MW1 (SYS_...)
MW20 (SYS_...)
MW21
MW22
MWy (SYMBY)
MWy+1
MWy+2
MW(y+b-1)
MWx (SYMBX)
MWx+1
MWx+2
MW(x+d-1)
MW2047
MF0
MF1
MF2
MF2047

EXPLOITATION

Le lancement de l'application se fait via la commande " **Run** " du menu principal.
Les options de configuration d'échanges et d'application doivent être réglées au préalable.

Sur **passage en RUN** la variable SYS_INIT_APPLI passe à 1 pendant le 1er cycle.
Sur **passage en STOP** la variable SYS_CYCLE_PROSIMUL prend la valeur **-1** pendant le dernier tour de cycle pour sauvegardes éventuelles des états de variables (voir WRITEINI).

Des actions sont possibles en RUN depuis tous les écrans via les touches fonction associées à des variables binaires. Des fonctions de forçages et réglages de variables sont accessibles via les différents écrans de Graphique, Variables, Ladder, Grafcet et Fonctions en visualisation dynamique.

Commandes disponibles dans les menus

Init Appli ([ALT][CTRL][F05]) : provoque la réinitialisation complète de l'application.
Init Grafcet : provoque l'initialisation du Grafcet seul.
Init Temps : provoque la Raz des indicateurs de temps maximum affichés dans la barre d'état.

Forçage -> Raz ([ALT][CTRL][F09]) : annule l'ensemble des forçages actifs.
Forçage -> Liste : accès à la liste des forçages actifs

Dictionnaire : permet de basculer dynamiquement les libellés d'objets graphiques selon les langues présentes dans le dictionnaire.

Priorité : permet de modifier dynamiquement la priorité Windows des Threads PROSIMUL.

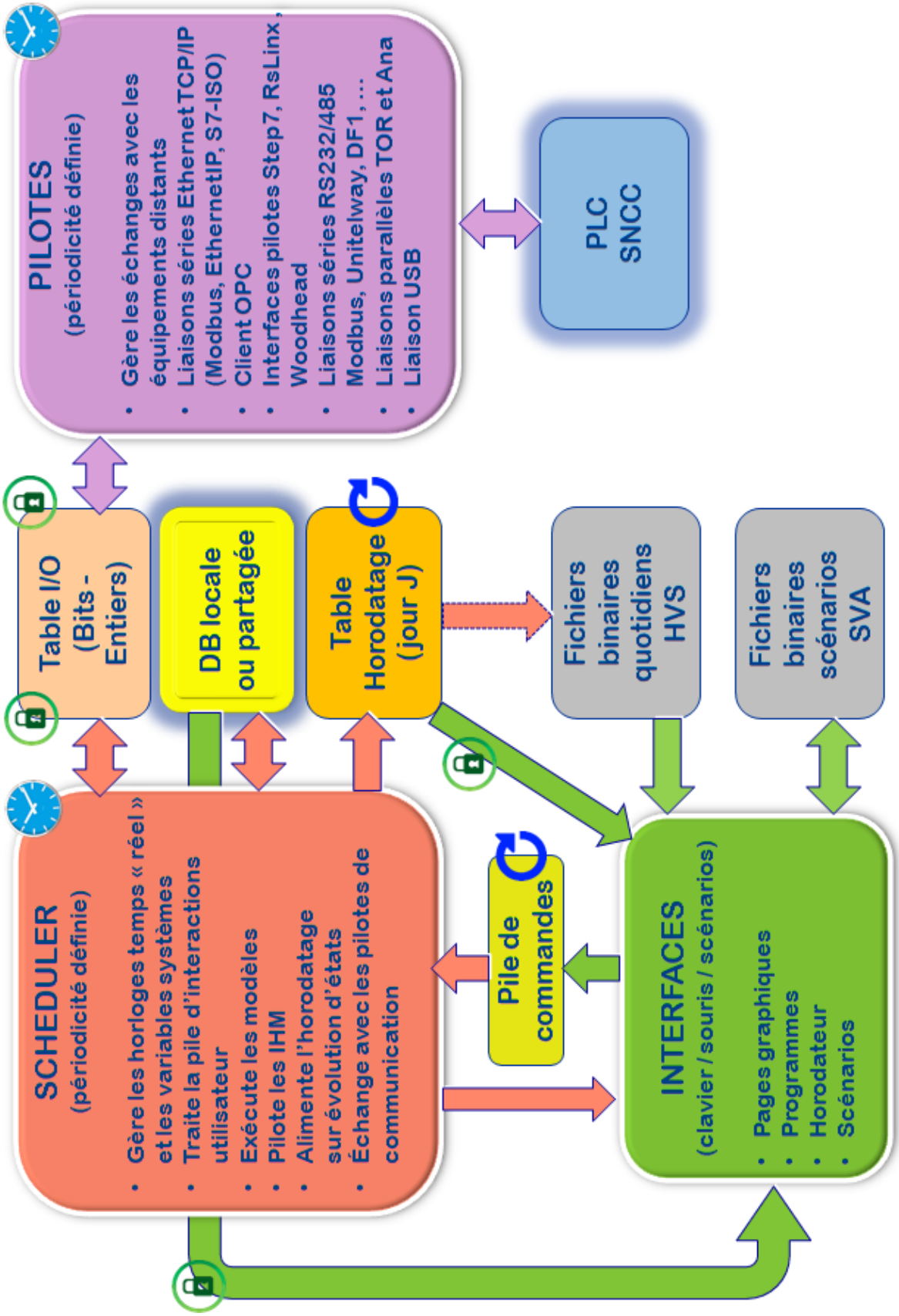
Run/Stop : sortie du mode RUN (possible avec 2 appuis rapprochés (<1s) sur la touche [**ESC**]).
Le passage en RUN avec la touche [SHIFT] lance l'application sans le navigateur.

Window : permet un basculement entre applications PROSIMUL ouvertes (jusqu'à 3 applications en simultané). **Sélection écran** déplace la fenêtre principale sur l'écran suivant si disponible.

Lock / UnLock : permet de verrouiller / déverrouiller les actions sur l'application courante. Si l'application est protégée, le mot de passe est demandé.

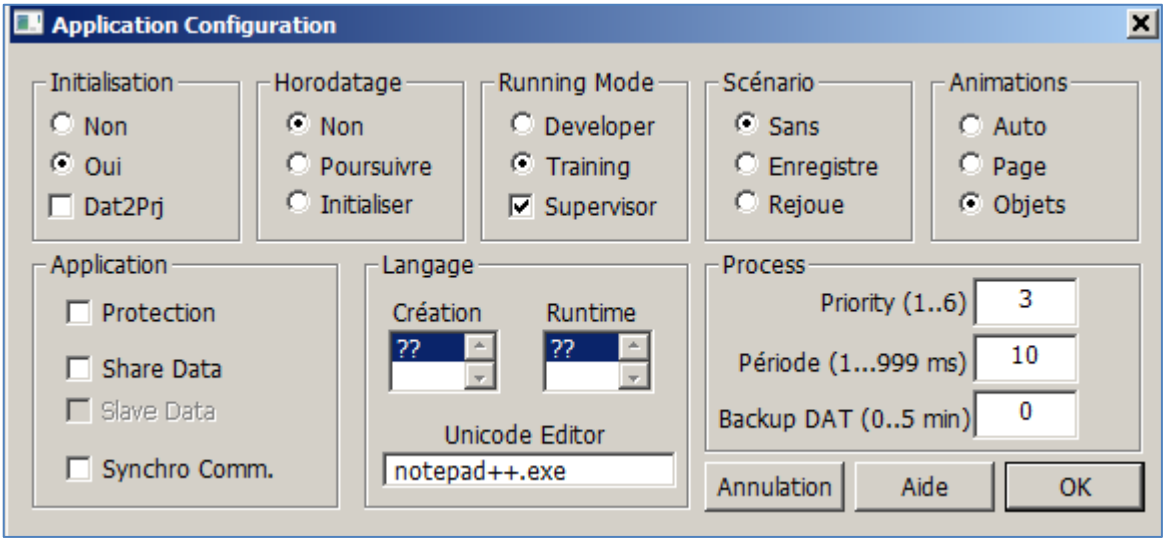
Barre d'état : différents états sont visualisés chaque seconde dans la barre d'état.
Temps de fonctionnement depuis le passage en RUN : 00:00:40 [F15] Uc[13.6%]
+ Nombre de variables forcées : **[Fxx]** → en cas de forçage présent le fond passe en jaune
+ Pourcentage d'usage du processeur.
Périodes du programme et de communication courantes et maxi : P[250<251ms] C[240<260ms]
(fond rouge en cas de défaut de communication). La commande « Init.temps » réinitialise les temps maxi affichés.

Visudyn (menus secondaires) : accès à la liste des variables surveillées ou à leur enregistrement si actif (accessible aussi via le navigateur « Outils » « Horodatage » ou via un clic droit sur une zone vide d'une page graphique).


PILOTES
(périodicité définie)
• Gère les échanges avec les équipements distants
• Liaisons séries EthernetTCP/IP (Modbus, EthernetIP, S7-ISO)
• Client OPC
• Interfaces pilotes Step7, RsLinx, Woodhead
• Liaisons séries RS232/485 Modbus, Unitelway, DF1, ...
• Liaisons parallèles TOR et Ana
• Liaison USB
PLC
SNCC
Table I/O
(Bits - Entiers)
DB locale
ou partagée
Table
Horodatage
(jour J)
Fichiers
binaires
quotidiens
HVS
Fichiers
binaires
scénarios
SVA
SCHEDULER
(périodicité définie)
• Gère les horloges temps « réel » et les variables systèmes
• Traite la pile d'interactions utilisateur
• Exécute les modèles
• Pilote les IHM
• Alimente l'horodatage sur évolution d'états
• Échange avec les pilotes de communication
Pile de
commandes
INTERFACES
(clavier / souris / scénarios)
• Pages graphiques
• Programmes
• Horodateur
• Scénarios

Configuration de l'application

L'accès à la configuration de l'application se fait à partir du navigateur de l'application " Configuration->Application ".



- **Initialisation** : possibilité d'initialiser les variables avec leurs valeurs initiales ou de lancer l'application avec les valeurs courantes (reprise de tests). La case à cocher « **Dat2Prj** » permet, à l'ouverture du fichier, d'initialiser les variables internes non forcées avec les valeurs contenues (reconnaissance par symbole) dans le fichier NOMAPPLI.DAT du répertoire courant.

- **Horodatage** : possibilité de valider l'enregistrement sur changement d'état des variables configurées via le menu horodatage. Dans la positive un fichier **NOMAPPLI_date.HVS** est créé chaque jour et pourra être exploité. Ce fichier peut être écrit en continu ou réinitialisé au passage en RUN.

- **Scénario** : possibilité d'enregistrer ou de rejouer des scénarios. Ceux-ci contiendront l'ensemble des actions manuelles effectuées sur l'application (réglages / forçages / ...).

- **Running Mode** : possibilité de démarrer l'application en mode Développement ou Formation.
MODE DEVELOPER : mode normal. Affichage de tous les objets et du navigateur.
Le passage en RUN avec la touche [SHIFT] lance l'application sans le navigateur.
MODE FORMATION : les objets graphiques paramétrés « Dev.only » ne seront pas affichés. Le navigateur n'est pas affiché non plus et les changements de page se feront par les touches [PgUp] et [PgDn] ou la variable SYS_PAGE_PROSIMUL. Le lancement de l'application .SIM par double clic provoque son passage en RUN automatique.
Supervisor mode : évite le rafraichissement des objets « Image » en arrière-plan (gain de temps de cycle) et n'affiche pas le menu « Graphique » en RUN. Les objets superposés doivent être immobiles. La navigation [PgUp] et [PgDn] ne fonctionne pas et doit être prévue par programme via la variable SYS_PAGE_PROSIMUL.

- **Process priority** : permet de régler la priorité « Windows » des Threads PROSIMUL.
- 1 : priorité minimale pour le Runtime et la Communication
- 2 : priorité inférieure à la normale pour le Runtime et la Communication
- **3** : priorité **normale** pour le Runtime et la Communication
- 4 : priorité supérieure à la normale pour le Runtime et la Communication
- 5 : priorité maximale pour le Runtime et la Communication
- 6 : priorité « temps réel » pour le Runtime et pour la Communication – précision d'une ms visée pour le programme mais **le % d'usage du CPU augmente**.

- **Process period** : permet de régler la périodicité d'exécution de l'application (1 = maxi ; 999 = mini). Cela permet notamment de dégager du temps pour d'autres applications. Il est conseillé de régler la période « Process » sur la moitié de la période « Communication ».

- **Animations** : permet de régler le mode de rafraîchissement des objets qui évoluent selon le meilleur temps de réponse. Le mode « Page » rafraîchit l'ensemble des objets de la page (intéressant pour les grandes images). Le mode « Objets » ne redessine que les objets évoluant et leurs arrière-plans (intéressant pour les petits objets). Le mode « Auto » sélectionne le meilleur mode au vu des 40 premières animations. En mode « Page » ou pour les objets sur fond invisible l'antialiasing est dégradé mais l'affichage est plus rapide.

Si une taille limite de synoptique est indiquée (outil Grille) et que le mode choisi est « Page », la taille des graphique s'adaptera automatiquement à la taille de l'écran utilisé.

- **Langage** : permet de régler les langages en mode Création et en mode Runtime selon les langues du dictionnaires intégré (FR,EN, DE, …). Si les langues de Création et Runtime sont différentes une traduction est effectuée au lancement du Runtime (voir Menu Principal / Dictionnaire).

- **Éditeur Unicode** : permet d'indiquer l'exécutable utilisé pour l'affichage du dictionnaire (en format Unicode) et des exports de données (par exemple : Wordpad, Word, Excel, Notepad++, …).

- **Protection** : Permet de valider ou d'invalider la protection des projets .SIM par mot de passe (6 caractères maxi). Si la protection est choisie, le mot de passe est demandé à chaque ouverture d'application.

- **Share Data** : Permet de partager la base de données de variables entre plusieurs applications PROSIMUL, 1 maître et 2 esclaves (**Slave Data**) maxi. Les applications lancées travailleront en R/W sur la même base mais seul le maître y rangera ses symboles (normalement identiques sur chaque application).

- L'option « **Synchro Comm.** » permet de cadencer les **échanges** via un jeton circulaire. Disponibles pour pilotes Ethernet seulement.

- **Process backup** : permet de régler la périodicité du backup des données en fichier NOM_APPLI.DAT (0 = pas de backup ; 1 à 5 minutes). Ces données peuvent être utilisées pour initialiser les données d'une application modifiée ayant les mêmes symboles.

Visualisations Graphiques (accès par SYS_PAGE_PROSIMUL= 0 à 15)

En RUN, la page 0 s'affiche par défaut. Des changements de pages peuvent être opérés via le Navigateur Application ou automatiquement via la variable système " SYS_PAGE_PROSIMUL " prenant les valeurs 0 à 15.

- Les touches fonctions associées aux symboles sont opérantes.

Les **clics** droit et gauche peuvent être **inversés** en mettant le paramètre « **Runtime.Invert_Mouse_Click** » des préférences à la **valeur 1**.

- Pour les objets avec option « **forçage** » **:**

. un **clic gauche** sur un objet provoque le **forçage** de sa 1ère variable d'animation. Si celle-ci est de type binaire elle est directement complémentée. Si elle est de type mot (entier ou réel) la saisie de sa valeur de forçage est proposée. Dans les deux cas la variable n'est plus modifiable par programme. Si le paramètre « Cmd_Mode » est >= 100 et que la variable est booléenne, la variable est complémentée durant le temps de ce paramètre (exprimé en ms).

. un **clic gauche** sur un objet déjà forcé **annule** son **forçage**. Une recherche des variables forcées est possible via la boîte de dialogue des Variables.

. un **clic droit** ou un **clic gauche associé avec [SHIFT]** sur un objet agit sur sa 1ère variable d'animation associée. Un complément est réalisé si la variable est de type binaire ou la saisie de sa valeur est proposée si elle est de type mot. Son **forçage** (si présent) est **annulé** (la variable est modifiable par programme).

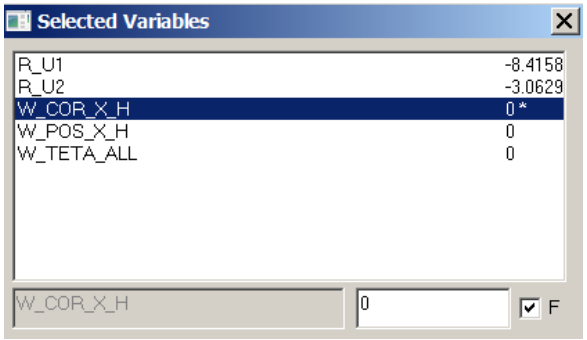
. un **double clic gauche** sur un objet « Courbe » provoque son impression.

. un **clic avec [ALT]** sur un objet provoque le **forçage** de sa 1ère variable d'animation. La variable n'est pas modifiée (**figeage** de la valeur).

- Pour l'objet « INTER_3P » : un clic sur la partie gauche de l'objet active la première variable associée et désactive la seconde, un clic sur la partie droite active la seconde variable et désactive la première. Les deux variables sont remises à zéro par un clic en position centrale.

- L'objet « CURSEUR » peut être piloté par les flèches gauche et droite, le mouvement du curseur ou le clic sur la valeur.

- Un clic droit sur une zone vide permet d'afficher la liste des variables surveillées (paramétrées via la fonction Horodatage).



Visualisations Variables

- Les touches fonctions associées aux symboles sont opérantes.

- Les variables sont visualisées en dynamique.

- Les valeurs courantes peuvent être modifiées (forcées ou non).

- Le forçage peut être activé (case à cocher). Dans cet écran la modification de valeur de variables forcées reste possible.

Visualisations Ladder

- Les touches fonctions associées aux symboles sont opérantes.

- Les instructions peuvent être visualisées en dynamique (accès aux instructions par **double-clic**). Les variables binaires vraies ont leur texte d'information sur fond vert. Pour les variables mots, leur valeur courante est affichée en lieu et place du texte d'information. La variable peut être accédée par un clic sur son texte d'information.

- Les commandes " Suivant " et " Précédent " restent actives.

- La commande « **VisuDyn** » du menu Ladder ou le clic droit sur la fenêtre de programme permet de visualiser en dynamique la liste des variables utilisées dans le programme courant. Un double-clic sur une variable permet de retirer la variable de la liste. Les valeurs peuvent être modifiées en validant une nouvelle valeur dans la zone de saisie. Les forçages peuvent être modifiés via la case à cocher prévue à cet effet.

Visualisations Littéral

- La commande « **VisuDyn** » du menu Ladder ou le clic droit sur la fenêtre de programme permet de visualiser en dynamique la liste des variables utilisées dans le programme courant. Un double-clic sur une variable permet de retirer la variable de la liste. Les valeurs peuvent être modifiées en validant une nouvelle valeur dans la zone de saisie. Les forçages peuvent être modifiés via la case à cocher prévue à cet effet.

Visualisations Grafcet (accès par SYS_PAGE_PROSIMUL= 20 à 27)

- Les touches fonctions associées aux symboles sont opérantes.

- Les graphes sont visualisés en dynamique :

. étape active sur fond vert
. transition valide indiquée en vert
. variable associée forcée indiquée en rouge.

- **Action Souris**

. un **simple clic gauche** sur une étape / transition provoque le **forçage** de sa variable associée. Elle est de plus directement complémentée.

. un **simple clic gauche** sur une étape / transition déjà forcée **annule** son **forçage**. Une recherche des variables forcées est possible via la boîte de dialogue des Variables.

. un **clic droit** ou un **clic gauche associé avec [SHIFT]** sur une étape / transition agit en complément sur sa variable associée. Son **forçage** (si présent) est **annulé** (la variable est modifiable par programme).

. un **clic avec [ALT]** sur une étape / transition provoque le **forçage** de sa variable associée. La variable n'est pas modifiée (**figeage** de la valeur).

Visualisations Fonctions (accès par SYS_PAGE_PROSIMUL= 30)

- Les touches fonctions associées aux symboles sont opérantes.

- Un simple **clic gauche** sur un bloc " **sous-page** " provoque l'entrée dans la sous-page.

- Un simple **clic droit** à partir d'une **sous-page** provoque la sortie vers le niveau supérieur.

- Les valeurs sont affichées en dynamique et les blocs " lampes " animent leurs sorties en vert si le seuil en entrée est atteint.

- Un **double-clic** sur bloc fonction " **Courbe** " provoque l'affichage des courbes associées en entrées en fonction du maximum indiqué. Un **clic droit** referme cet affichage. Un second double-clic provoque son impression.

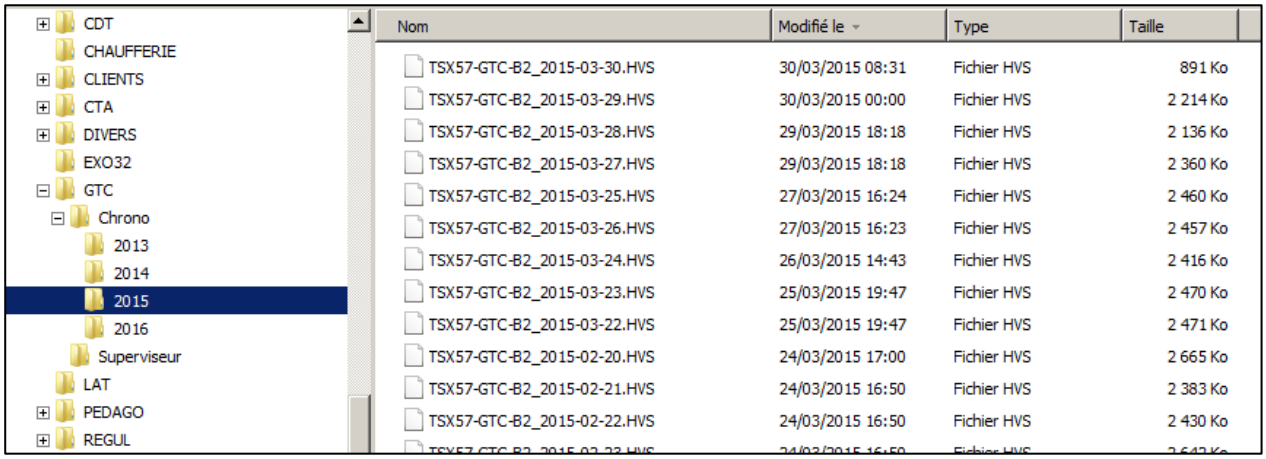
Connexions Veyon

Si le paramètre « **Runtime.Minimize_if_Veyon** » est à 1 dans les préférences et si un évènement d'affichage (EraseBkGnd) intervient et que Veyon est connecté (sur le **port 11100**), les applications se minimisent pour éviter les risques de conflits d'affichage.

HORODATAGE

Cette fonction accessible via le Navigateur “ Outil -> Horodatage ”ou via la commande « Visudyn » permet de gérer une liste de variables (100 maxi) à enregistrer sur changement d’état. L’option d’enregistrement doit être déclarée au préalable en Configuration Application.

Des fichiers **NOMAPPLI_AAAA-MM-JJ.HVS** sont créés chaque jour dans ce cas.
Si un sous-répertoire « **Chrono** » existe, les fichiers y seront stockés et recherchés.
Si des sous-répertoires « **Chrono\AAAA** » (années) existent, les fichiers y seront stockés par année et recherchés.



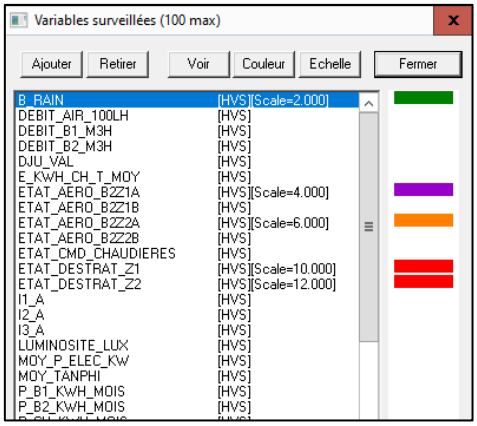
Les références des variables enregistrées sont incluses quotidiennement dans les fichiers HVS.
La référence est réalisée grâce aux symboles des variables et à leurs types. Si des modifications ont été apportées aux noms de symboles dans l’application .SIM après les enregistrements, le fichier « NOMAPPLI_SYMB-HISTORY.TXT » contient l’historique des modifications. Si ce fichier existe (il peut être supprimé), les modifications incluses sont prises en compte dans les fichiers HVS.

En mode RUN, par défaut le fichier de la date courante est recherché et est affiché en dynamique (les 10.000 dernières valeurs).
La page peut être appelée si SYS_PAGE_PROSIMUL= 31

Menu Horodatage

Capturer : copie les courbes dans le Presse-papier Windows.

Gestion Liste : active la boîte de dialogue de sélection d’enregistrement / affichage des variables.
→ Ajouter / Retirer : des variables surveillées de la liste
→ Marqueur [HVS] : la variable existe en fichier
→ Voir (ou double-clic) : affichage ou non de la variable
→ Marqueurs [Scale] & couleur : la variable est affichée
→ Couleur : choix de coloris
→ Echelle : choix de l’échelle (ex : 0,01)



J-1 ou ← : sélectionne le fichier du J-1 (ou à défaut recherche jusqu'à J-10).
Date : permet la sélection d'un fichier *.HVS précédemment créé.
J+1 ou → : sélectionne le fichier du J+1 (ou à défaut recherche jusqu'à J+10).

Variables : permet la sélection en affichage des variables consignées dans le fichier *.HVS.

Affichage → Grille : affichage ou masquage de la grille.
Affichage → Barre Verticale : affichage d'une barre verticale lors du déplacement du curseur.
Affichage → Barre Horizontale : affichage d'une barre horizontale lors du déplacement du curseur.
Affichage → Pinceau large : double l'épaisseur du pinceau.
Affichage → Point à point : affichage et export des valeurs en point indépendants.
Affichage → Temps réel : affichage des valeurs en temps horaire ou en temps relatif à l'origine.
Affichage → Moyenne : permet de moyenner les valeurs (0=sans…99).
Affichage → Seuil Haut : seuil maximum d'affichage (plus grande valeur sinon).
Affichage → Seuil bas : seuil minimum d'affichage (plus petite valeur sinon).
Affichage → Extrêmas : permet l'affichage des extrêmas sous forme de carrés situés sur les courbes.
Affichage → Inflexions : permet l'affichage des points d'inflexion sous forme de cercles et tangentes situés sur les courbes.

Paramètres -> Sauver : permet de sauvegarder dans un fichier **NOMAPPLI_*.INI** l'ensemble des paramètres de variables et d'affichages sélectionnés.
Paramètres -> Charger : permet de charger un fichier **NOMAPPLI_*.INI** contenant des paramètres précédemment sauvegardés. Les listes disponibles sont affichées en dessous de cette commande.

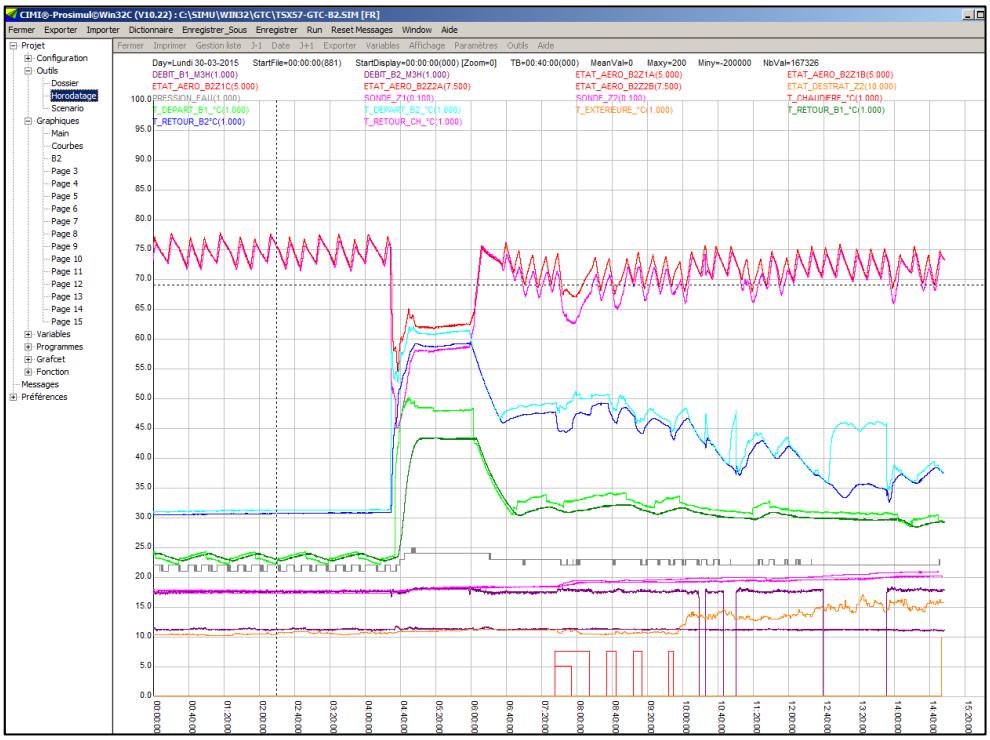
Outils → Polynômes : si <u>une seule variable est affichée</u> cette fonction permet de calculer les coefficients d'un polynôme de degré 2 à 12 selon la méthode des moindres carrés. Ces coefficients peuvent être utilisés dans un bloc fonction " Polynôme " (à la fermeture de la boîte de dialogue indiquant le résultat, le polynôme peut être collé dans la page fonction sélectionnée <u>avant</u> cette opération).

Exporter → Jour : permet la création d'un fichier **NOMAPPLI_AAAA-MM-JJ.HVX** pour les variables sélectionnées sur la plage de temps affiché (voir Zoom) avec la moyenne de valeurs choisie.
Exporter → Mois : permet la création d'un fichier **NOMAPPLI_AAAA-MM.HVX** pour tout le mois intégrant le jour affiché avec la moyenne de valeurs choisie.
Exporter → An : permet la création d'un fichier **NOMAPPLI_AAAA .HVX** pour toute l'année intégrant le jour affiché avec la moyenne de valeurs choisie.

Format d'exportation
Year MonthDay Time MilliSecond Symbol Value(10) Forçage
où Value(10) indique que les valeurs sont moyennées sur 10 valeurs.
Un outil sous EXCEL peut être fourni pour l'analyse des données des fichiers HVX.

Affichage des valeurs

- Une mise à l'échelle des ordonnées est effectuée automatiquement selon les seuils retenus (initialement les maximas des valeurs enregistrées).



- La **base de temps** réglée automatiquement et correspondant aux abscisses est reportée en haut de l'écran.
- Les symboles des variables sélectionnées sont indiqués dans leurs couleurs respectives en haut de l'écran.
- Le **déplacement du curseur** sur la grille est reporté sous forme X=abscisse Y=ordonnée dans la zone de message droite.

*Les variables et options d'affichage sélectionnées sont mémorisées dans le fichier *.HVS et dans le fichier global NOMAPPLI.INI qui est relu (si existant) à chaque changement de jour.*

Déplacement / Zoom Temps

La **sélection d'une zone** à la souris permet de réduire la plage de temps affiché. Cette fonction est disponible sur 10 niveaux reportés en haut de l'écran (Zoom=0...9).
Un **clic droit** ou la commande [**ESC**] provoque un Zoom Arrière du temps.

En mode Run, l'échelle de temps est automatiquement décalée selon le temps écoulé.

Les **touches gauche et droite** permettent en mode zoom de déplacer l'échelle de temps en amont ou en aval d'une base de temps de l'affichage courant.

La **molette de la souris** permet de décaler les valeurs affichées d'une journée en amont ou en aval de la date courante.

SCÉNARIO

L'option d'Enregistrement ou de Relecture doit être déclarée au préalable en Configuration Application avant passage en RUN.

Dans la phase **d'enregistrement** un fichier **NOMAPPLI.SVA** est créé qui contiendra l'ensemble des commandes manuelles effectuées sur des variables (base de variables, touches fonctions, forçages,...).

La fonction scénario accessible via le navigateur " Outils -> Scénario " permet d'éditer les valeurs enregistrées sous forme d'un fichier texte **NOMAPPLI.TXT**. Les valeurs et variables peuvent être alors modifiées (notamment par collage de valeur issues d'horodatage). Le format texte du fichier (**séparateur = tabulation**) est compatible pour les heures et valeurs avec le format texte horodatage et est le suivant :

Time MilliSecond Symbol Value Forçage(Oui=FORC)

Exemple

Time	MilliSecond	Symbol	Value	Forçage
00:00:02	000	W_POLYMER_AFF	1	
00:00:03	000	R_HEAD_PC_EDT_SP	70.0	
00:00:03	100	O_BP_HEAD_ADJUST	1	FORC
00:00:03	200	O_BP_HEAD_ADJUST	0	
00:00:04	000	O_P_42MB_W_RUN	1	
00:00:08	000	O_BP_M_SS2_START	1	FORC
00:00:08	200	O_BP_M_SS2_START	0	
00:00:20	000	R_V_FCV5201A_SP	25.0	FORC

Éditer : permet de modifier le scenario courant (NOMAPPLI.SVA) via le fichier texte NOMAPPLI.TXT. A la fermeture de l'éditeur texte, le scenario NOMAPPLI.SVA est reconstitué.

Charger : permet d'éditer n'importe quel fichier *.TXT correctement formaté et de l'utiliser comme scenario courant.

Le fichier *.SVA peut être **rejoué** automatiquement (Configuration Application / Horodatage / Rejouer). Le coefficient temps système est aussi appliqué aux valeurs enregistrées. L'ensemble des opérations présentes en fichier seront alors exécutées.

IMPORT / EXPORT

Prosimul permet l'échange de fichiers texte (.TXT) pour les définitions de symboles (avec valeurs initiales) et des fichiers Graphiques+Modèles (.BIM) pour échanges entre applications (.SIM). Les mêmes commandes import/export serviront aux deux usages selon l'ouverture ou non de la boîte de dialogue des Variables.

Export de variables (depuis boîte de Variables)

- Sélectionner dans la boîte de dialogue Variables, les variables à exporter.
- Choisir la commande exporter dans le menu principal.
- Donner un nom au fichier (extension .TXT)
- Le fichier exporté s'affiche avec le bloc-notes Windows (notepad.exe).
- Format du fichier :

  SYMBOLE**[tabulation]**valeur(valeurs réelles reconnues grâce au point séparateur des décimales).

Import de variables (depuis boîte de Variables)

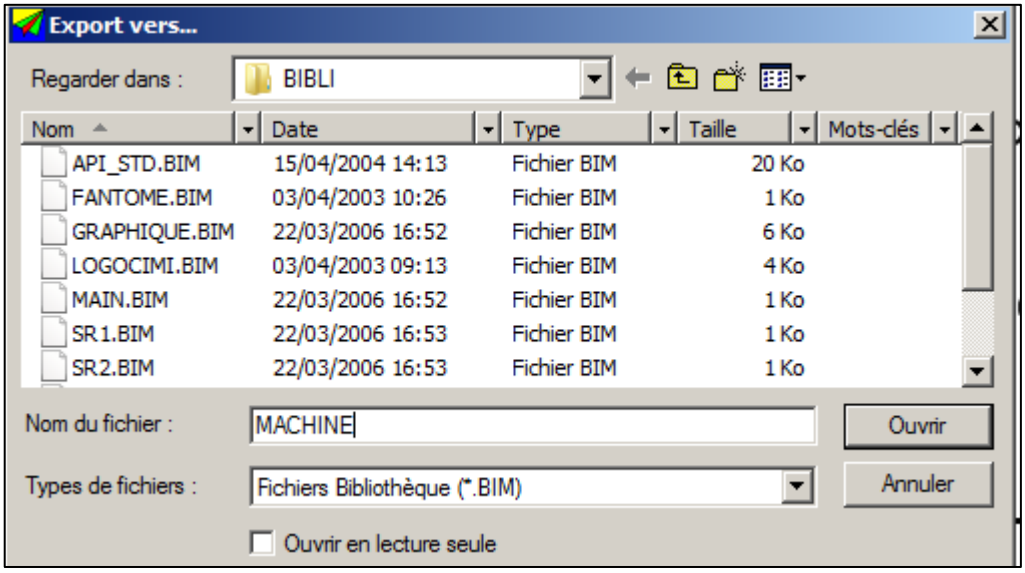
- Sélectionner dans la boîte de dialogue Variables, la première variable recevant l'importation.
- Choisir la commande importer dans le menu principal.
- Choisir un nom de fichier (extension .TXT).
- Si des symboles existent déjà consécutivement à la première sélectionnée le choix d'annulation ou d'écrasement est proposé. Un contrôle d'unicité des symboles est effectué. Si la valeur initiale existe (SYMBOLE**[tabulation]**valeur) celle-ci est récupérée.

Export de Graphiques et Modèles (.BIM)

- Le principe est d'exporter tous les objets **sélectionnés au préalable** à travers les différents éditeurs. Cette fonction est disponible pour les éditeurs Graphique / Ladder / Grafcet et Fonctions.

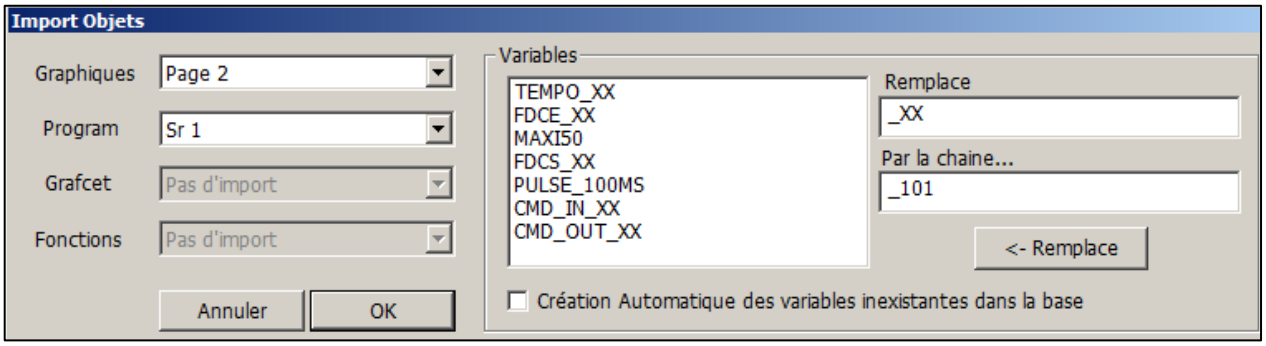
- Pour exporter des graphiques, instructions Ladder, Grafcet et blocs fonctions dans le même fichier .BIM, répéter l'opération de sélection / export depuis chaque éditeur en sélectionnant le même fichier destination .BIM.

- L'ensemble des composants sélectionnés sera exporté dans un fichier binaire **.BIM** contenant de plus les symboles associés aux objets. Vous pouvez remplacer des éléments déjà présents en fichier .BIM en répétant les opérations de sélection/export. Un message vous demande de confirmer ou non le remplacement dans ce cas. Veillez dans tous les cas à exporter des éléments exploitants la même base de variables.



Import de Graphiques et Modèles (.BIM)

- Le principe est d'importer sur des **pages choisies** les éléments sélectionnés contenu dans le fichier .BIM (graphique et modèles).



- Il est possible de ne pas importer certains éléments. Dans ce cas sélectionner l'option « No Import » pour les éléments non désirés.

- L'import de Grafcet et Fonctions nécessitent des pages vierges.

- Le dépassement du nombre d'objets maximal autorisé par éditeur se soldera par un échec avec message d'erreur.

- Pour les pages graphiques, les objets importés sont groupés par défaut.

- Une **création / remplacement des symboles** présent dans le fichier .BIM est proposé à travers une boîte de dialogue. Vous pouvez :

. importer au préalable dans la base les variables nécessaires via fichier .TXT et simplement valider.

. créer ou modifier tout ou partie des variables. Un double-clic sur chacune des variables en liste projette dans la boîte de dialogue Variable sur le type de variable adéquat. Créer ou sélectionner le symbole choisi (le clic droit rappelle les 10 derniers symboles utilisés).

. remplacer une chaîne de caractère dans tous les symboles de la liste (ex : _01 par _02). Utiliser dans ce cas le bouton « Remplacer » après avoir compléter les champs " Ancienne chaîne " et " Nouvelle chaîne ".

. choisir l'option « Création automatique des variables inexistantes dans la base ». Dans ce cas les variables sont automatiquement créées sur les premières variables disponibles en fin de listes. Les types de variables sont respectés. Les étapes déjà existantes sont remplacées.

CONVERTISSEUR

Fichier DOS nécessaire

- Seuls les fichiers .IMA (graphiques), .CFG (variables), .ANI (animations), .ACT (Ladder), .XTR (ladder Grafcet), .GRF (Grafcet) et .FCT (blocs fonction) sont nécessaires pour la conversion des applications DOS.
Il est possible de permuter les couleurs noires et blanches (faire attention aux programmes Ladder utilisant les codes de couleur).

Contraintes sur les applications DOS

- les variables “ BIT A 1 ”, “ TOP DIXIEME ”, “ TOP CENTIEME ” et “ INIT GRAFCET ” doivent exister sur les bits 0 à 3 de l’application DOS.

- les bits SYS_TOP_DIXIEME et SYS_TOP_CENTIEME sont désormais des signaux carrés. Prévoir l’usage d’un front montant (**P#**) dans les programmes convertis.

- les variables “ RIEN ”, “ HORLOGE ”, “ CYCLE PROSIMUL ”, “ PAGE PROSIMUL ” et “ COEFF TEMPS(%) doivent exister sur les mots 0 à 4 de l’application DOS.

- Le bit B1023 doit être libre d’utilisation.

Variables

- La variable “ BIT A 1 ” est remplacée par la valeur binaire 1 (0 pour son complément).
- La variable “ RIEN ” est remplacée par la valeur entière 0.

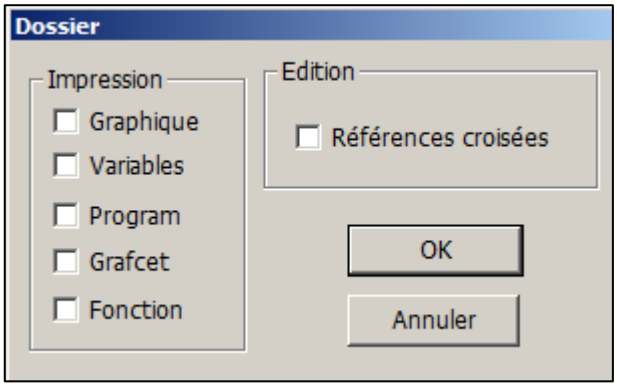
Ladder

- Dès que 10 instructions seules restent disponibles dans le “ Main ”, les instructions sont introduites dans les SR (1, 2,...) et un appel inconditionnel (CALL) est effectué depuis le “ Main ”.

- Les JUMPS relatifs sont substitués par des étiquettes (attention toutefois au découpage en SR pouvant provoquer des césures de boucles).

MENU DOSSIER

- Le dossier permet d'imprimer les éléments sélectionnés dans la zone "impression"' (graphiques, listes de variables, programmes Ladder, Grafcet et/ou Fonctions) sur l'imprimante choisie et d'éditer les références croisées sous éditeur de texte si sélectionnées.



- L'accès au dossier ne peut se faire qu'en mode STOP.

- L'imprimante et l'orientation du papier peuvent être choisies.

EVOLUTIONS

V12.07

12.07A4 (05/08/2026)

NEW **Variables** : possibilité de copier un symbole en RUN (pour l'outil recherche et le presse-papier).

12.07A2 (20/07/2026)

NEW **Communication** : nouveau pilote « ENIP (CIP-TAGS) » - Adressage des datas par Tags.

NEW **Graphiques** : nouvel objet « CUVE_TETA » (cuve rotative, contenant exprimé en volume).

NEW **Graphiques** : nouvel objet « TABLE_TETA » (bargraphe rotatif).

CORR **Graphiques** : la couleur de fond de page pouvait ne pas s'afficher correctement.

12.06C2 (15/03/2026)

NEW **Editeur graphique** : nouvel objet « Trait_Bezier » .

12.06C1 (22/12/2025)

MOD **Pilote ENIP client** : UC configurable jusqu'au slot 19.

12.06B2 (15/11/2024)

NEW **Littéral** : nouvelles fonctions « ConnectToTcp », « Disconnect », « ModbusReadRegister ».

CORR **Horodatage** : correctif Exports data.

CORR **Blocs Fonctions** : correctif liaisons inter-blocs.

12.06B1 (22/10/2024)

NEW **Graphiques** : 2 nouveaux objets utiles pour des cotations (Cote et Surface).

CORR **Fonction SendMail** : le corps de message était omis (V12.06 seulement).

CORR **Modbus TCP** : problème de configuration pour cible client Modbus (V12.06A2 seulement).

12.06A2 (10/07/2024)

NEW **Horodatage** : création de points de navigation tactiles ← ESC →

NEW **Modbus Client** : possibilité d'utiliser la requête FC04 en plus de FC03 et FC23.

CORR **Graphiques** : affichage d'objets superposés (uniquement pour V12.06A1).

CORR **Boîte variables** : correctif d'affichage des variables système.

12.06A1 (21/06/2024)

NEW **Littéral** : nouvelle instruction « GetTcpPortState »

MOD **Graphiques** : les appels GDI et GDI+ sont récrits pour éviter les problèmes avec VEYON.

CORR **Graphiques** : divers correctifs sur les boîtes de dialogue (dont propriétés et police).

CORR **Littéral** : correctifs mineurs des instuctions InfoApp(4 ;0) et MsgLog(;)

V12.05

12.05A9 (12/12/2023)

MOD **Horodatage** : nouvelle version de la boîte de gestion.

CORR **Graphiques** : les anciens groupes (applications converties) n'ont pas le bon type de variable.

12.05A3 (23/10/2023)

CORR **Littéral - Box** : sur certains OS les DLL ne sont pas multi-instances + passage à 12 box maxi. La DLL fournie est modifiée.

NEW **Horodatage** : nouvelle fonction de capture d'écran.

MOD **Vidéos** : nouvelle version de Pro32Video.exe utilisable seulement avec PROSIMUL V12.05.

CORR **Littéral** : la fonction « Remplacer_tous » opère sur l'ensemble des programmes Littéral.

MOD **Littéral** : nouveaux paramètres pour les fonctions InfoApp et MouseCtrl.

V12.04

12.04A6 (13/09/2023)

NEW **Scenario** : nouvelle fonction « Charger ».

NEW **Graphiques** : nouvel objet « LISTBOX2 » (animation en couleur du texte).

NEW **Blocs Fonctions** : nouveaux blocs « Share » et « Screw ».

NEW **Runtime** : en mode Développement, le passage en RUN avec la touche [SHIFT] permet de lancer l'application sans le navigateur.

CORR **Littéral** : fonction « Rechercher suivant » stoppait sur programme vide.

CORR **Graphiques** : correctif du plantage aléatoire de la boîte de Propriétés des objets.

NEW **Graphiques** : nouvelle courbe « COURBE_PC » (valeurs normées en %, avec / sans abscisse).

NEW **Littéral** : nouvelles fonctions de conversion « Fahrenheit » ←→ « Celsius ».

NEW **Ladder** : nouvelle commande « Coller + Suivant ».

NEW **Littéral** : la fonction « InfoApp » peut retourner les origines X, Y du synoptique.

NEW **Littéral - Box** : possibilité de gérer 10 exécutables externes.

NEW **DialogBox** : nouveau bouton (code 64) avec Bitmap (256 couleurs max).

V12.03

12.03A3 (12/01/2023)

CORR **Graphique** : correctif affichage du nouveau convoyeur rotatif.

CORR **Graphique** : correctif affichage des courbes après visualisation programme.

12.03A2 (05/01/2023)

MOD **Pilotes Ethernet :** le paramètre « Retries » disparaît au profit d'un paramètre « Delay before retry (s) ». Le paramètre « Period (ms) » devient « Connexion timeout(s) » seulement pour les pilotes serveurs. Les pilotes Ethernet sont relancés en cas de timeout.

NEW **Graphiques** : création d'un objet convoyeur rotatif avec produit intégré.

CORR **Littéral** : détection du mot-clé « then » en début de ligne.

CORR **Runtime** : l'activité des applications est détectée pour éviter une fermeture de toutes les applications en cas de plantage d'une d'entre elles.

CORR **Ladder** : les raccourcis « Shift » « / P N F » ne sont plus pris en compte dans les commentaires.

NEW **Grafcet** : création de liaisons orientées basses (uniquement derrière des transitions).

CORR **Littéral :** correctif de retour des valeurs des instructions GetSX, GetX1, …

V12.02

12.02C2 (23/05/2022)

CORR **Pilotes Ethernet** : correctif pour version anglaise du paramètre « Retries » (-1).

12.02C1 (17/05/2022)

MOD **Pilotes Ethernet** : si le paramètre « Retries » prend la valeur -1, le pilote de communication reste actif même en cas de défaut de communication.

12.02B6 (04/04/2022)

NEW **Graphiques** : nouvel outil « Capturer » pour faire une copie d'écran.

NEW **Graphiques** : nouvel outil pour Redimensionner l'ensemble des objets.

NEW **Graphiques** : nouveau paramètre de Box avec alignement vertical sur le cadre de la fenêtre.

NEW **Graphiques** : nouvelle vanne 3 voies.

MOD **Graphiques / Cuves** : si le commentaire débute par le caractère | la cuve est fermée.

CORR **Serveur EIP** : pas de redémarrage dans certains cas si maxi tentatives atteint (sortie nécessaire).

CORR **Editeur Littéral** : problème avec BackSpace et Return (suppressions intempestives).

CORR **Runtime** : problème lors de clic droit (déforçage) sur des objets VALEUR_T2 et LISTBOX.

12.02A5 (08/11/2021)

CORR **Littéral** : fonction « Remplacer tous » inopérante dans certains cas.

CORR **Ladder Timer** : si BT à 0, une boucle sans fin arrête l'exécution du programme.

CORR **Modbus TCP** : correctif FC23 (timeout). V12 seulement.

12.02A4 (11/05/2021)

NEW **Communication** : une adresse IP locale fixe (intégrée à l'application SIM) est disponible dans la configuration des échanges.

NEW **Littéral** : après une recherche en Run, le dernier programme visualisé s'affiche lors du passage en Stop.

NEW **Littéral** : nouvelle fonction « Remplacer tous ». Recherche et remplace toutes les occurrences de textes sélectionnés dans l'ensemble des programmes en Littéral.

NEW **Littéral** : un double tiret « -- » en fin de ligne permet d'écrire un commentaire.

NEW **Runtime / Graphiques** : les clics droit et gauche peuvent être inversés en mettant le paramètre « Runtime.Invert_Mouse_Click » des préférences à la valeur 1.

NEW **DataBase** : les adresses des I/O peuvent être réglées en notation X.Y en réglant le paramètre « DataBase.IO_HexaMode » à la valeur 1 dans les préférences.

NEW **DataBase**: rajout des touches [Ctrl]C/V/X/Insert et [Shift]Suppr/Insert (copier/coller/suppr).

NEW **Horodatage** : possibilité d'afficher ou cacher la grille.

MOD **Graphiques** : boîtes Couleurs/Police/Styles : seuls les paramètres modifiés ou revalidés sont pris en compte, notamment dans le cas de sélection multiples.

MOD **Graphiques** : la fonction magnétisme ne modifie plus la position du point 2. Celle-ci peut toujours être calée sur la grille lors du déplacement du point 2 seul (touche [CTRL] ou souris).

MOD **Littéral – Fenêtre Variables** : le double-clic ne supprime plus les variables de la liste.

MOD **Graphiques** : la molette de la souris pilote désormais les fonction « Zoom+ » et « Zoom- ». Pour modifier les dimensions de plusieurs objets il faut les grouper et déplacer le point 2.

CORR **Littéral** : une recherche infructueuse ramène sur le programme initial.

CORR **Graphiques – Animations** : en cas de renommage de symbole et si la boîte de Propriétés est ouverte, les nouveaux symboles, utilisés dans la boîte, sont pris en compte. Les paramètres /#, P#, …, restent conservés sauf en cas de sélection d'un nouveau symbole.

CORR **Communication** : le pilote S7-ISO-TCP ne démarrait pas (en V12.01D4 uniquement).

CORR **Graphiques** : rares risques de plantage lors de l'affichage de valeurs décimales trop longues (objet Valeurs).

V12.01 (Upgrade)

12.01D4 (16/03/2021)

NEW **Horodatage** : affichage et export des valeurs en simple « points ».

NEW **Littéral** : La fonction MsgLog admet une variable par ligne.

NEW **Runtime** : affichage toutes les 5s du % d'usage processeur par application en Run.

MOD **Runtime** : en priorité 6, le programme ne suspend plus son cycle mais cède seulement son droit d'exécution. La période fixée est garantie d'être tenue à la ms au détriment de l'usage de l'UC.

CORR **Littéral - Fenêtre Variables** : tri des variables dynamiques (version anglaise seulement).

CORR **Runtime** : précision du temps (1ms) pour les tâches Runtime & Comm pour **Windows 10**.

CORR **Communication** : tous pilotes. Défaut ponctuel de transmission vers API dans de rares cas.

12.01C2 (07/01/2021)

NEW **Littéral** : nouveaux paramètres 6 et 7 pour la fonction StopApp.

MOD **Runtime** : amélioration de la précision des temps de cycle (runtime & échanges).

MOD **Blocs Fonctions** : les fonctions Multiplicateur et Ordre 1 ont une valeur initiale et un Minimum.

CORR **Horodatage** : de l'export des données horodatées en temps relatif.

CORR **Horodatage** : Horodatage : plantage possible si plusieurs changements de liste de variables.

CORR **Graphiques** : du format de l'affichage des valeurs de l'objet « Curseur ».

CORR **Littéral** : Fonction WinExec avec Param2 différent de 1.

12.01B2 (09/10/2020)

NEW **Communication** : **Modbus** TCP-IP : requêtes FC01 / FC15 supportées.

CORR **Communication** : pilote **DF1** série (V12 uniquement).

CORR **Runtime** : forçage des variables binaires pour des objets multi-formats.

12.01A5 (03/08/2020)

NEW **Graphiques** : nouvel objet « OBJ2D » dans le groupe « Rotation 2D ».

CORR **Windows 10** : accès au registre 64 bits pour Windows 10.

CORR **Littéral** : envoi des emails. Les valeurs numériques étaient omises.

12.01A1 (20/01/2020)

NEW **Communication** : pilote EthernetIP serveur pour répondre aux modules Ethernet Rockwell

NEW **Communication** : l'adresse **IP locale** de l'application peut être indiquée dans la configuration des échanges.

NEW **Graphiques** : nouvel objet graphique de type « Objet3D » (formats .ASE, .DAE, .OBJ) utilisé avec le « viewer3d » Pro3D.exe.

MOD **Communication** : optimisation des clients Ethernet (tous protocoles)

MOD **Littéral** : l'envoi d'email utilise maintenant l'utilitaire SendMail.

CORR **Horodatage** : défauts d'affichage mineurs corrigés.

CORR **Navigateur** : défaut d'affichage du projet lors de la création de pages graphique / programme.

DEL **Conversion** : la fonction de conversion des applications MS-DOS est retirée.

DEL **Clés** : les clés sur port parallèle ne sont plus reconnues.

V11.06

11.06H6 (14/09/2020)

CORR **Communication** : pilote DF1 série.

CORR **Runtime** : forçage des variables binaires pour des objets multi-formats.

11.06H1 (10/12/2019)

CORR **Runtime** : plantage au passage en Run après que le Littéral ait été ouvert puis fermé.

CORR **Littéral** : multiples beeps lors de la compilation et consultation du littéral en Run.

11.06G5 (17/04/2019)

MOD **Graphiques** : éditeur graphique, outil « Nettoyer » : améliorations.

CORR **Communication** : Drivers Ethernet : correctif de l'attente possible de quelques secondes avant relance.

CORR **Graphiques** : de la gestion de l'objet « Son » (*.wav).

11.06G3 (21/02/2019)

NEW **Runtime** : la page « Horodateur » peut être appelée si SYS_PAGE_PROSIMUL vaut 31.

NEW **Runtime / Fonctions** : les actions Zoom In et Out sont enregistrés en scénario.

11.06G2 (21/01/2019)

CORR **Navigateur** : position d'insertion des nouvelles pages vierges et intermédiaires.

CORR **Graphiques** : défaut de sélection avec clic droit lorsque le zoom était différent de 100%.

11.06F1 (18/11/2018)

MOD **Graphiques** : pour les nouveaux objets, les couleurs d'animation sont par défaut la couleur de fond de page.

CORR **Runtime** : les groupes cachés initialement au démarrage ne s'affichent plus.

11.06E4 (29/08/2018)

NEW **Blocs Fonctions** : les valeurs P/I/D des régulateurs peuvent aussi être écrites en manuel.

CORR **Ladder** : défaut d'exécution si un commentaire existe juste devant la fin du programme.

11.06E2 (07/08/2018)

NEW **Blocs Fonctions** : les valeurs P/I/D des régulateurs peuvent être récupérées dans la DataBase.

MOD **Communication** : S7-ISO-TCP : emplacement UC n°1 accessible. Liaison à reconfigurer.

CORR **Graphiques** : format d'affichage des valeurs réelles des tables « Symbol » et « Value ».

11.06D1 (15/06/2018)

NEW **Graphiques** : nouvelles commandes « Coller Suivant » qui colle l'objet avec les variables suivantes de la base si la racine est la même (sauf variable système).

MOD **Graphiques** : objet « Dialog Box ». Le Code Box 32150 indique un séparateur de texte Tabulation.

11.06C3 (04/05/2018)

NEW **Graphiques** : nouvel objet graphique de type « Trait » elliptique.

CORR **Communication** : client OPC : la connexion au serveur est testée continument pour éviter un « plantage » si l'utilisateur stoppe le serveur.

CORR **Graphiques** : couleur de fond des traits hachurés.

11.06B2 (15/01/2018)

NEW **Ladder / Littéral** : nouvelles instructions d'ouverture / fermeture de boîtes de dialogue (DialogOpen, DialogClose).

NEW **Cartes I/O** : adjonction des cartes de sorties analogiques « IPCDAS / PIO-DAxU » avec pilote « UNIDAQ ». Les cartes « PISO-P32A32U » doivent aussi utiliser le pilote « UNIDAQ ».

CORR **Ladder** : la fermeture (X) des instructions Ladder en visu dynamique provoquait un défaut.

11.06A3 (04/12/2017)

CORR **Graphiques** : affichage objet « ListBox » en mode « Page ».

CORR **Aide** : n° de pages d'appel aux fichiers d'aide PDF.

11.06A2 (17/11/2017)

NEW **Graphiques** : nouvel objet de type « Dialog Box » paramétrable via un fichier texte.

NEW **Application** : configuration Application : création d'un mode « Supervisor » qui évite le rafraichissement des images en arrière-plan (temps de cycle réduit) et retire le menu « Graphique » en état Run.

NEW **Graphiques** : nouvel objet de type « ListBox ».

NEW **Graphiques** : les objets « Valeur » peuvent afficher l'heure selon le format prédéfini (idem objet texte). La valeur de temps est exprimée en millisecondes à partir de 0h.

NEW **Ladder / Littéral** : nouvelles instructions de contrôle de la position souris.

NEW **Graphiques** : nouvel objet « Valeur_T2 » dont la valeur est modifiable directement sur l'objet.

NEW **Graphiques** : pour tous les objets pilotables en binaire avec « Cmd_Mode ». La valeur 21 force toujours à 1 et la valeur 20 force toujours à 0.

NEW **Graphiques** : les objets « TEXTE » peuvent recevoir la valeur « Cmd_Mode » égale à 10 qui autorise la modification du texte (29 caractères maxi) directement sur l'objet. Les valeurs 30 et 31 permettent d'afficher l'heure et la date système.

NEW **Graphiques** : 2 nouvelles pompes + 2 nouveaux moteurs.

NEW **Blocs Fonctions** : nouveau bloc « Moyenne ».

MOD **Blocs Fonctions** : le gain du bloc « Convoyeur » est réparti sur l'ensemble des éléments.

MOD **Graphiques / Application** : les couleurs supplémentaires (16 à 31) sont stockées dans le .SIM.

CORR **Blocs Fonctions** : valeur initiale pour bloc convoyeur.

CORR **Littéral** : correctif des instructions GetSX, GetSY, …, pour les objets en mouvement.

V11.05

V11.05g2 (15/01/2018)

- **Création** : Objet « Rotation2D » : pour les objets « Rotation2D » type 3 et 4 création d'un « Mode contour » (1=contour non affiché).
- **Modification** : Éditeur graphique : la fonction « Undo » est basculée sur la combinaison de touches [CTRL][Z] (idem Microsoft Office). La touche [ESC] sert à désélectionner tous les objets.
- **Modification** : Pilote EIP : la limite maximum du total des I/O échangées passe de 127 à 254.
- **Correctif** : la fermeture (X) des instructions Ladder en visu dynamique provoquait un défaut.

V11.05f (01/08/2017)

- **Correctif** : instructions « Air Humide » : les valeurs de pression sont toutes exprimées en Pascal
- **Correctif** : Runtime : interception du message d'erreur de calcul qui provoquait une défaillance. Passage en Stop automatique au-delà de 4 erreurs de calcul dans le même cycle (voir fichier LOG).
- **Correctif** : position de boîtes de dialogue en multi-écrans.
- **Création** : éditeur graphique : nouvelles vannes « 3 voies ».
- **Création** : nouvelles fonctions « TahHsRs(kJ ; kg/kg as) » et « TrPv(Pa ;0.0) ».
- **Correctif** : Ladder / Littéral : fonction PVapeurPatmRs ajustée.
- **Modification** : Éditeur graphique : les objets contenus dans des groupes peuvent être sélectionnés individuellement avec la touche [ALT].
- **Correctif** : très rarement, la clé peut ne pas être reconnue sur passage RUN→STOP (V11.05e seulement).
- **Correctif** : pilote Modbus TCP-IP avec requête FC23 : des états de sorties à 0 pouvaient s'intercaler entre les échanges d'états corrects.

11.05e

- **Modification** : en RUN, lors de l'ouverture/déplacement de boîtes de dialogue, le temps de « figeage » des tâches programmes et communication est limité (notamment lors du maintien du clic gauche sur la barre de titre).

11.05d

- **Correctif** : très rarement, un changement d'état pouvait ne pas être transmis au PLC.
- **Correctif** : l'attribut « lecture seule » sur un fichier .SIM interdit sa modification.
- **Correctif** : en Run, l'affichage de page vide via la variable SYS_PAGE_PROSIMUL est bloqué et provoque l'affichage d'un message d'erreur.

- **Correctif** : la copie de variables dans les boîtes d'animation ne fonctionnaient pas si la sortie de la boîte de dialogue se faisait avec le bouton « OK » après la copie.
- **Correctif** : objets vérins : les animations pouvaient laisser une « trace » à l'écran.
- **Modification** : dictionnaire : si un dictionnaire est actif, les modifications des libellés d'objets « textes » sont prises en compte dans le dictionnaire dans la langue active.

11.05c

- **Création** : nouvelles fonctions « Frigo ».
- **Correctif** : mouvement Y des groupes corrigés (inversés).

11.05b

- **Correctif** : upgrade : le format des valeurs antérieur à V11 est conservé.
- **Création** : littéral : touches F1 & F2 pour faciliter la mise en commentaire.

11.05a

- **Création** : graphiques : nouvel objet « Notepad » permettant l'affichage d'un fichier texte.
- **Correctif** : fonctions Getxxx pour objets vérins sont limitées à la sortie de tige.
- **Correctif** : fonction « Undo ».
- **Création** : lancement Pro32xx.EXE avec paramètre M1 à M5 (écran 1 à 5).
- **Création** : graphiques : nouvel objet « Vidéo » utilisant le lecteur Pro32Video.EXE et l'instruction VideoLoad.
- **Création** : ladder/litteral : nouvelle fonction « ScreenParam ».
- **Correctif** : runtime : défaut d'affichage mineur en mode « objets ».
- **Correctif** : dossier application incomplet.
- **Création** : la taille de police des éditeurs Ladder et Littéral peuvent être réglées dans les préférences.
- **Création** : les couleurs des listes Ladder peuvent être réglées via les préférences.
- **Création** : une palette de 16 couleurs « User » peut être utilisée. Les couleurs se règle via les préférences et sont stockées dans PROSIMUL.INI. Dans la boîte de dialogue des couleurs, un bouton « palette » permet d'utiliser la palette « System » ou « User ».

V11.04

V11.04b (17/01/2017)

- **Correctif** : Pilote EIP : le n° de connexion est codé sur 2 octets et intègre l'octet bas de l'adresse IP cliente (cas de plusieurs PC clients communiquant avec le même serveur IP).
- **Correctif** : Runtime : si ouverture d'une boite de dialogue, les animations graphiques restent actives.

V11.03

V11.03d (15/12/2016)

- **Correctif** : Éditeur graphique : correctifs mineurs d'affichage (en mode page).
- **Création** : pilote pour cartes E/S parallèles sur port USB **Sealevel 8227**.
- **Correctif** : Pilote OPC : défaillance de l'application si le pilote OPC utilisé n'est pas installé.
- **Correctif** : Éditeur graphique : sur changement de page avec fond coloré le scintillement est évité.
- **Correctif** : Éditeur Littéral : risque de sauvegarde partielle en environnement virtuel.
- **Création** : protection contre les doubles ouvertures de projet sur un même ordinateur.

V11.02

V11.02 (09/11/2016)

- **Correctif** : pilote ENIP : adjonction des fonctions fin de session et reprise de session.
- **Création** : Éditeur graphique : l'objet trait peut être « caché » avec des couleurs d'affichage et/ou d'effacement à la valeur -1.
- **Création** : Éditeur graphique : fonction « Collage objet seul » (sans attributs d'animation).
- **Correctif** : animation des groupes (attributs CACHE et X/Y ne fonctionnaient pas en simultané).

- **Création** : Ladder/Litteral :
  nouvelle instruction « % » : équivalent à MOD (modulo)
  nouvelle instruction « >< » : proximité d'une valeur : OP1><OP2 équivaut Abs(OP1)<=OP2
  ex : B_CAPTEUR = (W_POSITION – 100) >< 10
  → B_CAPTEUR passe à 1 si W_POSITION est compris entre 90 et 110 inclus
  les opérateurs « + , - , *, / , % » peuvent être utilisés en Littéral dans les équations logiques
- **Modification** : Pilotes Ethernet : le temps de « repos » UC forcé devient variable (10% de la période de communication choisie). Évolution réalisée pour les PC peut puissants.
- **Modification** : Pilotes Ethernet : les trames reçues peuvent contenir plus de 255 octets.
- **Création** : Éditeur graphique : le pas d'affichage et de magnétisme de la grille peut être différent.
- **Correctif** : Éditeur graphique : correctifs mineurs d'affichage des boîtes « Couleurs », « Styles » et « Polices ».
- **Modification** : Graphiques/Ladder/Litteral/Grafcet : les items non utilisés n'apparaissent plus dans le navigateur. Un **clic droit** sur la racine permet de faire apparaitre le menu « **Ajouter** ».
- **Création** : Ladder/Litteral : les instructions « Call » peuvent faire référence aux noms des programmes. Ex Ladder : CALL#MONPROG Ex Littéral : Call 'MonProg' (entre simples guillemets).
- **Modification** : Ladder/Litteral : le nombre de programmes passent de 10 à 16.

V11.01 - Upgrade

V11.01 (09/10/2016)

- **Modification** : serveurs Ethernet Modbus/EIP/S7-ISO : la période de communication ne s'applique plus qu'à la lecture des sorties automate. L'écriture des entrées automate est réalisée dès qu'un changement d'état de celles-ci est détecté dans PROSIMUL. Le temps de cycle de communication est amélioré en moyenne de 30%.
- **Correctif** : pilote EIP : défaut de reprise possible en cas de coupure d'échange.
- **Modification** : pilote EIP et S7-ISO : la fonction « Synchro » des échanges est disponible.
- **Création** : si nécessaire, les adresses IP clientes peuvent être configurées dans PROSIMUL.INI.
- **Correctif** : Grafcet : défaut possible de positionnement de symboles lors de la construction des graphes.
- **Correctif** : Littéral : les tableaux numériques peuvent contenir plus de 100 éléments.
- **Correctif** : Littéral : bug possible lors du passage en Run si un défaut de syntaxe est détecté.
- **Modification** : Littéral : le nombre de lignes maxi par programme est doublé (1024) et le nombre d'opérandes maxi par ligne passe de 10 à 15.
- **Correctif** : bug possible lors du remplacement d'élément dans une librairie .BIM.
- **Création** : Éditeur Graphique : l'objet « Groupe » accepte 2 animations qui agissent sur l'ensemble des objets groupés (Cache et Déplacement en X, Y).
- **Création** : Éditeur Graphique : fonction « Zoom[+]/[-] » disponible en Stop.
- **Correctif** : Éditeur Graphique : pilotage objet « Curseur » hors limite supérieure.
- **Modification** : Éditeur Graphique : « Undo » fonctionne sur 5 niveaux.
- **Création** : Éditeur Graphique : fonction « arrière-plan ».
- **Modification** : Éditeur Graphique : la zone de sélection des traits droits est limitée autour du trait.
- **Modification** : la sélection de l'écran d'affichage est possible par déplacement de fenêtre sur un autre écran ou via la commande successive « Window→Sélection écran ».
- **Création** : le changement de résolution écran est pris en compte (sur changement d'écran).
- **Modification** : la fenêtre principale peut être redimensionnée.
- **Création** : l'état de la communication série est visualisé par un fond vert dans la partie droite de la barre d'état. L'erreur de communication apparait sur fond rouge. La présence de variables forcées est indiquée par un fond jaune. Format d'affichage :
hh :mm :ss [Temps de cycle maxi] [F : nombre var forcées] [C : nombre d'échanges].

V10.30 (11/07/2016)

- **Modification** : pilote EthernetIP : les fichiers et variables supérieurs à 254 sont accessibles.
- **Modification** : tracés des liens des blocs fonctions.
- **Création** : Runtime : auto-adaptation de la taille des images si le mode d'« Animations » est positionné sur « Page » (Configuration/Application) **et** qu'une limite d'écran est indiquée (Outil Grille).
- **Création** : Editeur Graphique : la touche [ALT] permet les sélections successives des objets en arrière-plan.
- **Création** : Editeur Graphique : outil Rotation d'un groupe d'objets (clic droit).
- **Correctif** : bug possible lors de l'appel de la boîte de dialogue « fichiers » (V10.29 seulement).
- **Correctif** : bug possible lors de l'appel Visudyn.
- **Création** : Editeur Graphique : nouvel objet « Table_Bargraphe ».
- **Création** : Ladder / Littéral : fonction AtgPi(x ;y) retourne la tangente entre -180° et +180°.

V10.29 (01/03/2016)

- **Création** : la variable SYS_INIT_APPLI passe à 1 au 1er tour de cycle.
- **Création** : en RUN un backup des données peut être réalisé dans un fichier nommé NOMAPPLI.DAT. Ce fichier peut servir à initialiser les valeurs de l'application.
- **Modification** : boîtes de dialogue « Fichiers » standards.
- **Correctif** : impression pour Windows 64 bits.

V10.28 (04/12/2015)

- **Modification** : Tous pilotes de communication : possibilité de permuter les octets de poids forts et faibles.
- **Correctif** : Objet « courbe auto » – Défaut d'affichage si échelle Y = 1.
- **Correctif** : Runtime stop – Défaut d'arrêt possible si communication HS.
- **Correctifs** : Horodatage – Gestion des listes + Export des données + Suppression variables.
- **Correctif** : Graphiques - Affichage incorrect des courbes avec des valeurs décimales.

V10.27 (09/09/2015)

- **Modification** : Éditeur Graphique – Fonction « Nettoyer ». Le type des variables d'animation est contrôlé.
- **Correctif** : bug possible lors de la création de programme littéral (V10.26 seulement).
- **Modification** : Pilote S7-ISO-TCP : permutation des octets de poids forts et faibles.
- **Correctif** : des relances de communication sur timeout (S7-ISO-TCP et ETHERNETIP).

V10.26 (30/07/2015)

- **Création** : Ladder/Littéral : possibilité de mettre en avant une autre application (fonction StopApp).
- **Création** : Ladder/Littéral : possibilité de stopper une autre application (fonction StopApp).
- **Création** : Ladder/Littéral : création de la fonction InfoApp (récupération d'information sur les applications Prosimul en mémoire).
- **Création** : Éditeur graphique : objet « Mobile » triangulaire orienté horizontalement.

- **Création** : Ladder/Littéral : pour les instructions d'information faisant référence à des objets graphiques (GetSX, GetLeftX, …), au lieu du numéro le 2ème paramètre peut être le libellé de l'objet entouré de guillemets, par exemple GetSX(0 ; "Tapis1").
- **Création** : Blocs Fonction : en Run les valeurs d'E/S des blocs sont affichées en dynamique.
- **Création** : La variable « SYS_PAGE_PROSIMUL » peut appeler par programme les pages Grafcet (20 à 27) et Blocs Fonctions (30).
- **Création** : Boîte de saisie de valeurs avec touches numériques.
- **Création** : Éditeur Graphique : nouvel objet « COURBE_AUTO » permettant d'afficher l'heure en abscisse.
- **Correctif** : défaut de sortie de PROSIMUL avec timeout série > 500ms.
- **Correctif** : Bloc fonction : risque de défaut d'exécution sur cycle PROSIMUL court (< 10ms).
- **Correctif** : risque de défaut d'affichage sur cycle PROSIMUL court (< 10ms).
- **Création** : Fonction GetDateTime. Si le 1er paramètre vaut 10, la fonction retourne le nombre de millisecondes cumulées depuis le début de la journée (0h).

V10.24-25 (30/05/2015)

- **Création** : Pilotes Ethernet TCP/IP : le numéro de port TCP peut être sélectionné.
- **Modification** : Runtime : le fonctionnement des priorités est révisé.
- **Création** : Applications Maître/Esclaves. La communication en **protocole Modbus** client peut être synchronisée (1 application communique après l'autre) → configuration Application (Synchro Data).
- **Modification** : Dossier : pour les imprimantes PDF, 1 seul fichier est utilisé en sortie.
- **Création** : Pilotage des objets graphiques binaires. La variable « Cmd_Mode », si >= à 100 spécifie une commande pulsatoire de durée >= à 100ms (uniquement pour les nouvelles créations d'objets).
- **Modification** : Configuration EthernetIP : l'adresse du « Channel » peut être saisie derrière l'adresse IP (ex : 192.168.2.66/3).
- **Modification** : blocs fonction « Convoyeur » : en cas de raccordement amont / aval de blocs « Convoyeurs » entre eux (MAV sur MAV et MAR sur MAR), les blocs sont synchronisés sur le même tour de cycle.
- **Modification** : les objets « Table_Fill » et « Table_Value » peuvent s'orienter de haut en bas et de bas en haut.
- **Correctif** : plantage éventuel lors de passages Stop->Run ou Run->Stop.
- **Correctifs** : éditeurs Fonctions et Grafcet : divers correctifs mineurs de sélections d'objets.
- **Correctif** : problème lors de la création d'objets graphiques (Curseur, Vumètre).

V10.22-23 (31/03/2015)

- **Modification** : l'aide au format HLP n'est plus mise à jour. Seul le fichier PDF est exploité.
- **Modification** : protocole Modbus. Le numéro d'équipement peut être indiqué derrière l'adresse IP (ex : 192.168.2.65/2 pour équipement Modbus 2).
- **Modification** : protocole Modbus. La transmission des poids faibles en tête est possible via des champs prévu dans la configuration – 1=Poids Fort en tête (Normal), 0=Poids faible en tête. Les applications Modbus doivent être reconfigurées.
- **Correctif** : Runtime. Défaut d'animation si l'option « Initialisation » n'est pas cochée.
- **Création** : Éditeur graphique. Fonction « Dupliquer » (=copier/coller).
- **Création** : Éditeur graphique. Les courbes maintiennent en mémoire les 1000 dernières valeurs.
- **Correctif** : Éditeur graphique. Valeur de retour des GIF animés.
- **Correctif** : Éditeur graphique. Défaut de création d'objets (version anglaise seulement).

V10.20-21 (08/01/2015)

- **Création** : Rajout des commandes [CTRL][C/V/X] (copier/coller/couper) dans éditeurs Graphiques/Ladder/Grafcet/Fonctions (déjà existants pour Littéral).
- **Correctif** : Objet « Curseur ». Défaut de pilotage avec les flèches de direction gauche / droite.
- **Création** : Fonction : nouvelle fonction « PID Mixte » (l'ancienne est de structure parallèle).
- **Création** : à l'ouverture d'une application APPLI.SIM le fichier APPLI.BAT est exécuté si existant.
- **Correctif** : Fonction BODE (uniquement sur V10.19).

V10.19 (09/09/2014)

- **Correctif** : Ladder : opérations TBCOP et NCOP (uniquement sur V10.18).
- **Création** : Ladder/Littéral : nouvelles instructions pour le traitement des fluides :« DeltaP », « Flow ».
- **Correctif** : Editeur Littéral : le second opérande de fonctions admet des opérateurs.

V10.18 (23/07/2014)

- **Création** : Ladder/Littéral : nouvelles instructions pour le traitement de l'air humide :« DeltaZero », « PVapeurPatmRs », « PSVapeurTah », « PVapeurTahHr », « RoAirPasTabs », « RoVapeurPvTabs », « RsPvPas », « HsTahRs », « HkDtRs ».
- **Modification** : Ladder/Littéral : uniformisation du jeu d'instructions et optimisation des temps d'exécution.
- **Création** : Editeur graphique : nouvel objet pupitre de type « Bargraphe » à N niveaux.
- **Correctif** : Editeur Grafcet : la renumérotation des étapes modifie les libellés existants en langage Littéral.
- **Correctif** : Editeur Variables : la fonction incrément modifie les libellés existants en langage Littéral.

V10.16-17 (23/06/2014)

- **Création** : la communication avec le pilote OPC CoDesys est intégrée.
- **Correctif** : horodatage. Limitation des valeurs affichées à 10.000 en dynamique (plantage pour de grandes quantités selon la configuration des applications).
- **Correctif** : bloc fonction PID. La saturation mini <> de 0 était erronée.
- **Correctif** : détection des imprimantes au lancement (cause de crash au lancement).
- **Correctif** : fonction horodatage : échelle de temps de plus de 24h corrigée.
- **Correctif** : fonction horodatage : accès concourants aux enregistrements en RUN.
- **Correctif** : blocs fonction : traits « aléatoires » lors du passage en STOP supprimés. Reprise graphique des traits de liaison.
- **Création** : Test des éditeurs graphiques avec Windows 64 bits (hors communication).

V10.15 (13/05/2014)

- **Création** : Editeur graphique : le menu grille ou la touche [L] permettent d'afficher des limites d'écran utilisateur (sans, 800x600, 1024x768, 1280x1024).
- **Correctif** : Editeur Ladder : correctif instruction « EBM » (écriture bit de mot - obsolète).

V10.14 (01/04/2014)

- **Création** : Editeur graphique : la touche [0] permet de modifier toutes les couleurs d'animation des objets sélectionnés.
- **Création** : Horodateur : les listes de paramètres NOMAPPLI_*.INI sont gérées et accessibles directement via le menu « Paramètres ».
- **Création** : Horodateur : les enregistrements sont cherchés à +- 10 jours de la date courante.
- **Création** : Editeur graphique : nouvel objet « GIF animé ».
- **Création** : Littéral/Ladder : nouvelle fonction « StopApp » (état Lock, arrêt ou redémarrage application).

V10.12x → V10.13 (04/03/2014)

- **Correctif** : configuration OPC : saisie de l'adresse cible contenant des lettres P/N/R/S/U.
- **Correctif** : Editeur graphique : mauvais affichage de valeurs réelles pour l'objet TAB_SYMB.
- **Correctif** : EthernetIP : inversion des octets de poids fort et faible + bug ouverture socket.
- **Correctif** : la fonction horodatage pouvait modifier la table des échanges.
- **Création** : l'accès aux boitiers Usb FESTO Easyport est intégré (64DIO+16AI+8AO max).
- **Création** : nouvelle fonction « MsgLog » pour écrire le fichier LOG par programme (voir chapitre Littéral).

V10.11 (31/01/2014)

- **Modification** : Saisie variables. Création de raccourcis clavier + clic droit souris.
- **Correctif** : Littéral : boucle for avec index décroissant.
- **Création** : sur passage en STOP la variable SYS_CYCLE_PROSIMUL prend la valeur -1 pendant 1 tour de cycle pour détection de passage en STOP.
- **Création** : en cas d'erreur de communication la variable SYS_COMPTEUR_ECHANGE prend la valeur -1 et peut être testée.
- **Modification** : Horodatage. Le stockage des fichiers HVS peut être organisé par année (\Chrono\AAAA).
- **Correctif** : client OPC : plantage lors d'un double passage STOP->RUN (obligeait à sortir du logiciel).

V10.10 (10/01/2014)

- **Correctif** : textes Unicodes mal dimensionnés. Pouvait provoquer un « crash » de PROSIMUL.
- **Modification** : Editeur Graphique. Création de raccourcis clavier.
- **Modification** : Editeur Graphique. Création d'objets « TUYAU » avec animation de couleurs contour / fond.

V10.06/07/08/09 (02/01/2014)

- **Correctif** : les paramètres par défaut des textes/valeurs (police) sont mémorisés dans le projet .SIM.
- **Correctif** : éditeur graphique : les objets images de taille nulle (RATIO_X.Y) conservent une taille de 10 pixels en Stop pour pouvoir être sélectionnés.
- **Correctif** : la période d'exécution de PROSIMUL est réglée par défaut à 10ms et ne peut descendre en dessous de 1ms.
- **Correctif** : de la fonction import/export (défaut lié seulement à la version 10.05).

- **Correctifs** : éditeurs Grafcet et Fonction (déplacement de cellules et curseurs).
- **Modification** : les blocs fonctions « Intégrateur », « Sommateur » et « Polynôme » disposent désormais des paramètres « seuil minimum » et « valeur initiale ».
- **Modification** : Horodatage : les flèches gauche et droite permettent de décaler les courbes d'une base de temps en amont ou en aval de l'affichage courant.
- **Création** : d'un objet graphique « TABLE_SYMB » (rubrique « Divers ») permettant l'affichage direct d'une liste de symboles et de leurs valeurs (forçables en RUN).
- **Modification** : pour les objets graphiques « TABLE » avec paramètre « Forçage » validé, en mode Run, la variable forcée sur clic souris est celle située sous le curseur de la souris.
- **Modification** : éditeur LADDER : rajout de la molette souris pour déplacement dans les instructions.
- **Modification** : éditeur LADDER/Littéral : le programme principal peut être écrit en Littéral.
- **Création** : éditeur Fonction : nouveau bloc « Split-Range » (fonction Chaud/Froid ou fonction rapide/précis).
- **Modification** : Boîte de dialogue de configuration des échanges + trames Ethernet passe de 256 octets à 1024 maximum.
- **Création** : éditeur Graphique : nouvel objet « pupitre » POUSSOIR (double rond).
- **Modification** : éditeur Graphique : l'appui sur la touche [CTRL] permet de sélectionner une zone d'objet en évitant de sélectionner un objet directement sous le curseur.
- **Modification** : éditeur Graphique : la touche [M] permute la fonction magnétisme.

V10.05 (01/08/2013)

- **Création** : pilote de communication « **S7-ISO on TCP** » pour automate S7 sur liaison Ethernet.
- **Correctif** : de la fonction « copier / coller » de l'éditeur graphique via presse-papier.
- **Création** : fonction horodateur : les variables affichées et les paramètres d'affichage peuvent être sauvegardés et lus depuis des fichiers .INI différents.

V10.04 (28/06/2013)

- **Création** : les fonctions copier/coller des éditeurs fonctionnent désormais via le « Presse-papier ». Ces fonctions peuvent donc fonctionner entre 2 applications .SIM avec bases identiques.
- **Correction** : de la fonction horodatage en visualisation dynamique (échelle de temps incorrecte ou plantage application selon les cas).
- **Création** : Editeur Graphique : possibilité d'afficher les textes en taille fixe (Fixed VSize) ou auto adapté selon la longueur du texte (Auto HSize).
- **Création** : Fonction Horodatage : possibilité d'Export du jour affiché et du mois ou de l'année complète en fichier HVX. Un outil sous EXCEL peut être fourni pour l'analyse des données des fichiers HVX.
- **Création** : Editeur Graphique : un clic droit avec des valeurs sélectionnées permet de reformater l'ensemble des valeurs.
- **Modification** : lors du passage en RUN PROSIMUL démarre sur la dernière page graphique affichée.

V10.03 (23/05/2013)

- **Création** : fonction Horodatage : si un sous-répertoire « Chrono » existe, les fichiers HVS y sont stockés et recherchés.
- **Création** : fonction Horodatage : les variables et paramètres sélectionnés sont transmis dans un fichier NOM_APPPLI.INI qui est relu à chaque changement de jour ou réouverture de la fonction

Horodatage.
- **Création** : fonction Horodatage : un outil « Moyenne » de valeurs est disponible (0=off).
- **Correction** : les valeurs entières sont affichées selon le nombre de digits choisis.
- **Modification** : amélioration de l'outil plan de Bode (Fonctions).
- **Création** : les applications esclaves (Database partagée) sont automatiquement fermées en cas de passage en stop de l'application maître.
- **Création** : bloc fonction PID avec paramètres Kp, 1/Ti, Td, Td/N et limites de saturation d'intégrale.
- **Modification** : passage du nombre de variables de la base à 2048 au lieu de 1024.

V10.02 (02/04/2013)

- **Corrections** : de défauts mineurs (Editeur Grafcet, blocs fonction, objet graphique CURSEUR associé avec des entiers, ...).
- **Création** : le fichier d'horodatage quotidien HVS peut être réinitialisé à chaque lancement de l'application.
- **Correction** : configuration des échanges via le pilote S7-ONLINE.

V10.01 (22/03/2013) - Upgrade

- **Création** : Fonction **Horodatage**.
Les fichiers HVS sont enregistrés par jour en continu. Les sélections de variables se transmettent d'un jour à l'autre. Le format des noms doit être conservé : NOMAPPLI_YYYY-MM-DD.HVS.
Le menu Horodatage permet de naviguer d'un jour amont ou aval ou de charger une date précise. Par défaut le jour courant est chargé et est visualisé en dynamique pour le jour courant.
La liste de variable surveillé passe de 25 à 100 et est classée par ordre alphabétique.
- **Modification** : Tables de **VisuDyn** (FORCAGE / LITTERAL-LADDER / HORODATAGE) sont désormais classées par ordre alphabétique.
- **Création** :. Création d'une fonction SendMail (voir chapitre Littéral).
- **Création** : Création d'une fonction GetDateTime (voir chapitre Littéral).
- **Création** : Création d'une fonction WinExec (voir chapitre Littéral).
- **Création** : Les applications PROSIMUL peuvent fonctionner en maître-esclave (boîte de dialogue Configuration Application). Par exemple, 1 application est maître de la base de données les autres peuvent communiquer simplement avec d'autres automates. L'origine PROSIMUL des E/S TOR échangée peut être modifiée (boîte de dialogue Configuration Echanges).

V9.05 (XX/01/2013) – Non distribuée

- **Correctif** : corrections de défauts d'affichage dans la fonction horodatage.
- **Création** : création d'un objet graphique « Analogique-Vumètre ».
- **Création** : création d'un objet graphique « Curseur » pouvant être piloté.
- **Modification** : le format d'enregistrement de l'horodatage est modifié (fichier .HVS) pour inclure la date et l'heure de début d'enregistrement.
- **Création** : éditeur graphique : création d'une fonction Zoom géré avec la **molette** de la souris + touche CTRL.

V9.04 (12/10/2012)

- **Modification** : les objets « Texte », « Pupitre », « Tuyau », « Moteur », « Pompe » disposent maintenant de variables d'animation pour les mouvements X, Y.

- **Création** : accès au serveur OPC MATRIKON (Matrikon.OPC.Universal.1).
- **Création** : Pilote EthernetIP (Rockwell). Accès via Ethernet aux automates avec port Ethernet intégré ou via modules ENI. Accès aux fichiers Nx seulement (pour Tags configurer un mappage).
- **Création** : commande Lock en mode Run. Si un l'application est protégée, le mot de passe doit être saisi.
- **Modification** : lors de l'import d'un dictionnaire, les traductions non trouvées sont identifiées par le texte "-----Not Found------" dans le fichier PROSIMUL.LOG.

V9.03 (19/10/2011)

- **Création** : accès au serveur OPC Kepware.KEPServerEX.V5.

V9.02 (06/05/2011)

- **Correctif** : correction du nom du serveur OPC Rockwell.

V9.01 (15/04/2011)

- **Correctifs** : corrections de différents défauts mineurs d'affichage d'objets contenant des textes.

V9.00 (15/03/2011) - Upgrade

- **Création** : protection des applications .SIM par mot de passe. Le mot de passe est configuré dans la boîte de dialogue « Configuration application ».

V8.02 (xx/08/2010)

- **Création** : accès aux serveurs OPC distant. Indiquer le type de serveur et l'adresse IP du serveur. ATTENTION : **les configurations d'échange OPC existantes doivent être modifiées** (consulter les chapitre OPC dans cette documentation).
- **Création** : l'accès aux variables en MODBUS TCP-IP peut se faire soit par les FC03/16 soit par la FC23 (R/W multiples registers – non supportée par tous les équipements).
- **Correction** : en cas de groupement d'objets dans certains cas, un défaut d'affichage de l'image pouvait être rencontré. Un outil de « nettoyage » a été créé pour les applications rencontrant ce problème (voir menu Graphique / Nettoyer)
- **Correction** : de l'affichage des textes centrés ou cadrés à droite.

V8.00/V8.01 (25/04/2010-10/05/2010) - Upgrade

- **Création** : les libellés des objets graphiques sont désormais traités en format **Unicode**. Ils peuvent donc être saisis avec les différentes polices de caractères internationales. Une fonction (menu principal) permet d'exporter/importer des **dictionnaires** multilingues et de lancer l'application en Run dans une autre langue que la langue de création (Configuration Application). Les dictionnaires doivent être aussi en format Unicode (2 octets par caractère). Les applications doivent être converties (Upgrade V7->V8).
- **Création** : mode de communication MODBUS TCP/IP en version serveur (attente de connexion d'un client).

V7.30/31 (18/07/2008)

- **Création** : Editeur graphique : la variable « Cmd Mode » est adjointe aux objets « Texte ». Elle permet de définir le mode de pilotage (0 :pousser/pousser, 1 : maintien)
- **Création** : Editeur Ladder et Littéral : création d'une instruction « Writeini(OP1 ;OP2) » permettant de stocker dans OP1 et son état initial la valeur courante de OP2.
- **Création** : Echanges : nouveau pilote de communication « MODBUS-TCP/IP » permettant l'accès direct aux automates via les WinSocks.
- **Modification** : Runtime – classement des listes de variables dynamiques par types.

V7.29x (21/04/2008)

- **Création** : Création d'un menu « Window » pour accéder aux fenêtres PROSIMUL.
- **Création** : Gestion de la molette de la souris dans les pages graphiques et variables.
- **Création** : Graphiques – gestion des mouvements de la fenêtre avec la molette de souris et le clavier (touches de direction, HOME et END).
- **Création** : variables – mémorisation des positions dans les tables de variables.
- **Correction** : Ladder – instruction de déforçage « U#BIT » opère désormais de manière conditionnelle.

V7.28x (17/03/2008)

- **Création** : Communication – Validation des interfaces (OPC/DataBase) avec Direct-Link SW1000 (Applicom / Woodhead).
- **Correction** : problème d'affichage des objets graphiques marqués « Dev.Only ».
- **Correction** : éditeur graphique – fonction premier plan échoue si plus de 400 objets sur page.
- **Modification** : éditeur littéral – la sélection par double-clique sélectionne l'ensemble du mot.
- **Correction** : éditeur graphique – défaut d'affichage des objets de type « tableaux ».
- **Correction** : plantage lors du passage en STOP si mouvement de souris sur une page graphique ayant des objets groupés.

V7.27 (Bêta1-4)

- **Correction** : Reprise de la fonction dossier (problème si page graphique non ouverte).
- **Correction** : plantage si des commentaires existent en Ladder dans certaines conditions.
- **Correction** : plantage lors du passage en STOP si affichage et mouvement de souris sur une page graphique ayant des objets groupés.
- **Création** : Editeur graphique : création de l'objet « IMAGES » dans la catégorie « Divers » (formats BMP/GIF/JPG/PNG/TIF/EMF acceptés). Cet objet peut être déplacé, réduit/agrandit et tourner selon un point de rotation indiqué. Il utilise la librairie GDIPLUS.DLL.
- **Création** : Editeur graphique : une variable « Cmd Mode » est adjointe aux objets « Pupitre », et « Tuyau ». Elle permet de définir le mode de pilotage (0 :pousser/pousser, 1 : maintien)
- **Création** : Editeur graphique : Modification directe de la taille d'un groupe d'objets.
- **Création** : Nouvelles fonctions permettant la récupération des dimensions, positions et coordonnées d'un objet (GetSX, GetSY, GetX1, GetY1, GetX2, GetY2, GetLeftX, GetTopY, GetRightX, GetBotY). Syntaxe utilisée : Get…(n°page, n° objet) selon informations obtenues en renseignant l'objet (dans la barre de titre).
- **Création** : Nouvelles fonctions de conversion Int(X ; Y) et Float(X ; Y) utilisables en Ladder et

principalement Littéral.
- **Création** : Editeur graphique : nouvel objet « Convoyeur » dans la famille « Rotation_2D ».
- **Création** : Editeur graphique : nouveaux objets « Tableaux » dans la famille « Divers ». Ils permettent d'afficher des tables de valeurs (le 1er élément est la dimension) sous forme de cases colorées (type objet REMPLI) et avec valeurs affichées.
- **Modification** : Blocs fonctions, « Retard pur » et « Convoyeur » : les valeurs issues des blocs peuvent être écrites dans la Database via le paramètre VALUES (à concurrence du nombre de périodes des blocs).
- **Correction** : instruction de copie de tableaux TAB1[N] = TAB2[N].
- **Correction** : Grafcet – cellules effacées pouvant pointer sur des variables non nulles.
- **Correction** : Littéral – l'écriture à 0 d'une chaîne de bits peut produire un mauvais résultat.
- **Modification** : Configuration Application – Le mode graphique d'affichage des objets en RUN peut être choisi selon le meilleur temps d'affichage ; mode « Page » : l'ensemble des objets de la page sont redessinés ; mode « Objets » : seuls les objets évoluant (et leurs arrière-plans) sont redessinés.

V7.26 (19/04/2007)

- **Modification** : Rajout des mouvements X et Y aux objets « ROTATION 2D ».
- **Création** : Création de la fonction Hypoténuse (Hypo) en Littéral et Ladder.
- **Création** : Création de l'objet Graphique « CHARNIERE » en mécanique.
- **Modification** : Rajout automatique des caractères « .0 » si omis pour les variables réelles.
- **Correction** : Editeur Littéral – saut intempestif de zone de programme affiché.
- **Modification** : Editeur graphique – La taille des objets peut être modifiée dans la boîte de paramétrage.
- **Correction** : Editeur Littéral – Instruction Logn corrigée.
- **Création** : Editeurs Grafcet et Fonctions – Les éléments peuvent être déplacés avec la souris après sélection.

V7.25 (03/04/2007)

- **Correction** : Taux d'occupation mémoire s'incrémentant en mode RUN.
- **Modification** : Lors du lancement par double-clic sur l'application .SIM et si l'application est configurée en mode « Formation », le passage en RUN est automatique.
- **Modification** : Le message « Pas d'imprimante » est dirigé vers le fichier PROSIMUL.LOG.

V7.24 (29/03/2007)

- **Correction** : Problème de mémorisation de résultat de compilation de programmes Littéral sur passage en RUN.

V7.22/23 (15/03/2007)

- **Modification** : Editeur Graphique – Animation des distributeurs T1.
- **Modification** : Editeur Graphique – Animation des vérins : la couleur de fond des vérins est utilisée pour animer la partie de fluide évacuée.
- **Modification** : Littéral – Instruction MOD – Rajout du test de modulo effectué avec valeur nulle.
- **Modification** : Littéral –Rajout de tests pour séparateurs « : » dans les structures.
- **Modification** : Littéral – Commentaires multi lignes - Un commentaire peut s'étendre sur plusieurs lignes. Il débute sur la ligne contenant (* et se finit après la ligne comportant les caractères *)

juxtaposés.
- **Création** : La périodicité d'exécution du Runtime peut être paramétrée dans la configuration Application.
- **Correction** : Client OPC – Modification pour défaut démarrage sur XP.
- **Correction** : Verrouillage des fonctions inaccessibles avec clé Exploitation.

V7.21 (17/12/2006)

- **Correction** : Editeur Littéral – Le remplacement des variables n'opérait pas sur les variables précédées de tabulations.
- **Correction** : Editeur Littéral – Le nouvel éditeur est susceptible d'apporter automatiquement des corrections de caractères de fins de ligne. Un message le signale à l'utilisateur.

V7.20 (13/12/2006)

- **Création** : Editeur Littéral – Les couleurs d'affichage des Mots-clés, Symboles, Commentaires, Opérateurs,.., sont réglables via les préférences accessibles depuis le navigateur ou via le fichier PROSIMUL.INI chargé au démarrage.
- **Modification** : Les messages sont désormais stockés dans un fichier PROSIMUL.LOG (et non dans le fichier Errors.txt).
- **Modification** : Editeur Variables – Fonctions Couper/Coller/Supprimer/Insérer reprises pour permettre des opérations entre les fonctions « Couper » et « Coller ».
- **Correction** : Sauvegarde des programmes littéral modifiée (pb si passage en RUN après validation).
- **Correction** : Editeur Graphique – Les objets indiqué en « Dev.Only » mais non animé s'affichent correctement.

V7.16 (22/11/2006)

- **Correction** : Editeur Variables – Les modifications, insertions et libérations sont prises en compte complètement dans le Littéral.
- **Correction** : Runtime – Le message « Perte de communication » clignote dans la barre de message.
- **Création** : Les types de messages dans la barre d'état sont différenciés par couleur.
- **Création** : Le clic droit dans les zones de saisie de variables permet d'aller sélectionner des variables dans la DataBase (dernier menu).

V7.15 (31/10/2006)

- **Correction** : Client OPC – Les objets VARIANT sont réinitialisés pour éviter la consommation de mémoire.
- **Correction** : Runtime – Exécution des Blocs Fonctions réactivée.
- **Correction** : Outil de traduction – La traduction via fichier .TRA opère le remplacement de texte sur les graphiques et en option sur les variables et programmes.
- **Correction** : L'information « Perte de communication » s'arrête correctement lors du passage en STOP.
- **Correction** : Runtime Visudyn Ladder – Problème lors de la sortie de la visualisation dynamique de Temporisation à très petite base de temps.
- **Création** : Littéral – Lors de la modification de symboles, les variables figurants dans les programmes Littéral sont automatiquement modifiés.
- **Création** : Boîte de dialogue « Configuration Application » – La priorité d'exécution est réglable.

V7.14 (18/10/2006)

- **Correction** : En cas d'insertion/suppression de variables (autres que mots), les mots sélectionnés dans la table d'échange sont désormais conservés.
- **Modification** : Client OPC – Reprise du code pour éviter l'augmentation d'allocation mémoire observé en mode RUN avec communication.

V7.13 (06/10/2006)

- **Création** : Littéral – rajout de la commande de sélection globale [CTRL][A].
- **Modification** : En RUN, l'annulation de forçage (UnF en littéral ou U# en Ladder) ,n'est réalisé que si la variable est forcée (pour limiter les temps d'affichage).
- **Correction** : Tables des variables – Mode RUN – La sélection d'une variable (par bouton ou par activation de la zone de saisie) fige son animation et permet sa modification. Les touches [ESC] ou [ENTER] annule la sélection.

V7.12 (26/09/2006)

- **Correction** : En RUN en cas « d'INIT_APPLI » problème de perte de programmes Littéral.
- **Correction** : En RUN, la fenêtre « Variables » peut provoquer un ralentissement des échanges.
- **Correction** : En cas de perte de communication, un clic en dehors du message provoque un verrouillage des commandes PROSIMUL -> message cyclique toutes les 2 secondes.
- **Correction** : En cas d'insertion/suppression de variables, les mots sélectionnés dans la table d'échange sont désormais conservés.

V7.11 (09/09/2006)

- **Modification** : Configuration échange OPC : indication des serveurs OPC installés.

V7.10 (09/06/2006)

- **Modification** : Littéral – Amélioration de la vitesse de compilation.
- **Correction** : Littéral – Problème de perte partielle de programme sur passage RUN -> STOP.
- **Création** : Editeur Graphique – Objet « LIAISON_COURBE » dans la famille Mécanique.
- **Création** : Editeur Graphique – Objet Interrupteur 3 positions dans la famille « PUPITRE ». En RUN cet objet peut piloter 2 variables en ou exclusif (un clic sur la partie gauche de l'objet active uniquement la première variable et un clic sur la partie droite la seconde variable. Les deux variables sont remises à zéro par un clic en position centrale).

V7.08 (03/05/2006)

- **Création** : Mode RUN – Accès à la liste des variables surveillées définies dans la fonction « Horodatage ». L'accès est possible en RUN via la fonction « Outils » « Horodatage » de l'explorateur ou via un clic droit sur une zone vide des pages graphiques.
- **Correction** : Editeur Littéral – Priorité des opérateurs *, / , +, - .

Les opérateurs * et / ont le même niveau de priorité et sont traités de gauche à droite.
Les opérateurs + et - ont le même niveau de priorité et sont traités de gauche à droite.

Les priorités de fonctions / opérateurs sont définies dans la section Littéral de la documentation.
- **Correction** : Editeur Littéral – Insertion automatique des espaces après les commentaires.

V7.07 (27/04/2006)

- **Création** : Mode RUN – Accès à la liste des forçages (Forçage -> Liste).
- **Création** : Ladder / Littéral – En RUN la commande VisuDyn permet la visualisation en dynamique des variables utilisées dans le programme courant.
- **Correction** : Editeur Littéral – Détection des phrases avec opérateur manquant.
- **Correction** : Editeur Littéral – Test des opérateurs / et * inversé.

V7.06 (04/04/2006)

- Modification : Editeur Graphique – Boîte animation / nouvel affichage des coordonnées des points.
- Correction : Editeur littéral – mauvaise validation de comparaisons sans opérateur (pas de problème à l'exécution).
- Correction : Editeur de variables – impossibilité de modifier / forcer les valeurs en RUN.

V7.05 (28/03/2006)

- Correction : Littéral – les calculs effectués avec les opérateurs PLUS ; MOINS délivrent des valeurs correctes.

V7.04 (24/03/2006)

- Modification : Littéral – Imbrication de fonctions possibles (toujours 10 opérateurs maxi par ligne) - ex : A = **Sqrt (** *Pow (B ; 2.0) + Pow (.C ; 2.0)* **; 1.0)**
- Création : Editeur Graphique – nouvel objet « Mécanique » de type LIAISON.
- Correction : Editeur Graphique – Animation des vérins verticaux.
- Correction : Editeur Graphique – Trait CNEG et CPOS de taille nulle.
- Correction : Littéral – Création de SR entraîne la recopie de programme.

V7.03 (15/03/2006)

- Correction : Littéral – Transfert d'INT vers FLOAT avec opérateurs.
- Correction : Littéral – Saisie des valeurs réelles négatives (ex : -1.0).
- Correction : Littéral – Mise à 1 ou à 0 possible des bits extraits de mots (ex : M10 . 4 = 1).
- Création : Ladder / Littéral – fonctions ArcSinus (Asin) et ArcCosinus (Acos).
- Création : Animation – un double-clic sur la boîte propriétés des objets graphiques permet de modifier l'ensemble des couleurs d'animation.
- Création : Convertisseur – possibilité de choisir une permutation des couleurs noire et blanc.
- Création : Editeur Graphique – nouveaux types d'objets orientés « mécaniques ».

V7.02 (08/03/2006)

- Correction : Convertisseur – Mise à jour en fonction des évolutions 7.01.
- Correction : Import – Remplacement de variable unitaire.
- Correction : Animation des objets « FILL » incorrecte lorsque le CADRE est renseigné à 0.

- Correction : Variables – Sélection impossible des variables SYS_ en version 7.01 uniquement.
- Modification : Import de variables – Contrôle des doublons et des caractères interdits.

V7.01 (27/02/2006) - Upgrade

- Correction : Littéral – interprétation des équations logiques avec double parenthèse.
- Correction : Mode RUN – la commande « Priorité - » permet de descendre en niveau 1.
- Correction : Mode RUN – problème de rafraîchissement d'écran sur changements de pages.
- Correction : passage en STOP via double Echap.
- Modification : en cas d'upgrade de fichier .SIM le remplacement des caractères s'effectue de la manière suivante :
Les caractères "**(**" "**)**" "**,**" "**;**" "**:**" "**#**" "**.**" sont remplacés par le caractère "**_**"
Les caractères "**<**" "**>**" "**=**" "**+**" "**-**" "*****" "**/**" "**%**" "**[**" "**]**" sont remplacés respectivement par les caractères "**L**" "**G**" "**E**" "**P**" "**M**" "**X**" "**D**" "**R**" "**{**" "**}**"
- Modification : Communication en client OPC – Les données cible doivent être désignées sous forme textuelle (ex : %MW pour Schneider, DB20 pour Siemens, ou TAGNAME pour Rockwell).
- Création : les couleurs sont créées par défaut en variable SYS_COLOR_BLACK…WHITE sur les mots 5 à 20.
- Correction : Littéral – les calculs « mathématiques » (COS, SIN,…) effectués avec des variables de type entier délivrent des valeurs correctes.

V6.02

- Correction : Editeur Variable : problème d'insertion / suppression de variable lorsqu'un programme Littéral existe devant un programme Ladder.
- Correction : Editeur Littéral, fonction Insertion de variable depuis la DataBase.
- Création : Editeur Littéral : liste des 10 dernières variables en menu contextuel.
- Correction : Littéral : possibilité de déforcer tout type de variable (Unf)
- Correction : fonctions intervalle ?1, ?2, ?5, ?10 en Ladder et Littéral.
- Correction : forçage possible sur objet IMAGE en mouvement.
- Création : en mode Formation (configuration application) l'explorateur n'est pas affiché et les changements de page s'effectuent par les touches [PgUp] et [PgDn] ou la variable SYS_PAGE_PROSIMUL.
- Modification : Littéral – L'instruction FOR fonctionne avec un paramètre optionnel STEP
- Création : Menu Run – Le process PROSIMUL peut être monté ou descendu de priorité Windows depuis les commandes « Priorité Plus / Moins ».
- Création : Editeur Graphique : 2 nouveaux objets de type texte acceptant 4 couleurs de police (T_VISU) ou 4 couleurs de fond (T_FOND).
- Modification : objets TEXTE : si l'épaisseur choisie du Style est supérieure à 1 le texte est affiché en gras.
- Correction : opérateurs booléens et numériques – propagation correcte des résultats intermédiaires.

V6.01 - Upgrade

- Modification : pour pilote DF1-Direct il n'est plus nécessaire de multiplier l'adresse des mots pour automates PLC par 2
- Création : les applications .SIM font désormais l'objet d'un **versioning**. Les applications antérieures doivent être converties à l'ouverture de manière irréversible.
- Correction : les fichiers de traduction .TRA sont désormais traités avec ou sans ligne vide.

- Correction : les objets graphiques déclarés comme « Dev.Only » ne sont plus forçables.
- Modification : pour tous les pilotes série, le paramètre « Retries » permet de configurer le nombre d'échanges défaillants avant arrêt du pilote.
- Correction : le bloc fonction CONVOYEUR gère désormais correctement l'accumulation de matière à l'arrêt. L'état initial est aussi correct. Les valeurs de vitesse (COEFF) admissibles sont des entiers de –24 à +24 (0 = arrêt ; 1 = avance d'un pas à chaque période).
- Modification : les cibles « PROSIMUL » et « PRO32 » sont indiquées pour les pilotes Rockwell et client OPC et peuvent être modifiées (sous la forme NAME.X).
- Correction : Editeur Graphique : une position d'objet supérieure à 999 est valide
- Correction : les valeurs initiales des variables SYSTEME sont déverrouillées.
- Modification : la désignation des touches fonctions doit se faire avec les accolades « **{** », « **}** » dans les symboles.
- Création : Langage LITTERAL (choix entre LADDER ou LITTERAL pour les Sri).
- Création : menu principal – Raz_Errors : permet d'effacer directement le fichier errors.TXT

V2005.12a

- Création : Gestion des clés de protection USB ALADDIN