



TUTORIEL MOTION

[Axe linéaire avec VisiLogic]



Retrouvez-nous sur www.pl-systems.fr



Sommaire

Introduction.....	2
Matériel	3
Déclaration matérielle.....	3
Déclaration de la communication	3
Programmation	4
Les blocs fonctions	4
Le bloc de mise sous tension et de paramétrage.....	5
Le bloc de mouvement manuel.....	6
Le bloc de position Home.....	7
Le bloc de mouvement absolu	8

Introduction

Nous allons aborder dans ce tutoriel, la configuration et la programmation d'un axe linéaire avec un automate V700 programmé sous VisiLogic

Matériel

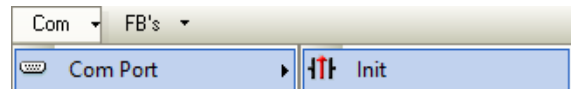
Déclaration matérielle

Contrairement à UniLogic, il n'y a pas de déclaration matérielle à faire dans le projet.

Déclaration de la communication

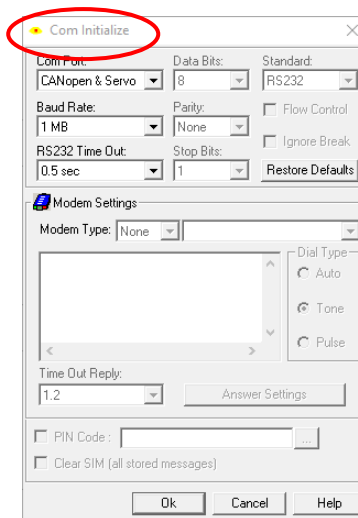
Il faut paramétrer la communication entre l'automate et le Drive via un module RS485 qui permettra de communiquer en CANopen.

Initialiser la communication avec un bloc « Init » se trouvant dans l'onglet « Com » → « Com Port ».

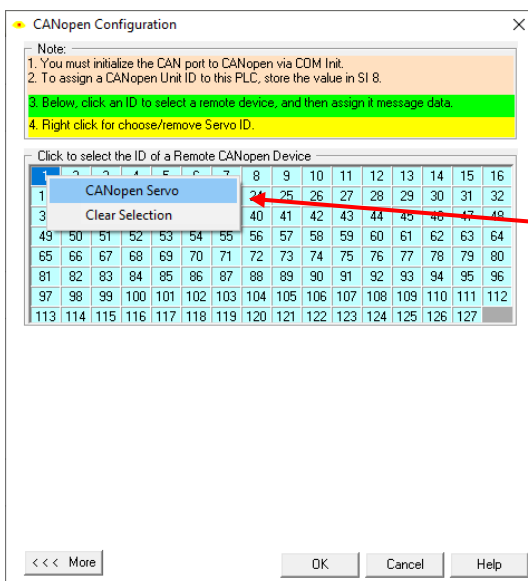
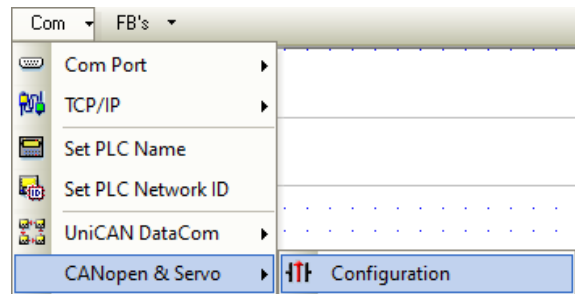


Le bloc se configure comme suit :

Bien choisir le protocole « Canopen & Servo »



Configurer ensuite la communication en ajoutant le bloc « Configuration » se trouvant dans l'onglet « Com » → « CANopen & Servo ».



Faites un clic droit sur le numéro de l'ID que vous souhaitez attribuer au Drive et cliquez sur « CANopen Servo » puis « OK »

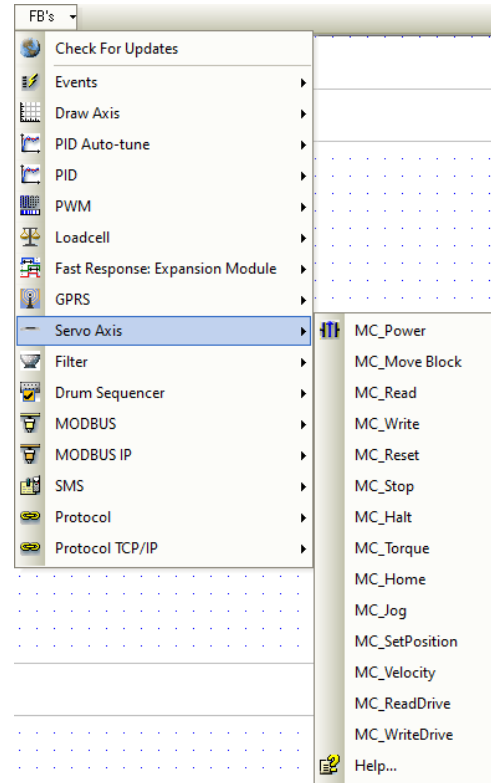
Retrouvez-nous sur www.pl-systems.fr

Programmation

Les blocs fonctions

Après avoir configuré la communication avec le Drive, il est maintenant temps de s'attaquer à la programmation des différents blocs qui permettront d'animer notre Servo.

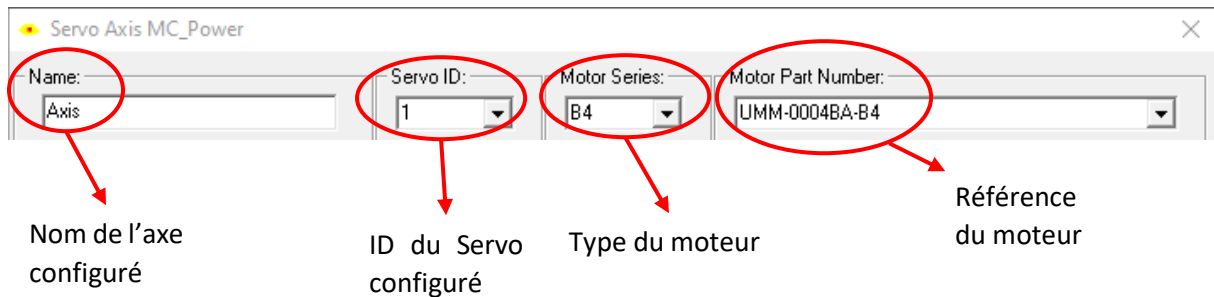
Pour cela, tous les blocs qui concernent la partie Servo se trouve dans l'onglet « FB's » → « Servo Axis ».



Le bloc de mise sous tension et de paramétrage

Ce bloc permet de mettre sous tension le moteur et de le paramétrer :

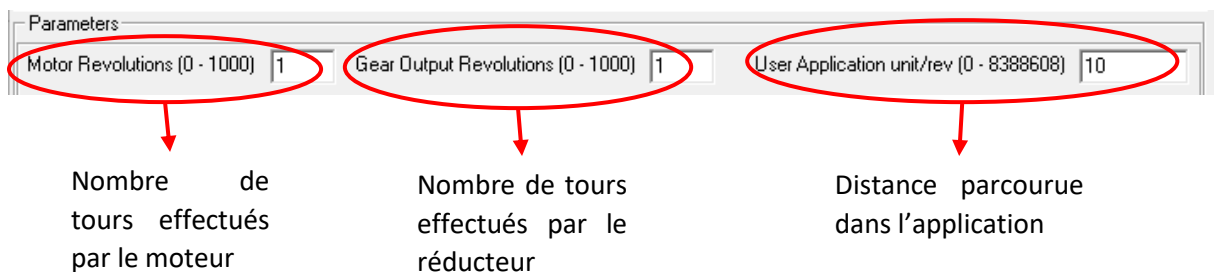
Dans un premier temps, des informations standards sont à renseigner :



Name: Servo ID: Motor Series: Motor Part Number:

Nom de l'axe configuré ID du Servo configuré Type du moteur Référence du moteur

Vous devez ensuite renseigner la partie concernant le ratio :

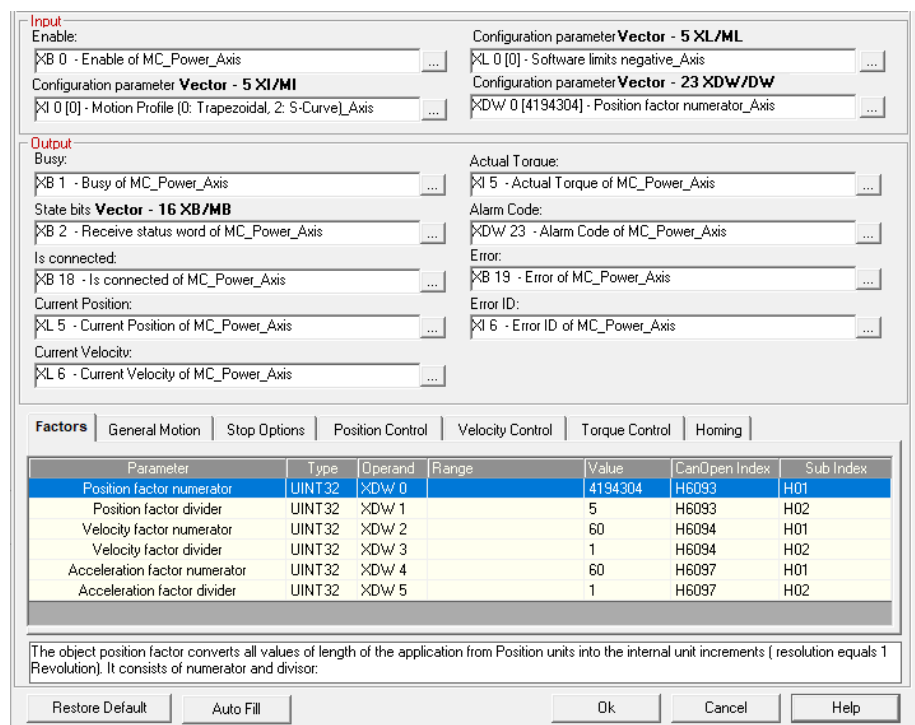


Motor Revolutions (0 - 1000) | 1 Gear Output Revolutions (0 - 1000) | 1 User Application unit/rev (0 - 8388608) | 10

Nombre de tours effectués par le moteur Nombre de tours effectués par le réducteur Distance parcourue dans l'application

Ici dans notre exemple, nous n'avons pas de réducteur, nous avons donc un rapport de 1 (1 tour réducteur fait par tour moteur). La distance que nous parcourons dans cet exemple est de 10. Cette valeur est souvent donnée par la mécanique, dans notre cas, nous parcourons 10mm par tour de moteur.

Tous les paramètres qui suivent sont les variables à allouer pour l'automate pour que celui-ci soit capable de gérer en tâche de fond le pilotage du drive correctement. Nous pouvons les attribuer un adressage libre en cliquant sur le bouton « Auto Fill » qui les allouera à des variables libres.

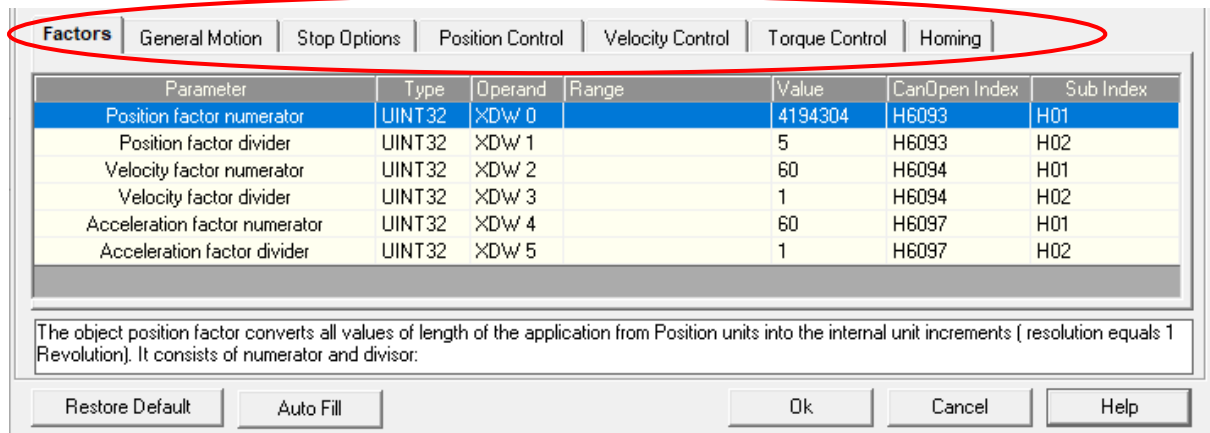


Parameter	Type	Operand	Range	Value	CanOpen Index	Sub Index
Position factor numerator	UINT32	×DW 0		4194304	H6093	H01
Position factor divider	UINT32	×DW 1		5	H6093	H02
Velocity factor numerator	UINT32	×DW 2		60	H6094	H01
Velocity factor divider	UINT32	×DW 3		1	H6094	H02
Acceleration factor numerator	UINT32	×DW 4		60	H6097	H01
Acceleration factor divider	UINT32	×DW 5		1	H6097	H02

The object position factor converts all values of length of the application from Position units into the internal unit increments (resolution equals 1 Revolution). It consists of numerator and divisor.

Retrouvez-nous sur www.pl-systems.fr

Tous les paramètres attrait au fonctionnement du moteur ont pris des valeurs « standard » et vous pouvez aller dans chaque onglet pour les affiner en fonction de vos besoins et de votre mécanique.



Parameter	Type	Operand	Range	Value	CanOpen Index	Sub Index
Position factor numerator	UJINT32	XDW 0		4194304	H6093	H01
Position factor divider	UJINT32	XDW 1		5	H6093	H02
Velocity factor numerator	UJINT32	XDW 2		60	H6094	H01
Velocity factor divider	UJINT32	XDW 3		1	H6094	H02
Acceleration factor numerator	UJINT32	XDW 4		60	H6097	H01
Acceleration factor divider	UJINT32	XDW 5		1	H6097	H02

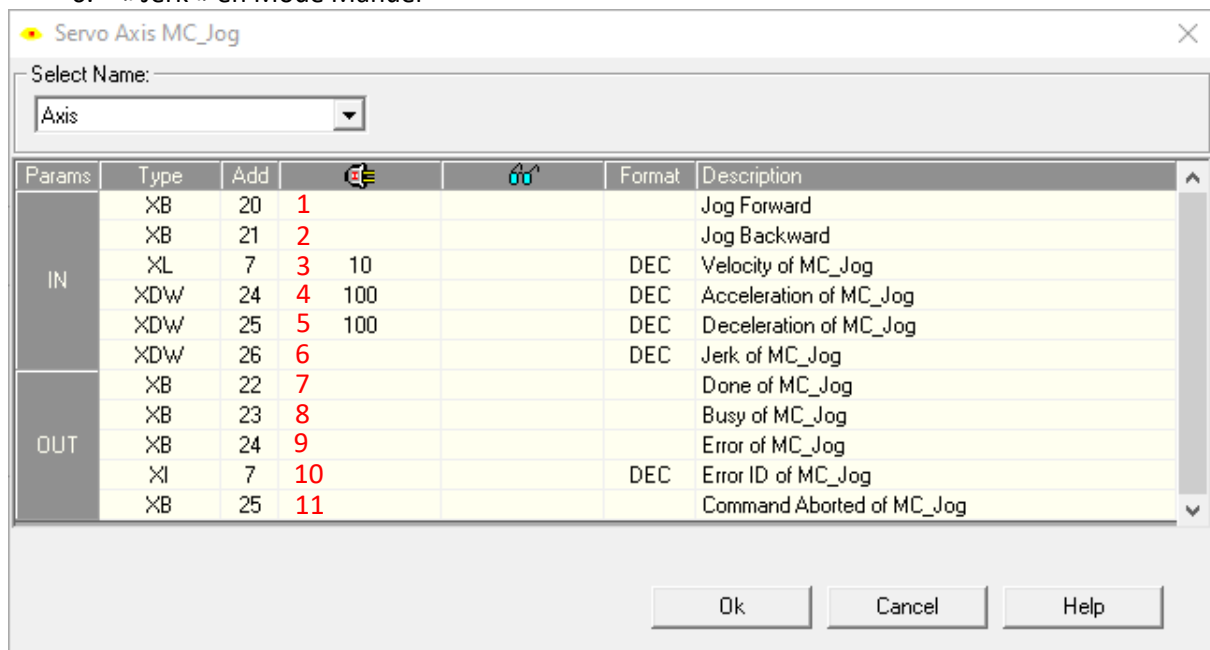
The object position factor converts all values of length of the application from Position units into the internal unit increments (resolution equals 1 Revolution). It consists of numerator and divisor.

Le bloc de mouvement manuel

Le bloc de mouvement manuel permet d'effectuer un mouvement manuel à une vitesse constante.

Ce bloc se nomme MC_Jog et accueille 6 entrées et 5 sorties :

- | | |
|--------------------------------|--------------------------|
| 1. Bouton Avance | 7. Fait |
| 2. Bouton Recule | 8. En fonctionnement |
| 3. Vitesse en Mode Manuel | 9. Erreur |
| 4. Accélération en Mode Manuel | 10. Erreur d'ID |
| 5. Décélération en Mode Manuel | 11. Commande interrompue |
| 6. « Jerk » en Mode Manuel | |



Params	Type	Add	Description
IN	XB	20	1 Jog Forward
	XB	21	2 Jog Backward
	XL	7	3 10 Velocity of MC_Jog
	XDW	24	4 100 DEC Acceleration of MC_Jog
	XDW	25	5 100 DEC Deceleration of MC_Jog
	XDW	26	6 DEC Jerk of MC_Jog
OUT	XB	22	7 Done of MC_Jog
	XB	23	8 Busy of MC_Jog
	XB	24	9 Error of MC_Jog
	XI	7	10 DEC Error ID of MC_Jog
	XB	25	11 Command Aborted of MC_Jog

Retrouvez-nous sur www.pl-systems.fr

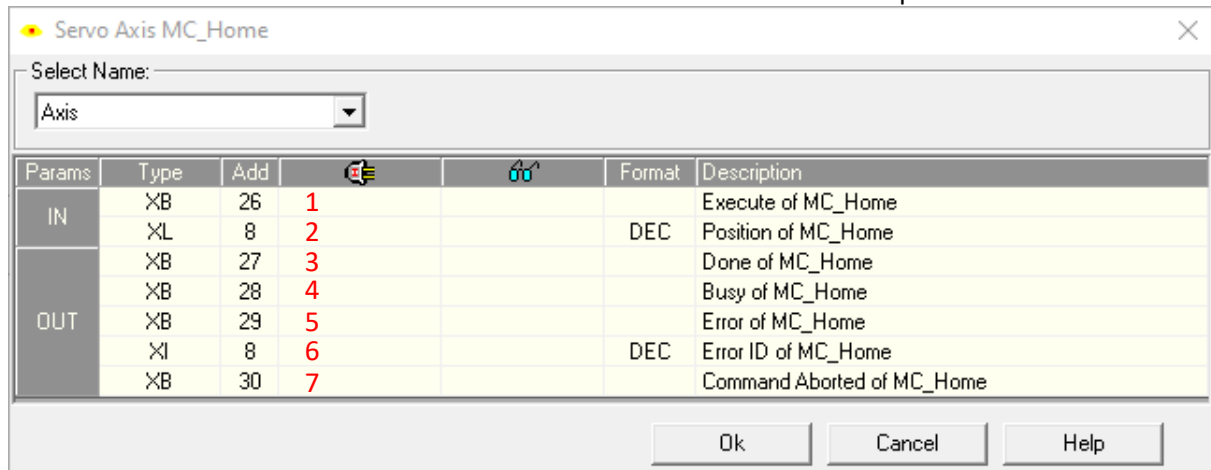


Le bloc de position Home

Ce bloc permet d'effectuer une prise d'origine du moteur qui servira de référence par exemple sur un mouvement absolu.

Ce bloc se nomme MC_Home et accueille 2 entrées et 5 sorties :

- | | |
|---------------------------------|-------------------------|
| 1. Bouton de prise d'origine | 3. Fait |
| 2. Position offset (si utilisé) | 4. En fonctionnement |
| | 5. Erreur |
| | 6. Erreur d'ID |
| | 7. Commande interrompue |



Lors de l'exécution du bloc et si le homing se fait via la méthode 35, alors la position passe à zéro et est enregistrée comme origine.

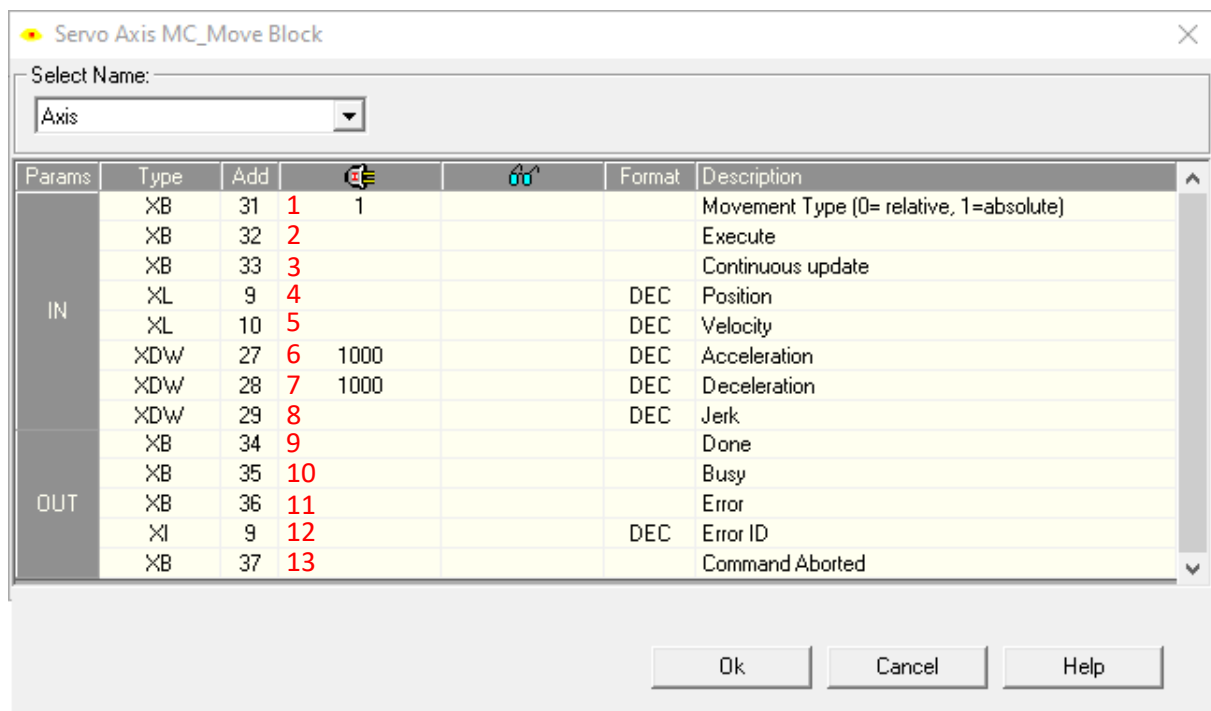


Le bloc de mouvement absolu

Le bloc de mouvement absolu permet d'effectuer un mouvement à une position absolue spécifiée et à vitesse constante.

Ce bloc se nomme MC_Move Block et accueille 8 entrées et 6 sorties :

- | | |
|-------------------------------------|--------------------------|
| 1. Type de mouvement | 9. Fait |
| 2. Bouton de lancement du mouvement | 10. En fonctionnement |
| 3. Continuous Update | 11. Erreur |
| 4. Position à atteindre | 12. Erreur d'ID |
| 5. Vitesse du mouvement | 13. Commande interrompue |
| 6. Accélération du mouvement | |
| 7. Décélération du mouvement | |
| 8. « Jerk » du mouvement | |



Params	Type	Add			Format	Description
IN	XB	31	1	1		Movement Type (0= relative, 1=absolute)
	XB	32	2			Execute
	XB	33	3			Continuous update
	XL	9	4		DEC	Position
	XL	10	5		DEC	Velocity
	XDW	27	6	1000	DEC	Acceleration
	XDW	28	7	1000	DEC	Deceleration
	XDW	29	8		DEC	Jerk
OUT	XB	34	9			Done
	XB	35	10			Busy
	XB	36	11			Error
	XI	9	12		DEC	Error ID
	XB	37	13			Command Aborted