

**Automate programmable
easyControl
EC4-200**

DRAFT

**Matériel, étude et
description des fonctions**

03/07 AWB2724-1584F

MOELLER 

Think future. Switch to green.

Tous les noms de produits sont des marques ou des marques déposées des différents titulaires.

DRAFT

1ère édition 2007, date de rédaction 03/07

© 2007 by Moeller GmbH, 53105 Bonn

Auteur : Peter Roersch
Rédaction : Thomas Kracht
Traduction : Christine Carayon

Tous droits réservés, y compris de traduction.

Toute reproduction de ce manuel sous quelque forme que ce soit (impression, photocopie, microfilm ou autre procédé) ainsi que tout traitement, copie ou diffusion par des systèmes électroniques sont interdits sans autorisation écrite de la société Moeller GmbH.

En raison de l'évolution des matériels, les caractéristiques données dans ce manuel sont susceptibles de modifications.



Avertissement ! Tension électrique dangereuse !

Avant de commencer les travaux d'installation

- Mettre l'appareil hors tension
- Prendre les mesures nécessaires pour interdire tout réenclenchement
- Vérifier l'absence de tension
- Effectuer les mises à la terre et en court-circuit nécessaires
- Protéger par un écran les pièces voisines sous tension
- Respecter impérativement les directives contenues dans les notices de l'appareil (AWA)
- Les interventions sur cet appareil ou ce système ne doivent être exécutées que par du personnel qualifié selon EN 50 110-1/-2.
- Lors des travaux d'installation, veillez à vous décharger de l'électricité statique avant de toucher l'appareil.
- Raccorder la terre fonctionnelle (TE) au conducteur d'équipotentialité ou à la terre de protection (PE). La réalisation de ce raccordement est sous la responsabilité du personnel effectuant les travaux d'installation.
- Les conducteurs de raccordement et de signaux doivent être installés de telle manière que les parasites inductifs et capacitifs ne perturbent pas les fonctions d'automatisation.
- Les appareils d'automatisation et leurs organes de commande doivent être montés de manière à être protégés contre tout actionnement involontaire.
- Pour éviter que la rupture d'un câble ou d'un conducteur véhiculant des signaux n'entraîne des états indéfinis dans l'appareil d'automatisation, il convient de prendre toutes les mesures de sécurité nécessaires sur le plan matériel et logiciel pour le couplage des entrées/sorties.
- Si l'appareil est alimenté en 24 V, veiller à assurer une séparation électrique sûre de la très basse tension. N'utiliser que des blocs d'alimentation conformes à CEI 60 364-4-41 ou HD 384.4.41 S2.
- Les fluctuations ou les écarts de la tension réseau par rapport à la valeur nominale ne doivent pas dépasser les seuils de tolérance indiqués dans les caractéristiques techniques car ils peuvent être à l'origine de défauts de fonctionnement et d'états dangereux.
- Les dispositifs d'arrêt d'urgence selon CEI/EN 60 204-1 doivent rester efficaces dans tous les modes de fonctionnement de l'appareil d'automatisation. Le déverrouillage du dispositif d'arrêt d'urgence ne doit pas provoquer de redémarrage incontrôlé ou indéfini.
- Les appareils à monter dans des coffrets ou des armoires ne doivent pas être exploités ou commandés autrement que sous enveloppe. Le boîtier des appareils de bureau ou portables doit impérativement être fermé.
- Prendre toutes les mesures nécessaires pour assurer la poursuite correcte d'un programme interrompu par une chute ou une coupure de tension et interdire l'apparition d'états dangereux, même fugitifs. Si nécessaire, faire intervenir un arrêt d'urgence.
- Si l'appareil d'automatisation présente un défaut ou une panne susceptibles de causer des dommages corporels ou matériels, il faut prendre des mesures sur l'installation garantissant ou forçant le fonctionnement sûr de l'appareil (p. ex. à l'aide de fins de course limites de sécurité, verrouillages mécaniques ou autres protecteurs).

DRAFT

Sommaire

Préface		7
	Documentation complémentaire	7
	Conventions de lecture	7
1 Utilisation		9
	Synoptique des références EC4-200	9
2 Réalisation	Entrées	11
	– Touches de fonction et de direction utilisées comme des entrées	12
	– Entrées de diagnostic	12
	– Entrées pour compteurs rapides	12
	Sorties	13
	Module mémoire (MMC)	13
	– Données du module mémoire	13
	– Accès aux données du module mémoire	13
	Visualisation d'état à l'aide des DEL RUN/STOP/SF et CAN/NET	13
	Horloge temps réel	14
	Interfaces	14
	– Interface de programmation destinée au raccordement d'un PC	14
	– Interface multifonctions (MFI)	15
	– Liaisons à l'aide de câbles	16
	Interfaces CAN/easyNet	16
3 Appareils d'extension	Entrées	17
	– Entrées de diagnostic	17
	Sorties	17
		17
4 Montage		19
	Montage sur profilé chapeau	19
	Fixation sur platine de montage	19
5 Installation		21
	Raccordement à la tension d'alimentation	21
	Raccordement des entrées tout-ou-rien	21
	Raccordement des entrées analogiques	21
	– Raccordement: Potentiomètre d'entrée de consignes	22
	– Raccordement: Sonde de température	22
	– Raccordement d'un capteur 20 mA	22
	Raccordement d'un générateur d'impulsions/d'un codeur incrémental	23
	– Raccordement d'un générateur d'impulsions	23
	– Raccordement d'un codeur incrémental	23
	Raccordement des sorties	24
	– Raccordement des sorties à relais	24
	– Raccordement des sorties à transistors	25
– Raccordement d'une sortie analogique	26	

	Carte mémoire, CAN/easyNet, liaison PC	27
	– Enfichage/retrait du module mémoire	27
	– CAN/easyNet, liaison PC	27
	Raccordement d'extensions/de modules de couplage réseau	28
	– Extension centralisée	28
	– Extension décentralisée	28
6	Utilisation	29
	Touches de commande	29
	Dialogue par menus et saisie de valeurs	29
	Choix d'options menu ou passage à d'autres options	29
	– Représentation du curseur	29
	– Réglage d'une valeur	29
	Sélection du Menu principal et du Menu spécial	30
	– Affichage d'état	30
	– Affichage d'état avec heure	30
	Structure des menus	31
	– Menu principal sans protection par mot de passe	31
	– Menu principal protégé par mot de passe	32
	– Menu spécial	32
	– Menu spécial	33
7	Description des réglages	35
	Protection par mot de passe	35
	– Paramétrage d'un mot de passe	35
	– Sélection de la plage de validité du mot de passe	35
	– Activation du mot de passe	35
	– Accès possibles en cas de protection par mot de passe	36
	– Modification ou effacement du mot de passe ou d'une plage	36
	Modification du choix de la langue des menus	37
	Réglage de la date et de l'heure	37
	Comportement au démarrage	37
	– Réglage du comportement au démarrage	37
	Réglage du contraste et du rétro-éclairage de l'afficheur LCD	38
8	Configuration des entrées/sorties (I/O)	39
	Représentation des entrées/sorties au niveau de la configuration	39
	Affichage des entrées/sorties locales	39
	Modification de la fonction des dossiers	39
	Affichage des entrées/sorties des appareils d'extension	40
9	Fonctionnement	41
	Généralités	41
	– Synoptique des capacités mémoire	41
	– Liste des différentes mémoires	41
	Comportement à la mise sous tension	41
	– Comportement à la mise sous tension avec un projet de démarrage sur le module mémoire	41
	Réglage du comportement au démarrage dans le logiciel de programmation	43
	Programme START/STOP	43
	– Lancement du programme (STOP → RUN)	43

– Comportement après coupure/interruption de la tension d'alimentation	43
– Arrêt du programme (RUN → STOP)	44
– Démarrage/Arrêt du programme à l'aide d'interrupteurs externes	44
Traitement du programme et temps de cycle du système	44
Surveillance du temps de cycle	44
Reset (remise à zéro)	44
– Reset à chaud (Reset warm)	44
– Reset à froid (Reset cold)	44
– Reset origine (Reset Hard)	44
– Rétablissement des réglages usine (factory set)	44
– Comportement des variables après une RAZ (Reset)	45
Test et mise en service	45
– Mode Point d'arrêt/Pas à pas	45
– Mode Cycle unique	45
– Forçage de variables et d'entrées/sorties	45
– Affichage d'état dans le logiciel de programmation	45
Compteurs rapides (Counter)	45
– Fonctions de comptage (entrées/sorties)	46
Compteur de valeurs incrémentales (Incremental Input)	47
– Signification des signaux d'entrée/sortie (I/Q)	47
– Synoptique des signaux des entrées/sorties (I/Q)	48
– Fonctions des signaux d'entrée/sortie	48
– Référencement	48
Evénements système	49
– START, COLDSTART, WARMSTART, STOP	49
– Entrées d'interruption I1 à I4	50
– Interruption au niveau d'un compteur (Counter-Interrupt)	50
– Interruption au niveau d'un temporisateur (Timer-Interrupt)	50
Traitement des interruptions	52
– Les étapes du traitement d'une interruption	52
– Exemple de traitement d'une interruption	52
Accès direct aux E/S	53
– Description des fonctions	53
Code d'erreur en cas d' "accès direct aux E/S"	54
Génération et transfert d'un projet de démarrage (bootproject)	55
– Sauvegarde du projet de démarrage sur un module mémoire	55
– Projet de démarrage et système d'exploitation (SE) sur module mémoire	55
– Effacement d'un projet de démarrage	55
Téléchargement/Actualisation du système d'exploitation	56
– Transfert du système d'exploitation du PC vers l'automate	56
– Transfert du système d'exploitation du PC vers un module mémoire	57
– Transfert du système d'exploitation du module mémoire vers l'automate	57

10 Commandes navigateur		59
	– Réglage des paramètres Ethernet	59
	Description des principales commandes navigateur	60
	– canload	60
	– setrtc	60
11 Bibliothèques, blocs fonctionnels et fonctions		61
	Utilisation des bibliothèques	61
	Installation de bibliothèques supplémentaires	61
	Fonctions spécifiques au EC4-200	62
	– Bibliothèque EC_Util.lib	62
	– Bibliothèque EC_Visu.lib/EC_Visu2.lib	62
12 Etablissement liaison PC – EC4-200		63
	Etablissement de la liaison via l'interface RS232	63
	Définition/modification des paramètres de communication du PC	63
	Modification des paramètres de communication (vitesse de transmission) de l'UC	64
	Etablissement de la liaison via Ethernet	64
	Scrutation/Modification de l'adresse IP	66
13 Saisie des paramètres système via le fichier STARTUP.INI		67
	Vue d'ensemble	67
	Structure du fichier INI	67
	Création du fichier Startup.INI	67
	Mise sous tension de l'automate dans le cas d'un module mémoire avec fichier Startup.INI enfiché	67
	Modification des paramètres	68
	Effacement du fichier Startup.INI	68
14 Programmation via le réseau CAN (routage)		69
	Conditions préalables	69
	Propriétés du routage de l'automate	69
	Routage via XC200	69
	Remarques sur le routage	70
	Réglage de l'easyNet de nœud (Node easyNet) et de l'easyNet de routage (Routing easyNet)	70
	Réglage du participant Maître	71
	Réglage du participant de type Appareil (Device)	71
	Combinaison d'automates pour le routage	72
15 Interface RS-232 en mode Transparent		73
16 Afficheur interactif		75
	Type de représentation (affichage)	75
	– Commutation entre Affichage d'état et Mode saisie/affichage	75
	– Synoptique des fonctions/blocs fonctionnels	76
	Description des principales fonctions/blocs fonctionnels	77
	– FUNCTION Disp_EnableDisplay: BOOL (*Commutation Affichage d'état <-> Mode Saisie/Affichage*)	77
	– Déroulement général concernant l'élaboration d'un programme	80
	– Exemple d'affichage de textes et de valeurs	81

	– Exemple d'affichage d'une page avec textes et possibilités de saisie	83
	Afficheur multifonctions MFD-CP4 à l'EC4-200	86
	– Réalisation du MFD	86
17 Le réseau easyNet		87
	Vue d'ensemble	87
	Surveillance de easyNet	87
	Emission/réception des données de l'utilisateur	88
	– Fonction NET_UPDATE	88
	– Fonction NET_GET	88
	Transmission des données	89
	– Transmission de données entre un participant doté de easyNet1 et un appareil de type Remote I/O	89
	– Transfert des blocs de données de bits entre plusieurs participants	90
	– Transmission de doubles-mots (32 bits) entre plusieurs participants selon le principe PUT-GET	91
	Configuration d'un EC4-200 avec easy800 sur un réseau easyNet	92
	– Configuration dans easySoft	92
	– Configuration dans easySoft-CoDeSys	92
	Configuration pilotée par EC4-200	93
	Topologie du bus	94
18 Programmation via easyNet (Routage)		95
	Routage via EC4-200	95
19 Modules de couplage réseau pour EC4-200		97
	EASY205-ASI	97
	– Echange de données cyclique	97
	– Configuration	98
	– Paramétrage de l'adresse des participants	98
	EASY221-CO, EASY204-DP, EASY222-DN	98
	– Echange de données cyclique	98
	– Configuration	99
	– Paramétrage de l'adresse des participants	99
	– Echange de données acyclique	99
Annexe		103
	Réseau CAN/easyNet	103
	– Accessoires	103
	Exemple de programme pour le démarrage/l'arrêt (START/STOP) de l'automate à l'aide d'un interrupteur externe	104
	Câble de liaison EASY800-PC-CAB	105
	Encombrements et poids	105
	Caractéristiques techniques	106
	– Sorties à transistors	111
	– Sortie analogique	113
	Jeux de caractères	114
Index des mots clés		117

DRAFT

Préface

Documentation complémentaire

Le présent manuel renvoie en différents endroits à d'autres documents comportant des descriptions complémentaires ou plus détaillées sur certains points. L'enregistrement de ces documents sur votre PC a lieu sous forme de fichiers PDF lors de l'installation du CD relatif au produit.

Pour trouver rapidement le document recherché, sélectionnez les options suivantes dans le menu Démarrer de Windows :

Programmes → Moeller Software → easy Soft CoDeSys → Documentation....

Vous avez par ailleurs la possibilité de télécharger les fichiers PDF via le serveur FTP. Vous y trouverez toujours les données les plus à jour.

https://es-assets.eaton.com/DOCUMENTATION/AWB_MANUALS/.

Pour obtenir des informations concrètes sur la communication avec des participants CAN et leur configuration, reportez-vous aux documents suivants :


- AN27K19GB : Communication via CAN entre deux automates avec des variables réseau (AN2700K19GB.pdf)
- AN27K20GB : Couplage via CANopen de plusieurs automates autonomes (appareils CAN, AN2700K20GB.pdf)
- Etude de participants CAN (AN2700K27GB.pdf)
(Figure dans le menu Démarrer de Windows, sous Programmes → Moeller Software → easy Soft CoDeSys → Exemples d'application...)


- AWB2786-1554 : Description des bibliothèques CANUser.lib, CANUser_Master.lib. Les différentes fonctions des bibliothèques CANUser.lib et CANUser_Master.lib vous permettent d'accéder directement aux divers objets CAN.
(Figure dans le menu Démarrer de Windows, sous Programmes → Moeller Software → easy Soft CoDeSys → Documentation...)


Conventions de lecture

Sélectionnez «File → New» signifie : activez la commande "New" du menu "File".

→ Attire votre attention sur des conseils et des informations complémentaires.

 **Attention !**
Met en garde contre les risques de dommages matériels légers.

 **Avertissement !**
Met en garde contre des risques de dommages matériels importants et de blessures légères.

 **Danger !**
Met en garde contre des risques de dommages matériels importants et de lésions corporelles graves susceptibles d'entraîner la mort.

Pour une meilleure vue d'ensemble, les pages de gauche comportent en en-tête le titre du chapitre considéré et les pages de droite le titre du paragraphe traité au sein de ce chapitre. Font exception les pages en tête de chapitre et les pages vierges en fin de chapitre.

DRAFT

1 Utilisation

Les automates de la gamme EC4-200 sont des appareils de connexion et de commande programmables. Ils sont conçus pour résoudre des tâches dans les domaines de la domotique et de la construction d'appareils et de machines. Tout automate EC4-200 peut être utilisé de manière autonome et relié à des appareils d'entrée/sortie décentralisés, via l'interface CANopen. Cette interface vous permet également de communiquer avec d'autres automates (équipés d'une interface CANopen).

Les automates référencés EC4P-222-... possèdent en outre une interface Ethernet.

A partir de la version 2.0 du système d'exploitation, les automates présentent les propriétés suivantes :

- Raccordement d'appareils d'extension/d'automates via easyLink
- Raccordement de l'afficheur multifonctions MFD-CP4 via l'interface multifonctions
- Mode Transparent via l'interface multifonctions
- Accès direct aux entrées/sorties locales et aux "compteurs rapides"
- Intégration dans le réseau easyNet, via l'interface easyNet/CAN

Les automates à partir de la version 2.10 peuvent également être raccordés aux réseaux ASI, PROFIBUS-DP, CAN ou DeviceNet, via des modules de couplage réseau.

La programmation de l'automate s'opère à l'aide du logiciel easySoft CoDeSys. Ce logiciel peut être installé sur un PC doté d'un système d'exploitation Windows NT, 2000 ou XP. Pour toute information complémentaire sur ce logiciel, reportez-vous au manuel du XSoft (AWB2700-1437GB).

Ce logiciel vous permet de vous familiariser aisément avec les différents langages de programmation CEI tels que :

- Liste d'instructions (IL)
- Langage en blocs fonctionnels (FBD)
- Langage à contacts (LD)
- Langage littéral structuré (ST)
- Langage séquentiel (SFC).

Vous disposez d'une multitude d'opérateurs. Exemples .:

- Opérateurs booléens (AND, OR, ...)
- Opérateurs arithmétiques (ADD, MUL, ...)
- Opérateurs de comparaison (<, =, >)

Le logiciel de programmation vous permet d'élaborer un projet, de le tester et d'en établir la documentation. Les blocs fonctionnels (temporisateurs, compteurs, ...) et les fonctions destinées au traitement des valeurs analogiques et à l'élaboration de régulateurs sont conçus pour faciliter votre programmation.

Synoptique des références EC4-200

La gamme EC4-200 comporte des automates qui se différencient par leur afficheur ainsi que par le type et le nombre d'entrées/sorties.

Référence	Propriétés				
	Touches/Afficheur	Sorties à transistors	Sorties à relais	Sortie analogique	Port Ethernet
EC4P-221-MTXD1	×	8	–	–	–
EC4P-221-MTXX1	–	8	–	–	–
EC4P-221-MRXd1	×	–	6	–	–
EC4P-221-MRXX1	–	–	6	–	–
EC4P-221-MTAD1	×	8	–	×	–
EC4P-221-MTAX1	–	8	–	×	–
EC4P-221-MRAD	×	–	6	×	–
EC4P-221-MRAX1	–	–	6	×	–
EC4P-222-MTXD1	×	8	–	–	×
EC4P-222-MTXX1	–	8	–	–	×
EC4P-222-MRXd1	×	–	6	–	×
EC4P-222-MRXX1	–	–	6	–	×
EC4P-222-MTAD1	×	8	–	×	×
EC4P-222-MTAX1	–	8	–	×	×
EC4P-222-MRAD	×	–	6	×	×
EC4P-222-MRAX1	–	–	6	×	×

DRAFT

2 Réalisation

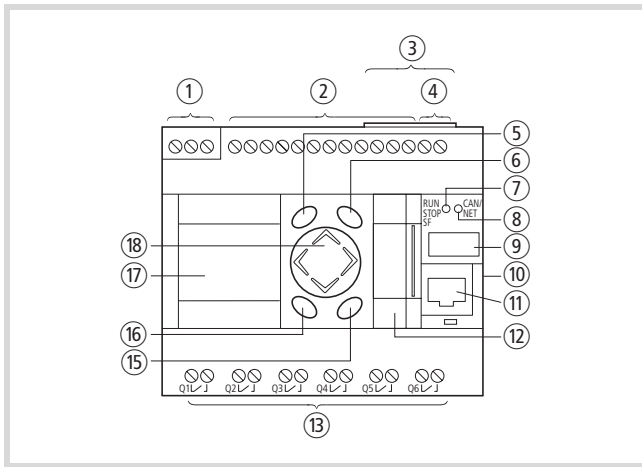


Figure 1 : Vue de face de l'EC4P-221-MRAD1, Légende → figure 2

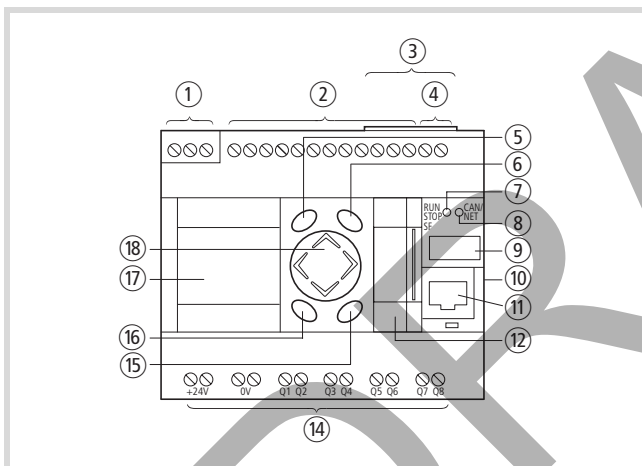


Figure 2 : Vue de face de l'EC4P-221-MTAD1

- ① Tension d'alimentation 24 V DC
- ② Entrées
- ③ Interface pour raccordement au réseau CAN
- ④ Sortie analogique, 0 – 10 V (non active)
- ⑤ Touche DEL
- ⑥ Touche ALT
- ⑦ DEL RUN/STOP/SF
- ⑧ DEL CAN/NET
- ⑨ Zone réservée au repérage de l'appareil
- ⑩ Interface easyLink destinée à une extension
- ⑪ Interface de programmation destinée au raccordement d'un PC
- ⑫ Interface multifonctions
- ⑬ Sorties à relais
- ⑭ Sorties à transistors
- ⑮ Touche OK
- ⑯ Touche ESC
- ⑰ Afficheur à cristaux liquides (EC4P-22x-M...D1)
- ⑱ Touches de direction P1...P4

Entrées

Tableau 1 : Type et nombre d'entrées

Tout-ou-rien	12 (I1...I12)	24 V DC
dont utilisables de manière analogique	4 (I7, I8, I11, I12)	24 V DC/0...10 V

Les entrées I7, I8, I11, I12 sont également utilisables comme des entrées analogiques. Sélectionnez-les dans le programme utilisateur, en utilisant la syntaxe adaptée (qui est indiquée dans le configurateur de l'automate).

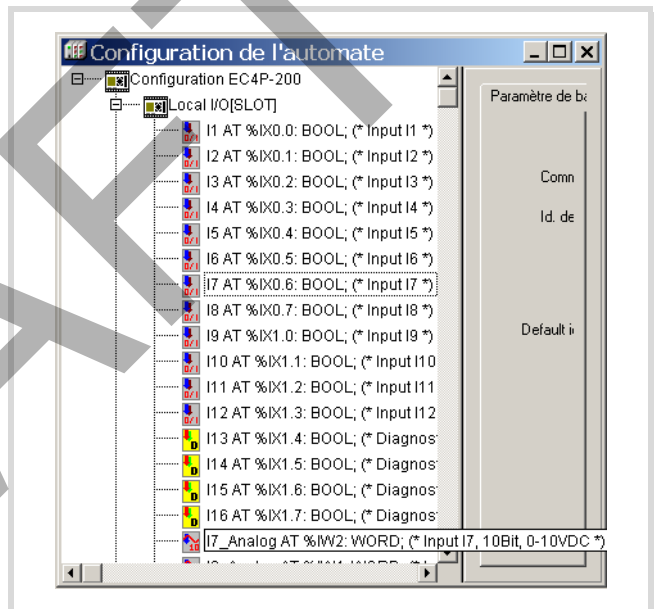


Figure 3 : Sélection entre une entrée TOR et une entrée analogique (dans cet exemple : I7)

Si vous programmez les entrées en tant qu'entrées TOR dans le programme utilisateur, la tension d'entrée de 8 V correspondra à la valeur-limite pour les signaux TRUE/FALSE.

Tension [V]	Etat
≤ 8	FALSE
> 8	TRUE

Caractéristiques techniques: → page 106

Vous pouvez utiliser les entrées I1, I2, I3, I4 :

- pour générer des interruptions (entrées I1, I2, I3, I4)
- pour activer des compteurs rapides tels que :
 - des compteurs 16 ou 32 bits, pour dénombrer des impulsions (I1, I2), compteurs/décompteurs
 - des compteurs de valeurs incrémentales, 32 bits, pour le traitement des signaux d'un codeur incrémental (I1, I2, I3, I4).

Le choix de la fonction s'opère dans le configurateur de l'automate. L'utilisation simultanée de plusieurs fonctions est cependant impossible.

Exemple : Si vous utilisez l'entrée I1 pour un compteur rapide (16 bits), l'entrée I2 est utilisable pour un autre compteur rapide (16 bits), mais pas pour la génération d'une interruption. De même, les entrées I3 et I4 ne peuvent pas être utilisées pour générer une interruption.

Description relative aux raccords → figure 22, page 23.

Touches de fonction et de direction utilisées comme des entrées

Le plastron de l'appareil comporte 4 touches de fonction (DEL, ALT, ESC, OK) disposées autour de la grosse touche centrale qui regroupe les 4 touches de direction (également désignées par les symboles P1 à P4). Dans la Configuration de l'automate, les touches de fonction et de direction sont représentées sous forme d'entrées. La scrutation de ces entrées dans le programme s'opère selon les règles de syntaxe habituelles. Il convient toujours d'actionner une seule touche de fonction et de direction, sous peine de provoquer des états non définis lors de la scrutation de ces touches dans le programme.

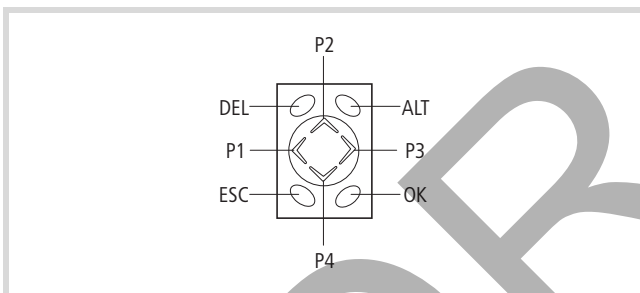


Figure 4 : Touches de fonction et touches de direction P1, P2, P3, P4

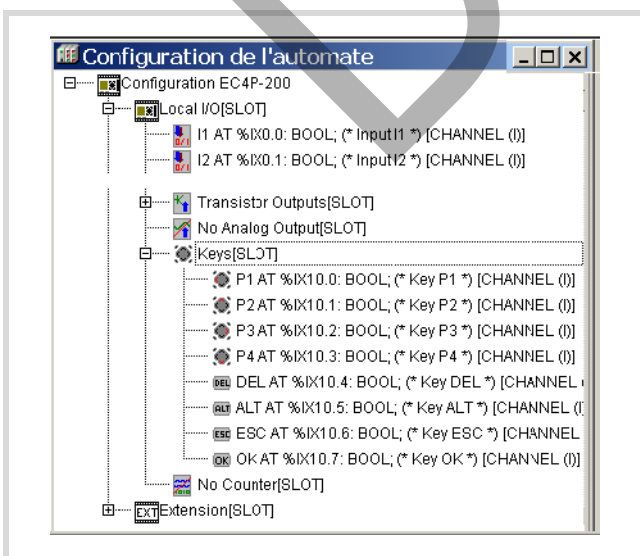


Figure 5 : Entrées des touches de direction et des touches de fonction

Le module fonctionnel "Disp_GetDisplayInfo" de la bibliothèque "EC_Visu2.lib" vous permet de procéder à la scrutation des touches en fonction du menu de l'automate, → paragraphe « Bibliothèque EC_Visu.lib/EC_Visu2.lib », page 62.

Entrées de diagnostic

Les entrées I13, I14, I15, I16 vous permettent de disposer d'informations complémentaires :

Entrée	Fonction
I13	Sans signification
I14	Liaison vers l'appareil d'extension, via easyLink (encore inactive dans le système d'exploitation relevant de la version 1.x) : 0 : ok ; 1 : non ok
I15	Sorties Q1, Q2, Q3, Q4 : 0 : pas de court-circuit ; 1 : court-circuit
I16	Sorties Q5, Q6, Q7, Q8 : 0 : pas de court-circuit ; alternance 1-0-1-0... : court-circuit

Vous pouvez scruter les entrées au sein du programme à l'aide des opérandes symboliques.

Entrées pour compteurs rapides

Vous avez le choix entre plusieurs fonctions :

- 1 × compteur 32 bits, pour compter des impulsions (compteur/décompteur)
- 2 x compteur 16 bits, pour compter des impulsions (compteurs/décompteurs) ; le sens de comptage (comptage/décomptage) est réglable au sein du programme, à l'aide de l'opérande DIRECTION.
- 1 × compteur de valeurs incrémentales, 32 bits, pour le traitement des signaux d'un codeur incrémental ; le sens de comptage est déterminé par la succession des fronts du codeur.

Le choix du type de compteur s'opère au niveau de la configuration de l'automate.

La fonction "compteur rapide" requiert l'activation des entrées et la scrutation des sorties au sein d'une UOP (PLC_PRG, par exemple). Cette UOP ne doit en aucun cas être appelée par une interruption générée par un compteur.

Pour plus d'informations, reportez-vous au paragraphe « Compteurs rapides (Counter) », page 45.

Sorties

Tableau 2 : Type et nombre de sorties

Sorties à transistors EC4P-221/222-MT...	8 (Q1...Q8)	24 V DC/0.5 A
Sorties à relais EC4P-221/222-MR...	6 (Q1...Q6)	250 V AC/8 A

Les sorties à transistors possèdent une surveillance de court-circuit. Toute apparition de court-circuit au niveau de l'une des sorties est signalée par le biais de l'entrée de diagnostic I15/I16. I15 est positionnée à 1 lorsqu'un court-circuit survient au niveau des sorties Q1 à Q4. La survenue d'un court-circuit aux sorties Q5 à Q8 provoque une alternance entre 0 et 1 au niveau de l'entrée I16.



Avertissement !

Procédez à la scrutation de I15/I16 au sein du programme. En cas de court-circuit, positionnez les sorties à 0 pour éviter une surcharge thermique du circuit de sortie.

Module mémoire (MMC)

Le module mémoire sert de mémoire de masse et supporte le système de fichiers FAT16.

Données du module mémoire

Le module mémoire vous permet de stocker les données suivantes :

Données	Possibilité de transmission
Projet de démarrage (bootproject)	Commande navigateur : copyprojtommc
Fichier STARTUP.INI	Commande navigateur : createstartupini
Système d'exploitation (SE)	Actualisation du système d'exploitation, → page 56
Code source du projet	Fonctionnement en ligne/Menu en ligne : charger le code source
Fichiers généraux	Fonctionnement en ligne/Menu en ligne : Ecrire le fichier dans l'automate Charger le projet à partir de l'automate

Vous trouverez une description succincte des commandes navigateur à partir de la page 59.



Attention !

Pour éviter toute perte de données, assurez-vous d'avoir fermé tous les fichiers du programme avant d'enficher ou de retirer le module mémoire ou de procéder à la mise hors tension.

Accès aux données du module mémoire

Les fonctions telles que "FileOpen" ou "FileRead" vous permettent d'accéder aux fichiers du module mémoire à partir du programme utilisateur. Ces fonctions figurent dans la bibliothèque "EC_File.lib" et sont décrites dans le manuel relatif aux blocs fonctionnels ("Function Blocks", AWB2786-1456GB).

Visualisation d'état à l'aide des DEL RUN/STOP/SF et CAN/NET

Après mise sous tension, l'unité centrale (UC) peut présenter les états suivants, signalés par les différentes LED :

Tableau 3 : Affichage d'état DEL

LED		Signification/Etat de l'UC
RUN/STP/SF	CAN/NET	
rouge	rouge ¹⁾	Test du système en cours (jusqu'à 6 secondes après le démarrage ; après 6 secondes, s'il n'existe aucun programme utilisateur) ; UC à l'état NOT READY !
orange	orange ¹⁾	Actualisation du système en cours
rouge	Eteinte ¹⁾	Test du système achevé : aucune erreur
Rouge, clignotant	rouge, clignotant ¹⁾	Test du système achevé : une erreur
orange	Eteinte	Il n'existe aucun programme utilisateur. UC à l'état NOT READY
vert, clignotant	–	Programme utilisateur chargé UC à l'état STOP
verte	–	Programme utilisateur chargé UC à l'état RUN
rouge	–	Temps de cycle dépassé; UC à l'état STOP
orange clignotant	–	Boucle sans fin détectée au sein du programme UC à l'état STOP
rouge, clignotant	rouge, clignotant	Une erreur fatale est survenue.

1) LED pertinente uniquement lors de la mise sous tension/du test du système

Si l'UC se trouve à l'état RUN, la LED CAN/NET indique les états suivants :

Tableau 4 : Visualisation d'état par LED pour CAN/easyNet

LED		Signification
RUN/STP/SF	CAN/NET	
Verte	éteinte	Communication non activée
verte	rouge	Etat du bus : STOP
verte	orange	Etat du bus : PREOPERATIONAL Possibilité d'initialiser le participant ; aucune transmission de données relatives au processus
verte	verte	Etat du bus : OPERATIONAL Les données du processus sont transmises.

Horloge temps réel

L'automate possède une horloge temps réel que vous pouvez adresser dans le programme utilisateur à l'aide des fonctions de la bibliothèque "SysLibRTC". Ces fonctions sont décrites dans le fichier PDF "SysLibRTC". Après installation, vous trouverez ce fichier dans le menu Démarrer de Windows, sous <Programmes → Moeller Software → easySoft CoDeSys → Documentation → Automation Manuals>.

Vous pouvez également lire l'heure ou activer l'horloge à l'aide des commandes navigateur "getrtc" et "setrtc". Pour toute information complémentaire, reportez-vous au paragraphe « setrtc », page 60.

En cas de disparition de la tension, l'horloge est secourue par pile durant au moins 72 heures.

Interfaces

Interface de programmation destinée au raccordement d'un PC

Composée d'un connecteur RJ45, l'interface de programmation permet d'assurer la communication entre l'automate et l'appareil de programmation.

Pour la programmation, le connecteur comporte une interface RS232 et en outre, sur les automates de type EC4P- 222-..., une interface Ethernet.

Tableau 5 : Affectation des signaux de l'interface de programmation

RJ45	EC4P-221...		EC4P-222-...	
	Signal		Signal	
	RS232		RS232	Ethernet
1	–	–	Tx+	
2	–	–	Tx-	
3	–	–	Rx+	
4	GND	GND	1)	
5	TxD	TxD	–	
6	–	–	Rx-	
7	GND	GND	1)	
8	RxD	RxD	–	

1) Pour une liaison Ethernet, le signal GND n'est pas nécessaire. Utilisez par suite un câble dont les broches 4 et 7 ne sont pas affectées !

Mode transparent

Pour réaliser une liaison point à point (sans lignes de Handshake) vers un autre appareil, procédez à la commutation en Mode transparent de l'interface RS232, à l'aide des fonctions de la bibliothèque EC_SysLibCom.lib. En Mode transparent, l'interface est adressée en tant que COM1.

→ chapitre « Interface RS-232 en mode Transparent », page 73.

Séparation des interfaces RS232/Ethernet

Le séparateur de câbles XT-RJ45-ETH-RS232 autorise la communication simultanée via l'interface RS232 et l'interface Ethernet. Le câble EASY-NT-30/80/130 vous permet d'établir la liaison entre l'automate et le séparateur de câbles. L'affectation des broches du connecteur femelle RS232 et Ethernet du séparateur de câbles correspond à l'affectation des broches de l'interface de programmation selon le tableau 5.

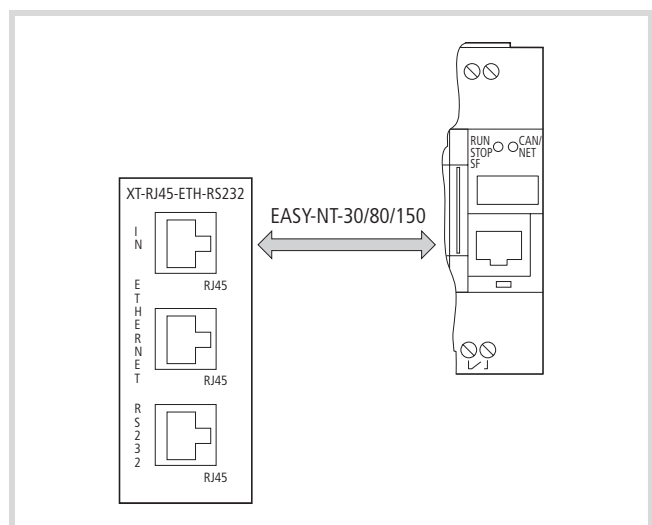


Figure 6 : Liaison entre automate et XT-RJ45-ETH-RS232

Voir également :

→ paragraphe « CAN/easyNet, liaison PC », page 27

→ chapitre 12 « Etablissement liaison PC – EC4-200 », page 63

Interface multifonctions (MFI)

Cette interface permet la communication de l'automate avec les appareils suivants :

- Module mémoire

Enfichez le module mémoire dans un adaptateur, puis enfichez ce dernier à cet emplacement.

- Afficheur multifonctions (MFD) MFD-CP4

Le MFD est un afficheur décentralisé par rapport à l'automate et offre des possibilités de commande opérateur. Il autorise la visualisation du contenu de l'afficheur de l'automate. Les touches intégrées vous permettent d'envoyer des signaux vers l'automate et de piloter le déroulement du programme. Le MFD peut être encastré dans une porte d'armoire, à 5 m max. de l'automate. Les appareils sont reliés à l'aide du câble MFD-CP4-800-CAB5.

- Terminal/Imprimante

L'utilisation d'un terminal vous permet d'afficher ou de saisir des caractères alphanumériques. L'émission de données peut également s'opérer par le biais d'une imprimante. Le terminal équipé d'une interface RS232 est relié à l'automate à l'aide du câble EASY800-PC-CAB. Le câble, doté d'éléments conçus pour l'adaptation aux signaux de l'automate, doit être alimenté séparément côté terminal. Les signaux et l'affectation des broches de l'interface sont réalisés selon les spécifications RS232. Pour l'alimentation des éléments situés dans le câble, le signal RTS doit être activé au sein de l'appareil (terminal) : → paragraphe « Câble de liaison EASY800-PC-CAB », page 105.

Afin de pouvoir émettre des données vers le terminal ou en recevoir, l'interface RS232 (qui est adressée par le biais de COM2) doit être positionnée en Mode Transparent.

→ chapitre « Interface RS-232 en mode Transparent », page 73.

Les fonctions destinées à l'ouverture et à la fermeture de l'interface ainsi qu'à l'émission et à la réception de données sont décrites dans la bibliothèque EC_SysLibCom.lib.

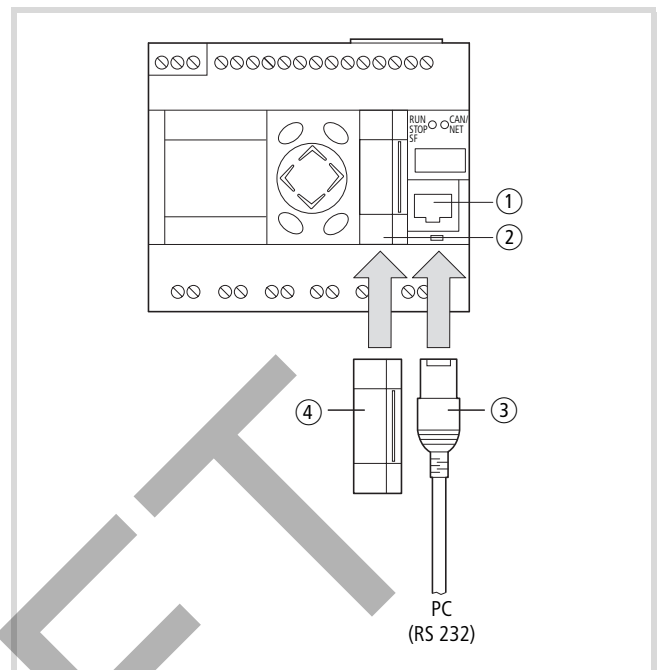


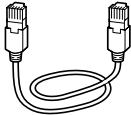

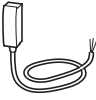


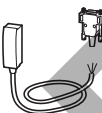


Figure 7 : Interfaces

- ① Interface de programmation destinée au raccordement d'un PC
- ② Interface multifonctions
- ③ Câbles de programmation, p. ex. : EU4A-RJ45-CAB1
- ④ Adaptateur avec module mémoire ou connexion pour câble

Liaisons à l'aide de câbles

Le synoptique suivant montre les types de câble qui peuvent être raccordés à l'automate et leur fonction.

Interface	Type de câble	Appareil	Fonction
RJ45 			
RS232	EU4A-RJ45-CAB1 	PC, terminal/imprimante	Programme, mode transparent (COM1)
Ethernet	XT-CAT5-X-2 	PC	Programme
MFI 	MFD-CP4-800-CAB5 	MFD-CP4	Rallonge pour afficheur
	easy800-USB-CAB 	PC	Programme
	easy800-PC-CAB 	Terminal/imprimante	Mode transparent (COM2)
	easy800-MO-CAB 	PC, terminal/imprimante	Programme, mode transparent (COM1)

Interfaces CAN/easyNet

L'automate possède une interface CAN/easyNet avec deux emplacements dont les connexions sont reliées de manière interne.

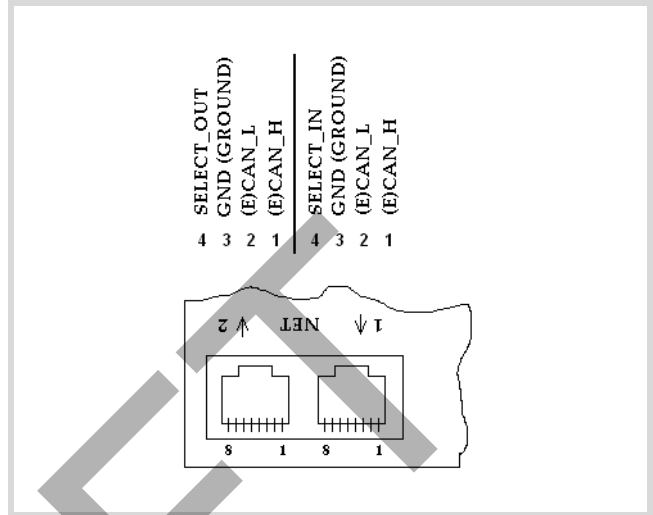


Figure 8 : Interfaces CAN/easyNet

CANopen

L'interface CAN est conçue en tant qu'interface CANopen, selon les spécifications DS301V4.0 du CiA. L'automate est exploitable aussi bien comme maître NMT (NMT-Master) que comme appareil CAN (CAN-Device) au niveau des réseaux CAN. En cas d'utilisation comme appareil CAN, l'automate requiert une adresse (= Node-Id) destinée à son identification sur le bus. Les adresses (Node-Id) possibles sont 1...127. C'est dans le configurateur de l'automate que vous pouvez procéder à la configuration du maître et de l'appareil.

→ paragraphe « Réseau CAN/easyNet », page 103.

3 Appareils d'extension

Raccordez les appareils d'extension directement à l'automate, via l'interface easyLink. Les appareils d'extension suivants vous permettent d'augmenter le nombre d'entrées et de sorties de l'automate.

Synoptique des appareils d'extension

RÉFÉRENCE	Tension d'alimentation	Entrées	Sorties
EASY618-AC-RE	100...230 V AC	12 AC	6 relais
EASY618-DC-RE	24 V DC	12 DC	6 relais
EASY620-DC-TE	24 V DC	12 DC	8 transistors
EASY202-RE	–	–	2 relais avec alimentation commune pour plusieurs sorties

L'appareil de couplage EASY200-EASY permet de relier un appareil d'extension de manière décentralisée avec l'automate, à l'aide d'un câble à 2 ou plusieurs brins de 30 m de longueur.

Synoptique des entrées/sorties

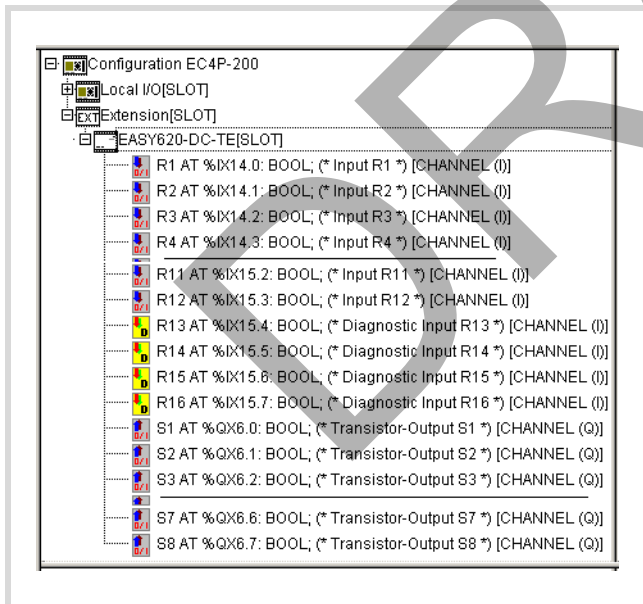


Figure 9 : E/S des appareils EASY620-DC-TE

Entrées

Tableau 6 : Nombre d'entrées et opérandes symboliques

RÉFÉRENCE	Nombre	Opérande
EASY6...-...-...	12	R1...R12
EASY620-DC-TE	4 (Diagnostic)	R13...R16

Faites votre choix dans le programme utilisateur, en utilisant la syntaxe adaptée indiquée dans le configurateur de l'automate.

Entrées de diagnostic

Les entrées R15, R16 vous permettent de disposer d'informations complémentaires :

Tableau 7 : Fonctions des entrées de diagnostic

Entrée	Fonction
R13,R14	Sans signification
R15	Sorties S1, S2, S3, S4: 0 : pas de court-circuit ; alternance 1-0-1-0... : court-circuit
R16	Sorties S5, S6, S7, S8: 0 : pas de court-circuit ; alternance 1-0-1-0... : court-circuit

Vous pouvez scruter les entrées au sein du programme à l'aide des opérandes symboliques.

Sorties

Tableau 8 : Nombre de sorties et opérandes symboliques

RÉFÉRENCE	Nombre	Opérande
EASY618	6	S1 ... S6
EASY620	8	S1...S8
EASY202-RE	2	S1, S2

Les sorties à transistors possèdent une surveillance contre les courts-circuits. L'apparition d'un court-circuit au niveau de l'une des sorties est signalée par le biais des entrées de diagnostic R15/R16. R15 est positionnée à 1 lorsqu'un court-circuit survient aux sorties S1 à S4. La survenue d'un court-circuit aux entrées S5 à S8 provoque une alternance entre 0 et 1 au niveau de l'entrée R16.

DRAFT

4 Montage

Intégrez l'automate dans une armoire, dans un tableau modulaire de distribution terminale ou dans un coffret de manière que les bornes de raccordement à la tension d'alimentation et les différentes connexions soient protégées contre les contacts directs lors du fonctionnement.

L'automate accepte aussi bien un encliquetage en position verticale ou horizontale sur un profilé chapeau selon IEC/EN 60715 qu'une fixation par vis sur une platine de montage, à l'aide de pattes de montage.

Pour faciliter le câblage, respectez côté bornes une distance d'au moins 3 cm par rapport au mur ou aux appareils voisins.

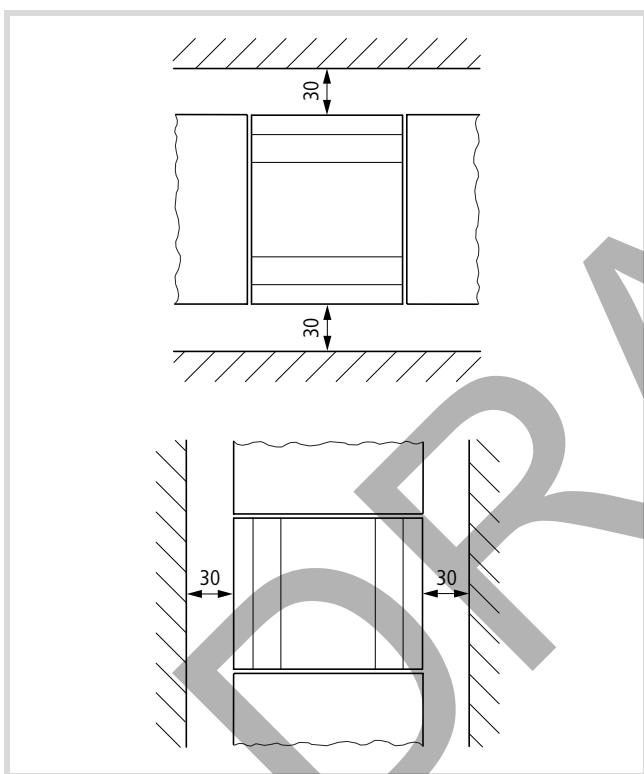


Figure 10 : Distances à respecter pour le câblage

Montage sur profilé chapeau

- ▶ Engagez l'appareil en l'inclinant sur l'arête supérieure du profilé chapeau. Pressez ensuite légèrement l'appareil vers le bas, contre le profilé chapeau, jusqu'à ce qu'il vienne s'encliquer sur l'arête inférieure de ce dernier. Un mécanisme à ressort assure l'encliquetage automatique.
- ▶ Vérifiez la bonne fixation de l'appareil.

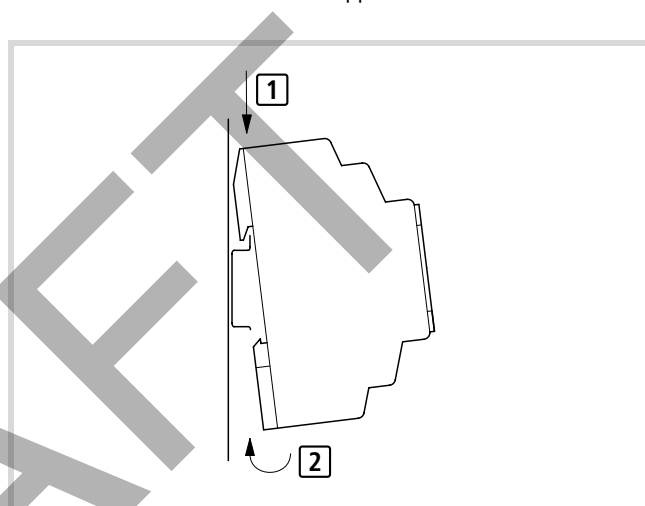


Figure 11 :

Le montage vertical sur profilé chapeau s'effectue de la même manière.

Fixation sur platine de montage

Pour la fixation par vis, il est nécessaire de recourir à des pattes de montage que vous pouvez placer au dos de l'appareil. Les pattes de montage sont des accessoires à commander séparément.

- Pour un appareil avec quatre points de fixation, trois pattes de montage suffisent.

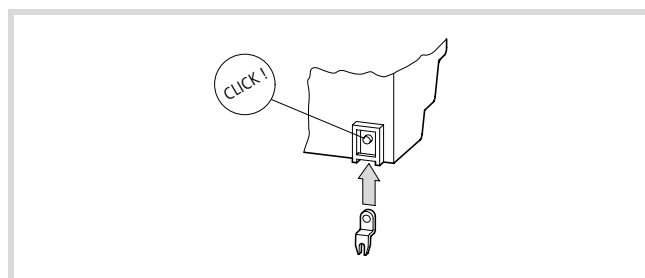


Figure 12 : Utilisation de pattes de montage

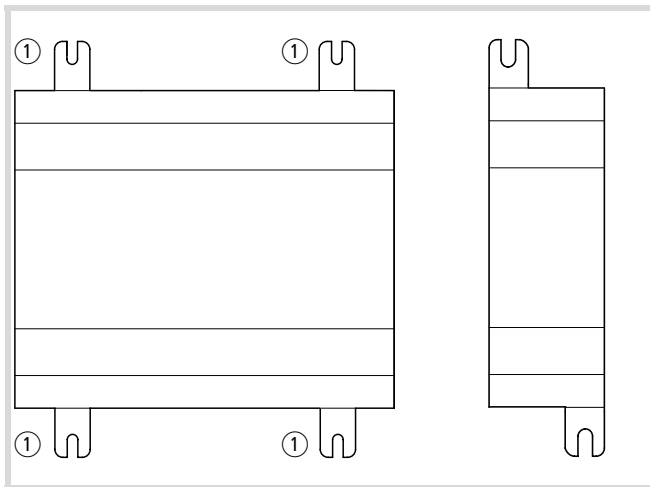


Figure 13 : Vissage des appareils

① Pattes de montage

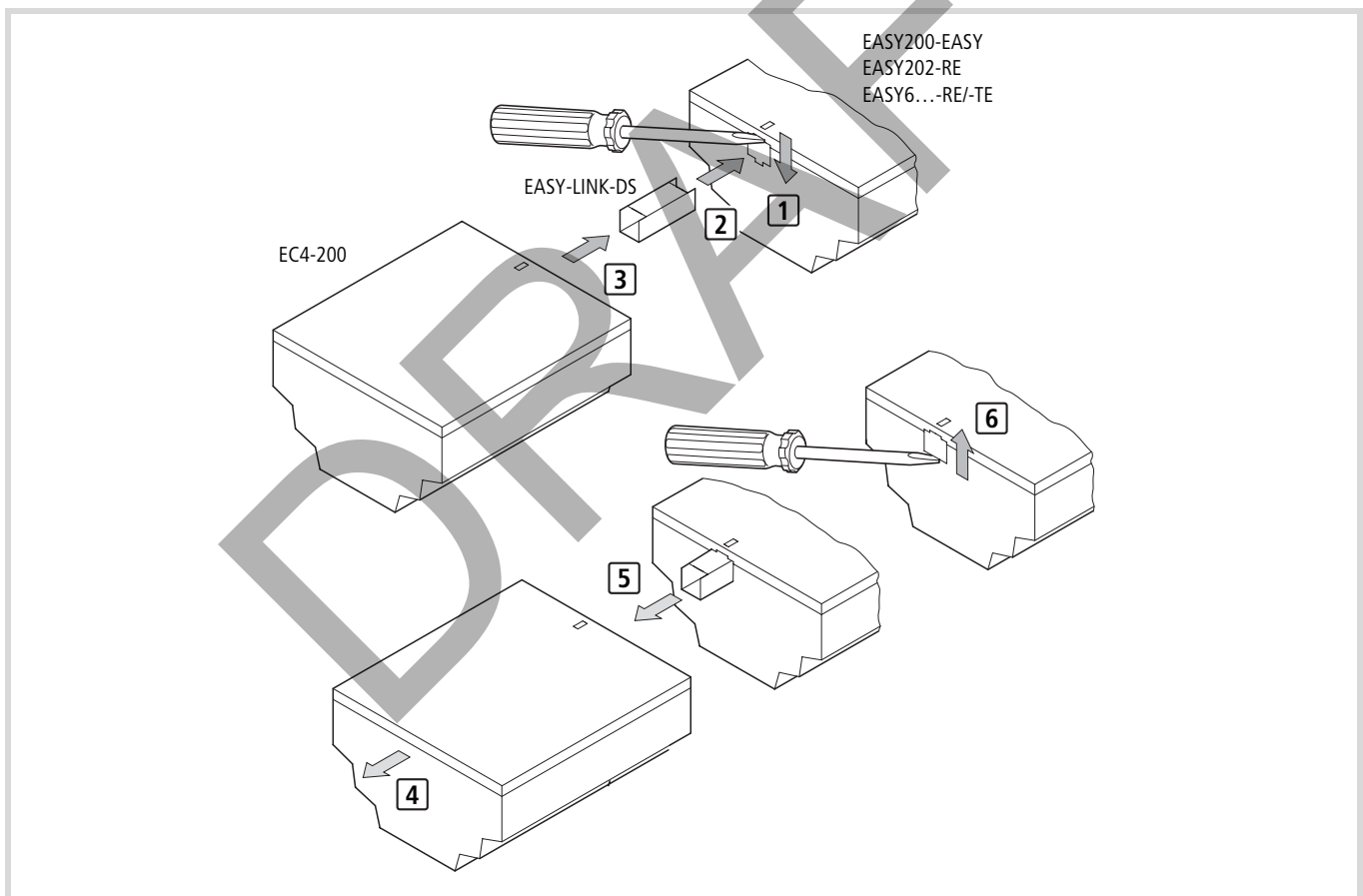


Figure 14 : Raccordement de l'extension/du module de couplage réseau à EC4-200

5 Installation

Raccordement à la tension d'alimentation

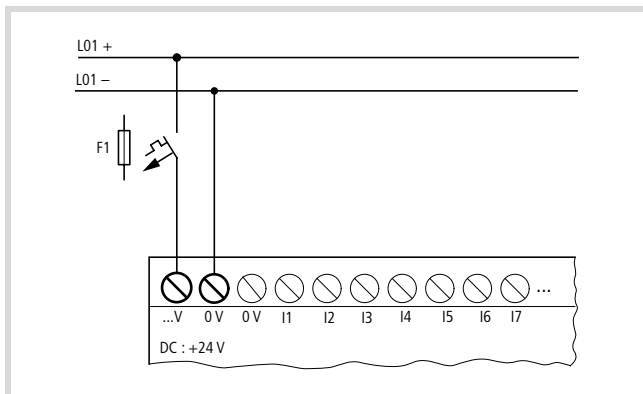


Figure 15 : Raccordement à la tension d'alimentation
Les deux bornes 0 V sont reliées l'une à l'autre de manière interne!

→ L'EC4-200 est protégé contre l'inversion de polarité.

→ Les données relatives aux raccordements figurent dans le chapitre « Caractéristiques techniques », page 106.

Protection des lignes

Protégez les câbles d'alimentation à l'aide d'un dispositif de protection de ligne (F1) d'au moins 1 A (lent).

→ A la première mise sous tension, l'automate se comporte de manière capacitive. L'appareil de connexion et l'appareil d'alimentation destinés à la mise sous tension doivent être conçus pour ce cas de figure : en d'autres termes, ils ne doivent pas comporter de contacts à relais Reed, ni de détecteurs de proximité.

Raccordement des entrées tout-ou-rien

Raccordez les boutons-poussoirs, les interrupteurs et les détecteurs de proximité à 3 ou 4 fils aux bornes d'entrée I1 à I12. En raison du courant résiduel élevé, n'utilisez pas de détecteurs de proximité à 2 fils.

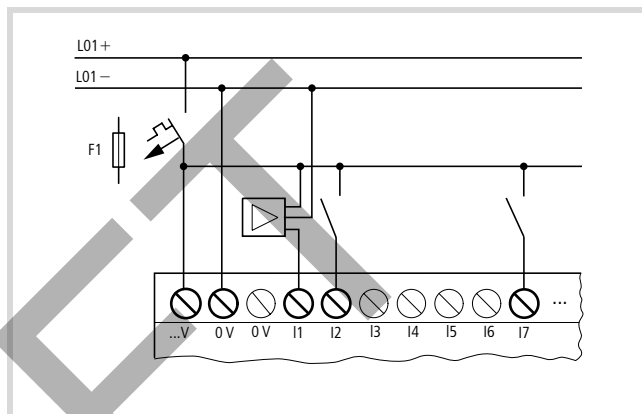


Figure 16 : Raccordement des entrées tout-ou-rien

Raccordement des entrées analogiques

Les entrées I7, I8, I11 et I12 vous permettent également de raccorder des tensions analogiques dans une plage de 0 à 10 V.

La résolution est de 10 bits = 0 à 1023.



Avertissement !

Respectez les remarques suivantes lors de la pose et du raccordement des lignes analogiques :

- ▶ Pour éviter les couplages de parasites sur les signaux analogiques, utilisez des paires torsadées blindées.
- ▶ En cas d'utilisation de câbles de faible longueur, reliez à la terre le blindage des câbles des deux côtés et sur toute la surface de contact. A partir d'une longueur de câble de 30 m environ, une mise à la terre aux deux extrémités peut engendrer une circulation de courants entre les deux points de mise à la terre et perturber ainsi les signaux analogiques. Dans ce cas, ne reliez le câble qu'à une seule extrémité.
- ▶ Ne disposez pas les câbles de signaux parallèlement aux câbles destinés au transport de l'énergie.
- ▶ Raccordez les charges inductives (commandées à l'aide des sorties de l'automate) à une tension d'alimentation séparée ou utilisez un circuit de protection aux bornes du récepteur. L'exploitation de charges telles que des moteurs, des électrovannes ou des contacteurs raccordées à la même tension d'alimentation peut provoquer lors de la commande une perturbation des signaux d'entrée analogiques.

Les schémas suivants montrent des exemples d'utilisation en matière d'acquisition de valeurs analogiques.

→ Créez une liaison équipotentielle au niveau du potentiel de référence. Reliez le 0 V des potentiomètres ou capteurs (présentés dans les exemples suivants) au 0 V de la tension d'alimentation de l'automate.

Raccordement: Potentiomètre d'entrée de consignes

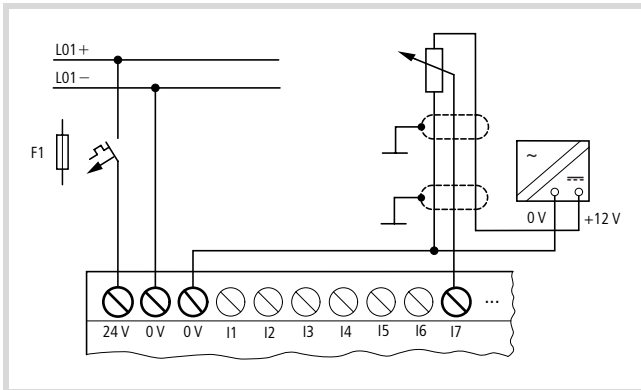


Figure 17 : Potentiomètre d'entrée de consignes

Faites appel à un potentiomètre présentant une résistance $\cong 1 \text{ k}\Omega$ (1 k Ω , par exemple) et 0,25 W.

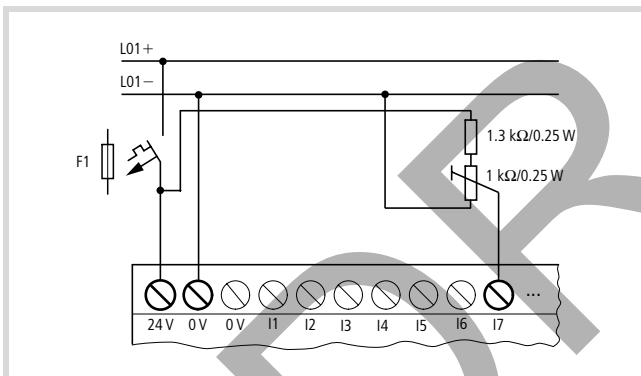


Figure 18 : Potentiomètre d'entrée de consignes avec résistance amont

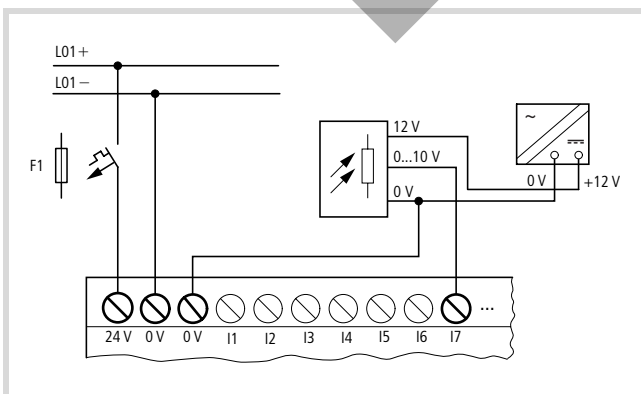


Figure 19 : Capteur de luminosité

Raccordement: Sonde de température

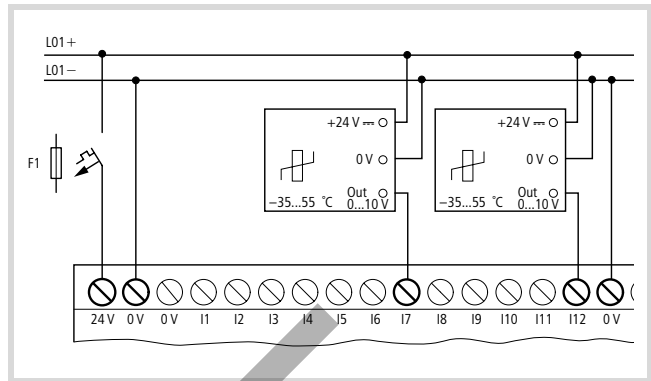


Figure 20 : Sonde de température

Raccordement d'un capteur 20 mA

Il est possible de raccorder sans problème un capteur de 4 à 20 mA (0 à 20 mA) à l'aide d'une résistance externe de 500 Ω .

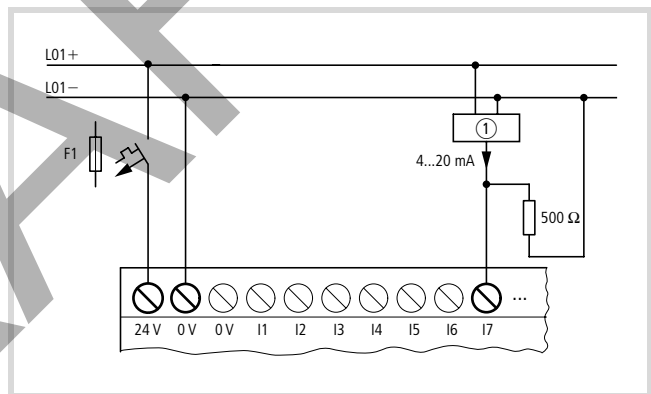


Figure 21 : Capteur 20 mA

① Capteur analogique

Il s'ensuit les valeurs suivantes :

- 4 mA = 1,9 V
 - 10 mA = 4,8 V
 - 20 mA = 9,5 V
- (d'après $U = R \times I = 478 \Omega \times 10 \text{ mA} \sim 4.8 \text{ V}$)

Raccordement d'un générateur d'impulsions/d'un codeur incrémental

Les entrées I1 à I4 sont conçues de manière à permettre le comptage de signaux rapides à l'aide de générateurs d'impulsions/de codeurs incrémentaux.

Vous avez le choix entre les possibilités de raccordement suivantes :

- 1 × générateur d'impulsions (32 bits)
- 2 × générateurs d'impulsions (16 bits)
- 1 × codeur incrémental (32 bits).

Raccordement d'un générateur d'impulsions

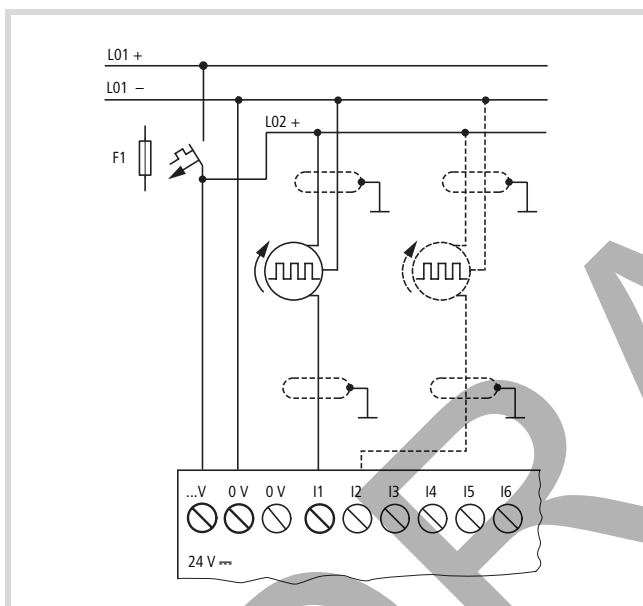


Figure 22 : Raccordement d'un générateur d'impulsions

Cette figure illustre le raccordement d'un générateur d'impulsions dont les impulsions sont amenées à l'entrée I1. Un compteur interne procède au traitement des impulsions. Vous avez le choix entre un compteur 16 bits (65535 max.) et un compteur 32 bits (4294967295 max.). Le générateur d'impulsions pour compteur 32 bits ne peut être raccordé qu'à I1. C'est uniquement si un compteur 16 bits est raccordé à I1 qu'il est possible de raccorder un autre générateur d'impulsions (de 16 bits) au niveau de I2.

Raccordement d'un codeur incrémental

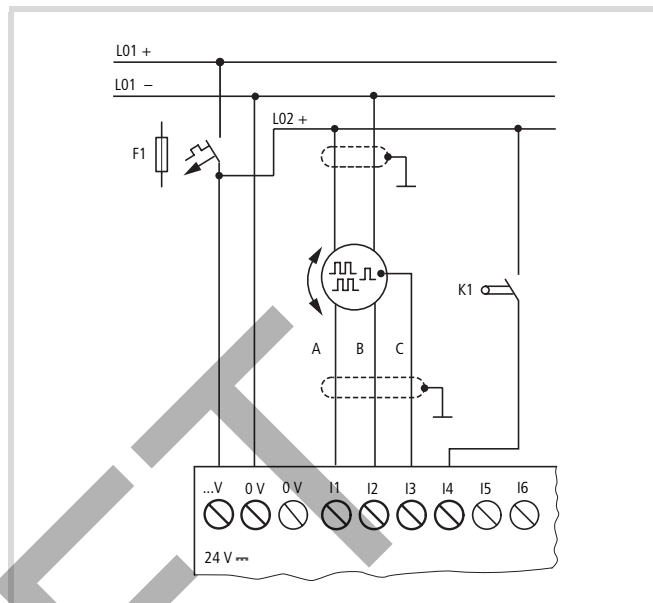


Figure 23 : Raccordement d'un codeur incrémental

A,B : signaux carrés incrémentaux déphasés de 90°
C : signal de référence
K1 : fin de course de fenêtre de référence

Raccordement des sorties

Les sorties à relais ou à transistors vous permettent de procéder à la commutation de charges telles que des tubes fluorescents, des lampes à incandescence, des contacteurs, des relais ou des moteurs. Avant l'installation, respectez les valeurs-limites et les caractéristiques techniques des sorties (→ page 110, 111).

Raccordement des sorties à relais

EC4P-221/222-MR..., EASY6...-DC-RE

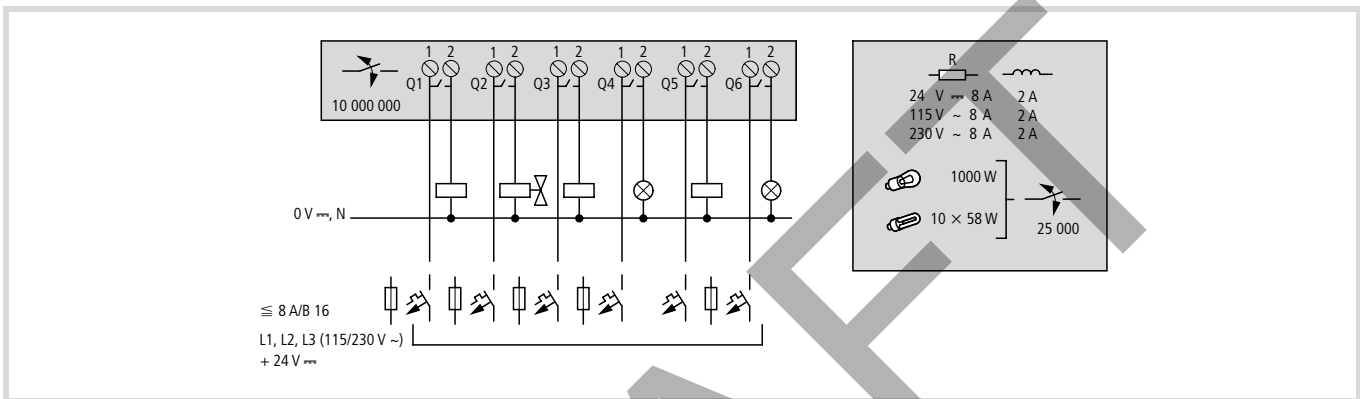


Figure 24 : Sorties à relais EC4P-221/222-MR...

Contrairement aux entrées, les sorties à relais EC4P-221/222-MR..., EASY6...-...RE autorisent le raccordement de phases différentes.



Avertissement !

Respectez la limite de tension maximale de 250 V AC au niveau du contact d'un relais. Une tension supérieure peut provoquer des décharges au niveau du contact et détruire ainsi l'appareil ou une charge raccordée.

Raccordement des sorties à transistors

EC4P-221/222-MT..., EASY6...-DC-TE

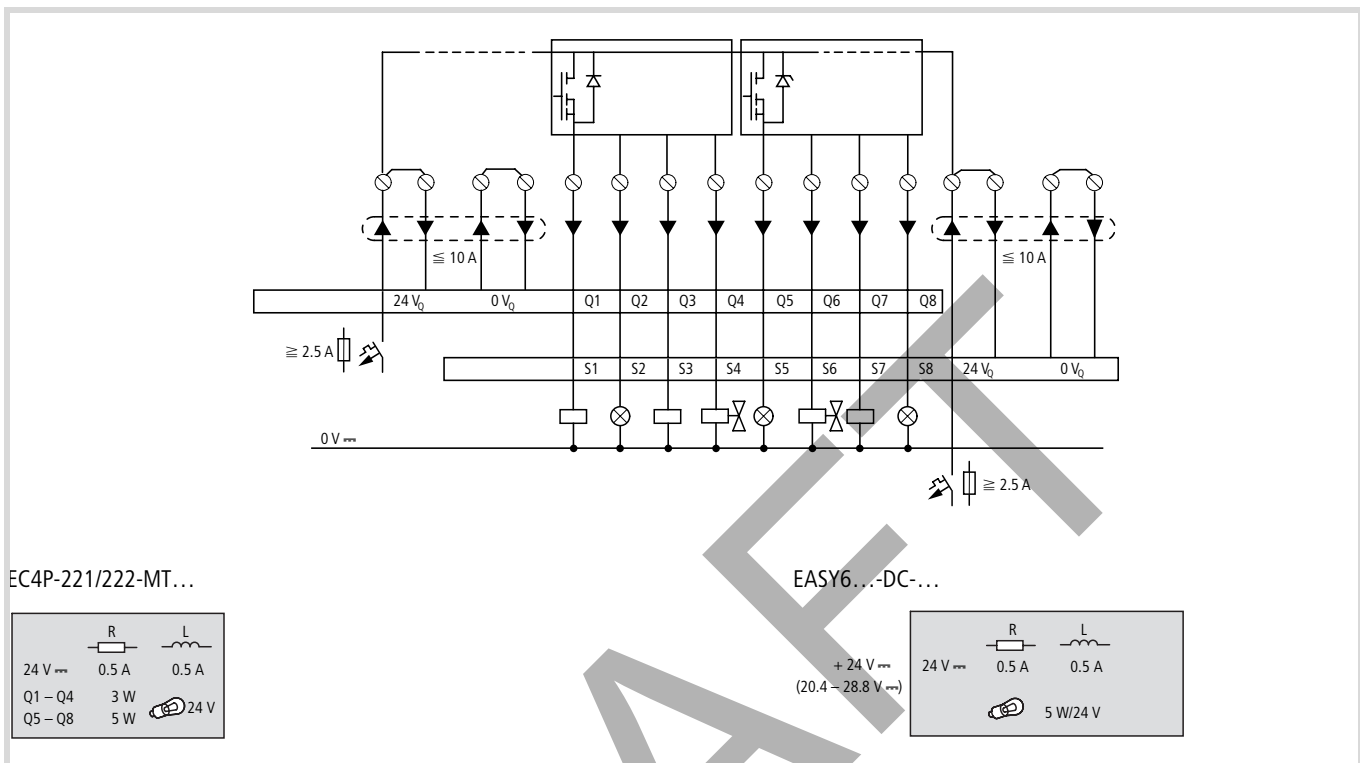


Figure 25 : Sorties à transistors EC4P-221/222-MT..., EASY6...-DC-TE

Montage en parallèle :
 Pour augmenter la puissance, il est possible de monter jusqu'à quatre sorties en parallèle. Le courant de sortie résultant atteint au maximum 2 A.

avec inductance risque d'être coupée, vous devez impérativement équiper les inductances d'un circuit de protection (→ figures suivantes).

⚠ Avertissement !
 En cas de coupure de charges inductives, il convient de tenir compte des points suivants :
 Les inductances avec circuit de protection engendrent moins de perturbations dans l'ensemble du système électrique. De manière générale, il est recommandé de raccorder le circuit de protection le plus près possible de l'inductance.

⚠ Avertissement !
 Seules les sorties appartenant à un même groupe (Q1 à Q4 ou Q5 à Q8) peuvent être montées en parallèle.
 Exemple . Q1 et Q3 ou Q5, Q7 et Q8. Les sorties montées en parallèle doivent impérativement être commandées simultanément.

Remarques valables lorsque les inductances ne possèdent pas de circuit de protection :
 Ne coupez pas simultanément plusieurs inductances, sous peine de provoquer dans le pire des cas un échauffement des modules pilotes. Si l'alimentation +24-V DC est coupée en cas d'arrêt d'urgence à l'aide d'un contact et que plus d'une sortie activée

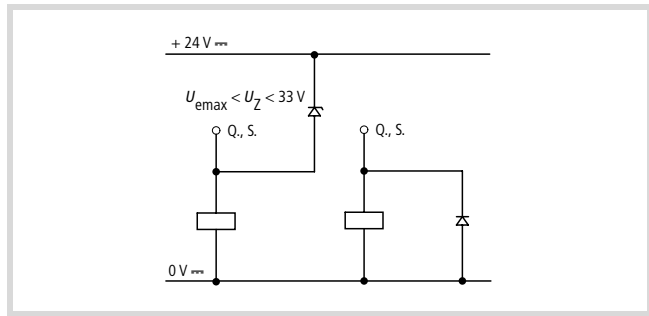


Figure 26 : Inductance avec circuit de protection

Comportement en cas de court-circuit/surcharge
 L'apparition d'un court-circuit ou d'une surcharge au niveau d'une sortie à transistors entraîne la coupure de cette sortie. A l'issue d'un temps de refroidissement qui est fonction de la température ambiante et de l'intensité du courant, le contact de sortie se referme jusqu'à ce que la température maximale soit atteinte. Si le défaut persiste, la sortie continue de s'ouvrir et de se fermer jusqu'à élimination du défaut ou jusqu'à la mise hors tension.

Raccordement d'une sortie analogique

L'EC4-200 possède une sortie analogique QA 01, 0 à 10 V DC, résolution 10 bits (0 à 1023). Cette sortie analogique vous permet de commander des servovalves ou d'autres actionneurs.

Raccordement d'une servovalve

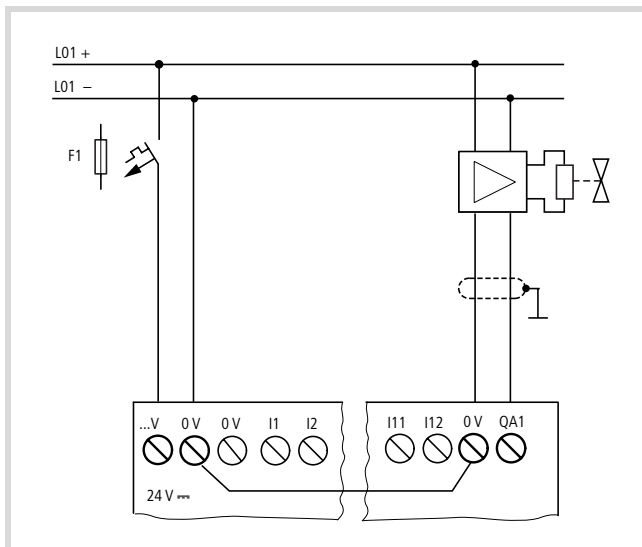


Figure 27 : Raccordement d'une servovalve

Envoi de consigne pour un entraînement

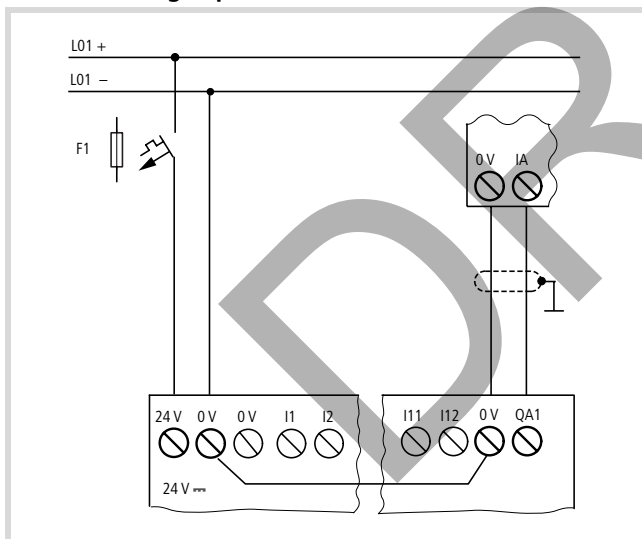


Figure 28 : Envoi de consigne pour un entraînement



Avertissement !

Les signaux analogiques sont plus sensibles aux parasites que les signaux tout-ou-rien : il convient donc de placer et de raccorder avec précaution les câbles véhiculant les signaux. Tout raccordement incorrect peut provoquer des états de commutation incontrôlés.

Carte mémoire, CAN/easyNet, liaison PC

Pour enficher un module mémoire ou créer une liaison PC ou CAN/easyNet, vous devez préalablement retirer le cache.

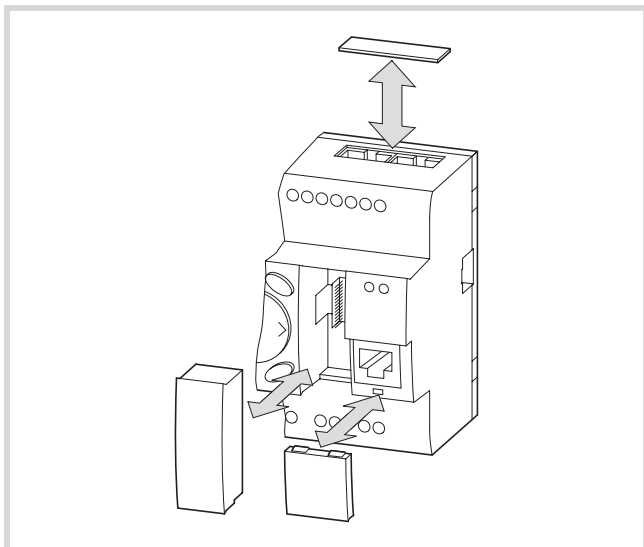


Figure 29 : Retrait du cache/de l'adaptateur : en haut : pour la liaison CAN/easyNet
en bas, à gauche : adaptateur pour module mémoire
en bas à droite : liaison PC

Enfichage/retrait du module mémoire

Le module mémoire se trouve dans l'adaptateur ③.

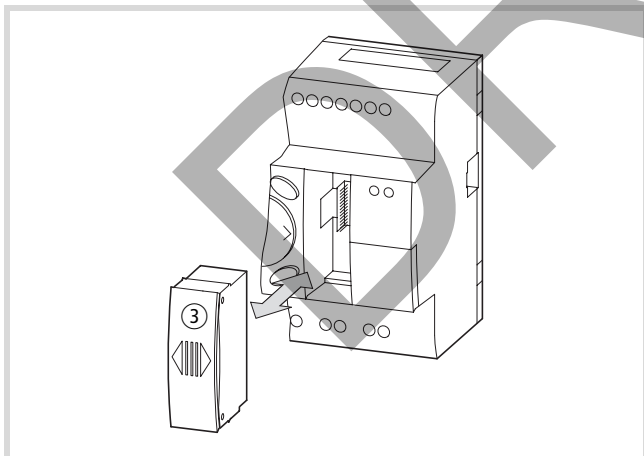


Figure 30 : Adaptateur avec module mémoire

- ▶ Pour enficher le module mémoire, appuyez dessus jusqu'à l'encliquetage.
- ▶ Pour retirer le module mémoire, vous devez appuyer dessus jusqu'à ce qu'il soit libéré par le mécanisme d'encliquetage.

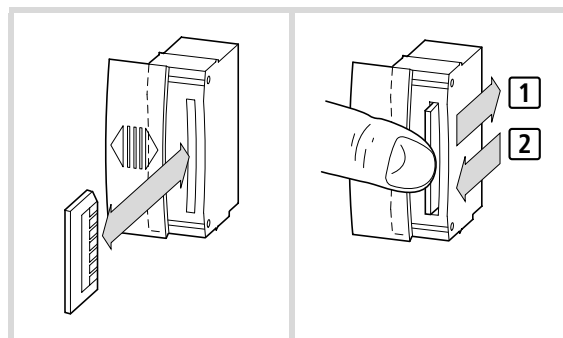


Figure 31 : Enfichage/Retrait du module mémoire

CAN/easyNet, liaison PC

- ▶ Enfichez le connecteur destiné à la liaison CAN/easyNet dans l'ouverture située à la partie supérieure de l'appareil ①.
- ▶ Enfichez le connecteur destiné à la liaison PC dans l'ouverture située à la partie inférieure droite de l'appareil ②.

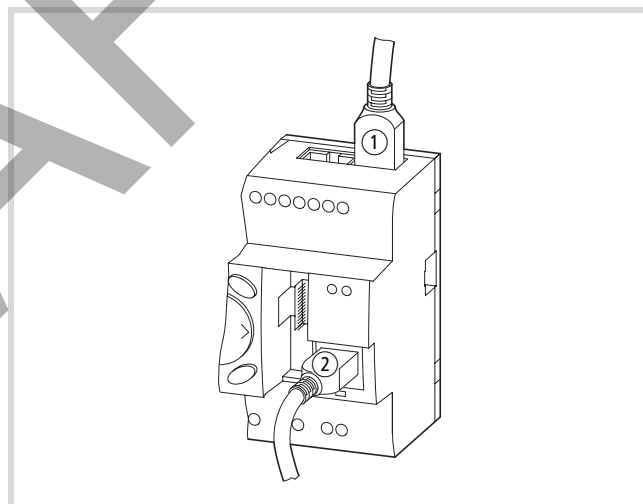


Figure 32 : Connecteurs pour liaison CAN/easyNet ① et liaison PC ②

→ Pour plus d'informations → paragraphe « Réseau CAN/easyNet », page 103.

**Attention !**

Protégez l'EC4-200 et le module mémoire comme suit contre les décharges électrostatiques : avant d'enficher ou de retirer le module mémoire, déchargez-vous de l'électricité statique dont vous êtes porteur en touchant une surface reliée à la terre.

Raccordement d'extensions/de modules de couplage réseau

Extension centralisée

- Raccordez les appareils à une extension ou à module de couplage réseau à l'aide d'un connecteur de liaison EASY-LINK-DS.

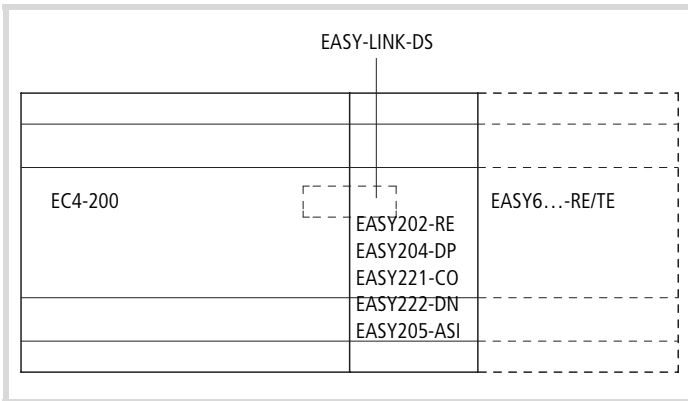


Figure 33 : Raccordement des appareils d'extension à EC4-200

- L'appareil de base et l'appareil d'extension peuvent être alimentés avec différentes tensions DC.

Extension décentralisée

En cas d'extension décentralisée, vous pouvez installer et exploiter les appareils d'extension jusqu'à une distance de 30 m par rapport à l'appareil de base.



Danger !

Les câbles bifilaires ou multibrins reliant les appareils doivent respecter la tension d'isolement requise pour l'environnement relatif à l'installation. En cas de défaut (défaut à la terre, court-circuit), le non-respect de cette consigne risque d'entraîner des dommages corporels ou la détérioration des appareils.

Un câble du type NYMO, par exemple, admettant une tension assignée d'emploi de $U_e = 300/500$ V AC suffit dans la majorité des cas.



Il existe une séparation électrique (séparation toujours située au niveau du raccordement local de l'extension) entre l'appareil de base et un appareil d'extension :

- séparation simple 400 V AC (+10 %)
- séparation sûre 240 V AC (+10 %)

Le dépassement de la valeur 400 V AC +10 % peut entraîner la destruction des appareils et un mauvais fonctionnement de l'installation ou de la machine.

- Les bornes E+ et E- de EASY200-EASY sont protégées contre les courts-circuits et l'inversion de polarité. Le fonctionnement n'est possible que si E+ est reliée à E+ et que E- est reliée à E-.

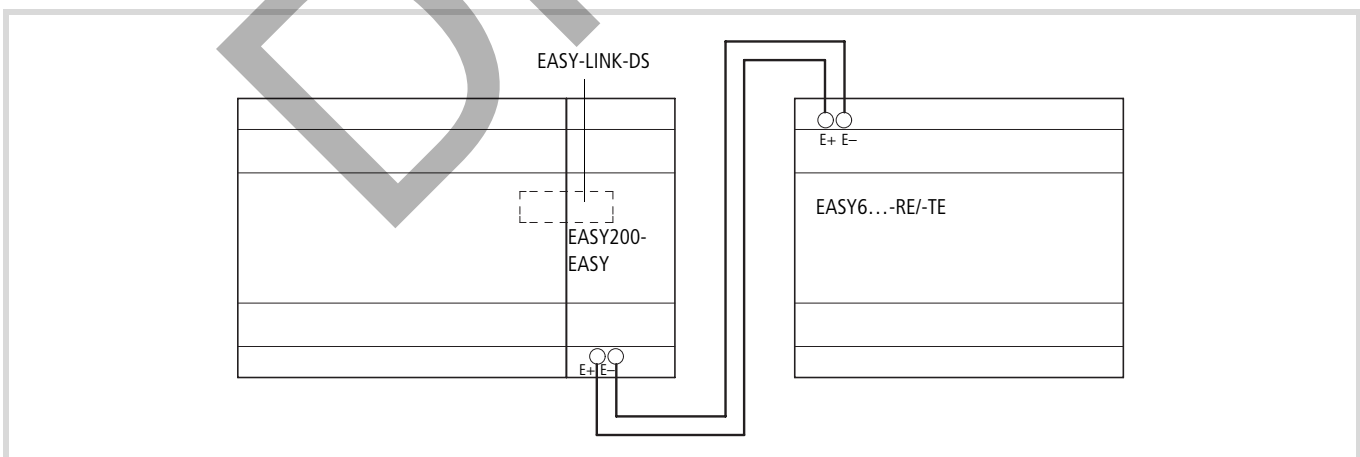
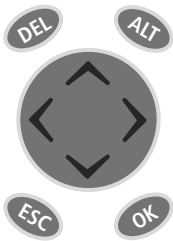


Figure 34 : Raccordement d'extensions décentralisées à EC4-200

6 Utilisation

Ce chapitre porte sur l'utilisation des touches et les différents affichages (afficheur) au niveau du plastron.

Touches de commande



DEL : effacement

ALT : fonctions spéciales, visualisation d'état

Touches de direction < > ^ v :
pour déplacer le curseur
pour sélectionner les options des menus
pour paramétrer des chiffres, des valeurs

OK : poursuite, enregistrement

ESC : retour en arrière, annulation

Dialogue par menus et saisie de valeurs

DEL et **ALT**

Appel d'un menu spécial

OK

Passage au niveau menu suivant
Appel d'une option menu
Activation, modification, enregistrement des saisies

ESC

Passage au niveau menu précédent
Annulation des saisies effectuées depuis le dernier **OK**

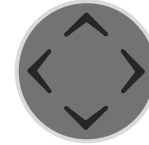


^ Passage à une autre option menu
v Modification d'une valeur
< > Changement d'emplacement

Fonction « Touches P » :

< Entrée P1, ^ Entrée P2
> Entrée P3, v Entrée P4

Choix d'options menu ou passage à d'autres options



Touches de direction
^ v

OK

Sélection ou passage à une autre option

Représentation du curseur



Le curseur clignote lors d'un changement.

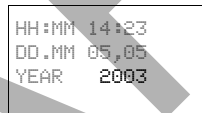
Curseur plein ████ :

- Déplacer le curseur à l'aide de < > ^ v.

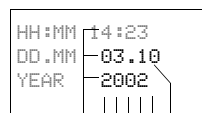
Valeur M/M

- Changement d'emplacement à l'aide de < >
- Modification des valeurs à l'aide de ^ v

Les valeurs qui clignent sont représentées en gris dans le présent manuel.



Réglage d'une valeur



Valeurs
Emplacements
Valeur à cet emplacement



Sélection d'une valeur à l'aide de ^ v.

Sélection d'un emplacement à l'aide de < >.

Modification à l'aide de ^ v de la valeur indiquée à l'emplacement sélectionné.

OK

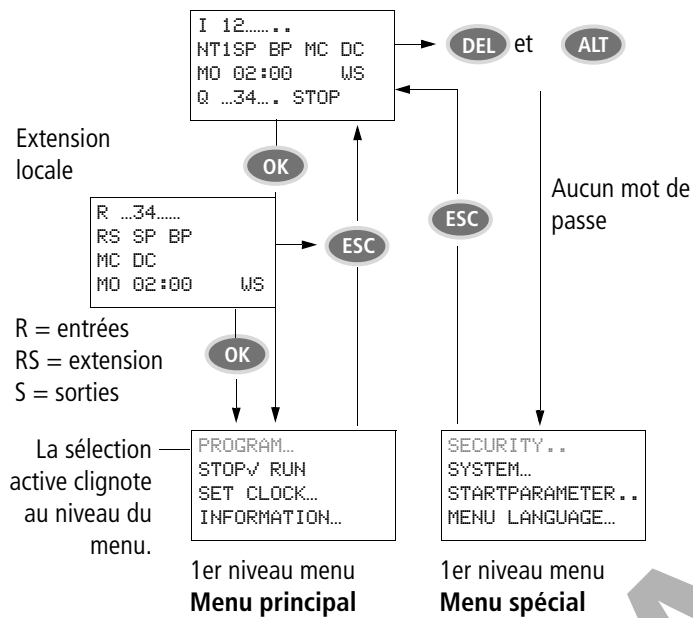
Enregistrement du réglage

ESC

Conservation de la valeur précédente

Sélection du Menu principal et du Menu spécial

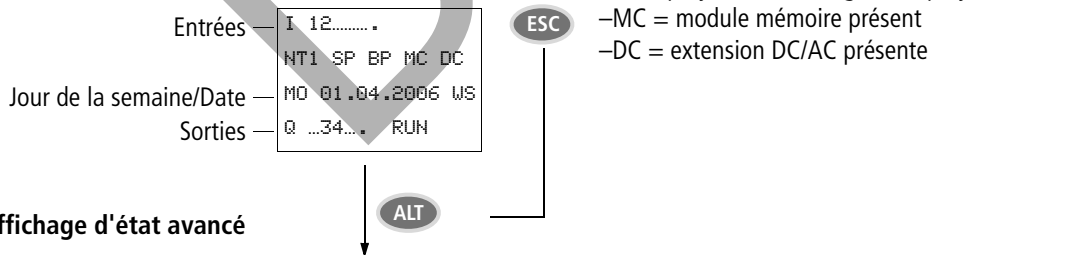
Affichage d'état



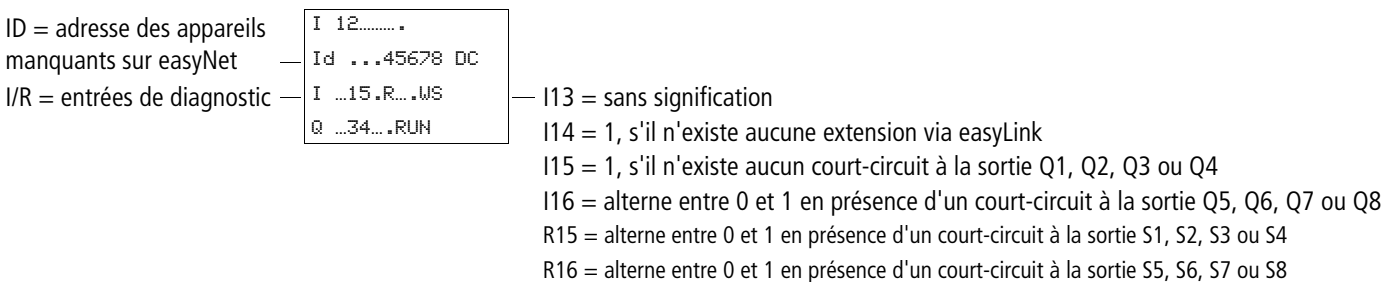
Affichage d'état avec heure



Affichage d'état avec date



Affichage d'état avancé



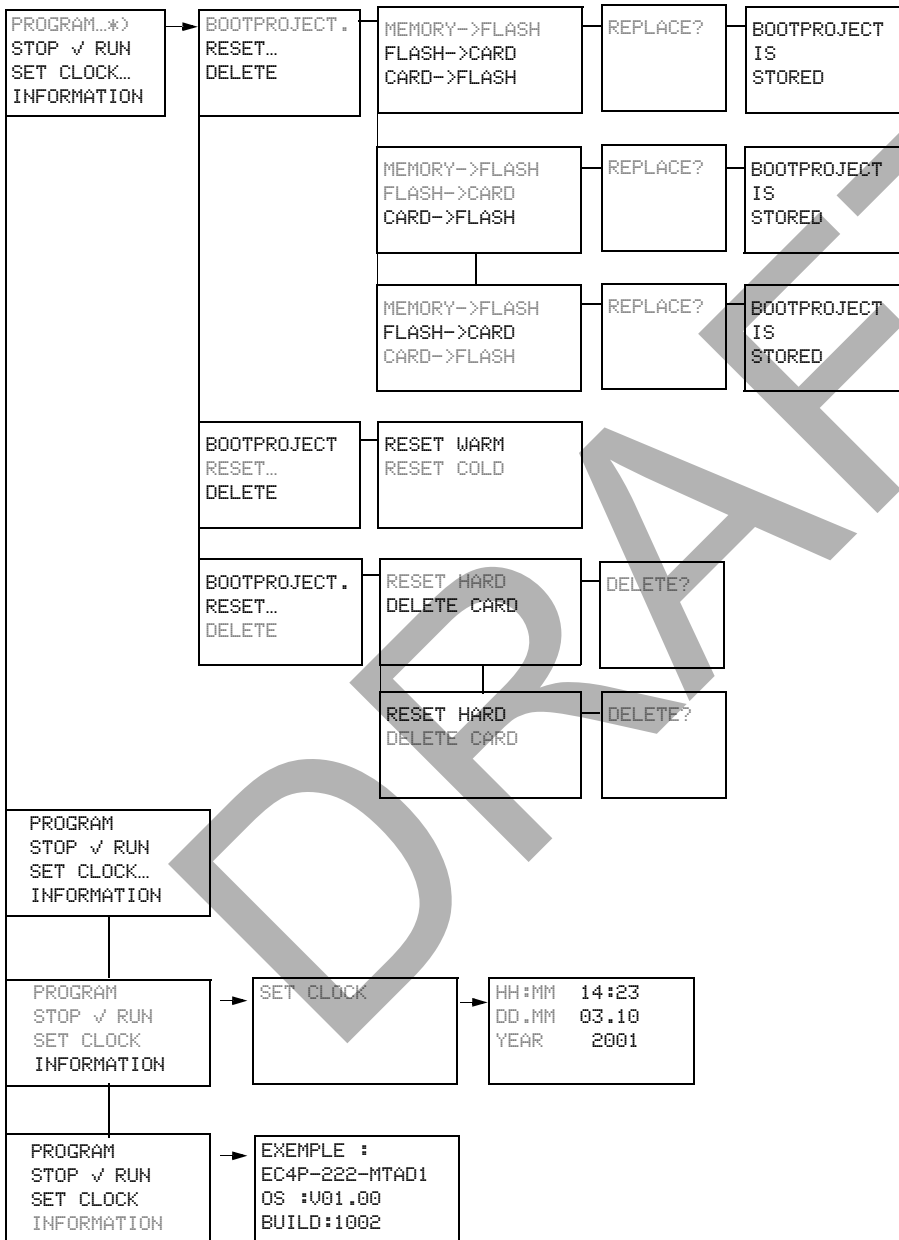
Structure des menus

Menu principal sans protection par mot de passe

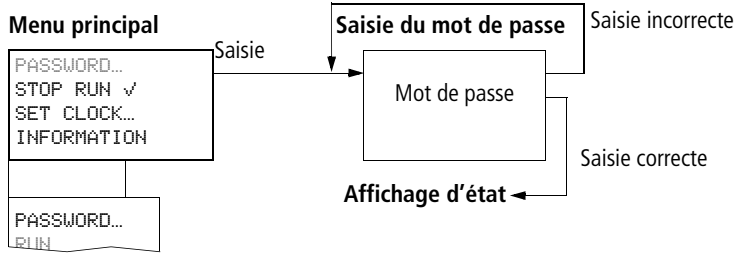
- L'actionnement de la touche OK vous permet d'accéder au menu principal.

Menu principal

* à n'utiliser qu'en mode STOP

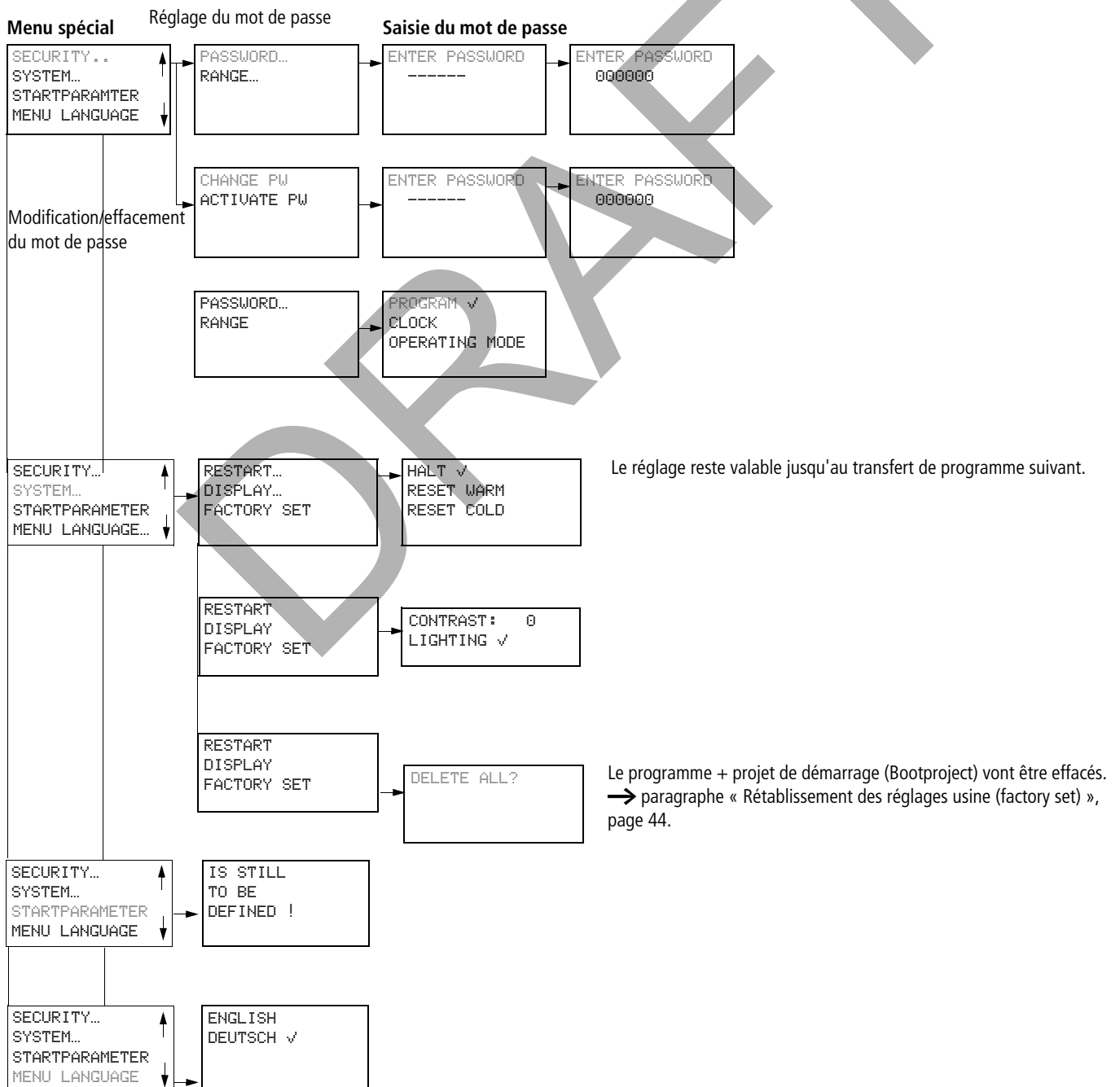


Menu principal protégé par mot de passe



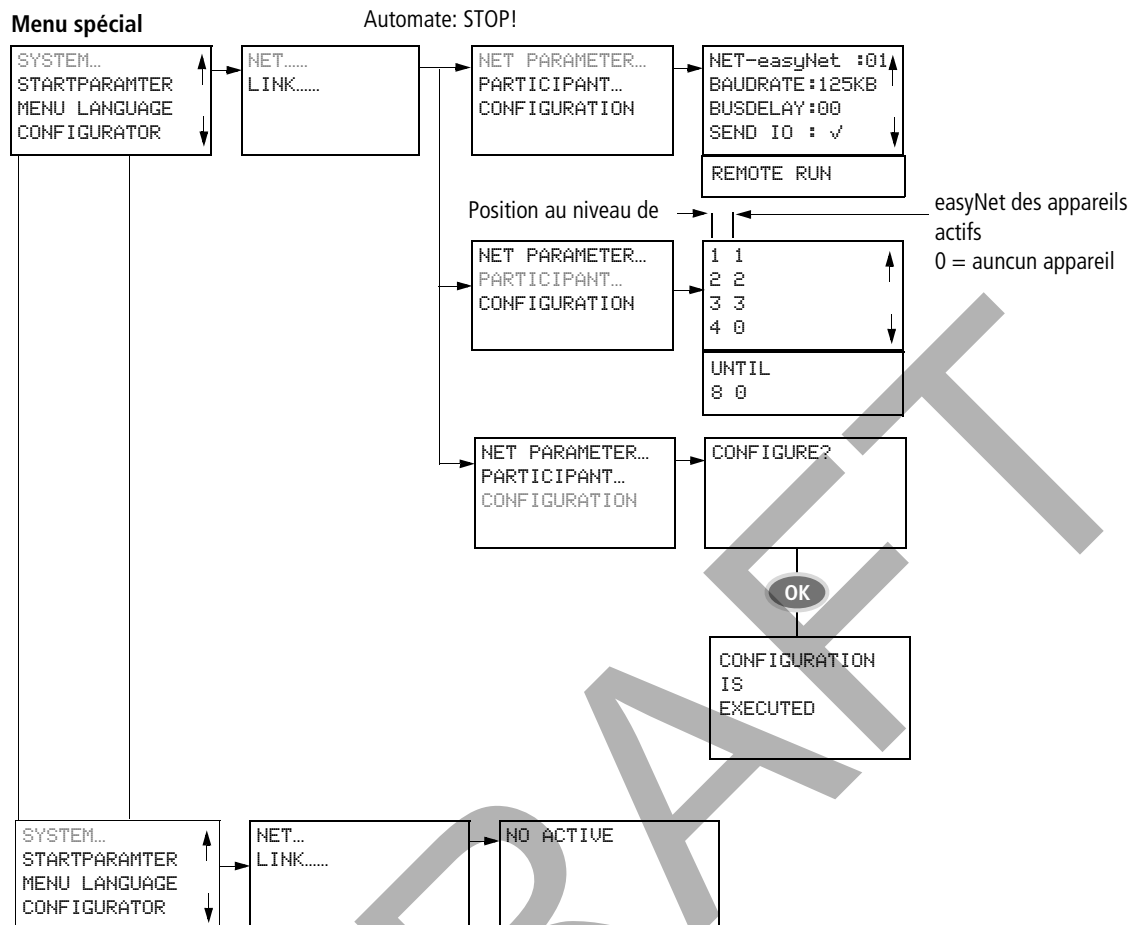
Menu spécial

- L'actionnement simultané des touches DEL et ALT vous permet de passer au Menu spécial.



Menu spécial

Menu spécial



DRAFT

7 Description des réglages

Tous les réglages s'opèrent à l'aide des touches de commande de l'automate.

Protection par mot de passe

Vous pouvez protéger par mot de passe l'accès au Menu principal et au Menu spécial, le réglage de l'horloge et du mode d'exploitation (RUN/STOP). C'est via le menu <Security → Range> que vous activez les différentes possibilités de réglage.

Le menu spécial est toujours protégé en cas de mot de passe activé.

Le mot de passe à saisir doit avoir une valeur comprise entre 000001 et 999999. La combinaison numérique 000000 permet d'effacer un mot de passe.

Paramétrage d'un mot de passe

Vous pouvez paramétrer un mot de passe via le menu spécial, indépendamment du mode RUN ou STOP. Dans le cas où un mot de passe a déjà été activé, vous ne pouvez pas passer au Menu spécial.

- ▶ Appelez le Menu spécial à l'aide des touches **DEL** et **ALT**.
- ▶ Lancez la saisie du mot de passe via l'option menu SECURITY...
- ▶ Appuyez sur la touche **OK** et passez au menu PASSWORD...
- ▶ En actionnant une nouvelle fois la touche **OK**, vous passez à la saisie du mot de passe.

Si aucun mot de passe n'est saisi, six tirets s'affichent : cela signifie qu'il n'existe encore aucun mot de passe.

```
ENTER PASSWORD
█-----
```

- ▶ Si vous actionnez la touche **OK**, six zéros s'affichent.
- ▶ Paramétrez le mot de passe à l'aide des touches de direction:
 - Touches < > : pour sélectionner l'emplacement dans le mot de passe,
 - Touches ^ v : pour saisir une valeur comprise entre 0 et 9.

```
ENTER PASSWORD
000042
```

- ▶ Enregistrez le nouveau mot de passe à l'aide de la touche **OK**.

Appuyez sur la touche **OK** pour quitter l'affichage du mot de passe ; actionnez ensuite les touches **ESC** et **✓** pour vous rendre au menu RANGE...

La plage de validité du mot de passe n'a pas encore été prise en compte. Le mot de passe est valable, mais n'a pas encore été activé.

Sélection de la plage de validité du mot de passe

- ▶ Appuyez sur la touche **OK**.
- ▶ Sélectionnez la fonction ou le menu à protéger.
- ▶ Appuyez sur la touche **OK** pour protéger cette fonction ou ce menu (coche présente = protection activée).

```
PROGRAM ✓
CLOCK
OPERATING MODE
```

→ La protection standard se situe au niveau du programme. Il convient de protéger au moins une fonction ou un menu.

- PROGRAM : le menu PROGRAM est protégé.
- CLOCK : la date et l'heure sont protégées par mot de passe.
- OPERATING MODE : la possibilité de commutation entre les modes RUN et STOP est protégée par mot de passe.

Activation du mot de passe

L'activation d'un mot de passe existant peut s'opérer de quatre manières :

- automatiquement, lors de la remise sous tension de l'automate ;
- automatiquement, lors du chargement du programme ;
- automatiquement, lorsqu'aucun télégramme n'a été envoyé à l'interface PC dans les 30 minutes suivant la saisie du mot de passe ;
- via le menu relatif au mot de passe.

- ▶ Appelez le Menu spécial à l'aide des touches **DEL** et **ALT**.
- ▶ Ouvrez le menu relatif au mot de passe, via l'option SECURITY...

Le menu relatif au mot de passe ne s'affiche qu'en cas d'existence d'un mot de passe.

```
CHANGE PW
ACTIVATE
```

La protection standard se situe au niveau du programme.

→ Prenez bien note de votre mot de passe avant de l'activer. En cas d'oubli du mot de passe, vous ne pouvez plus passer au Menu spécial.

- ▶ Sélectionnez **ACTIVATE PW** et appuyez sur la touche **OK**. Le mot de passe est alors actif. L'appareil passe à l'affichage d'état.

Vous devez saisir le mot de passe avant de pouvoir accéder à un menu protégé, exécuter une fonction protégée ou passer au Menu spécial.

Accès possibles en cas de protection par mot de passe

Après saisie du mot de passe, la protection par mot de passe se trouve désactivée. Vous pouvez réactiver cette protection ultérieurement, à l'aide du menu relatif au mot de passe ou par coupure puis rétablissement de la tension d'alimentation.

- Passez au Menu principal à l'aide de la touche OK.

L'indication PASSWORD... clignote.

```
PASSWORD...
STOP RUN ✓
PASSWORD...
SET CLOCK...
```

- Passez à la saisie du mot de passe à l'aide de la touche OK.

→ Si le menu principal affiche PROGRAM... au lieu de PASSWORD..., cela signifie qu'aucune protection par mot de passe n'est activée.

La zone réservée à la saisie du mot de passe s'affiche.

```
ENTER PASSWORD
XXXXXX
```

- Paramétrez le mot de passe à l'aide des touches de direction.
- Validez ensuite votre choix par **OK**.

Si le mot de passe est correct, vous revenez à l'Affichage d'état.

L'option menu PROGRAM... est déverrouillée.

```
PROGRAM...
STOP
PARAMETER
SET CLOCK...
```

Vous avez également accès au Menu spécial.

Modification ou effacement du mot de passe ou d'une plage

- Saisissez le mot de passe.
- Appelez le Menu spécial à l'aide des touches **DEL** et **ALT**.
- Ouvrez le menu Mot de passe via les options SECURITE et PASSWORD...

L'indication ACTIVATE PW clignote.

```
CHANGE PW
ACTIVATE PW
```

L'appareil n'affiche ce menu que lorsqu'un mot de passe a été préalablement saisi.

- Appelez la saisie du mot de passe à l'aide de la touche OK.
- Passez à l'aide de la touche **OK** à la zone de saisie à 6 chiffres.
- Le mot de passe actif s'affiche.
- Modifiez les six positions du mot de passe à l'aide des touches de direction.
- Validez ensuite votre choix par **OK**.

```
ENTER PASSWORD
XXXXXX
```

```
ENTER PASSWORD
100005
```

Appuyez sur la touche ESC pour quitter la plage de sécurité.

Effacement

La combinaison numérique « 000000 » permet d'effacer un mot de passe.

```
ENTER PASSWORD
-----
```

Si aucun mot de passe n'a été saisi, six tirets s'affichent.

Mot de passe saisi incorrect ou oublié

Vous avez saisi un mot de passe incorrect ?

```
ENTER PASSWORD
XXXXXX
```

- Saisissez de nouveau le mot de passe.

Vous pouvez le faire autant de fois que souhaité.

Si vous actionnez la touche ESC, vous revenez au menu du départ.



En cas d'oubli du mot de passe, vous pouvez appeler la commande navigateur "factoryset". Le mot de passe, le programme utilisateur et le projet de démarrage (bootproject) sont alors effacés et l'automate est initialisé avec les paramètres par défaut : → paragraphe « Reset (remise à zéro) », page 44.

Modification du choix de la langue des menus

Les menus sont disponibles en deux langues : le choix s'opère via le Menu spécial.

Langue	Affichage
Anglais	ENGLISH
Allemand	DEUTSCH

→ L'accès au choix de la langue n'est possible que si l'automate n'est pas protégé à l'aide d'un mot de passe.

- ▶ Appelez le Menu spécial à l'aide des touches **DEL** et **ALT**.
- ▶ Pour modifier la langue des menus, sélectionnez MENU LANGUAGE...

Le choix de la langue (ENGLISH) pour la première saisie s'affiche.

```
ENGLISH
DEUTSCH ✓
```

- ▶ A l'aide des touches \wedge ou \vee , sélectionnez éventuellement la nouvelle langue pour les menus.
- ▶ Validez ensuite votre choix par **OK**. La langue retenue comporte alors une coche.
- ▶ Quittez ce menu à l'aide de la touche ESC.

La nouvelle langue choisie pour les menus devient alors active.

La touche ESC vous permet de revenir à l'Affichage d'état.

Réglage de la date et de l'heure

Les appareils sont dotés d'une horloge temps réel avec date et heure. Lors de la première mise en service, réglez l'heure, les minutes, le jour, le mois et l'année.

- ▶ Dans le Menu principal, sélectionnez SET CLOCK...

Le menu destiné au réglage de l'heure s'affiche.

```
SET CLOCK
```

- ▶ Sélectionnez SET CLOCK.

- ▶ Réglez les valeurs correctes au niveau de l'heure, du jour, du mois et de l'année.

```
HH:MM: 00.27
DD.MM: 05.05
YEAR : 2002
```

- ▶ Actionnez la touche **OK** pour passer au Mode saisie.
 - Sélectionnez l'emplacement à l'aide des touches $\langle \rangle$.
 - Modifiez les valeurs à l'aide des touches $\wedge \vee$.
 - Appuyez sur la touche **OK** pour enregistrer le jour et l'heure.
 - Appuyez sur la touche **ESC** pour conserver le réglage antérieur.

La touche ESC vous permet de quitter l'Affichage destiné au réglage de l'heure.

Comportement au démarrage

Réglage du comportement au démarrage

Le menu RESTART vous permet de sélectionner l'une des possibilités de démarrage suivantes :

- HALT
- RESET WARM
- RESET COLD.

- ▶ Passez au Menu spécial.

→ Si l'automate est protégé par un mot de passe, le Menu spécial n'est disponible qu'après saisie de ce mot de passe (→ paragraphe « Accès possibles en cas de protection par mot de passe », page 36)

- ▶ Réglez le comportement au démarrage.

Réglage du contraste et du rétro-éclairage de l'afficheur LCD

Il est possible de désactiver le rétro-éclairage de l'afficheur à cristaux liquides (afficheur LC). Le contraste de l'afficheur est réglable selon 5 niveaux. L'afficheur n'est pas nécessaire lors du fonctionnement. Le rétro-éclairage n'est utile qu'en cas de maintenance ou lorsqu'il convient d'afficher des textes.

- Passez au Menu spécial.

→ Si l'automate est protégé par un mot de passe, le Menu spécial n'est disponible qu'après saisie de ce mot de passe (→ paragraphe « Accès possibles en cas de protection par mot de passe » page 36).

- Sélectionnez le menu SYSTEM.
- Appuyez sur la touche OK.

```
SECURITY
SYSTEM...
STARTPARAMETER...
MENU LANGUAGE...
```

- A l'aide de la touche de direction \downarrow , sélectionnez le menu DISPLAY, puis actionnez la touche **OK**.

```
RESTART...
DISPLAY...
FACTORY SET
```

Les menus relatifs au réglage du contraste et du rétro-éclairage s'affichent alors.

```
CONTRAST 0
LIGHTING ✓
```

- Actionnez la touche **OK** et passez ainsi au réglage du contraste.

Les touches de direction \wedge et \vee vous permettent de modifier le contraste en choisissant une valeur comprise entre -2 et +2.

```
CONTRAST: +1
LIGHTING ✓
```

- Choisissez votre valeur de réglage.

- Confirmez votre réglage à l'aide de la touche **OK**.

```
CONTRAST: +1
LIGHTING ✓
```

Le réglage du contraste demeure inchangé jusqu'à la prochaine modification.

- Utilisez les touches de direction \wedge et \vee pour passer au menu LIGHTING (rétro-éclairage).
- Appuyez sur la touche OK.

```
CONTRAST: +1
LIGHTING ✓
```

- Le rétro-éclairage est désactivé.

```
CONTRAST: +1
LIGHTING
```

- Si vous souhaitez réactiver le rétro-éclairage, actionnez la touche **OK**.

```
CONTRAST: +1
LIGHTING ✓
```

- La présence de la coche \checkmark signale que le rétro-éclairage est activé.

→ Le réglage de base à la livraison est le suivant :
Le contraste est réglé sur 0.
Le rétro-éclairage est activé en permanence. Réglage du menu : LIGHTING \checkmark

8 Configuration des entrées/sorties (I/O)

Représentation des entrées/sorties au niveau de la configuration

Au niveau de la Configuration de l'automate, des noms symboliques sont déjà affectés aux adresses directes des entrées/sorties.

Opérande symbolique	Opérande physique	Type de donnée
I1	AT % IX0.0	BOOL

Les opérandes symboliques sont directement utilisables dans le programme.

Affichage des entrées/sorties locales

- ▶ Pour afficher les entrées/sorties locales, cliquez d'abord sur le symbole + situé devant "Configuration EC4P-200", puis sur le symbole + placé devant "Local I/O".
- ▶ Sous le dossier "Local I/O" s'affichent également les dossiers suivants, dont il convient d'adapter la fonction au type d'automate effectivement retenu.
 - Transistor Outputs (sorties à transistors)
 - No Analog Outputs (aucune sortie analogique)
 - No Keys (aucune touche de commande)
 - No Counter (aucun compteur).

Dans le cas d'un automate avec sorties à relais, la fonction du dossier "Transistor Outputs" (sorties à transistors) doit être modifié en "Relay Outputs" (sorties à relais), → paragraphe « Modification de la fonction des dossiers ».

- ▶ Pour afficher les entrées/sorties de chaque dossier, cliquez à l'aide du bouton gauche de la souris sur le symbole + situé devant le dossier.

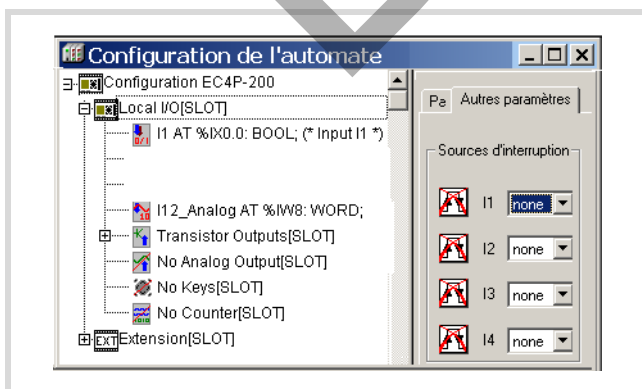


Figure 35 : Entrées/sorties sélectionnables

Modification de la fonction des dossiers

Transistor Outputs ↔ Relais Outputs (sorties à transistors ↔ sorties à relais)

Les sorties à transistors (Transistor Outputs) s'affichent dans le cadre de la configuration par défaut d'un automate. Si vous utilisez un automate équipé de relais, vous devez modifier le type de sortie ("Output-Typ") :

- ▶ A l'aide du bouton droit de la souris, cliquez sur l'indication "Transistor Outputs".
- ▶ Amenez le curseur sur l'option menu "Remplacer élément", puis cliquez sur "Sorties à relais" (Relais Outputs).

De manière générale : pour afficher les adresses directes et symboliques des sorties, cliquez sur le symbole "xxx Output".

No Analog Output (aucune sortie analogique)

La configuration par défaut "No Analog Output" peut être remplacée par "Analog Output" à partir de la version 2.0 du système d'exploitation.

No Keys (aucune touche de commande)

Le terme "Keys" désigne ici les touches situées en face avant de l'automate (touches de fonction ALT, DEL, ESC et OK, ainsi que les 4 touches de direction). Dans la configuration de l'automate, ces touches sont représentées sous forme d'entrées.

Si vous souhaitez scruter l'état des touches au sein du programme, vous pouvez programmer les adresses directes ou symboliques des entrées.

- ▶ A l'aide du bouton droit de la souris, cliquez sur l'indication "No Keys".
- ▶ Amenez le curseur sur l'option menu "Remplacer élément", puis cliquez sur "Keys".
- ▶ Cliquez sur le symbole + situé devant "Keys".

No Counter (aucun compteur)

Si votre application requiert des compteurs rapides, vous devez les activer.

- ▶ A l'aide du bouton droit de la souris, cliquez sur l'indication "No Counter".
- ▶ Amenez le curseur sur l'option menu "Remplacer élément", puis cliquez sur l'une des 3 fonctions de comptage.

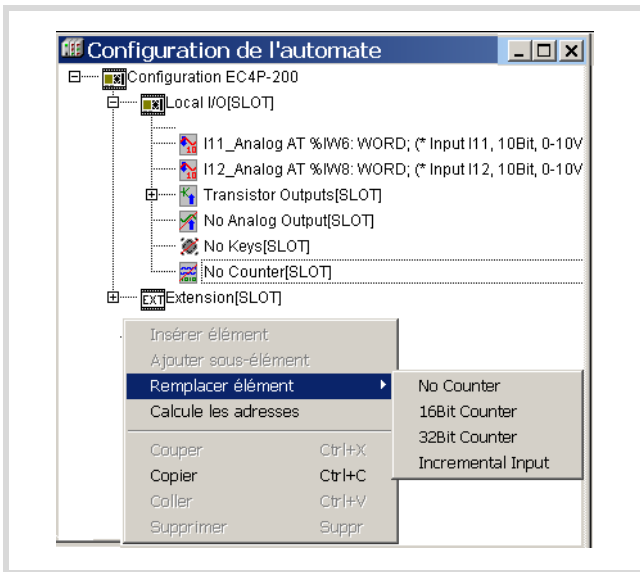


Figure 36 : Choix du compteur

Le sous-menu apparaît :

- ▶ Sélectionnez un type de compteur (un compteur 32 bits, par exemple).
- ▶ L'indication "No Counter" est alors remplacée par "32 Bit Counter".
- ▶ Un autre clic sur le symbole + permet d'afficher les entrées/sorties du compteur.

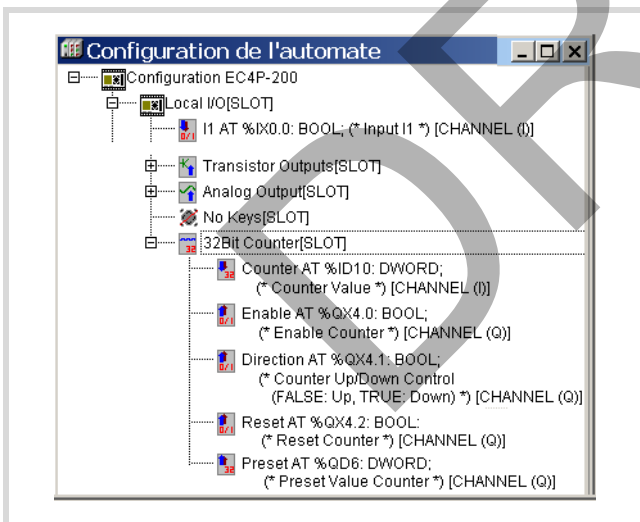


Figure 37 : Configuration d'un compteur (de 32 bits)

Affichage des entrées/sorties des appareils d'extension

- ▶ Cliquez sur le symbole + situé devant le dossier "Extension".
- ▶ A l'aide du bouton droit de la souris, cliquez sur le dossier "No Extension".
- ▶ Sélectionnez un appareil indiqué dans le menu "Remplacer élément".

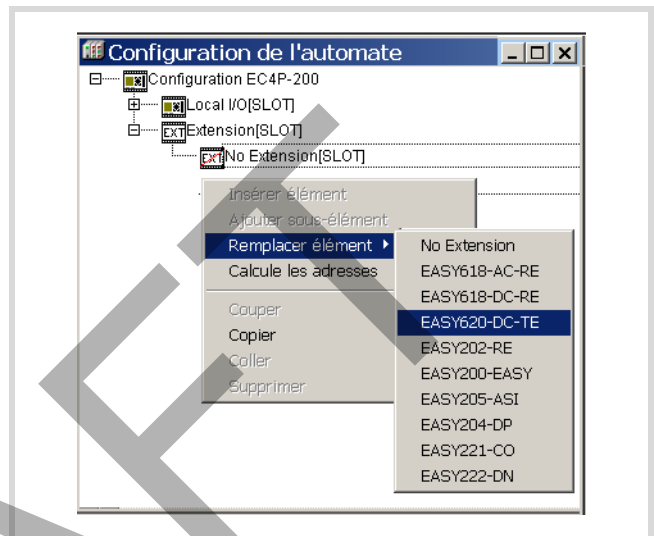


Figure 38 : Sélection d'un appareil d'extension

- ▶ Cliquez sur le symbole + placé devant le nouveau dossier, afin d'afficher les entrées, les sorties et les entrées de diagnostic.

9 Fonctionnement

Généralités

Synoptique des capacités mémoire

Vous pouvez disposer des modules mémoire/programme maximum suivants :

Programme (Code)	256 Koctets
Variables globales (Global)	224 Koctets
Mémoire de données (Memory)	16 Koctets
Image des entrées (Input)	4 Koctets
Image des sorties (Output)	4 Koctets
Variables rémanentes (Retain)	8 Koctets
Nombre max. de modules (UOP ; POU, en anglais)	2000 env.

→ Si une variable est déclarée comme RETAIN dans un module fonctionnel (FB en anglais), toutes les variables de ce module passent à l'état RETAIN.

Liste des différentes mémoires

L'automate possède les mémoires suivantes :

- Mémoire de travail (SRAM), non secourue par pile.
 - Exemple de contenu : programmes, données
- Mémoire système (FLASH), secourue par pile.
 - Exemple de contenu : projet de démarrage (bootproject)
- Module mémoire
 - Exemple de contenu : projet de démarrage (bootproject), système d'exploitation.

Comportement à la mise sous tension

L'automate ne possède pas de pile pour la sauvegarde de la mémoire de travail (qui comporte le programme). Pour sauvegarder le programme en cas de coupure de tension, vous devez générer un projet de démarrage (bootproject) de ce programme ; ce projet est stocké dans la mémoire système (qui est protégée contre les coupures de tension).

Après la mise sous tension, l'unité centrale (UC) procède à un auto-test du système. En cas de défaut, les diodes RUN/STOP/SF et CAN/NET clignotent (clignotement rouge). Après un auto-test exempt de défaut, l'automate contrôle si :

- Une mise à jour du système d'exploitation existe sur un module mémoire enfiché. Le cas échéant, il convient de la charger.
- un projet de démarrage est disponible. Si c'est le cas, ce projet est alors chargé dans la mémoire de travail de l'automate et lancé en fonction du comportement paramétré. S'il n'existe aucun projet de démarrage, l'automate demeure à l'état NOT READY.

Comportement à la mise sous tension avec un projet de démarrage sur le module mémoire

Lors de la mise sous tension de l'automate, tout projet de démarrage présent sur le module mémoire a priorité sur un projet enregistré dans la mémoire système. Dans le cas de projets de démarrage différents, c'est le projet de démarrage du module mémoire qui est copié dans la mémoire système puis exécuté. L'opération de copie rallonge de plusieurs secondes la phase de mise en route de l'automate.

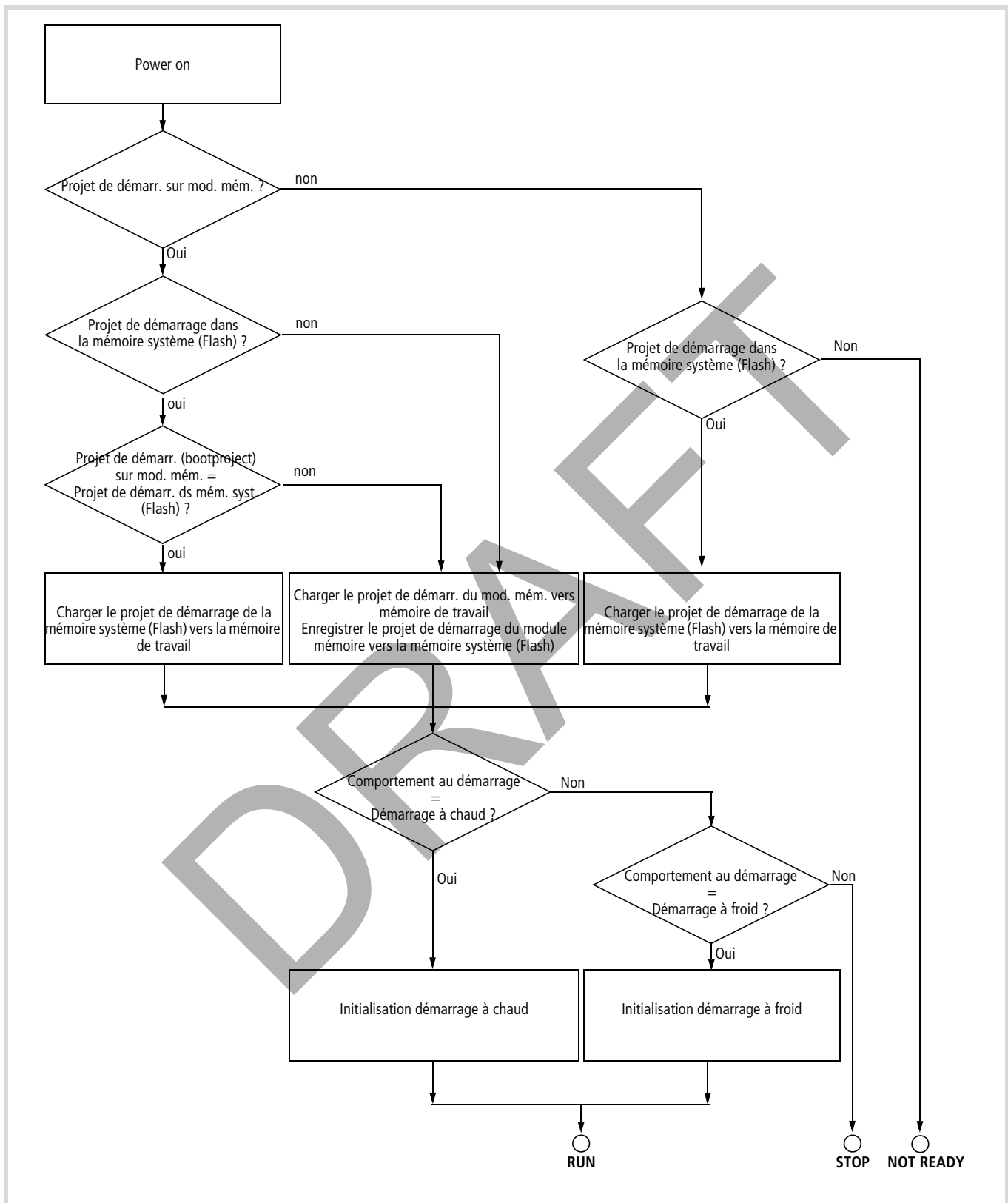


Figure 39 : Comportement à la mise sous tension avec projet de démarrage

Réglage du comportement au démarrage dans le logiciel de programmation

Procédez au réglage du comportement au démarrage afin de définir celui de l'automate à la mise sous tension.

Vous pouvez procéder à ce réglage dans le configurateur de l'automate ou à l'aide des éléments de commande de l'automate. Les possibilités de réglage ne sont pas priorisées. C'est la dernière saisie qui est valable.

Dans le configurateur de l'automate, activez l'onglet "Paramètres généraux" et sélectionnez dans la zone de liste la condition de démarrage souhaitée.

- HALT
- RESET WARM
- RESET COLD.

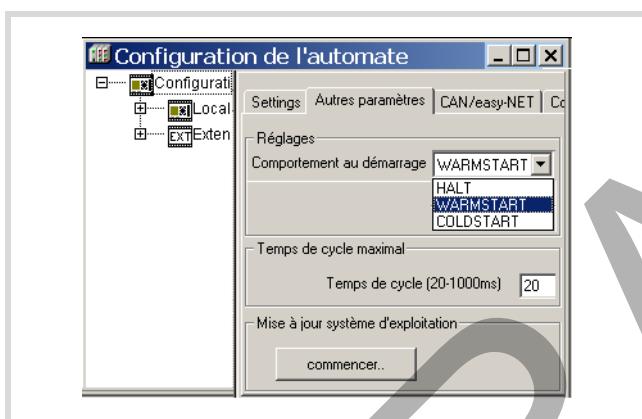


Figure 40 : Définition du comportement au démarrage

Programme START/STOP

Lancement du programme (STOP → RUN)

Vous avez deux possibilités pour lancer le programme :

- Sélectionner la commande START en mode En ligne (après le chargement d'un programme, par exemple).
- Utiliser les éléments de commande de l'automate.
 - Dans le Menu principal, au niveau du menu Programme, sélectionnez "START".

Comportement après coupure/interruption de la tension d'alimentation

La coupure ou l'interruption de la tension entraîne l'interruption immédiate du cycle du programme. Le traitement du programme n'est pas poursuivi jusqu'à la fin du cycle. Il n'est pas non plus poursuivi au retour de la tension. Le traitement reprend au tout début du programme. Conséquences possibles : certaines données rémanentes (des variables au format double-mot, par exemple) peuvent présenter des incohérences, selon l'emplacement où a été interrompu le programme !

Si des données incohérentes sont incompatibles avec votre application, vous pouvez par exemple recourir à une alimentation en courant avec sauvegarde par pile, donc protégée contre les coupures.

En cas de disparition de la tension, toutes les sorties sont positionnées à 0 ou coupées.

Le comportement des variables rémanentes en fonction de la condition définie pour le démarrage est indiqué dans le tableau 9.

Le démarrage de l'automate s'opère conformément aux réglages définis dans le configurateur de l'automate : → figure 40.

Tableau 9 : Comportement des variables au démarrage

Conditions de démarrage	Type de variable	
	Non rémanente	Rémanente (Retain)
RESET COLD	Activation des valeurs initiales	
RESET WARM	Activation des valeurs initiales	Conservation des valeurs
Charger et lancer le programme en mode En ligne	Activation des valeurs initiales	
Start/Stop/Start...	Conservation des valeurs	

Arrêt du programme (RUN → STOP)

Vous avez deux possibilités pour stopper le programme :

- Sélectionner la commande STOP en mode En ligne.
- Utiliser le menu de l'automate.
 - Dans le Menu principal, au niveau du menu Programme, sélectionnez STOP.

Lorsque vous exécutez la commande STOP, l'UC passe à l'état STOP dès que le cycle du programme est achevé. Les sorties sont positionnées à 0.

Démarrage/Arrêt du programme à l'aide d'interrupteurs externes

Vous pouvez lancer ou stopper le traitement du programme à l'aide d'un interrupteur externe amené au niveau d'une entrée. Quelques instructions de programme supplémentaires sont nécessaires : reportez-vous à cet égard à l'exemple indiqué dans l'annexe (→ page 104). Pour le lancement, vous avez besoin de la fonction SysStartPlcProgram et pour l'arrêt, de la fonction SysStopPlcProgram, toutes deux issues de la bibliothèque SysLibPlcCtrl.lib.

Dans ce cas, le comportement au démarrage de l'automate doit être positionné sur WARMSTART dans le configurateur de l'automate, sous <Autres paramètres → Réglages>.

Il vous est également toujours possible de positionner l'automate à l'état START ou STOP via le PC, en mode En ligne.

Traitement du programme et temps de cycle du système

Le programme utilisateur fait l'objet d'un traitement cyclique. Avant chaque cycle du programme, l'image des entrées est lue et, à la fin du cycle, l'image des sorties vient s'inscrire au niveau des sorties physiques.

Le temps de cycle dépend de la longueur et de la structure du programme utilisateur. Pour permettre une réaction rapide à certains événements, il est possible de programmer des routines qui sont lancées lors de la survenue d'événements système : → paragraphe « Événements système », page 49.

Surveillance du temps de cycle

Le temps de cycle du programme utilisateur fait l'objet d'une surveillance. Si le temps de cycle dépasse le temps paramétré, l'automate est positionné à l'état STOP et les sorties sont coupées.

Le réglage du temps max. admissible s'opère dans le configurateur de l'automate, sous "Autres paramètres". La valeur minimale est 20 ms (valeur par défaut) et la valeur maximale est 1 000 ms.

Reset (remise à zéro)

Vous pouvez procéder à une remise à zéro via le PC en mode En ligne ou via le menu de l'automate. Pour cela, sélectionnez l'option menu au niveau du configurateur ou du menu de l'automate.

Le menu comporte les ordres de RAZ suivants :

Configurateur (menu En ligne)	Menu de l'automate
Reset à chaud	Reset warm
Reset à froid	Reset cold
Reset origine	DELETE-> RESET HARD

Reset à chaud (Reset warm)

- Le programme est stoppé.
- Les variables non rémanentes sont initialisées, tandis que les variables "Retain" (rémanentes) sont conservées.
- Il est possible de relancer le programme.

Reset à froid (Reset cold)

- Le programme est stoppé.
- Toutes les variables sont initialisées.
- Il est possible de relancer le programme.

Reset origine (Reset Hard)

- Le programme situé dans la mémoire de travail est effacé, de même que le projet de démarrage (bootproject) situé dans la mémoire système de l'automate.
- Si le module mémoire est enfiché :
 - L'effacement concerne toutes les données spécifiques au projet enregistrées sur le module mémoire, le système d'exploitation et le projet de démarrage.
 - Toutes les données spécifiques à l'utilisateur ainsi que le fichier Startup.ini restent inchangés.
- L'automate est positionné à l'état NOT READY.

Rétablissement des réglages usine (factory set)

La commande navigateur "factoryset" ou l'option menu <SYSTEM → FACTORY SET> de l'automate vous permet de procéder à un "Reset origine" (→ paragraphe « Reset origine (Reset Hard) »). Le fichier Startup.ini situé sur le module mémoire et les paramètres système situés dans l'automate sont par ailleurs effacés. Après un démarrage, l'automate travaille de nouveau avec les données STARTUP. Les interfaces sont initialisées avec leurs valeurs par défaut.

Comportement des variables après une RAZ (Reset)

Remise à zéro (Reset)	Type de variable	
	Non rémanente	Rémanente (Retain)
Reset à chaud (Reset warm)	Activation des valeurs initiales	Conservation des valeurs
Reset à froid (Reset cold)	Activation des valeurs initiales	
Reset origine (Reset Hard)1)	Plus aucune variable n'est présente ; le programme est effacé.	

1) Après un "Reset origine" (Reset-Hard), il convient de recharger le programme. Vous pouvez ensuite procéder au redémarrage de l'automate, en mode En ligne.

Test et mise en service

L'automate gère les fonctions de test et de mise en service suivantes :

- Mode Point d'arrêt/Pas à pas
- Mode Cycle unique
- Forçage
- Modification en ligne
- Affichage dynamique (power flow).

→ Remarques valables pour le mode Point d'arrêt/Pas à pas et le mode Cycle unique :

N'utilisez pas ces fonctions de mise en service dans les programmes de routine (pour le démarrage, par exemple). Un comportement défectueux peut conduire à un état non défini de l'automate.

S'il s'avère impossible de procéder à l'exécution de ces fonctions de mise en service, activez la fonction de débogage (état par défaut) : dans le menu <Projet → Options → Options de compilation>, cliquez sur le champ "Débogage".

Mode Point d'arrêt/Pas à pas

Des points d'arrêt (Breakpoints) peuvent être activés au sein du programme utilisateur. Lors de l'exécution d'une instruction dotée d'un point d'arrêt, le programme est stoppé à cet emplacement. Il est alors possible d'exécuter le programme en mode Pas à pas. La surveillance du temps de cycle est désactivée.



Avertissement !

Les sorties activées à ce moment-là le restent !

Mode Cycle unique

En mode Cycle unique, un seul cycle de programme est exécuté en temps réel. Les sorties sont libérées durant le cycle. La surveillance du temps de cycle est active.



Avertissement !

Les sorties activées à ce moment-là le restent !

Forçage de variables et d'entrées/sorties

Toutes les variables d'un programme utilisateur peuvent être positionnées par forçage sur des valeurs fixes à l'aide de l'option "Forcer". Les sorties locales forcées ne sont connectées aux périphériques qu'à l'état RUN.



Le forçage des entrées/sorties raccordées via le bus de terrain CANopen n'est pas possible.

Affichage d'état dans le logiciel de programmation

- L'état du signal des entrées booléennes physiques est affiché, aussi bien à l'état RUN de l'UC qu'à l'état STOP.
- L'état du signal des entrées booléennes physiques n'est affiché qu'à l'état RUN de l'UC ; à l'état STOP, ces entrées sont repérées par FALSE.
- Toutes les autres variables sont affichées avec la valeur actuelle de la variable.

Compteurs rapides (Counter)

Pour chaque fonction de comptage, l'entrée de l'automate destinée au traitement des impulsions est présentée au paragraphe « Raccordement d'un générateur d'impulsions/d'un codeur incrémental » (page 23). Les autres entrées/sorties d'un compteur sont représentées dans la Configuration de l'automate à l'aide de symboles (tels que Reset, par exemple) une fois que vous avez sélectionné le type de compteur (32BitCounter, par exemple) ; (→ chapitre « Configuration des entrées/sorties (I/O) », page 39). Le paramétrage des entrées/sorties symboliques doit être effectué dans le programme (fonctionnement reposant sur des symboles).

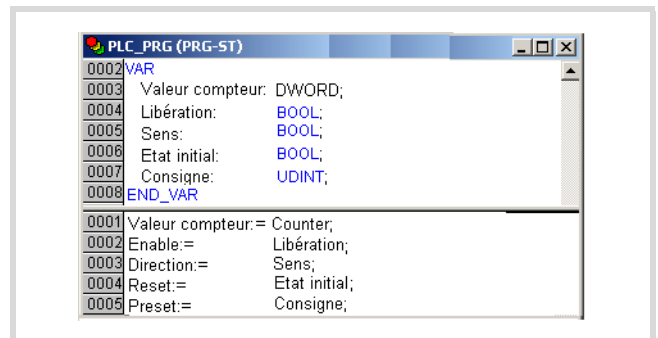


Figure 41 : Programmation des entrées/sorties d'un compteur 32 bits

Il convient à cet égard de tenir compte du fait que les opérations telles que la RAZ, par exemple, sont exécutées via le registre image du programme. La sortie "Reset AT%QX1.2" n'est activée qu'à la fin du programme.

Vous pouvez également atteindre des temps d'accès plus courts en contournant le registre image, comme cela peut par exemple être nécessaire dans une routine d'interruptions. Dans ce cas, utilisez des blocs fonctionnels au lieu de travailler avec des symboles. La bibliothèque EC_Util2.lib propose pour chaque type de compteur un bloc fonctionnel. Le module fonctionnel conçu pour le compteur 32 bits présente les paramètres suivants :



Figure 42 : Module fonctionnel pour compteur 32 bits (32 Bit Counter)

Les entrées/sorties des blocs fonctionnels correspondent pour l'essentiel aux entrées/sorties indiquées dans la Configuration de l'automate.

→ Pour les compteurs 16 bits et 32 bits, vous devez programmer non seulement les entrées du module fonctionnel, mais également l'entrée "Enable" dans la Configuration de l'automate, afin de libérer le compteur.

Fonctions de comptage (entrées/sorties)

La description des fonctions des entrées/sorties ci-après vaut pour les E/S des blocs fonctionnels et de la Configuration de l'automate.

Compteur 32 bits (32 Bit Counter)

Vous disposez d'une seule fonction de type "compteur 32 bits" (32 Bit Counter). Le générateur d'impulsions doit être relié à l'entrée externe I1. Il reçoit les impulsions avec une fréquence max. de 50 kHz. L'UC compte ces impulsions et les met à disposition sous forme de valeur réelle (= Counter). Vous pouvez scruter la valeur réelle au sein du programme utilisateur. C'est dans le programme utilisateur, par l'activation de la sortie "Direction", que vous décidez si la valeur réelle sera incrémentée ou décré- mentée lors de la survenue d'une impulsion de comptage.

→ Lorsque la valeur réelle est égale à la valeur de consigne, il est possible de générer une interruption. Cette dernière a pour effet de déclencher le traitement d'une routine. Pour cela, vous devez activer l'interruption dans la configuration des tâches et affecter la routine → paragraphe « Traitement des interruptions », page 52.

Le programme vous permet de définir les propriétés suivantes pour le compteur :

- Libération (Enable) :

- TRUE : les impulsions sont comptées.
- FALSE : les impulsions ne sont pas comptées.
- Un signal "1" à l'entrée "Enable" provoque la libération du compteur: Les impulsions entrantes sont comptées. Un front 0 → 1 au niveau du signal Enable entraîne le positionnement à 0 de la valeur réelle et la prise en compte de l'état à l'entrée "Direction" et à l'entrée "Mettre à disposition consigne" (Preset). Aucun changement de direction en cours de fonctionnement n'est détecté.

- Direction :
 - Incrémentation (Direction = FALSE) : le compteur compte jusqu'à la valeur de consigne prédéfinie (PRESET). Une fois la valeur de consigne atteinte, l'interruption paramétrée est activée et débouche sur une routine (→ page 52). Lors de l'impulsion de comptage suivante, le compteur recommence à 0.
 - Décrémentation (Direction = TRUE) : lors de la première impulsion de comptage, la valeur réelle passe de 0 à la valeur de consigne. Si une interruption est paramétrée, la routine qui lui est associée est appelée (→ page 52). A chaque nouvelle impulsion, la valeur réelle est décré- mentée, jusqu'à ce qu'elle atteigne 0. Lors de l'impulsion de comptage suivante, la valeur de consigne est reprise en compte et la routine à nouveau appelée.
- Remise à zéro (Reset) :
 - Un front 0 → 1 au niveau de l'entrée "Reset" provoque – indépendamment de l'état du signal de libération – le positionnement à 0 de la valeur réelle et la prise en compte de la direction et de la valeur de consigne.
- Mise à disposition de la valeur de consigne (Preset).

Exemple : programme avec module fonctionnel (FB en anglais) pour compteur 32 bits (32 Bit Counter)

```

PLC_PRG (PRG-ST)
0002 VAR
0003   COMPT:      Acc32BitCounterI
0004   Sens:       BOOL;
0005   Etat initial:  BOOL;
0006   Consigne:   UDINT;
0007   Valeur compteur: UDINT;
0008   ERREUR:    UINT;
0009   Libération:  BOOL;
0010 END_VAR
0001 Enable:=Libération;
0002
0003 COMPT(
0004   xDrection:= Sens,
0005   xReset:=   Etat initial,
0006   udiPreset:= Consigne,
0007   udiCounter=>Valeur compteur,
0008   uiError=> ERREUR);
    
```

Figure 43 : Programme avec module fonctionnel (FB en anglais) pour compteur 32bits (32 Bit Counter)

Compteur 16 bits (16 Bit Counter)

La fonction de type "compteur 16 bits" est disponible en double exemplaire : l'ensemble correspond à la fonction du compteur rapide (32 bits). Pour permettre de différencier les deux compteurs 16 bits, les opérandes symboliques sont dotés d'un numéro : 0 ou 1. Les opérandes dotés du numéro 0 commandent les impulsions de comptage présentes à l'entrée I1 ; ceux dotés du numéro 1 procèdent à l'acquisition des impulsions de comptage de I2.

Entrées externes :

Numéro du compteur	Entrée d'impulsion
0	I1
1	I2

Le numéro du compteur est visible à partir des opérandes symboliques du configurateur de l'automate, dans le dossier "16Bit Counter".

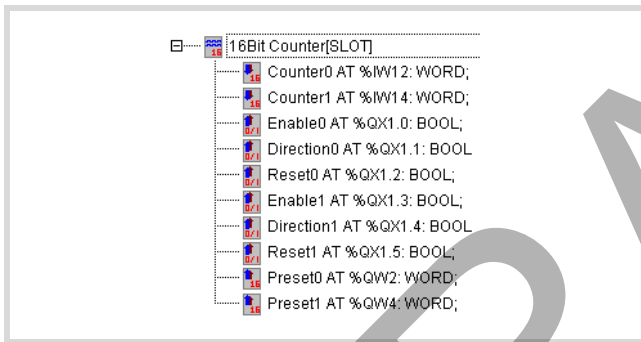


Figure 44 : Entrées/Sorties des compteurs 16 bits 0 et 1

→ Lorsque la valeur réelle est égale à la valeur de consigne, il est possible de générer une interruption entraînant le traitement d'une routine. Pour cela, vous devez activer l'interruption dans la configuration des tâches et affecter la routine → paragraphe « Traitement des interruptions », page 52.

Compteur de valeurs incrémentales (Incremental Input)

Vous disposez d'une seule fonction de type "compteur de valeurs incrémentales". Au niveau des entrées externes I1 et I2 sont amenés les signaux incrémentaux A et B du générateur, et à l'entrée I3 le signal de référence que produit le générateur une fois par rotation. Au niveau de l'entrée I4 est raccordé le fin de course de prise d'origine (qui, à l'état fermé, constitue la fenêtre de référence dans laquelle est traité le signal de référence).

En vue de déterminer le sens de comptage, les signaux incrémentaux A et B sont déphasés de 90 degrés. Les fronts montants et descendants sont analysés (4 analyses). La fréquence d'entrée maximale est de 40 kHz. La fréquence totale est par suite de 160 kHz. Le compteur ne génère aucune interruption.

Les signaux suivants vous permettent de commander le compteur et de l'adapter à l'application. Au sein du programme, vous scrutez les entrées de signaux et positionnez les sorties de signaux. La désignation de chaque signal est déterminée au niveau de la configuration de l'automate.

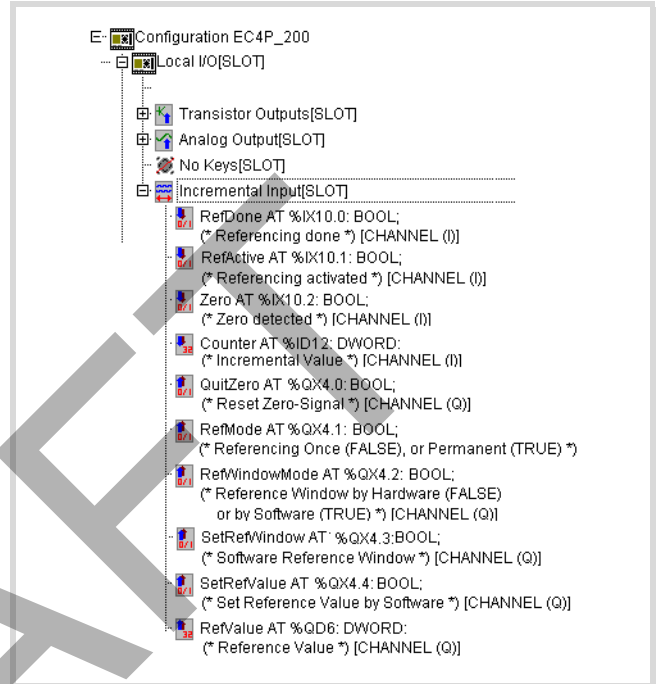


Figure 45 : Signaux d'entrée/sortie du compteur de valeurs incrémentales

Signification des signaux d'entrée/sortie (I/Q)

Signal	I/Q	Signification
RefDone	I	Référencement exécuté (message en retour de SetRefWindow)
RefActive	I	Référencement activé (message en retour de SetRefWindow ou I4)
Zero	I	Passage par zéro de la valeur réelle
Counter	I	Valeur réelle du compteur
QuitZero	Q	Acquittement du signal ZERO
RefMode	Q	Nombre de référencements 0 = une seule fois 1 = en permanence
RefWindowMode	Q	Activation de la fenêtre de référence par 0 = entrée externe I4 1 = au sein du programme, avec "SetRefWindow"
SetRefWindow	Q	Activation de la fenêtre de référence, si "RefWindowMode" = 1
SetRefValue	Q	La valeur de référence dépasse la valeur réelle (Reset).
RefValue	Q	Valeur de référence

Synoptique des signaux des entrées/sorties (I/Q)

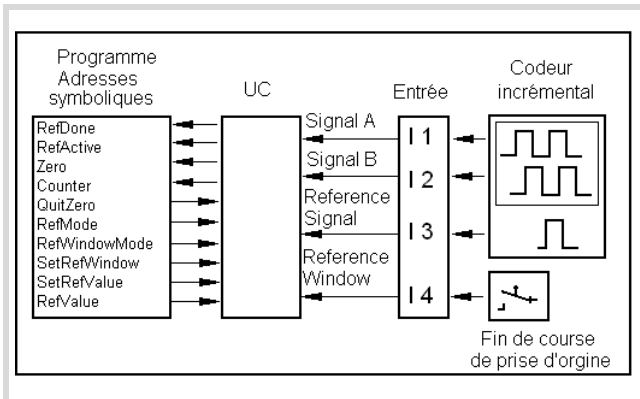


Figure 46 : Entrées/Sorties du compteur de valeurs incrémentales

Fonctions des signaux d'entrée/sortie

Le changement d'état HALT → RUN de l'UC entraîne la libération du compteur: Les impulsions entrantes sont alors comptées.

SetRefValue (Reset)

En cas de front 0 → 1 au niveau de l'entrée, la valeur réelle est écrasée par la valeur présente à l'entrée "RefValue".

Counter (valeur réelle)

La valeur réelle du compteur se trouve à l'entrée "Counter".

Zero (passage par zéro)

Lorsque la valeur réelle atteint la valeur 0, la sortie Zero est activée. Elle le reste jusqu'à ce qu'elle soit acquittée par un front 0 → 1 à l'entrée "QuitZero".

RefWindowMode (activer la fenêtre de référence)

Ce signal vous permet de décider si le signal destiné à activer la fenêtre de référence s'opérera via l'entrée I4 ou via le programme utilisateur, à l'aide du signal "SetRefWindow".

RefMode (type de référencement)

Ce signal placé au niveau de l'entrée vous permet de décider si le référencement aura lieu une seule fois (0 à l'entrée) ou de manière permanente (1 à l'entrée). La valeur réelle est écrasée par la valeur de référence lorsque la fenêtre de référence est activée et qu'une impulsion de référence survient à l'entrée I3. Selon l'indication du signal RefMode, cette opération a lieu une seule fois (lorsque les conditions sont réunies après le démarrage de l'automate) ou de manière permanente (lors de chaque impulsion de référence dans la fenêtre de référence).

Référencement

Dans de nombreuses commandes de positionnement, un point de référence est choisi au début du positionnement. Pour cela, un chariot porte-outil (par exemple) va à sa position initiale. Dans cette position, un "fin de course de prise d'origine" (dont le signal est amené à l'entrée I4) est mécaniquement fermé. Le signal du fin de course peut également être remplacé par le signal "SetRefWindow", que l'utilisateur peut commander dans le programme. En retour, le signal "RefActive" est activé. Pour indiquer avec précision la position du chariot, un codeur incrémental relié à ce dernier génère une impulsion de référence. Celle-ci est détectée à l'entrée I3, lorsque le fin de course de prise d'origine est fermé et que la fenêtre de référence est ouverte. L'impulsion de référence entraîne l'écrasement du compteur avec la valeur de référence que vous avez indiquée dans la configuration de l'automate. "RefActive" est remis à zéro et "RefDone" est activé jusqu'à la nouvelle ouverture de la fenêtre de référence.

→ Choisissez la taille de la fenêtre de référence de manière que le signal de référence n'apparaisse qu'une seule fois, mais qu'il puisse cependant être analysé fiablement.

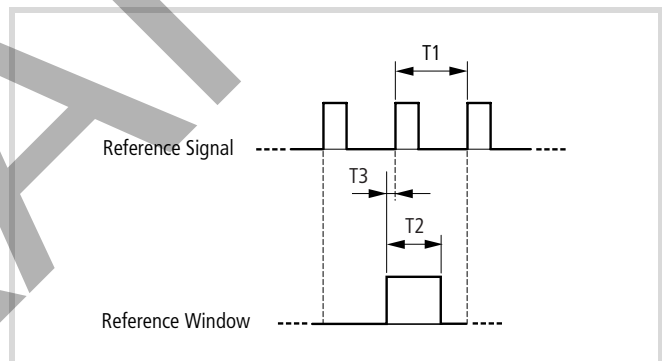


Figure 47 : Relation entre signal de référence et fenêtre de référence

T1 Temps de répétition entre deux impulsions de référence consécutives lors d'une rotation du codeur incrémental

T2 Durée maximale admissible de la fenêtre de référence. Elle doit impérativement être inférieure à T1, de manière à éviter toute détection d'une seconde impulsion de référence.

T3 Sa longueur doit être déterminée de manière à permettre la détection fiable du front L/H de l'impulsion de référence.

T2 et T3 dépendent de la fréquence de succession des impulsions de l'impulsion de référence et doivent éventuellement être déterminés par voie expérimentale, selon l'application.

Evénements système

Les événements système sont les suivants :

START	Lancement du programme utilisateur (démarrage à froid et démarrage à chaud)
RESET COLD	Démarrage à froid du programme utilisateur
RESET WARM	Démarrage à chaud du programme utilisateur
STOP	Arrêt du programme utilisateur (non valable pour un dépassement du temps de cycle ou un "chien de garde" matériel)
IO-Interrupt 1, 2, 3, 4	Changement de tension aux entrées I1, I2, I3, I4
Counter-Interrupt1	Valeur réelle = valeur de consigne, pour un compteur de type "16 Bit Counter 0"
Counter-Interrupt2	Valeur réelle = valeur de consigne, pour un compteur de type "16 Bit Counter 1" ou "32 Bit Counter"
TIMER-INTERRUPT	Un temporisateur paramétré par l'utilisateur déclenche une interruption.

Vous pouvez réagir aux événements système de l'automate en élaborant une routine (UOP) qui sera exécutée lorsqu'un événement particulier surviendra. L'exécution est surveillée sur le plan temporel. C'est la valeur maximale admissible paramétrée pour le temps de cycle qui est utilisée comme base de temps.

START, COLDSTART, WARMSTART, STOP

Lorsque l'événement considéré survient (Warmstart, par exemple) au niveau de l'automate, une interruption est générée (→ page 52) et appelle la routine qui lui est affectée. C'est dans la Configuration des tâches que vous devez procéder à l'affectation.

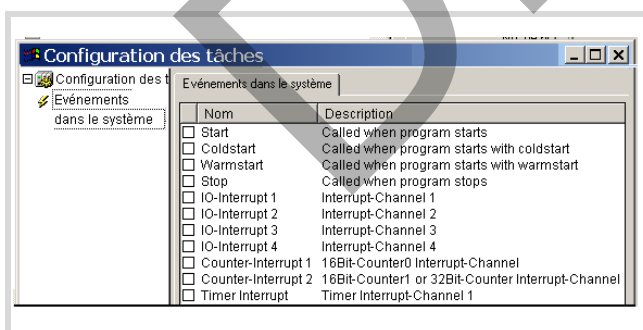


Figure 48 : Evénements système

Entrées d'interruption I1 à I4

Les entrées I1 à I4 peuvent être configurées en tant qu'entrées d'interruption. Un front à l'entrée génère une interruption (→ page 52) qui appelle la routine associée.

- Déterminez d'abord le type de front du signal d'entrée dans le configurateur de l'automate.
- Dans la configuration des tâches, affectez la routine à l'entrée.

Les entrées sont priorisées. I1 présente la priorité maximale ; suivent ensuite I2, I3 et I4.

Interruption au niveau d'un compteur (Counter-Interrupt)

Si vous utilisez la fonction "compteur rapide", l'automate compare en permanence la valeur réelle avec la valeur de consigne du compteur. Si les deux sont identiques, une interruption est générée (→ page 52) et appelle la routine (UOP) que vous avez élaborée.

Pour cela, vous devez d'abord déterminer le type de compteur dans le configurateur de l'automate. Dans la configuration des tâches, procédez ensuite à l'affectation entre l'entrée qui reçoit les impulsions de comptage et l'UOP.

Interruption au niveau d'un temporisateur (Timer-Interrupt)

Vous pouvez élaborer une routine qui sera appelée dans un intervalle fixe. Le lancement de la fonction `TIMERINTERRUPTENABLE` s'opère à l'aide d'une variable booléenne ou d'une entrée externe. L'affectation de la routine à l'interruption temporisateur (Timer-Interrupt) s'opère dans la configuration des tâches. La durée de la période peut être réglée dans la plage 500 – 2 500 000 microsecondes. Pour programmer la durée de la période, insérez dans votre programme utilisateur la fonction `TIMERINTERRUPTENABLE` située dans la bibliothèque `EC_Util.lib`.



Figure 49 : Fonction `TimerInterruptEnable`

Indiquez la durée de la période au niveau de l'entrée "dwTimerTickUS".

La valeur est prise en compte lors du démarrage du temporisateur et ne peut pas être modifiée durant le fonctionnement de ce dernier. En cas de passage de la valeur en deçà de 500 ou au-delà de 2 500 000, la fonction délivre en retour la valeur `FALSE` et le temporisateur n'est pas activé.

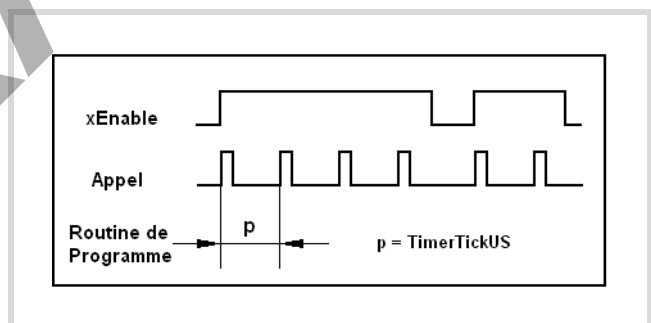


Figure 50 : Déroulement périodique de la routine

Si la durée de période de 2 secondes (par exemple) doit être activée par l'entrée externe `I0.0`, vous devez écrire dans le programme utilisateur la ligne de programme suivante :

```
TimerInterruptEnable(%IX0.0,2000000)
```

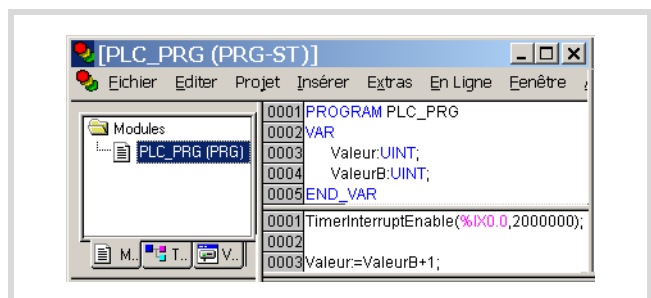


Figure 51 : Insertion d'une fonction dans le programme

Exemple

• Elaboration d'un programme avec appel de fonction
Elaborez un programme avec la fonction
TIMERINTERRUPTENABLE selon la figure 51.

- Elaboration d'une routine
- ▶ Dans le répertoire "Ressources", ouvrez le sous-répertoire "Configuration des tâches", en effectuant un double-clic.
- ▶ Cliquez sur le dossier "Événements système". L'onglet "Événements système" est actif.
- ▶ Activez l'interruption temporisateur en cliquant sur la case située à gauche, à côté du nom "Timer-Interrupt".
- ▶ Dans le champ "UOP appelée", indiquez le nom de la routine ("Int_Tempo", par exemple).

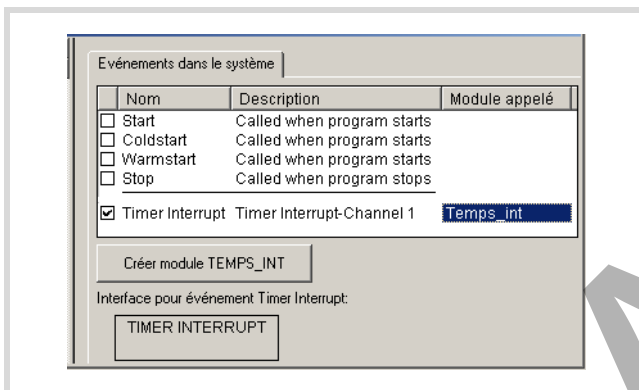


Figure 52 : Elaboration d'une routine

- ▶ Cliquez encore une fois sur le nom "Timer-Interrupt". Le bouton "Créer un module" devient alors actif et affiche le nom de l'UOP.
- ▶ Cliquez sur ce bouton. Dans la fenêtre MODULES, sous l'UOP "PLC_PRG", un dossier (UOP) dénommé "Int_Tempo" est inséré.
- ▶ Ouvrez l'UOP et écrivez votre routine :

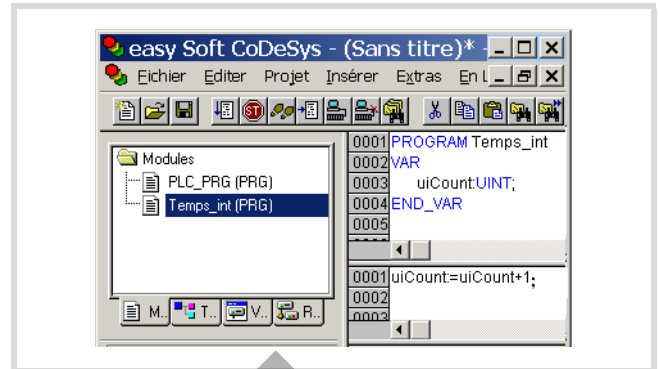


Figure 53 : Ecriture d'une routine

En cas d'application d'une tension à l'entrée IX0.0, l'UOP "Int_Tempo" est appelée périodiquement et la variable "uiCount" se trouve incrémentée.

→ L'interruption peut être interrompue par une interruption système de priorité supérieure. La surveillance du temps de cycle est active tout au long de l'exécution de l'interruption temporisateur (Timer-Interrupt).

La survenue d'interruptions temporisateur trop fréquentes peut conduire au dépassement du temps de cycle programme présélectionné. Dans ce cas, l'automate passe de l'état RUN à l'état STOP.

L'interruption temporisateur (Timer-Interrupt) peut être stoppée et lancée à partir du programme utilisateur. Pour ce faire, vous disposez des fonctions "DisableInterrupt" et "EnableInterrupt" situées dans la bibliothèque EC_UTIL.lib.

Traitement des interruptions

Lorsqu'une interruption survient, le programme utilisateur est interrompu et la routine reliée à l'événement système fait l'objet d'un traitement. Les sources d'interruption sont indiquées au niveau de la figure 54.

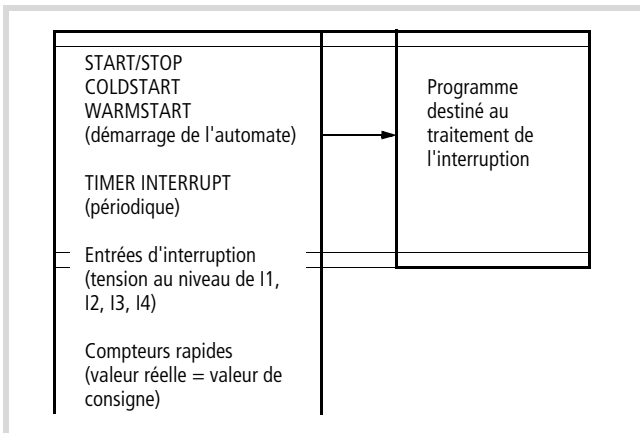


Figure 54 : Sources d'interruption

→ L'exécution des routines est surveillée du point de vue du temps.

La routine appelée par l'interruption peut être interrompue par une nouvelle interruption (autre canal).

Si une nouvelle interruption suit celle en cours (même canal), la nouvelle interruption n'est exécutée qu'à l'issue de celle en cours.

Les interruptions sont libérées à l'état RUN de l'UC et bloquées à l'état STOP. Les sources d'interruption non libérées dans la configuration ne provoquent aucune interruption.

Vous pouvez bloquer les entrées d'interruption I1...I4 et l'interruption temporisateur (Timer-Interrupt) à partir du programme, puis les libérer à nouveau. Vous disposez pour cela des fonctions "DisableInterrupt" et "EnableInterrupt". Un paramètre d'appel détermine si le blocage/la libération portera sur une seule ou sur l'ensemble des interruptions. La libération d'une interruption bloquée doit s'opérer avec le même paramètre que pour le blocage.

Les deux fonctions "DisableInterrupt" et "EnableInterrupt" font partie intégrante de la bibliothèque "EC_Util.lib". Dans le cas où elle n'est pas encore présente, vous devez intégrer cette bibliothèque dans votre projet – à partir du gestionnaire de bibliothèques du logiciel de programmation.

DisableInterrupt : cette fonction vous permet de désactiver à partir du programme utilisateur une interruption physique paramétrée.

EnableInterrupt : cette fonction vous permet de libérer et d'activer à nouveau l'interruption physique préalablement désactivée.

Les étapes du traitement d'une interruption

► Définissez les propriétés de l'interruption :

Comportement au démarrage	Choix du type
TIMER INTERRUPT	Appeler la fonction TIMERINTERRUPTENABLE
Entrées d'interruption	Déterminer les fronts
Compteurs rapides	Choisir le type

► Elaborez la routine (UOP).

Il convient d'élaborer pour l'UOP existante PLC_PRG une autre routine (UOP) de type PRG qui appelle une interruption.

► Affectez à la routine une source d'interruption :

- Pour cela, appelez le configurateur de l'automate et cliquez sur Configuration des tâches | Événements système. Les sources d'interruption (noms) sont indiquées sous l'onglet "Événements système" et une zone de saisie libre est destinée au nom de l' "UOP appelée".
- Libérez l'interruption en cliquant sur la case située à côté de l'interruption souhaitée et saisissez le nom de l'UOP sur la même ligne. Pour plus de détails, reportez-vous au point Exemple de traitement d'une interruption.

Exemple de traitement d'une interruption

Une UOP "PLC_PRG" doit être traitée en permanence. Une autre routine (UOP) "Fastprog" doit par ailleurs être traitée lorsqu'un front montant (L → H) à l'entrée I3 génère une interruption.

► Elaborez les UOP "PLC_PRG" et "Fastprog" selon la figure 55.

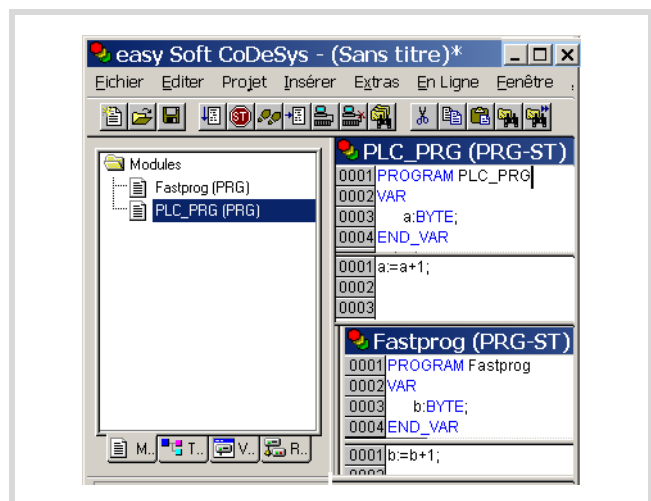


Figure 55 : Elaboration d'un programme

- Passez à la configuration de l'automate, cliquez sur le dossier Local I/O[SLOT] et ouvrez l'onglet "Autres paramètres".
- Affectez à l'entrée I3 le type "Front montant".

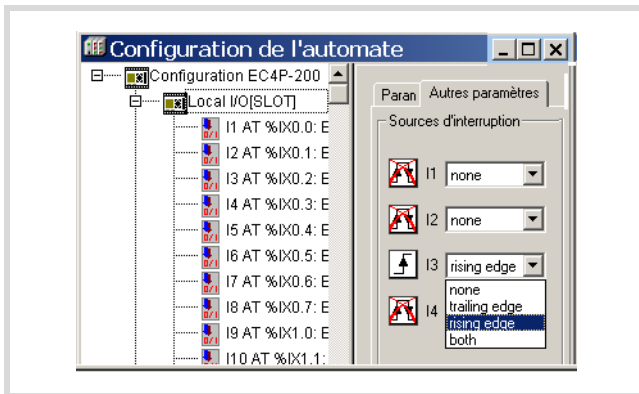


Figure 56 : Sélection d'un front d'interruption

- Passez à la Configuration des tâches et ouvrez le dossier "Événements système".

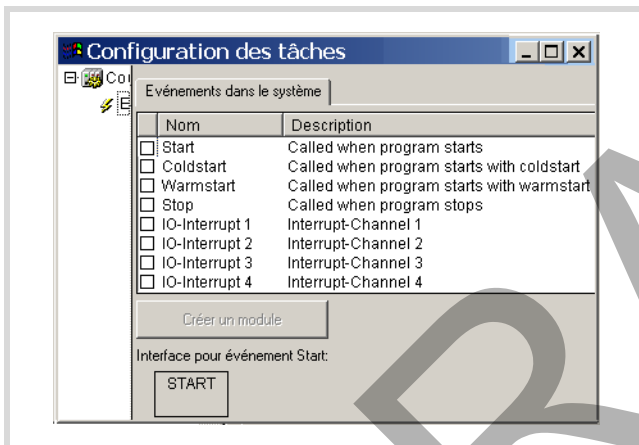


Figure 57 : Libération d'une interruption

- Libérez IO-Interrupt 3 en cliquant sur la case située à gauche, à côté du nom "IO-Interrupt3". La confirmation est visualisée par une coche.
- Sélectionnez la zone de la colonne "UOP appelée" et de la ligne "IO-Interrupt3".
- Positionnez le curseur sur la zone sélectionnée et actionnez la touche de fonction F2.

La fenêtre "Aide à la saisie" s'ouvre. Tous les programmes prédéfinis y figurent sous forme de liste :

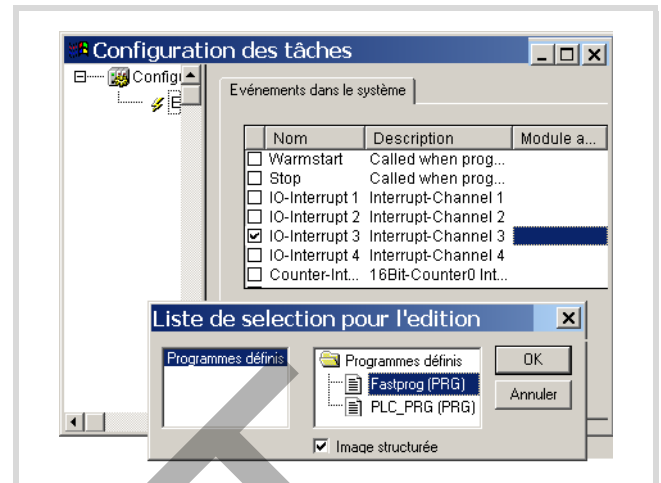


Figure 58 : Affectation Source d'interruption → UOP

- Sélectionnez l'UOP "Fastprog" et confirmez votre choix par OK.
- Enregistrez le projet. Vous pouvez à présent le tester.

A chaque front montant au niveau de l'entrée I3, la variable "b" est augmentée de 1.

Accès direct aux E/S

Les fonctions de la bibliothèque EC_Util.lib conçues pour l'accès direct aux E/S vous permettent d'accéder directement aux entrées et sorties locales de l'automate. L'accès a lieu immédiatement à partir du programme utilisateur, et non pas via le registre des entrées/sorties. L'accès n'est pas géré pour les entrées/sorties suivantes :

- Entrées/sorties des appareils d'extension
- Bits de diagnostic locaux
- Touches de fonction et de direction
- Entrées/sorties des appareils qui sont intégrés via un bus.

L'accès aux "compteurs rapides" peut s'opérer à l'aide des blocs fonctionnels de la bibliothèque EC_Util.Lib.

Description des fonctions

La fonction "ReadBitDirect" est décrite à titre d'exemple pour toutes les autres fonctions.

ReadBitDirect

Cette fonction vous permet de lire directement l'état d'un bit d'entrée. Ce dernier est mémorisé dans la variable indiquée par le pointeur "ptr_xValue" paramétré. La variable du pointeur n'est pas modifiée lorsqu'une erreur survient lors du traitement.

```

FUNCTION ReadBitDirect:UINT (* Returnvalue 0 or Errorcode > 0 *)
VAR_INPUT
  uiSlot :UINT; (* Slot 0..7 *)
  uiBit :UINT; (* Bitposition 0..63 *)
  ptr_xValue :POINTER TO BOOL; (* Pointer to read data value *)
END_VAR
VAR
  END_VAR
  
```

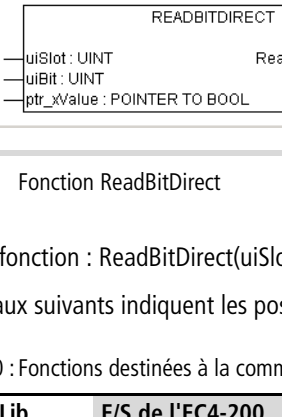


Figure 59 : Fonction ReadBitDirect

Appel de fonction : ReadBitDirect(uiSlot, uiBit, ptr_xValue)

Les tableaux suivants indiquent les possibilités d'accès.

Tableau 10 : Fonctions destinées à la commande des E/S

EC_UTIL.Lib	E/S de l'EC4-200			
	I1-I12 TOR	I7, I8, I11, I12 analogiques	Q1-Q8 TOR	QA1 analogiques
ReadBitDirect	Bits 0-11	–	–	–
ReadByteDirect	Octets 0+1	–	–	–
ReadWordDirect	–	Offsets 2,4,6,8	–	–
WriteBitDirect	–	–	Bits 0-7	–
WriteByteDirect	–	–	Octet 0	–
WriteWordDirect	–	–	–	Offset 2

Tableau 11 : Blocs fonctionnels (FB en anglais) pour "compteurs rapides"

EC_UTIL2.Lib	Compteurs rapides			
	Module fonctionnel	16 bits	32 bits	Incrémental
Acc16BitCounterDirect	Offset 0+1	–	–	–
Acc32BitCounterDirect	–	ok	–	–
AccIncremental InputDirect	–	–	ok	–

Code d'erreur en cas d' "accès direct aux E/S"

Dans la mesure du possible, toutes les fonctions vérifient la validité des paramètres d'appel. Lorsqu'une erreur survient, l'accès n'a pas lieu et un code d'erreur est émis.

Les valeurs en retour suivantes sont possibles :

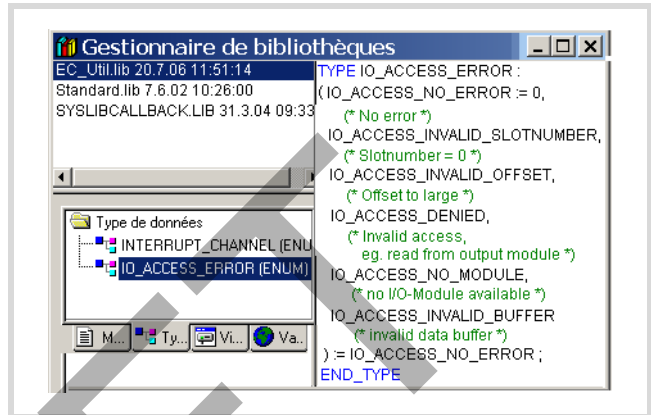


Figure 60 : Valeurs de retour des fonctions EC-UTIL.Lib

Tableau 12 : Synoptique des valeurs de retour

	IO_ACCESS_INVALID_SLOTNUMBER	IO_ACCESS_INVALID_OFFSET	IO_ACCESS_DENIED	IO_ACCESS_NO_MODULE	IO_ACCESS_INVALID_BUFFER
READBITDIRECT	X	X	X		X
READBYTEDIRECT	X	X	X		X
READWORDDIRECT	X	X	X		X
WRITEBITDIRECT	X	X	X		
WRITEBYTEDIRECT	X	X	X		
WRITEWORDDIRECT	X	X	X		
ACCESS16BITCOUNTERDIRECT		X	X		
ACCESS32BITCOUNTERDIRECT			X		
ACCESSINCREMENTALINPUTDIRECT			X		

Génération et transfert d'un projet de démarrage (bootproject)

L'UC procède au traitement du programme utilisateur enregistré dans la mémoire de travail. Du fait que la mémoire de travail n'est pas secourue par pile, le programme est effacé en cas de coupure de tension. Pour sauvegarder le programme de manière rémanente, créez un projet de démarrage (bootproject).

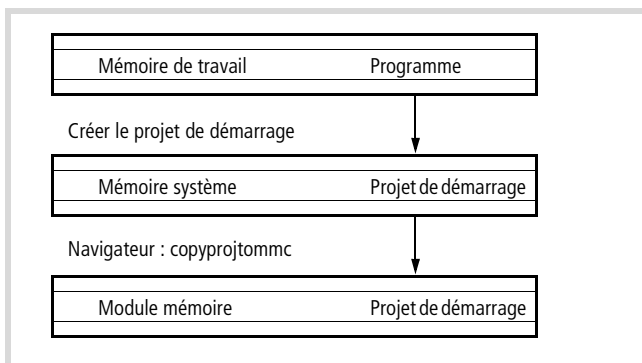


Figure 61 : Enregistrer le projet de démarrage

Vous pouvez créer le projet de démarrage en mode En ligne ou via le menu de l'automate. Le projet de démarrage est généré à l'aide du système d'exploitation actuel de l'automate.

En mode En ligne, les étapes suivantes sont nécessaires :

- ▶ Passez au menu "En ligne" et sélectionnez la commande "Accéder au système".
- ▶ A l'état RUN, vous êtes invité à arrêter l'automate.
- ▶ Sélectionnez la commande "Créer un projet de démarrage".

La demande de confirmation suivante s'affiche :



Figure 62 : Créer un projet de démarrage

- ▶ Répondez par "Oui" à cette question.

La fenêtre suivante s'affiche brièvement :



Figure 63 : Création d'un projet de démarrage

Le masquage automatique de la fenêtre indique que le projet de démarrage (bootproject) est créé. Vous pouvez alors procéder au redémarrage de l'automate.

Sauvegarde du projet de démarrage sur un module mémoire

Le projet de démarrage stocké dans la mémoire système (Flash) peut par ailleurs être sauvegardé sur un module mémoire. Cette opération peut s'effectuer par appel de la commande navigateur "copyprojtommc" en mode En ligne ou à l'aide des touches de commande de l'automate, par sélection à partir du menu principal : PROGRAM → BOOTPROJECT → FLASH → CARD.

Projet de démarrage et système d'exploitation (SE) sur module mémoire

Le projet de démarrage (bootproject) ne travaille qu'avec le système d'exploitation (SE) actif avec lequel il a été élaboré !

L'insertion du module mémoire avec un SE dans l'automate entraîne, après mise sous tension, l'actualisation du SE de l'automate et le chargement d'un projet de démarrage dans l'automate. Si le projet de démarrage n'a pas été créé avec ce SE, il ne sera pas reconnu par l'automate. Dans ce cas, chargez le programme et créez un nouveau projet de démarrage.

Effacement d'un projet de démarrage

La commande navigateur "Remove" efface aussi bien le projet de démarrage situé dans la mémoire système (Flash) que celui du module mémoire. La commande navigateur "removeprojfrommmc" entraîne l'effacement du projet de démarrage et du fichier Startup.INI sur le module mémoire. L'effacement du projet de démarrage enregistré sur le module mémoire peut également s'opérer via le menu de l'automate : PROGRAM → DELETE → DELETE CARD.

Téléchargement/Actualisation du système d'exploitation

Avec l'EC4-200, vous avez la possibilité de remplacer le système d'exploitation (SE) enregistré par un SE plus récent. Moeller vous propose de télécharger la version actuelle du SE sur Internet (<http://www.moeller.net/support>). Le SE actuel figure également sur le plus récent CD « easySoft CoDeSys ».

⚠ Attention !

Le téléchargement ne peut s'opérer qu'en mode Hors ligne, via l'interface RS232 ! Lors du téléchargement du SE, toutes les données présentes dans l'automate/sur le module mémoire sont effacées. L'automate exécute un "Reset origine", → page 44

Le transfert du SE peut s'opérer de deux manières :

- Directement du PC vers l'automate
- Du PC vers le module mémoire. Lors du démarrage de l'automate, le SE du module mémoire est copié vers l'automate.

Transfert du système d'exploitation du PC vers l'automate

- ▶ Ouvrez un projet et, sous « Ressources → Configuration de l'automate », activez l'onglet "Paramètres généraux".
- ▶ Dans le champ "Actualisation du système d'exploitation", cliquez sur le bouton "Démarrer".

La fenêtre "Transférer le système d'exploitation" s'ouvre.

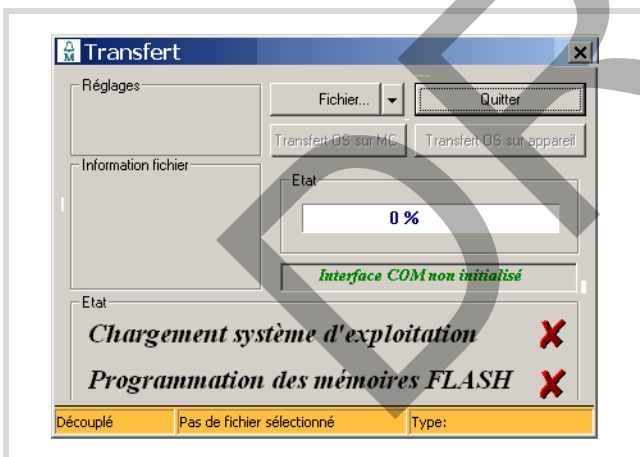


Figure 64 : Transfert du système d'exploitation

Le système signale que le port COM n'est pas initialisé.

- ▶ Cliquez sur le bouton "Fichier du système d'exploitation" et sélectionnez le fichier (*.hex) souhaité pour le système d'exploitation.

→ Vous pouvez sélectionner dans la liste déroulante (menu déroulant) les derniers fichiers ouverts.

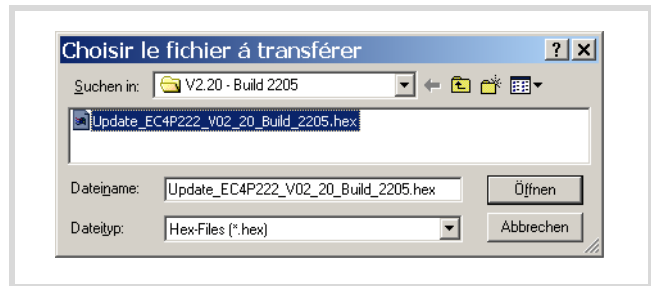


Figure 65 : Sélection du fichier du système d'exploitation

Après avoir sélectionné le fichier du SE, vous obtenez des informations concernant la référence de destination et la version du fichier.

- ▶ Actionnez le bouton "Transférer vers l'API".
- ▶ Sélectionnez l'interface RS232.

Le transfert commence. La programmation de la "Flash-Eprom" dure environ 20 à 30 secondes.

→ Si un symbole d'avertissement apparaît dans le champ "Programmation de la Flash-Eprom" au cours du transfert, la tension ne doit en aucun cas être coupée !

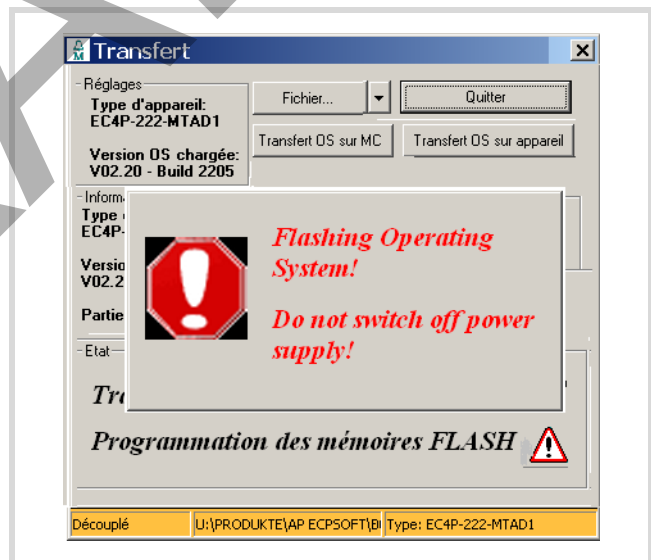


Figure 66 : Avertissement lors du téléchargement

Attendez jusqu'à obtention de l'affichage suivant.

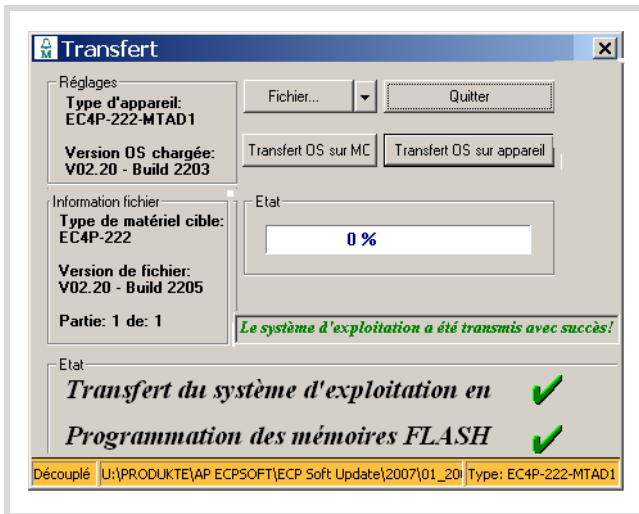


Figure 67 : Transfert du système d'exploitation vers un API

- Dans cette fenêtre, cliquez sur le bouton "Quitter".

Transfert du système d'exploitation du PC vers un module mémoire

Si vous chargez un système d'exploitation (SE) sur le module mémoire, le SE existant et le projet de démarrage présents sur le module mémoire ainsi que le programme utilisateur situé dans l'automate seront effacés. La procédure est analogue à celle décrite dans le paragraphe « Transfert du système d'exploitation du PC vers l'automate ». Cliquez dans ce cas sur le bouton 'Transférer vers le module mémoire', → figure 64, page 56.

Transfert du système d'exploitation du module mémoire vers l'automate

- A l'état hors tension, insérez le module mémoire dans l'automate.
- Mettez l'automate sous tension.

A la mise sous tension, le système d'exploitation (SE) de l'automate est actualisé et un projet de démarrage est chargé dans l'automate. Le transfert est susceptible de durer plus de 30 secondes car l'UC peut faire l'objet de plusieurs démarrages.

- N'interrompez pas l'opération en coupant par exemple la tension d'alimentation.

DRAFT

10 Commandes navigateur


Le navigateur "PLC-Browser" est un terminal d'automate reposant sur du texte. Sur une ligne de saisie, indiquez ici des commandes de type "demandes d'informations d'état à l'automate". Elles seront envoyées à l'automate sous forme de chaînes de caractères. La réponse s'affiche dans une fenêtre de résultat du navigateur. Vous pouvez utiliser cette fonctionnalité pour procéder à un diagnostic ou à un débogage.

→ Les commandes navigateur ne sont utilisables qu'en ligne.

Pour appeler ces commandes, les étapes suivantes sont nécessaires :

- ▶ Dans le logiciel de programmation, au niveau du répertoire "Ressources", sélectionnez le navigateur "PLC-Browser" en double-cliquant dessus.

Dans la zone de droite, une nouvelle fenêtre dénommée "PLC-Browser" apparaît.

- ▶ Cliquez sur le bouton .

La zone de sélection affiche les commandes navigateur disponibles.

- ▶ Sélectionnez la commande souhaitée à l'aide d'un double-clic.

La commande sélectionnée apparaît alors dans la fenêtre "PLC-Browser".

- ▶ Actionnez la touche Entrée pour visualiser dans la fenêtre de résultat la réponse de l'automate concernant la commande navigateur.

→ Pour obtenir de plus amples informations sur la commande navigateur choisie, placez devant cette commande un "?" suivi d'un espace, puis actionnez la touche ENTREE.

Pour toute description concernant ces commandes, reportez-vous au manuel relatif au logiciel de programmation (h1437f.pdf), chapitre «Ressources → PLC-Browser».


L'automate gère les commandes navigateur indiquées dans le tableau 13.

Réglage des paramètres Ethernet

Si vous exécutez une commande navigateur pour régler l'adresse IP/le masque de sous-réseau et l'adresse de la passerelle, cette commande ne sera suivie d'effet qu'avec l'une des actions suivantes :

- Mise hors puis sous tension de l'appareil
- Saisie de la commande "reboot" (redémarrage)
- Retrait puis nouvel enfichage du câble Ethernet au niveau de l'EC4-200.

Tableau 13 : Commandes navigateur

?	Aller chercher la liste des commandes implémentées
pinf	Afficher les informations relatives au projet
cycle	Afficher le temps de cycle
canload*	Afficher la charge du bus CAN
copyprojtommc	Copie du projet de démarrage actuel sur le module mémoire
createstartupini	Génération du fichier Startup.INI sur le module mémoire
factoryset	Prendre en compte les réglages usine
format	Formatage du module mémoire
GetNodeId	Affichage du nœud (Node-Id) CANopen de l'interface CAN
GetRoutingId	Affichage du nœud de routage (Routing Node-Id) et de l'interface de routage
getipgateway	Affichage de l'adresse réglée pour la passerelle (Adresse par défaut : 0.0.0.0)
getipconfig	Affichage de l'adresse IP réglée et du masque de sous-réseau; Affichage des valeurs par défaut : Adresse IP : 192.168.119.60 Masque de sous-réseau : 255.255.255.0
getmacaddress	Affichage de l'adresse MAC du contrôleur Ethernet. Exemple d'affichage : 00-80-99-05-11-22
getrtc	Lire l'horloge temps réel
metrics	Afficher les informations relatives à l'API
reboot	Redémarrage de l'EC4-200
	 Une fois la commande confirmée, l'exécution est immédiate – sans demande complémentaire !
reload	Charger le projet de démarrage de la mémoire FLASH vers l'automate
remove	Effacer le projet de démarrage dans la mémoire FLASH
removeprojfrommmc	Effacement du projet de démarrage (et du fichier Startup.INI) de sur le module mémoire
removestartupini	Effacement du fichier Startup.INI de sur le module mémoire
setipconfig	Réglage de l'adresse IP et du masque de sous-réseau Syntaxe (exemple) : setipconfig 192.168.119.60 255.255.255.0 -> ok ou message d'erreur
setipgateway	Réglage de l'adresse de la passerelle Syntaxe (exemple) : setipconfig 192.168.119.10 -> ok ou message d'erreur
setrtc*	Activer l'horloge temps réel

Vous trouverez ci-après des informations complémentaires sur les commandes repérées par un *.

Description des principales commandes navigateur

canload

Affiche la charge du bus de terrain CANopen.

Exemple :

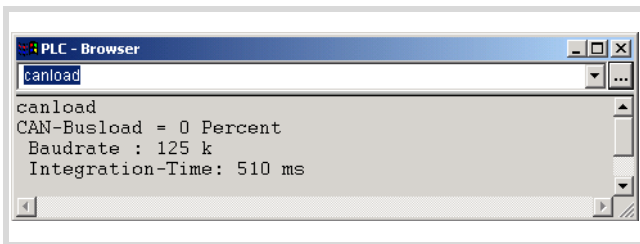


Figure 68 : Commande navigateur "canload"

Après appel de cette commande navigateur, vous obtiendrez par exemple l'information suivante :

- CAN-Busload = 0 Percent (zéro pourcent)
- Baudrate (vitesse de transmission) : 125 kBaud
- Integration-Time: 510 ms (temps d'intégration : 510 ms)



Attention !

Dans le cas d'une charge du bus de 75 % ou plus, l'avertissement suivant s'affiche par ailleurs : "ATTENTION: HIGH BUSLOAD". Associée à d'autres brefs pics de charge, la surcharge du bus CAN local peut provoquer une perte de données CAN.



Outre la commande navigateur, vous disposez à partir du programme utilisateur de la fonction CAN_BUSLOAD pour déterminer la charge du bus CAN ; voir → paragraphe « Fonction CAN_BUSLOAD », page 62.

setrtc

Règle ou modifie la date et/ou l'heure au niveau de l'automate.

Syntaxe :

```
<setrtc_YY:MM:DD:DW_HH:MM:SS>
```

Légende :

- Espace
- YY Les deux derniers chiffres du numéro de l'année (00 F YY F 99)
- MM Mois (01 F MM F 12)
- DD Jour (01 F DD F 31)
- DW Jour de la semaine (01 F DW F 07; 01 = Lundi, 07 = Dimanche)
- HH Heure (00 F HH F 23)
- MM Minute (00 F MM F 59)
- SS Seconde (00 F SS F 59)

11 Bibliothèques, blocs fonctionnels et fonctions

Les bibliothèques contiennent les blocs fonctionnels IEC et les fonctions que vous pouvez par exemple utiliser pour les tâches suivantes :

- Echange de données via le bus CANopen
- Utilisation de l'horloge temps réel
- Détermination de la charge du bus CANopen
- Exécuter une interruption
- Emettre/Recevoir des données via les interfaces.

Les bibliothèques se trouvent dans les répertoires :

- Lib_Common pour tous les automates
- Lib_EC4P_200 pour l'automate EC4-200

Utilisation des bibliothèques

Lors de l'ouverture d'un projet, les bibliothèques "Standard.lib" et "SYSLIBCALLBACK.lib" sont copiées dans le gestionnaire de bibliothèques. Si d'autres bibliothèques sont nécessaires pour l'application, vous devez les installer ultérieurement.

Les bibliothèques présentes dans le gestionnaire de bibliothèques sont affectées au projet à l'issue de l'enregistrement. Lorsque vous ouvrez à nouveau le projet, ces bibliothèques sont également appelées.

Le synoptique suivant dresse la liste des documents dans lesquels sont décrits les blocs fonctionnels et les fonctions.

Document	Bibliothèque
AWB 2700-1437	Standard.lib Util.lib XX_Util. Lib
Aide en ligne ou fichiers PDF (dans le menu Démarrer de Windows, sous <Programmes → Moeller Software → easySoft CoDeSys → Documentation → Automation Manuals>)	SysLib...pdf
AWB 2786-1456	XS40_MoellerFB. Lib/Visu. Lib/...
AN2700K20	3S_CANopenDevice. Lib 3S_CANopenManager. Lib
AN2700K19	3S_CANopenNetVar. Lib
AN2700K27	SysLibCan. Lib
AWB 2786-1554	CANUserLib. Lib CANUser_Master. Lib

Installation de bibliothèques supplémentaires

Les étapes suivantes sont nécessaires pour l'installation ultérieure de bibliothèques :

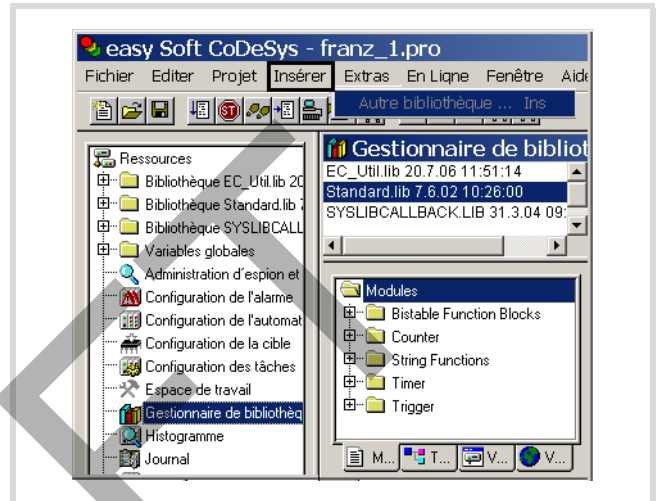


Figure 69 : Installation de bibliothèques supplémentaires

- ▶ Dans le projet créé, cliquez sur l'onglet "Ressources".
- ▶ A l'aide du bouton gauche de la souris, double-cliquez sur le répertoire "Gestionnaire de bibliothèques".
- ▶ Cliquez sur l'option menu <Insérer → Autres bibliothèques... Insérer>.

Les bibliothèques disponibles qui dépendent du système de destination s'affichent alors dans la nouvelle fenêtre.

- ▶ Sélectionnez la bibliothèque supplémentaire à installer et cliquez sur le bouton "Ouvrir".

La bibliothèque apparaît ensuite dans le gestionnaire de bibliothèques.

Fonctions spécifiques au EC4-200

Bibliothèque EC_Util.lib

Comme le montre la figure ci-dessous, cette bibliothèque propose les fonctions suivantes :

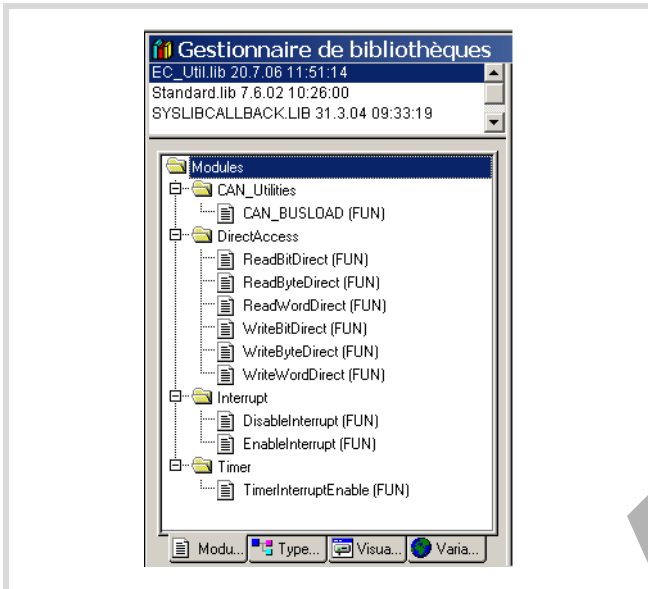


Figure 70 : Fonctions de la bibliothèque EC_Util.lib

Fonction CAN_BUSLOAD

Cette fonction peut être appelée de manière cyclique dans un programme utilisateur. A l'issue d'un cycle de lecture effectué avec succès, cette fonction délivre en retour la valeur TRUE et inscrit dans les adresses transmises les valeurs déterminées pour le temps d'intégration et la charge du bus.

Si le calcul de la charge du bus n'est pas achevé, ou si le contrôleur CAN n'est pas initialisé, cette fonction délivre en retour la valeur FALSE.

Pour toute information sur l'analyse de la valeur de retour, reportez-vous à la commande navigateur « canload », page 60.

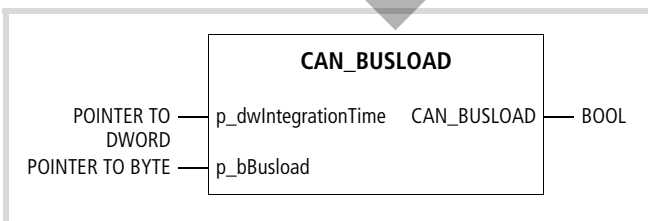


Figure 71 : Fonction CAN_BUSLOAD

Les autres fonctions sont décrites aux pages suivantes :

- Accès direct aux E/S (DirectAccess) → page 53
- TimerInterruptEnable → page 50
- DisableInterrupt/EnableInterrupt → page 52

Bibliothèque EC_Visu.lib/EC_Visu2.lib

La bibliothèque EC_Visu2.lib regroupe les blocs fonctionnels destinés à la commande de l'afficheur LCD.

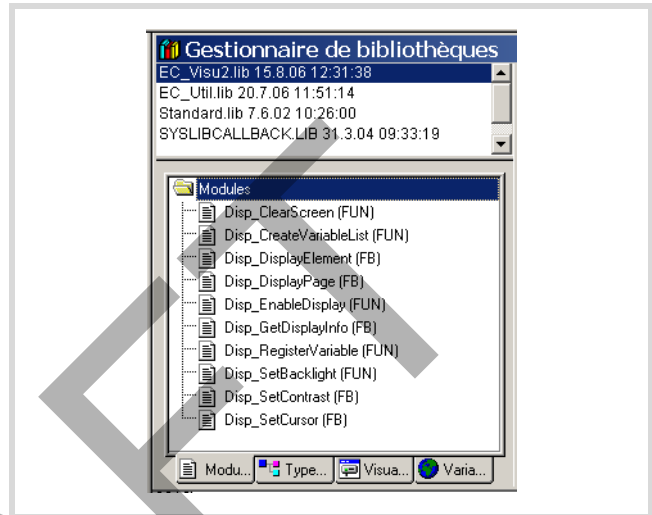


Figure 72 : Bibliothèque EC_Visu2.lib

→ Vous pouvez continuer à utiliser les fonctions/blocs fonctionnels déjà existants ("SetBacklight", "SetContrast" et "GetDisplayInfo") de la bibliothèque EC_Visu.lib. Ils seront toutefois remplacés par les fonctions/blocs fonctionnels contenus dans la bibliothèque EC_Visu2.lib (→ Tableau).

EC_Visu.lib	EC_Visu2.lib
Bloc fonctionnel SetBacklight	Disp_SetBacklight
Bloc fonctionnel SetContrast	Disp_SetContrast
GetDisplayInfo	Disp_GetDisplayInfo

Ces fonctions sont décrites page 76.

12 Etablissement liaison PC – EC4-200

Afin de pouvoir établir la liaison entre le PC et l'automate, les paramètres de communication des deux appareils doivent concorder. Sur les appareils pour lesquels il s'agit de la première mise en service, les paramètres par défaut sont réglés selon la figure 73. Il vous reste simplement à choisir l'interface PC (COM...). Aucun autre réglage n'est nécessaire.

→ Si vous obtenez un message d'erreur, cela signifie que les réglages standard de l'UC ont déjà été modifiés. Dans ce cas, faites un essai avec d'autres vitesses de transmission ou optez pour le réglage usine.

Vous pouvez ensuite redéfinir les paramètres de l'UC (→ figure 74). Il convient de réadapter par la suite ces modifications de paramètres pour le PC.

Vous pouvez réaliser la liaison entre le PC et l'interface de programmation de l'automate via :

- l'interface RS 232
- l'interface Ethernet (pour les appareils de type EC4P-222-..., en plus de l'utilisation de l'interface RS232).

Etablissement de la liaison via l'interface RS232

Adaptez d'abord les paramètres de communication du PC aux paramètres standards de l'automate → paragraphe « Définition/modification des paramètres de communication du PC ».

L'interface RS232 de l'automate (COM1) présente les paramètres standards suivants :

Vitesse de transmission	38400 bits/s
Parity	No
Bits de stop	1
Octets Motorola	No

Définition/modification des paramètres de communication du PC

Vous pouvez utiliser l'interface COM1 à COMx du PC. Dans le logiciel de programmation, déterminez les paramètres de communication de l'interface.

- ▶ Sélectionnez l'option menu « En ligne → Paramètres de communication ».
- ▶ Déterminez le port (interface COM1 ou COM2) → paragraphe « Modification des paramètres ».
- ▶ Validez les autres paramètres de la figure 73.
- ▶ Confirmez le choix des paramètres par OK.
- ▶ Etablissez une liaison avec l'automate ("Accéder au système").

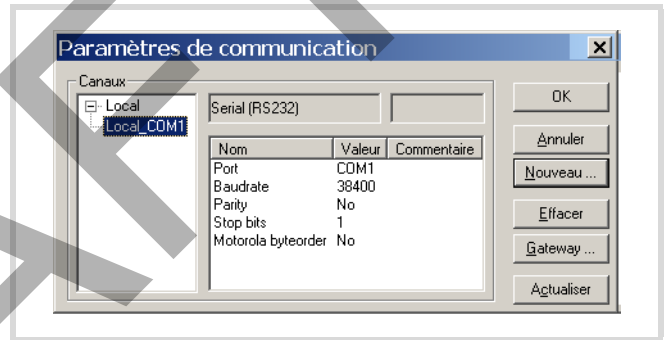


Figure 73 : Définition des paramètres de communication du PC

Modification des paramètres

Pour modifier des paramètres tels que la vitesse de transmission ou l'indication d'un port, par exemple, procédez comme suit :

- ▶ Double-cliquez sur la valeur (38400, par exemple). Un fond gris apparaît alors au niveau de ce champ.
- ▶ Indiquez la valeur souhaitée.

Le fait de double-cliquer un certain nombre de fois dans ce champ vous permet de sélectionner la vitesse de transmission souhaitée (exemple : 57600 bits/s).

Modification des paramètres de communication (vitesse de transmission) de l'UC

- ▶ Ouvrez la Configuration de l'automate.
- ▶ Sélectionnez l'onglet "Communication".
- ▶ Dans la zone de liste "Vitesse de transmission" (Baudrate), choisissez la vitesse souhaitée (exemple : 57600 bits/s, comme indiqué sur la figure 74).

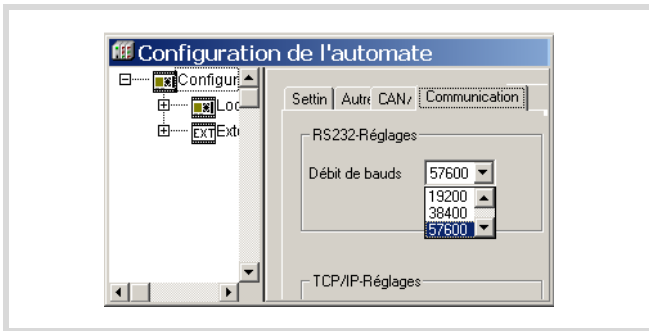


Figure 74 : Définition des paramètres de communication de l'UC

- ▶ Etablissez une liaison avec l'automate ("Accéder au système").

La question suivante s'affiche :

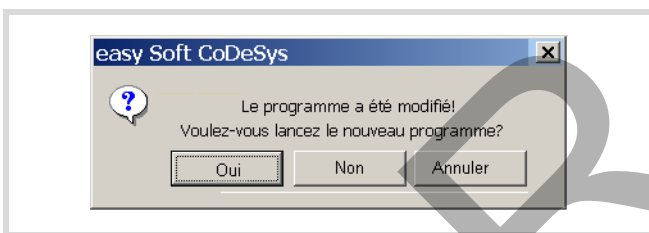


Figure 75 : Question concernant une modification du programme

- ▶ Répondez par "Oui" à cette question.

Le programme est alors chargé. Après une temporisation d'environ 2 minutes, un message d'erreur concernant la communication apparaîtra du fait que la vitesse de transmission entre l'UC et le PC ne concordent plus :



Figure 76 : Erreur de communication

- ▶ Confirmez le message d'erreur par OK.

Pour pouvoir reconnecter le PC, vous devez adapter de nouveau la vitesse de transmission (Baudrate) du PC à celle du projet.

Etablissement de la liaison via Ethernet

Après avoir relié le PC à l'automate à l'aide d'un câble Ethernet, sélectionnez dans le logiciel de programmation (au niveau de la fenêtre "Paramètres de communication") le canal de communication TCP/IP et indiquez l'adresse IP de l'automate. L'adresse par défaut de l'automate est 192.168.119.60.

Sélection du canal de communication et de l'adresse

- ▶ Appelez le menu «En ligne → Paramètres de communication».

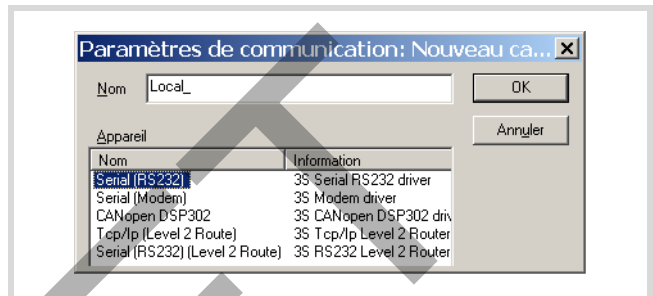


Figure 77 : Choix du canal

- ▶ Actionnez le bouton Nouveau...
- ▶ Dans le synoptique, sélectionnez le canal de communication TCP/IP (Level2Route) et modifiez le nom "Local" (choisissez par exemple "Test Ethernet").
- ▶ Confirmez à l'aide de la touche OK.



Figure 78 : Indication de l'adresse IP

- ▶ Double-cliquez sur le champ "localhost" et saisissez l'adresse par défaut 192.168.119.60.
- ▶ Confirmez vos indications en cliquant d'abord sur n'importe quel autre champ, puis ensuite seulement sur OK.

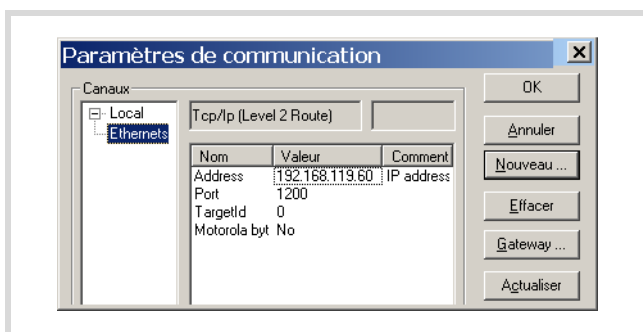


Figure 79 : Paramètres de communication avec adresse IP

- Compilez le programme puis établissez la liaison ("Accéder au système").

Transmission des données à l'aide du protocole TCP/IP (Level2Route)

Ce type de transmission de données est géré par les automates XC200, MFD4 et EC4-200 (EC4P-222... uniquement).

Les données entre le PC et l'automate sont transmises par blocs de données de 128 Koctets (réglage à la livraison de l'appareil). Restriction concernant les appareils EC4P-222... : la transmission de volumes de données importants est susceptible de générer un message d'erreur en raison du trop faible espace mémoire. Dans ce cas, choisissez une taille de blocs de 4 Koctets. Procédez à ce réglage dans le dossier "Configuration", au niveau de l'onglet "Communication" :

- Cliquez simplement sur le bouton "Adapter réglages" (Adjust settings). La taille des blocs de données est alors automatiquement réglée sur 4 Koctets.

Si le bouton est grisé (donc inaccessible) comme sur la figure 80, cela signifie que la taille des blocs est déjà réglée sur 4 Koctets.

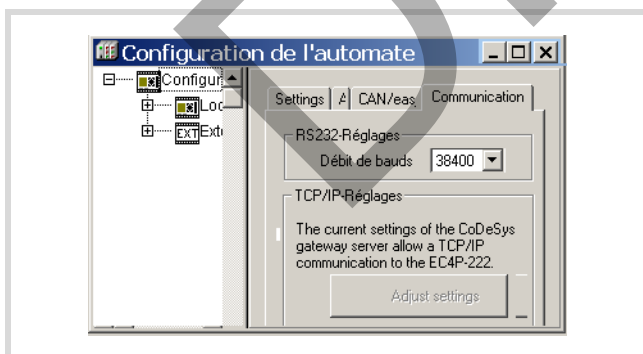


Figure 80 : Réglage de la taille des blocs de la liaison TCP/IP

Réglez à nouveau la taille des blocs sur 128 Koctets lorsque vous paramétrez un appareil XC200 ou MFD4. Ces deux automates possèdent un plus grand espace mémoire. Vous pouvez modifier à volonté le réglage via l'application "BlockSizeEditor.exe" (qui figure dans le menu Démarrer de Windows, sous Programmes → Moeller Software → easy Soft CoDeSys → Communication...).

Scrutation/Modification de l'adresse IP

Les commandes navigateur "setipconfig" et "getipconfig" vous permettent de modifier et de scruter l'adresse IP : → paragraphe « Commandes navigateur », page 59.

Après avoir modifié l'adresse IP, procédez au redémarrage de l'automate. Veillez à ce que l'adresse IP de l'appareil de programmation appartienne à la même famille d'adresses. En d'autres termes, l'adresse IP de l'appareil de programmation et de l'automate doivent concorder au niveau des groupes de chiffres suivants dans le cas d'un masque de sous-réseau de type 255.255.255.0 :

Exemple 1

Adresse IP de l'automate : 192.168.119.xxx

Adresse IP du PC: 192.168.119.yyy

Exemple 2

Adresse IP de l'automate : 192.168.100.xxx

Adresse IP du PC: 192.168.100.yyy

Les conditions suivantes s'appliquent aux exemples 1 et 2 :

- xxx différent de yyy
- Les adresses doivent être comprises entre 1 et 254.
- Les adresses doivent appartenir à la même famille d'adresses.

Si aucune liaison n'aboutit, vous pouvez vérifier le chemin de transmission à l'aide de la fonction "PING" : elle vous permet de vous assurer que l'échec de l'établissement de la liaison n'est pas dû au chemin de transmission. Procédez pour cela aux étapes suivantes :

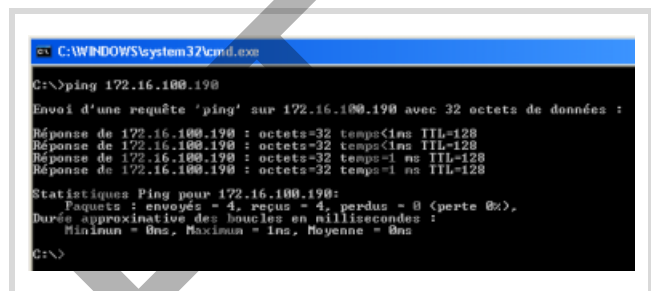
- ▶ Ouvrez la fenêtre DOS à l'aide du champ "Démarrer" et de la commande "Exécuter".
- ▶ Dans la zone de saisie, indiquez "CMD", puis validez cette entrée à l'aide de "OK".

Vous obtenez alors une fenêtre avec indication d'un lecteur et un curseur clignotant situé derrière le lecteur mentionné.

- ▶ Dans le cadre de l'exemple concret indiqué ici, saisissez le texte suivant : "ping 192.168.119.60" et validez-le par "OK".

Si le routage se déroule correctement, vous obtenez une réponse avec indication d'un temps de réponse. Dans le cas contraire, un message vous indiquera un dépassement de temps lors de l'établissement de la liaison.

La figure suivante montre le résultat d'une liaison qui s'est établie correctement.



```
C:\WINDOWS\system32\cmd.exe
C:\>ping 172.16.100.190
Envoi d'une requête 'ping' sur 172.16.100.190 avec 32 octets de données :
Réponse de 172.16.100.190 : octets=32 temps<1ms TTL=128
Réponse de 172.16.100.190 : octets=32 temps<1ms TTL=128
Réponse de 172.16.100.190 : octets=32 temps<1ms TTL=128
Réponse de 172.16.100.190 : octets=32 temps<1ms TTL=128
Statistiques Ping pour 172.16.100.190:
    Paquets : envoyés = 4, reçus = 4, perdus = 0 (perte 0%),
    Durée approximative des boucles en millisecondes :
    Minimum = 0ms, Maximum = 1ms, Moyenne = 0ms
C:\>
```

Figure 81 : Réponse PING en cas d'établissement correct d'une liaison Ethernet

13 Saisie des paramètres système via le fichier STARTUP.INI

Vue d'ensemble

Vous avez la possibilité de créer des paramètres système indépendants d'un projet et de les enregistrer sur le module mémoire. Ils y seront regroupés dans le fichier Startup.INI. Vous pouvez par ailleurs enficher le module mémoire dans d'autres automates. L'automate prend en compte les paramètres lors du démarrage. Le fichier Startup.INI est toujours créé avec l'ensemble des saisies spécifiques à l'automate (→ tableau 14).

Tableau 14 : Paramètres figurant dans le fichier Startup.INI

```
Saisies
COM_BAUDRATE: 4800,9600,19200,38400,57600
CAN1_BAUDRATE: 10,20,50,100,125,250,500
CAN1_NODEID: 1-127
CAN_ROUTINGID: 1-127
IP_ADDRESS=xxx.xxx.xxx.xxx
IP_SUBNETMASK=xxx.xxx.xxx.xxx
IP_GATEWAY=xxx.xxx.xxx.xxx
```

Les paramètres issus du fichier INI sont prioritaires par rapport aux paramètres issus de la Configuration de l'automate. Après un chargement du programme vers l'appareil ou après un chargement du projet de démarrage, les paramètres issus de la Configuration de l'automate ne sont pas pris en compte.

Structure du fichier INI

Un fichier INI est un fichier texte avec un format de données fixe. A partir d'une section désignée par un nom (indiqué entre crochets, tel que [STARTUP], par exemple), les paramètres système sont exécutés et suivis d'un signe "égal" et de leur valeur. La ligne se termine par CR/LF (retour chariot = Carriage/Return ; saut de ligne = Line Feed).

```
COM_BAUDRATE = 38400 (Carriage/Return)
```

L'automate interprète les lignes qui débutent par un point virgule comme un commentaire et les saute lors de la lecture :

```
; CAN_NODEID = 2
```

Vous pouvez modifier ou créer des paramètres avec un éditeur de texte lorsque vous insérez le module mémoire dans le slot (emplacement) prévu à cet effet sur le PC. Insérez d'abord le module mémoire dans l'adaptateur fourni, puis dans le slot du PC. Le fichier STARTUP.INI sera enregistré sur le module mémoire, dans le répertoire « MOELLER/EC4P_200/PROJECT/ ».

Création du fichier Startup.INI

Par principe, l'automate travaille lors de la première mise sous tension (état initial) avec les paramètres système par défaut, les données STARTUP. Si vous chargez un projet dans l'automate qui se trouve à l'état initial, ce dernier démarre aussitôt avec les paramètres système du projet.

La commande navigateur "createstartupini" vous permet de transférer à partir de l'automate les paramètres actuels du système vers le module mémoire. Le fichier Startup.INI (qui contient ces données) est alors simultanément créé. Condition préalable : le module mémoire doit être enfiché et formaté (autrement dit, il ne doit pas comporter de fichier Startup.ini).

Tableau 15 : Exemple : fichier STARTUP.INI pour EC4-200

```
[STARTUP]
TARGET = EC4P-200
IP_ADDRESS=192.168.119.60
IP_SUBNETMASK=255.255.255.0
IP_GATEWAY=0.0.0.0
COM_Baudrate = 38400
CAN1_Baudrate = 125
CAN1_NODEID = 2
CAN_ROUTINGID = 127
```

Un fichier existant ne peut pas être modifié ou écrasé via la commande navigateur "createstartupini". Si vous saisissez néanmoins cette commande, un avertissement s'affiche. Pour créer un nouveau fichier, effacez d'abord le fichier existant, voir → paragraphe « Effacement du fichier Startup.INI », page 68.

Mise sous tension de l'automate dans le cas d'un module mémoire avec fichier Startup.INI enfiché

A la mise sous tension de l'automate, les données du fichier Startup.INI du module mémoire sont transférées vers l'automate. Ces paramètres système restent actifs, même après chargement d'un nouveau programme.

Modification des paramètres

Les paramètres sont conservés jusqu'à ce que vous saisissez la commande navigateur "removestartupini" et procédez à la mise hors puis sous tension de l'automate. L'automate travaille alors avec les paramètres du projet.

Effacement du fichier Startup.INI

Les commandes navigateur suivantes à votre disposition permettent un accès au module mémoire.

- removestartupini :
Cette commande efface les paramètres système au sein de l'automate. Si un module mémoire est enfiché, le fichier INI présent sur le module mémoire est également effacé. A la mise sous tension suivante, les paramètres provenant du projet sont pris en compte.
- removeprojfrommmc :
Cette commande efface le projet de démarrage et le fichier INI présent sur le module mémoire. Les paramètres système présents dans l'automate sont conservés.
- format :
Cette commande efface l'ensemble du module mémoire, y compris le fichier INI.

Le comportement du fichier Startup.ini avec la commande menu "Reset origine", avec la commande de réglage usine (dans le menu de l'automate) et avec la commande navigateur "factoryset" est décrit de manière détaillée au paragraphe « Reset (remise à zéro) », page 44.

PROJET

14 Programmation via le réseau CAN (routage)

On désigne par "routage" (Routing, en anglais) la possibilité d'établir une liaison en ligne entre un appareil de programmation (PC) et un automate quelconque (apte au routage) au sein d'un réseau CAN, sans que l'appareil de programmation ait besoin d'être relié directement à l'automate de destination. L'appareil de programmation peut être raccordé à un autre automate au sein du réseau. La liaison de routage vous permet d'effectuer toutes les actions qui sont également disponibles dans le cadre d'une liaison en ligne directe entre un appareil de programmation et un automate :

- Transfert d'un programme vers l'appareil
- Modifications en ligne
- Test du programme (Débogage)
- Création de projets de démarrage
- Ecriture de fichiers dans l'automate
- Lecture de fichiers à partir de l'automate.

Le routage offre un avantage : il vous permet d'accéder, à partir d'un automate relié à l'appareil de programmation, à tous les automates du bus CAN aptes au routage. Par la sélection du projet, vous déterminez avec quel automate vous souhaitez communiquer. Il est ainsi possible d'utiliser facilement des automates décentralisés.

La transmission de données des liaisons de routage est toutefois beaucoup plus lente que celle des liaisons directes (série ou TCP/IP). Ceci est par exemple particulièrement visible au niveau des temps d'actualisation (qui s'avèrent plus longs) d'éléments de visualisation (ou variables) ou des vitesses de téléchargement (qui s'avèrent moins rapides).

Conditions préalables

Le routage ne peut être utilisé que si les conditions préalables suivantes sont remplies :

- L'automate procédant au routage ET l'automate de destination doivent tous deux gérer le routage.
- Les deux automates doivent être reliés à l'aide du bus CAN.
- Les automates doivent disposer de la même vitesse de transmission CAN active.
- Sur chacun des deux automates doit être réglé un nœud de routage (Routing-Node-Id) valable.

Propriétés du routage de l'automate

L'automate gère le routage via le bus CAN.

Le routage peut être effectué sans chargement préalable d'un programme utilisateur (valeurs par défaut : 125 kBaud, Node-Id 127). Pour cela, l'automate de destination ne doit en aucun cas être configuré en tant que maître CAN ou en tant qu'appareil CAN.

Vous pouvez par exemple charger un programme du PC vers l'EC4-200, via un automate de la gamme XC. Dans ce cas, affectez à l'EC4-200 (automate de destination) un nœud de routage (Routing-Node-Id).

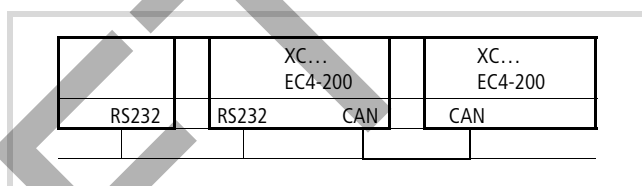


Figure 82 : Transfert d'un programme par routage

Routage via XC200

Si vous procédez, dans le cas d'une liaison entre XC200 et PC, à un routage ou à un transfert de programme via TCP/IP, vous devez régler la taille des blocs des données à transférer. La taille des blocs (4 ou 128 Koctets) dépend du type de transfert (routage ou transfert de programme) et du système d'exploitation, → tableau 16.

Tableau 16 : Taille des blocs pour le transfert des données

	Transfert de programme/de fichier		Routage	
	BTS < V1.03.03	BTS ≥ V1.03.03	BTS < V1.03.03	BTS ≥ V1.03.03
Taille des blocs	128 Koctets	4/128 Koctets	Routage impossible	4 Koctets
Valeur par défaut :	128 Koctets			
s				



Attention !

Le transfert d'un programme présentant des blocs d'une taille de 4 Koctets vers un automate équipé d'un système d'exploitation < V1.03.03 conduit à un comportement défaillant.

Lorsque vous procédez à un transfert de programme vers l'appareil, la barre de progression visible sur l'écran de l'appareil de programmation n'évolue que de manière brusque (toutes les 10 secondes environ).

Le roulage avec l'XC200 est possible à partir de la version V1.03.03 du système d'exploitation.

Le réglage de la taille des blocs (modification de la valeur dans la base de registre) est décrit ci-après :

→ Ce réglage ne peut être effectué qu'avec des droits d'administrateur (accès à la base de registre) !

Réglage de la taille des blocs :

- ▶ Fermez toutes les applications.
- ▶ Fermez le serveur pour passerelle CoDeSys.

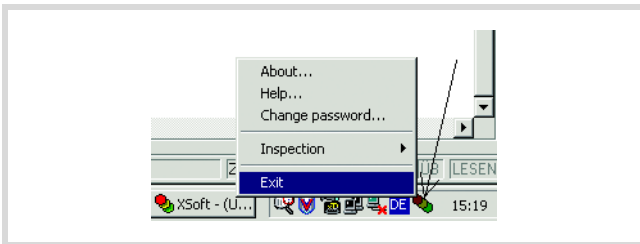


Figure 83 : Fermeture du serveur pour passerelle CoDeSys

- ▶ Modifiez la taille des blocs : indiquez la valeur souhaitée.

Appelez l'application BlockSizeEditor.exe dans le répertoire "easy Soft CoDeSys" du logiciel de programmation et choisissez la taille des blocs.

Autre possibilité :

Pour saisir la taille des blocs dans la base de registre, vous disposez dans le répertoire d'installation des fichiers *.reg suivants :

BlockSizeDefault.reg	Permet de saisir une taille de bloc (valeur par défaut) de 20000 _{hex} = 128 Koctets dans la base de registre.
BlockSizeRout.reg	Permet de saisir une taille de bloc de 1000 _{hex} = 4 Koctets dans la base de registre.

La taille de bloc relative au transfert est réglée à l'aide de la saisie suivante dans la base de registre :

```
[HKEY_LOCAL_MACHINE\SOFTWARE\3S-Smart Software Solutions
GmbH\Gateway Server\Drivers\Standard\Settings\Tcp/Ip (Level
2 Route)]
"Blocksize" = dword:00020000
```

La taille de bloc par défaut est de 20000_{hex} (= 128 Koctets) et la taille de bloc pour le roulage est de 1000_{hex} (= 4 Koctets).

Remarques sur le roulage

- Dans le cas de fichiers volumineux écrits dans l'automate de destination ou lus à partir de l'automate, une interruption de la liaison en ligne peut survenir à l'issue du transfert. Il est possible de procéder à un nouveau couplage.
- Si un programme avec un nœud de roulage (Routing-Node-Id) modifié est chargé dans l'automate de destination par le biais d'un automate procédant à un roulage, l'automate de destination prend en compte le Routing-Node-Id modifié ; la liaison de communication est cependant interrompue. Il est possible de procéder à un nouveau couplage avec un Routing-Node-Id rectifié.
- Si un automate reçoit un programme sans paramètres de roulage valables (Baudrate/Node-Id), il n'est pas possible de réaliser un couplage avec cet automate à l'aide d'une liaison de roulage.
- Le roulage est indépendant de la configuration (maître/appareil) : il est possible d'accéder à un automate de destination qui n'a été configuré ni en tant que maître, ni en tant qu'appareil. Il suffit que cet automate comporte les paramètres de base, tels que Node-Id (n° de nœud) et Baudrate (vitesse de transmission), ainsi qu'un programme simple.

Réglage de l'easyNet de nœud (Node easyNet) et de l'easyNet de roulage (Routing easyNet)

Tout automate raccordé au bus CAN peut être configuré en tant que maître (Master) ou en tant qu'appareil (Device). Pour assurer une identification claire lors de la communication de base, chaque automate reçoit un easyNet de nœud (Node easyNet), c'est-à-dire une adresse. Si vous souhaitez accéder à un automate (de destination) à l'aide d'une fonction de roulage, vous devez affecter à l'automate de destination un autre easyNet (de roulage), désigné en anglais par "Routing easyNet". Pour assurer la liaison entre le PC et l'EC4-200 (par exemple), il est possible de faire appel à l'interface RS232.

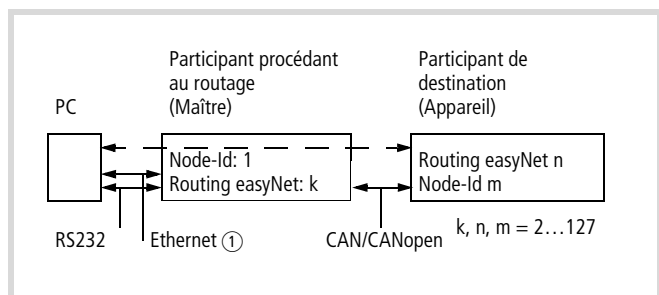


Figure 84 : Roulage via le participant doté de ID1 (XC..., EC4-200, MFD4)

① Liaison Ethernet possible dans le cas de XC200, MFD4 et EC4-222...

Tableau 17 : Exemple de réglage de l'Id de nœud (Node Id), de l'Id de routage (Routing Id) et de la vitesse de transmission (Baudrate)

Automate	Fonction	Node easyNet Routing easyNet	Vitesse de transmission	→ figure
Automate procédant à un routage	Maître	1 (Base) 127 (Routage)	125 Ko	86 85
Automate de destination	Appareil (Device)	3 (Base)	125 Ko	87 85
		54 (routage)		

→ Remarque valable pour les participants de type Appareil (Device) : l'easyNet de routage (Routing easyNet) doit impérativement être différent de l'easyNet de nœud/Node easyNet (communication de base) !

Exception relative aux XC100 avec système d'exploitation f V2.0 :

dans ce cas, l'easyNet de routage (Routing easyNet) doit être identique à l'easyNet de nœud (Node Id) !

Réglage du participant Maître

Déterminez deux easyNet de nœud (Node easyNet) au sein du participant Maître :

- un Id pour la fonction de routage.
- un Id pour la communication de base

Pour la fonction de routage :

Réglez l'easyNet de routage et la vitesse de transmission du bus CAN dans l'onglet "CAN/easyNET" du dossier "Configuration", comme indiqué sur la figure 85.

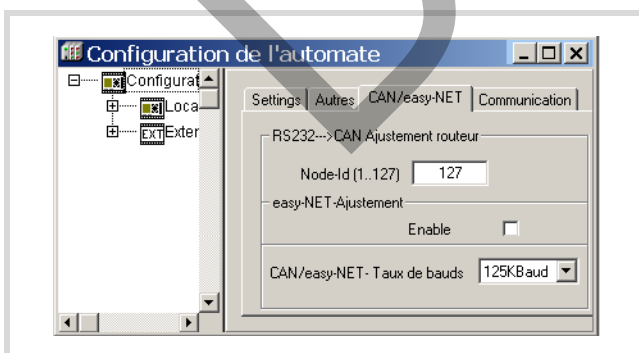


Figure 85 : Réglages relatifs au routage du maître CAN

Pour la communication de base :

Déterminez les easyNet destinés à la communication de base ainsi que la vitesse de transmission du bus CAN dans le dossier "CanMaster", au niveau de l'onglet "Paramètres CAN", comme indiqué sur la figure 86.

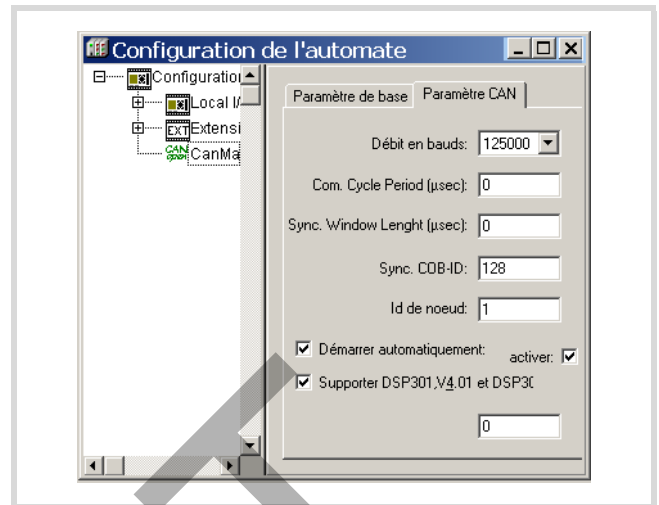


Figure 86 : Maître CAN, easyNet pour la communication de base

Réglage du participant de type Appareil (Device)

Déterminez deux easyNet de nœud (Node easyNet) au sein du participant (de destination) de type Appareil (Device) :

- un easyNet pour la fonction de routage : réglez l'easyNet de routage (Routing easyNet) et la vitesse de transmission CAN (en Baud) dans l'onglet "CAN/easyNET", comme indiqué sur la figure 85. Réglez par exemple l'easyNet de nœud (Node easyNet) sur "54".
- un easyNet pour la communication de base : déterminez l'easyNet destiné à la communication de base ainsi que la vitesse de transmission du bus CAN dans le dossier "Réglages CAN", comme indiqué sur la figure 87.

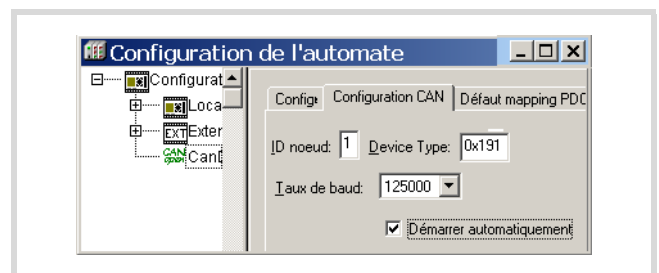


Figure 87 : Paramètres d'un appareil CAN

L'Id de nœud (Node-Id) et la vitesse de transmission sont transférés dans l'automate en même temps que le projet.

Exemple : accès à un programme d'automate

L'exemple suivant explique comment accéder à un programme d'automate.

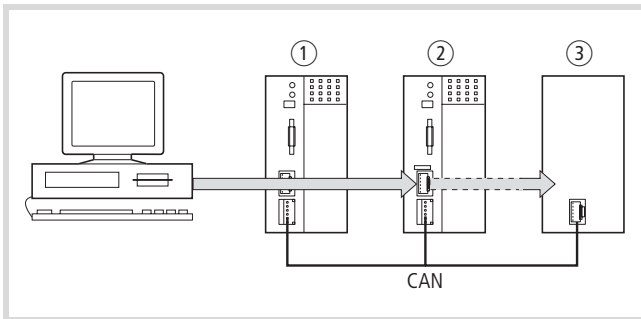


Figure 88 : Possibilités de diagnostic

- ① XC100 avec Node easyNet 1
- ② XC200 avec Node easyNet 2, Routing easyNet 127
- ③ Automate avec Node Id 3 et Routing easyNet 54
(exemples : XC100, XC200, XC121, EC4-200)

Vous avez raccordé le PC à l'automate doté de Node easyNet 2 et vous voulez accéder à l'automate de destination doté de Routing easyNet 54.

- ▶ Ouvrez le projet de l'automate de destination (Node-Id 3) dont vous voulez éditer ou tester le programme.
- ▶ Paramétrez d'abord la liaison matérielle PC ↔ Automate (Node-Id 2).
- ▶ Dans le menu En ligne, sélectionnez "Paramètres de communication".
- ▶ Sous Canaux "local", cliquez sur le bouton "Nouveau".

La fenêtre "Nouveau canal" s'affiche.

- ▶ Choisissez le canal dans la fenêtre "Appareil" : Serial [RS232] [Level 2 Route] ou TCP/Ip [Level 2 Route] (EC4P-222... uniquement).
- ▶ Dans le champ "Nom", vous pouvez attribuer un nouveau nom (exemple : "Rout_232").

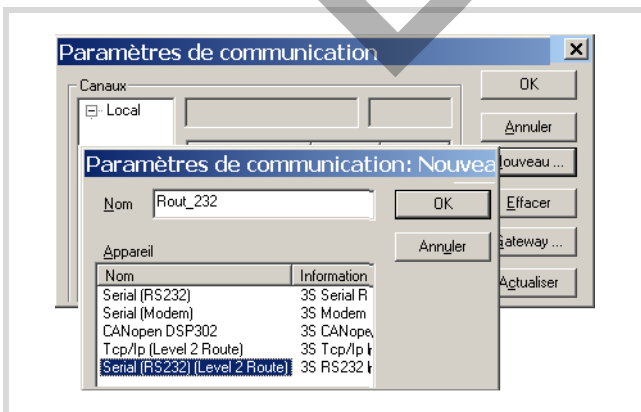


Figure 89 : Paramétrage d'un canal

Vous avez à présent défini les paramètres pour la liaison matérielle entre le PC et l'automate (Node easyNet 2).

- ▶ Indiquez le "Target Id" du participant de destination (le chiffre 54, dans notre exemple). Le "Target Id" est identique au "Routing easyNet" !
Pour indiquer le "Target Id", cliquez sur le champ de la colonne "Valeur" situé à droite du terme "Target Id". Saisissez-y le chiffre 54 et validez par OK.
- ▶ Etablissez la liaison ("Accéder au système") et effectuez l'action.

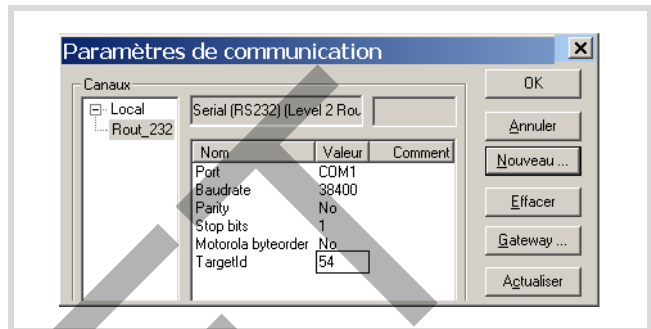


Figure 90 : Réglage du Target-Id de l'automate de destination

Combinaison d'automates pour le routage

Les automates suivants gèrent le routage :

De → Vers ↓	XC100	XC121	XC200	EC4-200	MFD4
XC100	×	×	×	×	×
XC121	×	×	×	×	×
XC200	×	×	×	×	×
MFD4	×	×	×	×	×
EC4-200	×	×	×	×	×

15 Interface RS-232 en mode Transparent

En mode Transparent, l'échange de données entre l'EC4-200 et les terminaux (écrans, imprimantes, PC, appareils de mesure, par exemple) s'opère sans interprétation des données. Il convient pour cela de procéder à la commutation de l'interface série RS 232 (COM1/COM2 = interface multifonctions) en mode Transparent, via le programme utilisateur.

Pour exécuter le mode Transparent, vous disposez de fonctions destinées à l'ouverture et à la fermeture de l'interface, à l'émission et à la réception de données, ainsi qu'à l'activation des paramètres de l'interface. Une fois l'ouverture réalisée, l'interface travaille avec les paramètres de communication actifs (que vous pouvez adapter en appelant la fonction "SysComSetSettings").

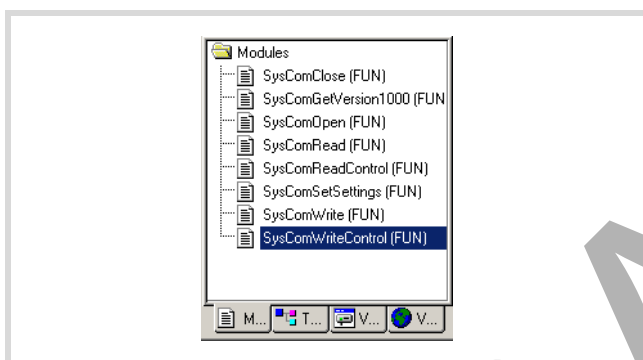


Figure 91 : Synoptique de la fonction

Les fonctions du mode Transparent se situent dans la bibliothèque "EC_SysLibCom.lib". Cette bibliothèque doit par suite être intégrée dans le gestionnaire de bibliothèques. Pour la description des fonctions, reportez-vous au manuel "blocs fonctionnels" (AWB2786-1452GB).

- Lorsque l'interface RS-232 (COM1) se trouve en mode Transparent, une programmation via cette interface est impossible. Il convient d'abord de fermer le mode Transparent. La fermeture du mode Transparent entraîne la réinitialisation des paramètres de communication initiaux.

COM1/COM2 :

La fermeture du mode Transparent s'opère obligatoirement lors d'un passage de l'automate à l'état STOP ou par appel de la fonction "SysComClose".

DRAFT

16 Afficheur interactif

L'utilisation de fonctions et de blocs fonctionnels (FB) vous permet de visualiser des variables (textes/valeurs) sur l'afficheur de l'automate et de saisir des valeurs à l'aide des touches de direction. Pour un affichage/une commande opérateur externe, raccordez à l'automate un MFD-CP4 qui exécute parallèlement ces fonctions.

Type de représentation (affichage)

L'afficheur de l'automate et du MFD-CP4 présente une trame formée de 4 lignes et 16 colonnes. Chaque ligne permet de représenter 16 caractères. Vous disposez de 3 jeux de caractères.

L'afficheur permet de représenter au max. 12 variables.

La longueur réservée à la représentation ou le nombre de caractères dépend du type de données. Si la variable comprend une valeur dont la représentation comporte une virgule, il convient de prévoir un emplacement supplémentaire pour la virgule. La valeur de la variable peut être actualisée en permanence. La saisie d'une valeur s'opère à l'aide des touches de direction.

Type de données	Valeur min. / max.	Nombre max. d'emplacements	Précision max.	Format
BYTE	255	4	2	1.23
WORD	65 535	6	4	1.2345
DWORD	4 294 967 295	11	9	1.234568
USINT	255	4	2	1.23
UINT	65 535	6	4	1.2345
UDINT	4 294 967 295	11	9	1.234568
SINT	-128/127	5	2	-1.23
INT	-32768/32767	7	4	-1.2345
DINT	-2 147 483 648/ +2 147 483 647	12	9	-1.234568

Commutation entre Affichage d'état et Mode saisie/affichage

A l'état initial, l'afficheur indique l'état de l'automate. Pour afficher des textes/variables spécifiques à l'application considérée ou pour saisir des valeurs/variables, il convient d'activer le mode Saisie/Affichage de l'automate. Pour une meilleure compréhension, représentez-vous deux afficheurs internes à l'automate et dont l'affichage est continuellement rafraîchi. Le premier sert à afficher l'état et le menu de l'automate. Le second sert à afficher des textes et des variables en mode Saisie/Affichage.

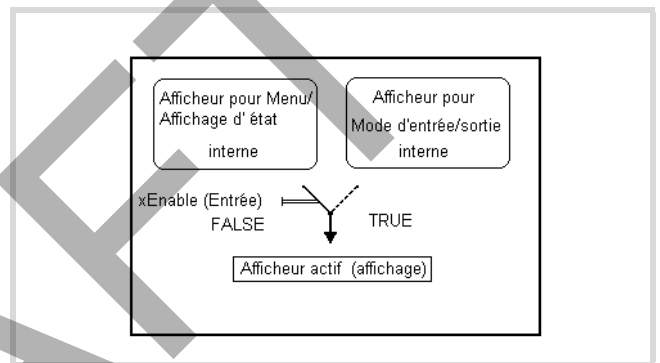


Figure 92 : Commutation entre Affichage d'état ↔ Mode saisie/affichage

Pour activer le mode Entrée/Sortie, la fonction "Disp_EnableDisplay" (→ page 77) de la bibliothèque "EC_Visu2.lib" doit faire l'objet d'un traitement continu dans le programme. C'est l'état au niveau de l'entrée de la fonction "xEnable" qui détermine le mode (→ figure 92) :

FALSE : affichage d'état de l'automate

TRUE : mode Saisie/Affichage

En mode Saisie/Affichage, l'afficheur indique les valeurs qui sont générées dans le programme utilisateur. Le programme actualise continuellement les valeurs et réceptionne les saisies via les touches (de direction). C'est à l'aide des fonctions et des blocs fonctionnels que vous indiquez la forme des variables, la représentation et le positionnement sur l'afficheur :

Le module fonctionnel "Disp_DisplayElement" vous permet de représenter une variable sur l'afficheur.

Le module fonctionnel "Disp_DisplayPage" vous permet de représenter une page comportant 8 variables.

Module fonctionnel "Disp_DisplayPage"

Pour chaque variable, déterminez l'utilisation souhaitée : affichage de textes ou saisie de valeurs.

Pour ce faire, vous devez paramétrer les entrées du module fonctionnel. Le module fonctionnel "Disp_DisplayPage" gère la commande du curseur. En cas de représentation de plusieurs variables nécessitant la saisie de valeurs, la première position destinée à la saisie est repérée par un curseur. A l'issue de la saisie, le curseur passe à la position suivante. Il est possible de réaliser dans le programme utilisateur une application avec représentation de plusieurs pages et appel d'une page. Utilisez pour cela les touches de direction P1, P2, P3, P4 et les touches de fonction ESC, DEL, ALT et OK, dont vous pouvez généralement scruter l'état au sein du programme.

Les informations concernant l'affichage des menus, la position actuelle du curseur et l'état des touches sont indiquées via les sorties du module fonctionnel "Disp_GetDisplayInfo".

Les blocs fonctionnels "Disp_DisplayElement" et "Disp_DisplayPage" vous permettent de définir des éléments. Le terme "Eléments" se rapporte aux blocs fonctionnels. Un élément est une variable dotée de propriétés complémentaires (positionnement sur l'afficheur, par exemple). La définition des propriétés complémentaires s'opère par le biais du paramétrage des entrées des blocs fonctionnels.

Les synoptiques suivants indiquent les autres fonctions/blocs fonctionnels.

Synoptique des fonctions/blocs fonctionnels

Les fonctions/blocs fonctionnels suivants (qui font partie de la bibliothèque "EC_Visu2.lib") vous permettent de créer des affichages ou d'influer sur eux en mode Saisie/Affichage.

Fonction/Module fonctionnel	Description
Disp_SetBacklight	Activation/désactivation du rétro-éclairage de l'afficheur de l'automate
Disp_RegisterVariable	Enregistrement de variables
Disp_CreateVariableList	Définition de la longueur de la liste pour variables
Disp_GetDisplayInfo	Scrutation des informations de l'afficheur
Disp_ClearScreen	Effacement de l'affichage
Disp_SetCursor	Définition de la position et du type de curseur
Disp_SetContrast	Définition du contraste de l'afficheur local
Disp_DisplayElement	Affichage d'un(e) seul(e) élément/variable
Disp_DisplayPage	Affichage d'une page avec 12 éléments max. pour textes, valeurs et saisie de plusieurs valeurs

Le tableau suivant montre au niveau de quel afficheur (afficheur de l'automate ou afficheur du MFD-CP4) agissent les fonctions/blocs fonctionnels.

Fonction/Module fonctionnel	Référence	Affichage sur l'afficheur de l'appareil ci-dessous
Disp_SetBacklight	Fonction	Automate
Disp_RegisterVariable	Fonction	–
Disp_CreateVariableList	Fonction	–
Disp_GetDisplayInfo	Module fonctionnel	–
Disp_ClearScreen	Fonction	Automate, MFD ¹⁾
Disp_SetCursor	Module fonctionnel	Automate, MFD ¹⁾
Disp_SetContrast	Module fonctionnel	Automate
Disp_DisplayElement	Module fonctionnel	Automate, MFD ¹⁾
Disp_DisplayPage	Module fonctionnel	Automate, MFD ¹⁾

1) Possible uniquement si la fonction Disp_EnableDisplay est activée

Description des principales fonctions/blocs fonctionnels

FUNCTION Disp_EnableDisplay: BOOL (*Commutation Affichage d'état <-> Mode Saisie/Affichage*)

```
VAR_INPUT
xEnable:      (* FALSE: affichage d'état, TRUE: mode Saisie/Affichage*)
xDisableESCKey: (*Libération de la touche ESC au niveau de l'afficheur local et du MFD-CP4:
                FALSE: libération
                TRUE:  touche verrouillée *)
END_VAR
(* Valeur en retour :
TRUE *)
```

A propos de xDisableESCKey :

Lorsque vous actionnez la touche ESC (condition préalable : la touche ESC est libérée) en mode Saisie/Affichage, il y a passage à l'Affichage d'état. Vous pouvez verrouiller la touche ESC en appliquant TRUE à l'entrée xDisableESCKey.

Si c'est de nouveau FALSE qui est appliquée à l'entrée "Enable", l'affichage d'état de l'automate réapparaît.

FUNCTION Disp_RegisterVariable : BOOL (* Définir une variable IEC en tant que variable de l'afficheur *)

```
VAR_INPUT
  sName:      (* Nom symbolique de la variable de l'afficheur *)
  dwAddress:  (* Adresse de la variable IEC appartenante *)
  eVarTyp:    (* Type de données de la variable IEC appartenante, voir DISP_VARTYP*)
END_VAR
(* Valeurs en retour:*)
(* TRUE: variable de l'afficheur enregistré avec succès*)
(* FALSE: liste de variables pleine *)

TYPE DISP_VARTYP :
( DISP_TYP_USINT := 0,
  DISP_TYP_UINT,
  DISP_TYP_UDINT,
  DISP_TYP_SINT,
  DISP_TYP_INT,
  DISP_TYP_DINT,
  DISP_TYP_BYTE,
  DISP_TYP_WORD,
  DISP_TYP_DWORD,
  DISP_TYP_STRING ) := DISP_TYP_UINT;
END_TYPE
```

Il est possible d'utiliser 50 variables. Si vous en avez besoin plus, veuillez l'indiquer via la fonction "Disp_CreateVariableList".

FUNCTION_BLOCK Disp_GetDisplayInfo (* Informations actuelles de l'état de l'afficheur *)

```

VAR_OUTPUT
byMenuLevel :(* Niveau menu : *)
(*0: Menu d'état *)
(*1: Menu principal *)
(*2: Menu principal / Programme *)
(*3: Menu principal / Régler Heure *)
(*4: Menu principal / Information *)
(*5: Menu spécial *)
(*6: Menu spécial / Sécurité *)
(*7: Menu spécial / Système *)
(*8: Menu spécial / Paramètres de démarrage *)
(*9: Menu spécial / Menu Langue *)
(*10: Menu spécial / Configuration *)
(*11- 14: inutilisé(e)(s)*)
(*15: Mode Saisie/Affichage *)
byActualLine: (*Position du curseur, ligne 1 - 4 *)
byActualColumn: (* Position du curseur, colonne 1 - 16 *)
xESCKeyDisabled: (* FALSE: actionner la touche ESC -> Menu d'état *)
(* TRUE: possibilité de scruter la touche ESC dans le programme utilisateur *)
xInputEnabled #) (* TRUE: lorsque les entrées xEnable et xEnableInput du module fonctionnel FB *)
Disp_DisplayPage = TRUE
(* FALSE: une entrée disabled *)
xInputActive #) lorsque les entrées xEnable et xEnableInput du module fonctionnel Disp_DisplayPage = TRUE et que la touche
ALT est actionnée*)
(* FALSE: saisie non active *)
(* #) En cas d'utilisation du module fonctionnel Disp_DisplayPage *)
END_VAR

```

FUNCTION_BLOCK Disp_DisplayElement (*Affichage d'un seul élément*)

```

VAR_INPUT
xEnable: (* Exécution lorsque l'entrée = TRUE *)
sName : (* Nom symbolique de l'élément *)
byLine : (* Afficher l'élément à la ligne 1 - 4 *)
byColumn : (* Afficher l'élément à la colonne 1 - 16 *)
eFont : (* Font, uniquement éléments du type STRING ! Voir DISP_FONTS*)
byDigits : (* Nombre de caractères, uniquement pour éléments numériques*)
byPrecision : (* Nombre de caractères après le point décimal, uniquement pour éléments numériques *)
eAttribut : (* Propriétés de l'élément : normal, reverse, blink. Voir DISP_ATTRIBUT*)
END_VAR
VAR_OUTPUT
eError (* Voir DISP_ERROR*)
END_VAR
(* Valeurs en retour:*)
(* DISP_ERROR_NO_ERROR:OK, aucune erreur*)
(* DISP_ERROR_INVALID_LINE: *)
(* DISP_ERROR_INVALID_COLUMN: en dehors de la plage de valeurs*)
(* DISP_ERROR_ELEMENT_NOT_FOUND: élément non trouvé*)
(* DISP_ERROR_INVALID_VARIABLE_TYP: en dehors de la plage de valeurs*)

TYPE DISP_FONTS :
(DISP_FONT_LATIN1 := 0,
DISP_FONT_LATIN2,
DISP_FONT_CYRILLIC ) := DISP_FONT_LATIN1;
END_TYPE
...

```

```

...
TYPE DISP_ATTRIBUT :
( DISP_ATTR_NORMAL := 0,
  DISP_ATTR_REVERSE,
  DISP_ATTR_BLINK ) := DISP_ATTR_NORMAL;
END_TYPE

```

FUNCTION_BLOCK Disp_DisplayPage (* Affichage d'une page *)

```

VAR_INPUT
xEnable:                (* TRUE: activer l'affichage *)
xEnableInput:           (* TRUE: activer la saisie*)
byNoOfElements:        (* Nombre d'éléments pour cette page 1 - 12*)
aElementDescription:ARRAY [1..12] OF (* Voir TYPE DISP_ElementDescription*)
DISP_ElementDescription:
END_VAR
VAR_OUTPUT
byError
END_VAR
(* Valeurs en retour :      *
(* 0:      OK, tous les éléments seront affichés*)
(* 1 - 12: erreur lors de l'affichage de l'élément "n" ou voir DISP_ERROR_INVALID_NO_OF_ELEMENTS*)

TYPE DISP_ElementDescription :      (* Description d'un élément d'affichage *)
STRUCT
    xEnable      : (* Réglage par défaut TRUE : l'élément sera affiché ; FALSE : l'affichage sera gelé*)
    xInputEnable : (* FALSE: Affichage de l'élément, voir figure 93; TRUE: Affichage valeur
                    (d'initialisation), saisie possible*)
    sName        : (* Nom symbolique de l'élément *)
    byLine       : (* Afficher l'élément à la ligne 1 - 4 *)
    byColumn     : (* Afficher l'élément à la colonne 1 - 16 *)
    eFont        : (* Font, uniquement éléments du type STRING ! Voir DISP_FONTS*)
    byDigits     : (* Nombre de caractères, uniquement pour éléments numériques*)
    byPrecision  : (* Nombre de caractères après le point décimal, uniquement pour éléments numériques *)
    diMinInputValue#: (* Valeur min. pour valeur de saisie, uniquement pour éléments numériques *)
    diMaxInputValue#: (* Valeur max. pour valeur de saisie, uniquement pour éléments numériques *)
    eAttribut    : (* Propriétés de l'élément : normal, reverse, blink. Voir DISP_ATTRIBUT*)
    xInputActiv#: (* TRUE : lorsque les entrées xEnable et xEnableInput du module fonctionnel
                    Disp_DisplayPage = TRUE*)
    xInputDone#: (* TRUE : à l'issue de la saisie de valeurs, avec actionnement de la
                    touche "OK". A repositionner sur FALSE par l'utilisateur !*)
    eError       : (*Voir DISP_ERROR*)
END_STRUCT
END_TYPE
END_VAR

# active, si xInputEnable = TRUE

(* Returnvalues: *)
(* DISP_ERROR_NO_ERROR,      OK, aucune erreur *)
(* DISP_ERROR_INVALID_LINE,  en dehors de la plage de valeurs : 1 - 4 *)
(* DISP_ERROR_INVALID_COLUMN, en dehors de la plage de valeurs : 1- 16*)
(*DISP_ERROR_ELEMENT_NOT_FOUND, (*DISP_ERROR_ELEMENT_NOT_FOUND, élément non trouvé*)
élément non trouvé*)

```

Relation entre DISP_DisplayPage.xEnable/ xEnableInput et DISP_ElementDescription.xInputEnable pour la saisie de valeurs

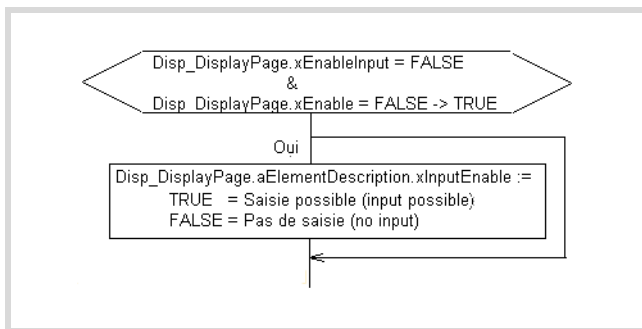


Figure 93 : Activation Affichage/Saisie de valeurs

Déroulement d'une saisie de valeurs

► Dans le programme, positionnez:

```

Disp_DisplayPage.xEnable = TRUE      (Affichage des
                                     valeurs/modifications
                                     visible)

Disp_DisplayPage.xEnableInput = TRUE (Saisie rendue
                                     possible)
  
```

► Actionnez la touche ALT au niveau de l'afficheur. Le curseur apparaît sur le premier élément "aElementDescription[1]" dont xInputEnable est positionnée sur TRUE.

► Appuyez sur la touche OK. La valeur est représentée sous la forme de base du type de données (exemple : TYP UINT : 00000).

► Vous pouvez modifier la valeur à l'aide des touches de direction.

- Choisissez l'emplacement de la valeur à l'aide des touches > ou <.
- Utilisez les touches ^ v pour modifier la valeur.

► Confirmez votre saisie à l'aide de la touche OK. Le curseur passe à la possibilité de saisie suivante (sur le second élément, par exemple).

Utilisez la touche ALT pour revenir au mode Saisie/Affichage.

Déroulement général concernant l'élaboration d'un programme

- Déclaration de la variable (de l'afficheur) destinée à l'affichage Saisie dans la liste "Globale_Variablen_Display" → figure 95
- Elaboration du programme
Elaboration de 3 programmes (pour chaque exemple) :
 - Programme de démarrage : génération d'une impulsion de démarrage (1er cycle)
 - PLC_PRG : programme utilisateur avec appel du programme "Visualisation"
 - Visualisation : programme destiné à la représentation des variables sur l'afficheur

Structure du programme "Visualisation"

- Durant le premier cycle :
 - Définir le nombre de variables de l'afficheur -> Fonction Disp_CreateVariableList (uniquement dans le cas où plus de 50 variables d'afficheur sont nécessaires !)
 - Enregistrer les variables de l'afficheur -> Fonction Disp_RegisterVariable (à exécuter de manière générale)
- Durant le cycle suivant (en fonction de l'application) :
 - Effacement de l'afficheur -> fonction Disp_ClearScreen
 - Eclairage de l'afficheur local -> fonction Disp_SetBacklight
 - Positionnement du curseur -> fonction Disp_SetCursor
 - Définition du contraste de l'afficheur local -> fonction Disp_SetContrast
 - Définition de propriétés (position, par ex.) des variables -> module fonctionnel Disp_DisplayElement ou Disp_DisplayPage
- Pour tous les cycles :
 - Démarrage de l'affichage au niveau de l'afficheur -> démarrage du module fonctionnel Disp_DisplayElement ou Disp_DisplayPage
 - Activation de l'affichage -> démarrage de la fonction Disp_DisplayEnable
 - Scrutation de l'état de l'afficheur -> fonction Disp_GetDisplayInfo

Exemple d'affichage de textes et de valeurs

(A l'aide du module fonctionnel Disp_DisplayElement)
 Il convient d'afficher les valeurs des variables "motor1" et "motor2". Ces deux valeurs sont modifiées en permanence par le programme utilisateur.

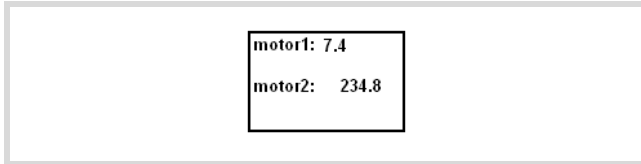


Figure 94 : Exemple d'affichage de textes et de valeurs

Actions exécutées via les entrées de l'automate

- I1 = FALSE: affichage d'état
- I1 = TRUE: mode Saisie/Affichage
- I2 = FALSE: touche ESC active
- I2 = TRUE: touche ESC verrouillée
- I3 = TRUE: représentation de la première ligne sur l'afficheur.
- I5 = TRUE: représentation de la troisième ligne sur l'afficheur.

Exécution

Cet exemple de programme comporte les programmes suivants :

- Programme de démarrage
 - Le programme de démarrage est appelé lors de l'événement système "Start".
 - La variable auxiliaire g_xFirstCycleAfterStartProgram est activée.
- PLC_PRG
 - 2 valeurs seront affectées d'un exposant
 - Le programme "Visualisation" est appelé.
- Visualisation
 - Enregistrement et positionnement des variables sur l'afficheur lors du premier cycle
 - La variable auxiliaire g_xFirstCycleAfterStartProgram est remise à zéro.
 - Activation du mode Saisie/Affichage (I1)
 - Lancer l'affichage (I3,I5)

Déclaration de variables

► Pour chaque élément de texte que vous souhaitez afficher ("motor1", par exemple), déclarez d'abord dans la liste "Globale_Variablen_Display" une variable de type "String", en suivant le modèle ci-dessous (voir également figure 95) :

```
VAR GLOBAL
    g_sDisp_String1 :STRING:='Motor1';
END_VAR
```

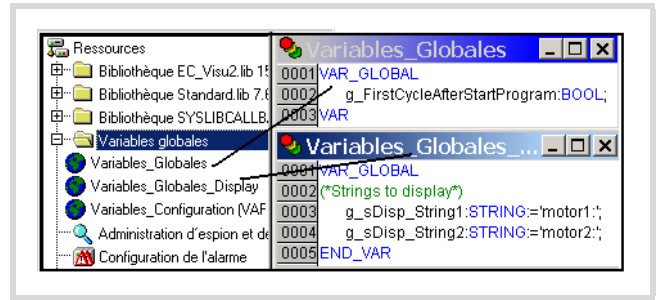


Figure 95 : Déclaration des variables au niveau de l'afficheur

Création de variables auxiliaires

- Pour le premier cycle de programme, appelez le programme "Startprogram" à l'aide de l'événement système "Start".
- Dans ce programme, activez une variable auxiliaire "g_xFirstCycleAfterStartProgram" que vous remettrez à zéro à l'issue du premier cycle. La variable auxiliaire doit être déclarée de manière globale. → figure 95

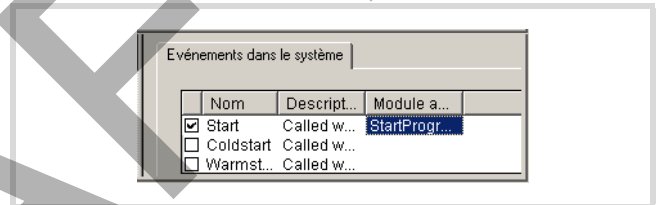


Figure 96 : définir événement système

Elaboration du programme "StartProgram"

► Ecrivez le programme "StartProgram" comme indiqué dans la → figure 97.

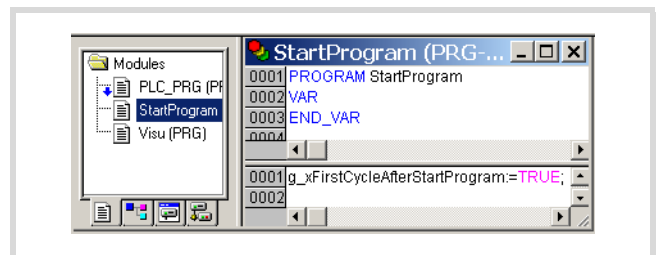


Figure 97 : Elaboration du programme de démarrage

Elaboration du programme "PLC_PRG"

```
PROGRAM PLC_PRG
VAR
    fbTimer1 :TON;
    (* Valeurs affichées concernant l'application *)
    byValue :BYTE;
    wValue :WORD;
END_VAR

fbTimer1(IN:=NOT fbTimer1.Q , PT:=t#50ms );
IF fbTimer1.Q = TRUE THEN
    byValue := byValue + 1;
    wValue := wValue + 1;
```

```
END_IF
```

```
Visualisation(); (* appeler la visualisation *)
```

Elaboration du programme "Visualisation"

C'est en fonction des variables auxiliaires

"g_xFirstCycleAfterStartProgram" que vous enregistrez les variables dont il convient d'afficher le texte/la valeur :

- Pour cela, programmez la fonction "Disp_RegisterVariable" à l'aide (par exemple) des paramètres suivants :
Disp_RegisterVariable ('S1',ADR(g_sDisp_String1), Disp_TYP_STRING). Le nom S1 est alors affecté à la variable.
- Pour afficher une valeur (de type octet), programmez une variable en appelant la fonction Disp_RegisterVariable (,V1', ADR(byValue), Disp_TYP_BYTE).

```
FUNCTION Disp_RegisterVariable : BOOL
(* Register one IEC-Variable for using as display variable *)
VAR_INPUT (* Symbolic name for display variable *)
sName :STRING(16); (* Address of corresponding IEC-Variable *)
dwAddress:DWORD; (* Datatyp of corresponding IEC-Variable *)
eVarTyp :DISP_VARTYP; (* Returnvalue: *)
END_VAR
```

Figure 98 : Fonction Disp_RegisterVariable

Dans cette partie du programme, vous pouvez également définir l'emplacement des variables sur l'afficheur, en indiquant la ligne (Line) et la colonne (Column). Pour cela, appelez le module fonctionnel "Disp_DisplayElement" et paramétrez les entrées sName, byLine, et byColumn. Exemple :

```
fbDisplayElement1.sName := 'S1';
fbDisplayElement1.byLine := 1;
fbDisplayElement1.byColumn := 1;
```

L'élément S1 avec le texte "motor1" s'afficherait sur la première ligne, à partir de la première colonne.

Pour l'affichage/la saisie de plusieurs éléments, appelez plusieurs fois dans la partie de programme suivante (qui fait l'objet d'un traitement continu) le module fonctionnel "Disp_DisplayElement" et attribuer aux entrées "xEnable" des entrées externes (I3, par exemple).

```
VAR
xIsDisplayEnabled: BOOL;
fbGetDisplayInfo: Disp_GetDisplayInfo;
fbDisplayElement1: Disp_DisplayElement;
fbDisplayElement2: Disp_DisplayElement;
fbDisplayElement3: Disp_DisplayElement;
fbDisplayElement4: Disp_DisplayElement;
byError: BYTE;
byValue: BYTE;
wValue: WORD;
END_VAR
```

```
(* Initialisation durant le premier cycle, après le lancement
du programme *)
```

```
IF g_xFirstCycleAfterStartProgram = TRUE THEN
Disp_ClearScreen(xEnable:=TRUE);
Disp_RegisterVariable('S1', ADR(g_sDisp_String1),
DISP_TYP_STRING);
Disp_RegisterVariable('S2', ADR(g_sDisp_String2),
DISP_TYP_STRING);
Disp_RegisterVariable('V1', ADR(PLC_PRG.byValue),
DISP_TYP_BYTE);
Disp_RegisterVariable('V2', ADR(PLC_PRG.wValue),
DISP_TYP_WORD);
```

```
fbDisplayElement1.sName := 'S1';
fbDisplayElement1.byLine := 1;
fbDisplayElement1.byColumn := 1;
```

```
fbDisplayElement2.sName := 'S2';
fbDisplayElement2.byLine := 3;
fbDisplayElement2.byColumn := 1;
```

```
fbDisplayElement3.sName := 'V1';
fbDisplayElement3.byLine := 1;
fbDisplayElement3.byColumn := 8;
fbDisplayElement3.byDigits := 4;
fbDisplayElement3.byPrecision := 1;
```

```
fbDisplayElement4.sName := 'V2';
fbDisplayElement4.byLine := 3;
fbDisplayElement4.byColumn := 8;
fbDisplayElement4.byDigits := 6;
fbDisplayElement4.byPrecision := 1;
```

```
(* Le premier cycle est terminé, remettre le drapeau à zéro *)
```

```
g_xFirstCycleAfterStartProgram := FALSE;
END_IF

xIsDisplayEnabled := Disp_EnableDisplay(I1, I2);
fbDisplayElement1( xEnable:= I3 );
fbDisplayElement2( xEnable:= I5 );
fbDisplayElement3( xEnable:= I3 );
fbDisplayElement4( xEnable:= I5 );
```

- Lancez les programmes.

Exemple d'affichage d'une page avec textes et possibilités de saisie

(A l'aide du module fonctionnel Disp_DisplayPage)

Il convient de réaliser l'affichage suivant.

Le contenu des variables MO11 et TEMP8 est modifié en permanence par le programme utilisateur.

MO11	3.5
TIM14	0
MOZ14	0
TEMP8	183

Figure 99 : Exemple d'une page destinée à la saisie et à l'affichage

Actions via les entrées de l'automate

- I1 = FALSE: affichage d'état
- I1 = TRUE: mode Saisie/Affichage
- I2 = FALSE: touche ESC active
- I2 = TRUE: touche ESC verrouillée
- I3 = TRUE: Les valeurs sont actualisées par le programme.
- I4 = TRUE: Saisie active.

Exécution

Cet exemple de programme comporte les programmes suivants :

- STARTPROGRAM : (il est appelé lors de l'événement système "Start")
 - La variable auxiliaire "g_xFirstCycleAfterStartProgram" est activée.
- PLC_PRG:
 - 2 valeurs seront dénombrées
 - Le programme "Visualisation" est appelé.
- Visualisation
 - Enregistrement et positionnement des variables sur l'afficheur lors du premier cycle.
 - La variable auxiliaire g_xFirstCycleAfterStartProgram est remise à zéro.
 - Activation du mode Saisie/Affichage (I1)
 - Libération touche ESC (I2).
 - Démarrer l'affichage (I3).
 - Démarrer la saisie (I4).

Déclarer les variables au niveau de l'afficheur

- Pour chaque élément de texte que vous souhaitez afficher ("MO11", par exemple), déclarez d'abord dans le dossier "Globale_Variablen_Display" une variable de type String, selon le modèle suivant :

```
VAR_GLOBAL
g_sDisp_String1 :STRING := 'MO11 :';
g_sDisp_String2 :STRING := 'TIM14 :';
g_sDisp_String3 :STRING := 'MOZ14 :';
g_sDisp_String4 :STRING := 'TEMP8 :';
END_VAR
```

- Créez une variable auxiliaire et écrivez le programme "Startprogram" comme indiqué dans l' "Exemple d'affichage de textes et de valeurs".
- Ecrivez les programmes PLC_PRG et Visualisation comme indiqué dans l'exemple suivant :

```
PROGRAM PLC_PRG (*****)
VAR
fbTimer1 :TON;
(* Valeurs affichées concernant l'application *)
byValue :BYTE;
wValue :WORD;
dwValue :DWORD;
usiValue :USINT;
siValue :SINT;
END_VAR

-----
fbTimer1(IN:=NOT fbTimer1.Q , PT:=t#50ms );
IF fbTimer1.Q = TRUE THEN
usiValue := usiValue + 1;
byValue:=byValue+1;
END_IF

Visualisation(); (* appeler la visualisation *)
```

```

PROGRAM Visualisation (*****)
VAR
    xIsDisplayEnabled      :BOOL;
    fbDisplayPage1        :Disp_DisplayPage;
    byError                :BYTE;
    siValue                :SINT;
END_VAR

-----
(* Initialisation durant le premier cycle, après le lancement du programme *)
IF g_xFirstCycleAfterStartProgram = TRUE THEN

Disp_RegisterVariable('S1', ADR(g_sDisp_String1), DISP_TYP_STRING);
Disp_RegisterVariable('S2', ADR(g_sDisp_String2), DISP_TYP_STRING);
Disp_RegisterVariable('S3', ADR(g_sDisp_String3), DISP_TYP_STRING);
Disp_RegisterVariable('S4', ADR(g_sDisp_String4), DISP_TYP_STRING);
Disp_RegisterVariable('V1', ADR(PLC_PRG.byValue), DISP_TYP_BYTE);
Disp_RegisterVariable('V2', ADR(PLC_PRG.wValue), DISP_TYP_WORD);
Disp_RegisterVariable('V3',ADR(PLC_PRG.dwValue),
DISP_TYP_DWORD);
Disp_RegisterVariable('V4', ADR(PLC_PRG.usiValue), DISP_TYP_USINT);

fbDisplayPage1.aElementDescription[1].sName      := 'S1';
fbDisplayPage1.aElementDescription[1].byLine    := 1;
fbDisplayPage1.aElementDescription[1].byColumn  := 1;
fbDisplayPage1.aElementDescription[2].sName      := 'S2';
fbDisplayPage1.aElementDescription[2].byLine    := 2;
fbDisplayPage1.aElementDescription[2].byColumn  := 1;
fbDisplayPage1.aElementDescription[3].sName      := 'S3';
fbDisplayPage1.aElementDescription[3].byLine    := 3;
fbDisplayPage1.aElementDescription[3].byColumn  := 1;
fbDisplayPage1.aElementDescription[4].sName      := 'S4';
fbDisplayPage1.aElementDescription[4].byLine    := 4;
fbDisplayPage1.aElementDescription[4].byColumn  := 1;

fbDisplayPage1.aElementDescription[5].sName      := 'V1';
fbDisplayPage1.aElementDescription[5].byLine    := 1;
fbDisplayPage1.aElementDescription[5].byColumn  := 13;
fbDisplayPage1.aElementDescription[5].byDigits  := 4;
fbDisplayPage1.aElementDescription[5].byPrecision := 1;
fbDisplayPage1.aElementDescription[5].xInputEnable := FALSE;
fbDisplayPage1.aElementDescription[5].diMinInputValue := 1;
fbDisplayPage1.aElementDescription[5].diMaxInputValue := 100;
fbDisplayPage1.aElementDescription[6].sName      := 'V2';
fbDisplayPage1.aElementDescription[6].byLine    := 2;
fbDisplayPage1.aElementDescription[6].byColumn  := 12;
fbDisplayPage1.aElementDescription[6].byDigits  := 5;
fbDisplayPage1.aElementDescription[6].byPrecision := 0;
fbDisplayPage1.aElementDescription[6].xInputEnable := TRUE;
fbDisplayPage1.aElementDescription[6].diMinInputValue := 0;
fbDisplayPage1.aElementDescription[6].diMaxInputValue := 33333;
fbDisplayPage1.aElementDescription[7].sName      := 'V3';

```



```

fbDisplayPage1.aElementDescription[7].byLine      := 3;
fbDisplayPage1.aElementDescription[7].byColumn   := 8;
fbDisplayPage1.aElementDescription[7].byDigits   := 9;
fbDisplayPage1.aElementDescription[7].byPrecision := 0;
fbDisplayPage1.aElementDescription[7].xInputEnable := TRUE;
fbDisplayPage1.aElementDescription[7].diMinInputValue := 0;
fbDisplayPage1.aElementDescription[7].diMaxInputValue := 4444444;
fbDisplayPage1.aElementDescription[8].sName      := 'V4';
fbDisplayPage1.aElementDescription[8].byLine     := 4;
fbDisplayPage1.aElementDescription[8].byColumn  := 13;
fbDisplayPage1.aElementDescription[8].byDigits  := 4;
fbDisplayPage1.aElementDescription[8].byPrecision := 0;
fbDisplayPage1.aElementDescription[8].xInputEnable := TRUE;
fbDisplayPage1.aElementDescription[8].diMinInputValue := 4;
fbDisplayPage1.aElementDescription[8].diMaxInputValue := 400;

(* Le premier cycle est terminé, remettre le drapeau à zéro *)
g_xFirstCycleAfterStartProgram :=
FALSE;
END_IF

xIsDisplayEnabled := Disp_EnableDisplay(I1, I2);
fbDisplayPage1( xEnable:= I3 , xEnableInput:= I4,
byNoOfElements:= 8, byError =>byError );

IF fbDisplayPage1.aElementDescription[7].xInputDone = TRUE THEN
siValue := PLC_PRG.siValue; (*Valeur pour le traitement interne*)
fbDisplayPage1.aElementDescription[7].xInputDone := FALSE;
END_IF

```

► Lancez les programmes.

Afficheur multifonctions MFD-CP4 à l'EC4-200

L'afficheur multifonctions (MFD-CP4) vous permet d'obtenir de manière externe les mêmes possibilités d'affichage et de commande de l'automate.

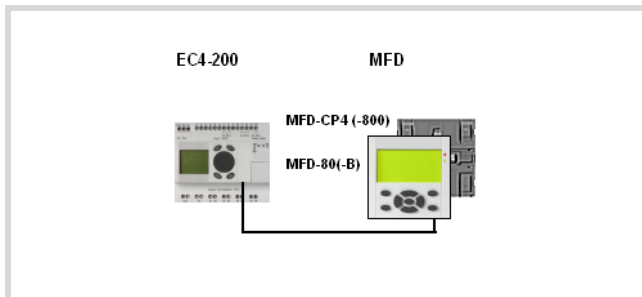


Figure 100 : EC4-200 avec MFD-CP4

Lorsque le MFD-CP4 raccordé à l'EC4-200 est mis sous tension, il se trouve en mode Terminal. Dans ce mode, il prend en compte l'information de l'afficheur de l'automate et en assure la visualisation sur l'afficheur (du MFD). Lorsque l'automate se trouve en mode Saisie/Affichage, il se comporte comme l'afficheur de l'automate.

Pour régler les paramètres du MFD, procédez à la commutation du MFD-CP4 en mode Local. Ces paramètres sont les suivants :

- Contrast
- Eclairage
- Langue de menus: adapter la désignation de paramètres à la langue du pays
- Interface COM
- easyNet=Identnumber (n° d'identification) 0, 1, ..., 8
 - 0 : le MFD-CP4 communique avec l'appareil réellement raccordé.
 - 1...8 : easyNet du participant easyNet : sélection du participant (-easyNet au niveau de easyNet
Si l'EC4-200 est un participant de easyNet, le MFD-CP4 peut communiquer via l'EC4-200 avec le participant sélectionné.
- Vitesse de transmission: 9600 (19200) Baud

Pour pouvoir commuter entre le mode Terminal et le mode Local du MFD-CP4, actionnez la touche "*" (présente uniquement sur le MFD-CP4).

Le passage en mode Terminal n'est possible qu'à partir du menu principal du mode Local.

(Voir également le manuel relatif au MFD-CP4 (AWB2528-1548), chapitre "Réglages")

Menu principal: COM...
LANGUE DES MENUS...
ECLAIRAGE: 80%
CONTRAST: +1

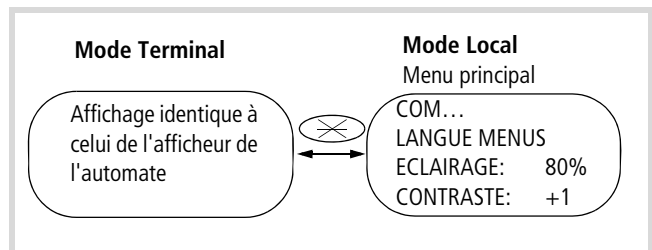


Figure 101 : Commutation entre mode Terminal ↔ mode Local

En mode Local, les touches de l'afficheur MFD sont actives. Reportez-vous au manuel d'utilisation relatif au module de communication/d'alimentation MFD-CP4 (AWB2528-1548GB).

Réalisation du MFD

Le MFD-CP4 est un appareil formé de plusieurs composantes. L'afficheur à proprement parler (unité d'affichage et de commande MFD-80(-B)) est conçu pour le montage encastré en face avant dans une porte d'armoire. Il se monte par encliquetage sur le module d'alimentation et de communication MFD-CP4(-800) à fixation arrière. Vous devez réaliser la liaison avec l'EC4-200 (interface multifonctions) à l'aide du câble MFD-CP4-800-CAB5.

Pour toute information complémentaire sur l'utilisation, le raccordement et les caractéristiques techniques de l'appareil, reportez-vous au manuel utilisateur relatif au module d'alimentation et de communication MFD-CP4 (AWB2528-1548D).

17 Le réseau easyNet

Vue d'ensemble

Le réseau easyNet repose sur le réseau CAN qui permet un échange de données système et de données du processus. Il est dimensionné pour 8 participants (automates). Chaque participant reçoit un n° easyNet allant de 1 à 8 et peut échanger des données avec tous les autres participants. Vous disposez des formats de données suivants : bit, octet, mot et double-mot. Le participant doté de easyNet1 doit toujours être présent. Il assure la gestion de la communication au sein du réseau. Utilisé en tant que participant unique, il peut communiquer avec des participants d'E/S décentralisés (Remote I/O).

Les participants de type "Remote I/O" sont des automates sans programme (un appareil easy800, par exemple) dont les sorties peuvent être activées par le participant doté de easyNet1 et dont les entrées peuvent être scrutées par ce même participant.

Si easyNet se compose d'automates XC200-/EC4-200/MFD4 et d'appareils easy800, vous devez effectuer la programmation et la configuration avec les logiciels de programmation easySoft et easySoft CoDeSys.

→ Dans la suite du présent document, les participants qui seront programmés à l'aide de easySoft-CoDeSys seront désignés par "participants IEC". Les participants programmés à l'aide de easySoft seront quant à eux désignés par "participants easy".

Le raccordement des participants IEC à easyNet est exposé en détail dans la documentation en ligne "Description de SysLibEasyNet.lib pour le couplage d'automates XC à un réseau easyNet" (SysLibEasyNet-D.PDF). Ce document aborde en particulier les fonctions conçues pour le transfert des données ainsi que les types de données et les structures. Le fichier PDF a été enregistré dans le répertoire CAA-Targets\Moeller\Lib_EC4P_200 lors de l'installation du CD relatif au produit.

Le réseau easyNet vous permet de transférer les données suivantes:

- Données de l'utilisateur
 - Etat des entrées et des sorties
 - Données provenant de l'automate (Id1) et à destination de l'automate avec E/S décentralisées (Remote I/O), en vue de l'activation des sorties
 - Emission ciblée de données à un participant ou réception de données d'un participant
 - Emission sur le réseau de données auxquelles peuvent accéder plusieurs participants (Broadcast).
- Données de système
 - Affichage d'état Run/Stop
 - Etat du programme (programme présent/absent)
 - Synchronisation des horloges des appareils

- Données pour la configuration des participants d'easyNet
- Données destinées à l'établissement d'une liaison en ligne entre le système de programmation (easySoft) et un participant NET quelconque.

→ Les chapitres relatifs à easyNet reposent essentiellement sur des exemples faisant intervenir des appareils easy800. Pour une fonction easyNet identique, l'appareil easy800 peut être remplacé par un appareil de visualisation MFD-Titan interconnectable en réseau. De la même façon, il est possible de remplacer le XC200 par un MFD4.

Surveillance de easyNet

Vitesse de transmission

Dans easyNet, chaque participant doit être réglé sur la même vitesse de transmission. Les participants IEC tels que les MFD4 ou les XC200, par exemple, peuvent être exploités sur le réseau easyNet avec une vitesse de transmission de 50 ou 125 Kbits/s et un EC4-200 avec une vitesse de transmission de 10, 20, 50, 125, 250, ou 500 Kbits/s. Les participants easy travaillent avec des vitesses de transmission de 10, 20, 50, 125, 250, 500 ou 1000 Kbits/s. Si vous exploitez conjointement des participants easy et IEC sur un même réseau, vous devez impérativement choisir une vitesse de transmission commune. C'est la seule possibilité !

Charge du bus

Veillez à ce que la charge du bus n'excède pas 70 %. Vous pouvez contrôler la charge du bus à l'aide de la commande "canload" du navigateur "PLC-Browser". La charge du bus s'affiche après exécution de la commande. Réitérez cette commande plusieurs fois pour pouvoir compenser d'éventuelles variations de charge du bus.

Réception de télégrammes CAN

Vous avez la possibilité de surveiller la réception de télégrammes CAN. Pour toute information complémentaire à ce sujet, reportez-vous à la documentation en ligne "Description de SysLibEasyNet.lib pour le couplage d'automates XC à un réseau easyNet", chapitre "Remarques sur la programmation/Recherche d'erreurs" (SysLibEasyNet-D.PDF). Le fichier PDF a été enregistré dans le répertoire CAA-Targets\Moeller\Lib_EC4P_200 lors de l'installation du CD relatif au produit..

Emission/réception des données de l'utilisateur

L'accès du EC4-200 aux données des participants easyNet s'opère via les fonctions NET_UPDATE et NET_GET, dans le programme utilisateur (Applicationprogram). Ces fonctions font partie intégrante de la bibliothèque SysLibEasyNet.lib. La transmission des données proprement dite entre un participant IEC et d'autres participants est assurée par la "tâche protocole" (Protcolltask), qui est non visible pour l'utilisateur et travaille indépendamment du programme utilisateur.

La tâche protocole assure également des tâches administratives telles que la surveillance des participants, la synchronisation des horloges et gestion des accès de programmation.

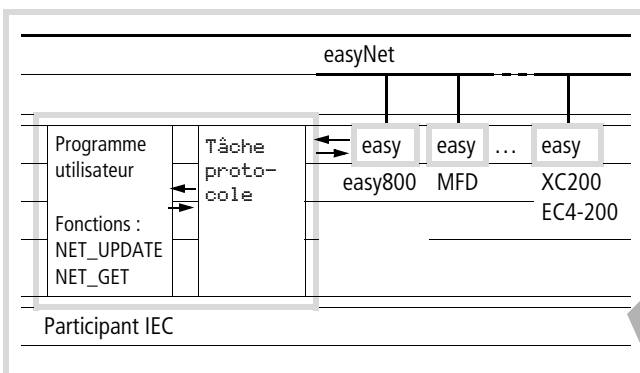


Figure 102 : Transmission de données entre un participant IEC et d'autres participants

Fonction NET_UPDATE

La fonction NET_UPDATE permet un échange de données entre le programme utilisateur et la tâche protocole.

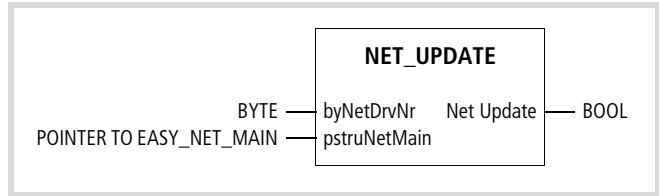


Figure 103 : Fonction NET_UPDATE

L'appel de la fonction NET_UPDATE doit s'opérer une seule fois par cycle du programme utilisateur. C'est l'assurance de disposer en permanence des données actuelles des participants easyNet ou de pouvoir alimenter les participants easyNet avec les données actuelles du participant local.

Fonction NET_GET

Cette fonction permet à un participant easyNet d'aller chercher une valeur (de donnée) déposée sur le bus à l'aide d'une commande PUT par un autre participant easyNet.

Il est possible d'aller chercher une valeur de donnée à chaque appel de la fonction NET_GET. La valeur de donnée "présente sur le réseau" est sélectionnée à l'aide des entrées byNET_easyNet et byModulNumber de la structure EASY_NET_GET. L'appel de la fonction entraîne l'inscription des données actuelles dans la structure par la tâche protocole. La fonction NET_GET peut être appelée autant de fois que vous souhaitez lire des valeurs de données.

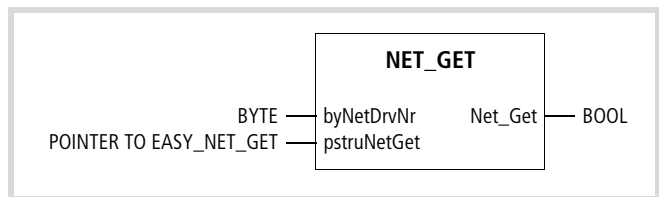


Figure 104 : Fonction NET_GET

Transmission des données

Il existe 3 possibilités relatives à la transmission des données :

- Transmission de données entre un participant doté de easyNet1 et un appareil de type Remote I/O, → page 89
- Transfert des blocs de données de bits entre plusieurs participants, → page 90
- Transmission de doubles-mots (32 bits) entre plusieurs participants selon le principe PUT-GET, → page 91

Transmission de données entre un participant doté de easyNet1 et un appareil de type Remote I/O

Les états des entrées et des sorties des automates ou des appareils de type Remote I/O (E/S décentralisées) peuvent être lus par tous les autres participants. Si un appareil easy800 (par exemple) est utilisé en tant qu'appareil Remote I/O, les participants easyNet peuvent scruter les entrées de ce easy800. Le participant doté de easyNet=1 peut en outre activer les sorties de cet appareil easy800. Ceci vaut également pour l'extension de easy800.

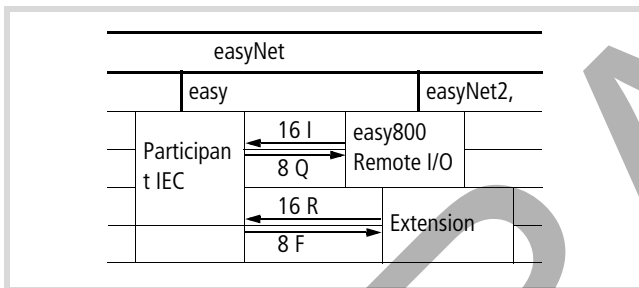


Figure 105 : Transmission des données entre un participant IEC et un appareil de type Remote I/O

Pour activer les sorties ou scruter les entrées de easy800 (Id2), appeler la fonction NET_UPDATE dans le programme utilisateur du participant portant l'Id1, comme indiqué dans l'exemple suivant.

A l'entrée "pstruNetMain", entrez un pointeur au niveau d'une structure du type EASY_NET_MAIN déclarée dans le programme utilisateur. Saisissez les données de transmission dans cette structure → figure 106.

Dans le programme, la valeur 5 (1 octet) est saisie via la variable de la structure:

```
NET_MAIN.SND.aToID[2].byQ:=5;
```

A l'état RUN du participant, la valeur est transmise après chaque modification aux sorties de l'appareil easy800.

Les 16 entrées de easy800 sont prises en compte par le participant via les variables de la structure NET_MAIN.RCV[2].wl.

```

0001 PROGRAM PLC_PRG (* IEC-station (ID1) <-> easy800 (ID2) *)
0002 VAR
0003   NET_MAIN: EASY_NET_MAIN;
0004   INPUTS: WORD;
0005 END_VAR
0001 Net_Update(byNetDrvNr:=0, PstruNetMain:=ADR(NET_MAIN));
0002
0003 NET_MAIN.SND.aToID[2].byQ:=5;
0004 (* IEC-station -> easy800 Output *)
0005
0006 INPUTS:= NET_MAIN.RCV[2].wl;
0007 (* IEC-station <- easy800 Input *)
0008
0009
    
```

Figure 106 : Exemple de programme pour la transmission des données entre un participant doté de easyNet1 et un appareil de type Remote I/O

La figure 107 illustre la représentation en ligne du programme : c'est de là que proviennent les données d'émission et de réception. Tenez compte du fait que la valeur de sortie 16#05 (aToID[2].byQ) envoyée à easy800 est reçue et automatiquement renvoyée par easy800 à l'entrée (RCV[2].byQ).

```

0001 NET_MAIN
0002   INFO
0003   SND
0004     byQ = 16#00
0005     byS = 16#00
0006     wl = 16#0000
0007     wR = 16#0000
0008   PUT
0009   aToID
0010     aToID[1]
0011     aToID[2]
0012       byQ = 16#05 (* to easy800 *)
0013       byS = 16#00
0014       dwSN = 16#00000000
0015     aToID[3]
0016     aToID[4]
0017     aToID[5]
0018     aToID[6]
0019     aToID[7]
0020     aToID[8]
0021     xStartClockSynchronisation = FALSE
0022     xAcknowledgeActiveError = FALSE
0023   RCV
0024     RCV[1]
0025     RCV[2]
0026       byQ = 16#05 (* to XC200 *)
0027       byS = 16#00 (*EC4-200*)
0028       wl = 16#2007
0029       wR = 16#0000
0030       dwRN = 16#00000000
0031     RCV[3]
0032     RCV[4]
    
```

1) easy800 Inputdata (16#05) return to XC200 /EC4-200
 2) Input Data: Hex. 007 = 7 dez.

Figure 107 : Représentation en ligne de la transmission des données entre un participant doté de easyNet1 et un appareil de type Remote I/O

Transfert des blocs de données de bits entre plusieurs participants

Chaque participant peut envoyer de manière ciblée un bloc de données de 32 bits à un autre participant, via easyNet. Pour cela, appelez la fonction NET_UPDATE dans le programme et indiquez le numéro de participant X du destinataire, via la structure (EASY_NET_MAIN) (SND.aToID[X].dwSN). Inscrivez dans la variable dwSN les données à envoyer.

Pour recevoir des données provenant d'autres participants, utilisez la structure (EASY_NET_MAIN).RCV[x], dans laquelle x correspond au n° NET-easyNet du participant dont vous souhaitez recevoir des données.

La figure 108 met en évidence que, pour les blocs de données, il existe dans chaque participant un tableau (Array) unidimensionnel à 8 éléments destiné à l'émission et à la réception. Il y a une affectation directe entre l'élément d'émission d'un participant et l'élément de réception d'un autre participant.

La position de l'élément (aToID) dans le tableau d'émission ou (RCV) dans le tableau de réception de l'EC4-200 correspond respectivement à l'easyNet du participant récepteur ou émetteur.

La valeur reçue se situe dans un élément du tableau de réception d'un EC4-200 dont la position est identique à l'easyNet du participant d'émission.

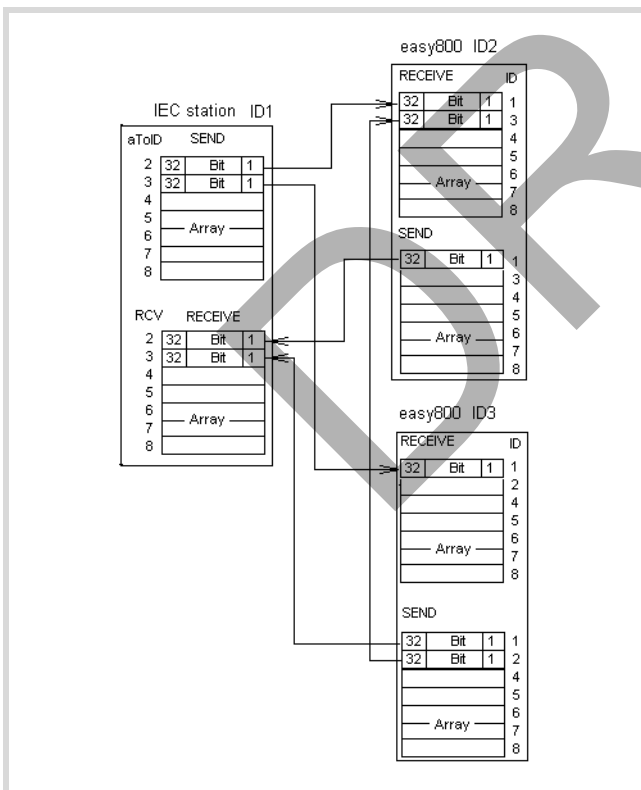


Figure 108 : Vue d'ensemble des blocs de données

1ère liaison:

La valeur "a" (32 bits) est transmise de l'EC4-200 (Id1) vers easy800 (Id2).

Il convient de programmer la structure suivante dans l'EC4-200 :

```
NET_MAIN.SND.aToeasyNet[2].dwSN:=a;
```

Le traitement de la valeur peut s'opérer dans easy800, par scrutation des bits d'entrée 1RN1 à 1RN32.

2ème liaison

Un bloc de données avec la valeur 7 (3_{hex}) doit être envoyé par l'automate easy800 (Id3) à l'automate EC4-200 (Id1).

Il convient de programmer et d'activer dans easy800 les 3 sorties 1SN1, 1SN2 et 1SN3.

Pour la scrutation dans l'EC4-200, il convient de programmer la variable de structure suivante :

```
INPVAL:=NET_MAIN.RCV[3].dwRN;
```

La variable INPVAL met la valeur à disposition.

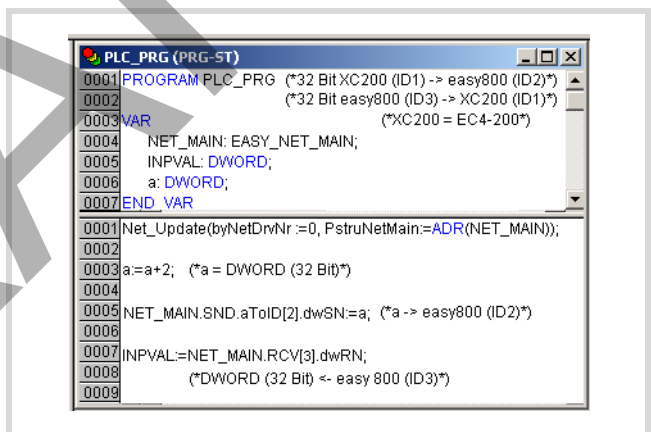


Figure 109 : Exemple de programme pour la transmission de blocs de données de type bit (possibilité d'utiliser également un MFD4 au lieu d'un XC200 doté de easyNet1)

La figure 110 illustre la représentation en ligne du programme de la figure 109 : c'est de là que proviennent les données d'émission et de réception.

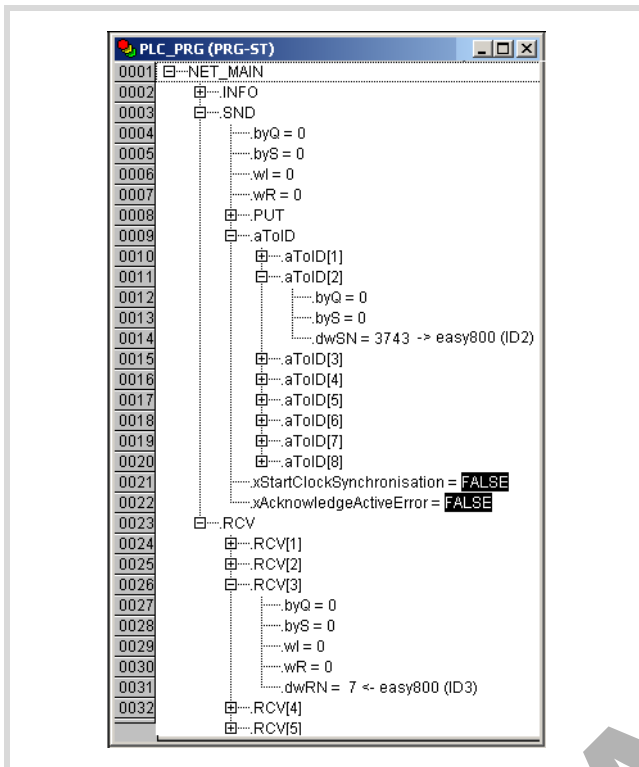


Figure 110 : Représentation en ligne du programme

Transmission de doubles-mots (32 bits) entre plusieurs participants selon le principe PUT-GET

PUT = Fournir la date sur le réseau ; GET = Capturer la date sur le réseau

Selon ce principe, un participant fournit une date (double-mot) sur le réseau (PUT), celle-ci étant repérée par un numéro de module (n° MN pour les participants IEC, n° PT pour les participants easy). Les autres participants peuvent scruter la date par indication de l'easyNet du participant émetteur et du n° de module (GET).

Chaque participant peut fournir des dates de façon séquentielle sur le réseau, à l'aide des numéros MN/PT 1, 2, ... 32 (PUT).

Pour les automates EC4-200, l'action PUT nécessite l'appel de la fonction NET_UPDATE et l'indication des éléments des variables des structures EASY_NET_PUT/byModuleNumber et /dwData.

Pour l'action GET, il convient d'appeler la fonction NET_GET et d'indiquer les éléments des variables des structures EASY_NET_GET/byModuleNumber et /dwGetData.

Sur les appareils easy800, les actions PUT et GET sont réalisés à l'aide de modules.

Exemple

Dans l'exemple suivant, l'EC4-200 (Id1) fournit la date 3747_{hex} (MN1) sur le réseau (PUT). Les automates easy800 scrutent cette date à l'aide d'un module GET. Le n° du module (PT) dans les automates easy800 (Id2 et Id3) dépend du n° MN de la date de l'émetteur (ici : MN1 → PT1).

L'appareil easy800 (Id3) fournit sur le réseau deux dates, chacune à l'aide d'un module PUT. Le module prévu pour la seconde date et comportant la valeur 9774_{hex} reçoit le n° PT 2.

L'EC4-200 scrute cette date. Il convient pour cela de saisir au niveau de la variable de structure :

```
(EASY_NET_GET).byModuleNumber := 2;
(EASY_NET_GET).byNET_ID := 3;
```

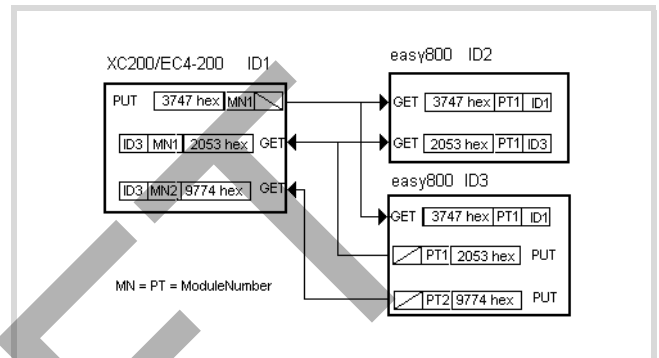


Figure 111 : Transfert des données selon le principe PUT-GET (également valable pour MFD4)

La séquence de programme suivante peut être utilisée pour exécuter une action PUT au niveau des EC4-200.

Si l'entrée IX0.0 est positionnée à 1 (=TRUE) une fois que xTransferPending est passé à FALSE, la séquence de programme est lancée. La valeur de la variable "Date" (MN1) est fournie sur le réseau.

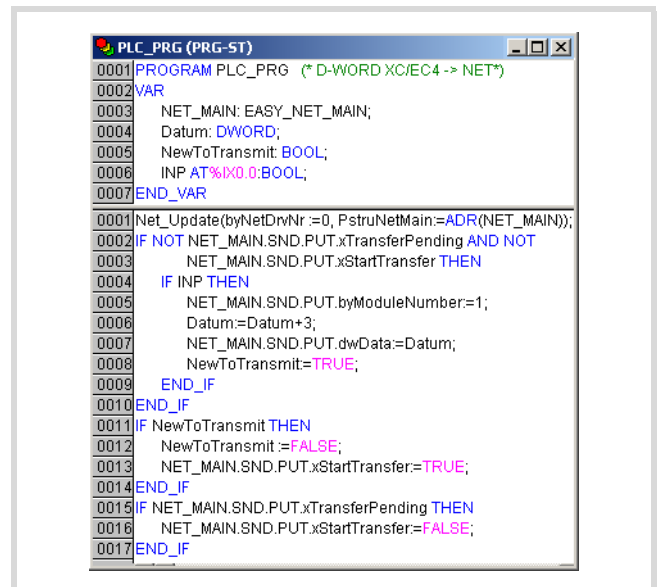


Figure 112 : Transfert des données selon le principe PUT-GET (PUT)

La figure 113 montre un exemple d'utilisation de la fonction NET_GET. Dans cet exemple, l'appareil doté de easyNet 3 scrute la valeur du module doté du numéro 2.

Il convient de saisir à l'entrée "pstruNetGet" un pointeur au niveau d'une structure du type EASY_NET_GET déclarée dans le programme utilisateur, à partir de laquelle vous lisez les données fournies sur le réseau par d'autres participants à l'aide d'une action PUT → figure 112.

```

PLC_PRG (PRG-ST)
0001 PROGRAM PLC_PRG
0002 (*D-WORD easy800(ID3)-> XC200 (ID1)*)
0003 VAR
0004   Getdatastruct: EASY_NET_GET;
0005   xSuccess: BOOL;
0006   RecData: DWORD;
0007   Errorprog: BYTE;
0008 END_VAR
0001 Getdatastruct.byNET_ID:= 3;
0002 Getdatastruct.byModuleNumber:= 2;
0003
0004 xSuccess := Net_Get(
0005   byNetDrvNr:=0,
0006   pstruNetGet:=ADR(Getdatastruct));
0007 IF xSuccess THEN
0008   IF Getdatastruct.xNewDataReceived THEN
0009     RecData:=Getdatastruct.dwGET_Data;
0010   END_IF
0011 ELSE
0012   (*Error handling*)
0013   Errorprog:=Getdatastruct.byError;
0014 END_IF
nm14
  
```

Figure 113 : Transmission de données selon le principe PUT-GET, ici : GET (possibilité d'utiliser également un MFD4 au lieu d'un XC200)

Configuration d'un EC4-200 avec easy800 sur un réseau easyNet

Chaque participant doit être configuré dans easyNet : il reçoit une adresse réseau (easyNet) et des paramètres (tels que SEND IO) qui déterminent le comportement du participant au sein du réseau.

La configuration s'opère dans le logiciel de programmation de chaque participant :

- easySoft pour easy800
- easySoft-CoDeSys pour MFD-CP8-..., MFD4 et EC4-200.

Configuration dans easySoft

easySoft vous guide tout au long de la configuration du réseau. Dès que vous ajoutez un participant à un participant jusque-là unique, les deux participants sont reliés symboliquement via le réseau. Le premier reçoit le n° easyNet1. Le réglage des easyNet destinés aux autres participants et de leurs paramètres s'opère comme suit :

- ▶ Cliquez sur l'onglet "Paramètres de communication".
- ▶ Sélectionnez l'adresse NET-easyNet via le menu NET-easyNet.
- ▶ Dans le champ "Configurateur NET", activez/désactivez les fonctions "Send IO" et "Remote RUN" (uniquement pour easyNet = 2...8).

Au sein d'un réseau faisant intervenir des appareils easy800 avec des automates XC200, MFD4 ou EC4-200, il convient d'intégrer dans le Configurateur easySoft tous les participants, y compris les participants IEC. Vous devez en outre paramétrer les participants IEC dans le Configurateur de easySoft-CoDeSys.

- ▶ Créez un schéma de commande pour chaque appareil easy800 et un programme pour les participants IEC.
- ▶ Reliez le PC à un participant IEC et chargez le programme/schéma de commande (y compris la configuration) vers l'appareil.

D'autres possibilités de configuration sont décrites dans le paragraphe « Programmation via easyNet (Routage) », page 95.

Configuration dans easySoft-CoDeSys

La configuration et la programmation d'un EC4-200 s'opèrent à l'aide du logiciel de programmation easySoft-CoDeSys.

- ▶ Pour cela, activez le réglage easyNet dans la Configuration, au niveau de l'onglet "Autres paramètres".
- ▶ Sous "easyNet easyNet", indiquez le n° easyNet destiné à la communication de base. Ce n° easyNet doit impérativement être identique au n° easyNet que vous attribuez au EC4-200 dans le Configurateur de easySoft !
- ▶ En option, activez les fonctions Remote RUN et Send I/O.

Remote RUN

Uniquement pour les automates dotés de easyNet 2...8 : Si cette fonction est activée, les automates dotés de easyNet 2...8 s'alignent sur l'état RUN/STOP du maître.

Send I/O

Fonction désactivée :

Les valeurs destinées aux entrées/sorties locales byQ; byS; wI; wR sont émises de manière cyclique (le temps de cycle du bus dépend de la vitesse de transmission CAN/easyNet réglée).

Fonction activée :

Les entrées/sorties locales sont émises de manière cyclique et en cas de modification.

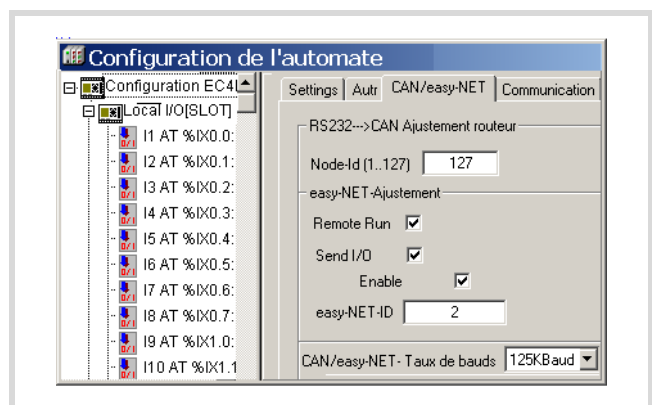


Figure 114 : Configuration d'un EC4-200 en tant que participant easyNet

Configuration pilotée par EC4-200

Vous pouvez configurer d'autres automates de type easy800, EC4-200 ou MFD4 via le réseau easyNet en recourant à un EC4-200 qui a été configuré avec le n° easyNet 1 :

- Via les saisies au niveau de l'afficheur de l'EC4-200
 - Au niveau de l'afficheur de l'automate, dans le menu spécial, sous CONFIGURATOR → NET → NET PARAMETER, saisissez les valeurs destinées au réseau : vitesse de transmission (Baudrate) et retard du bus (Busdelay).
 - Activez/désactivez les fonctions SEND IO et REMOTE RUN qui valent pour tout les participants.
 - Affectez un n° easyNet aux participants, dans le menu NET → PARTICIPANT (automate) : 2...8.

Les paramètres NET (NET PARAMETER) et les easyNet des participants (PARTICIPANT easyNet) sont transférés vers chacun des participants par confirmation de la sélection menu NET → CONFIGURATION.

Les menus sont décrits en détail dans le chapitre présentant la structure des menus.

- Via le paramétrage de l'EC4-200 dans le configurateur de easysoft-CoDeSys

La configuration s'opère en 2 phases :

1. Le paramétrage dans le configurateur

- ▶ Dans l'onglet "CAN/easyNet", cochez la case "Activer", au niveau du champ "Réglages easyNet".
- ▶ Réglez NET easyNet = 1.
- ▶ Activez/Désactivez les fonctions "Remote Run" (Id = 2...8) et "Send I/O".
- ▶ Affectez un n° easyNet aux autres automates : Cliquez sur le bouton "Configurer easyNet" et sélectionnez dans l'affichage ci-dessous les easyNet pour les participants.
- ▶ Dans le champ „CAN/easyNet-Baudrate“, saisissez la vitesse de transmission (Baudrate).

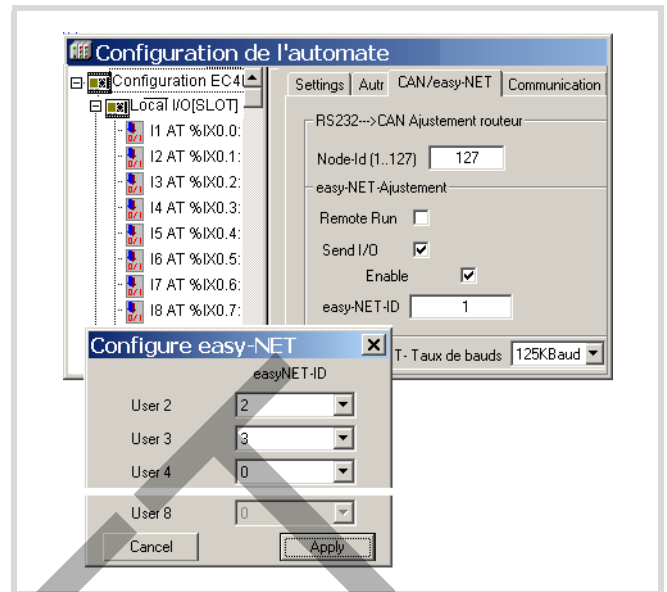


Figure 115 : Paramétrage

- ▶ Transférez le projet vers le participant.

2. Transmission des paramètres et affectation des easyNet à l'aide de la fonction NET_CONFIG.

L'appel de la fonction entraîne la transmission vers chacun des automates des paramètres que vous avez créés dans le configurateur. Cette fonction fait partie intégrante de la bibliothèque SysLibEasyNet.lib. Le mode opératoire de cette fonction et son intégration dans le programme utilisateur sont décrits dans le manuel "Description de SysLibEasyNet.lib ...".

Topologie du bus

La topologie du bus peut être une topologie linéaire avec des câbles de dérivation optionnels. Les extrémités du bus s'achèvent par des résistances de terminaison de bus (120 Ohm).

La figure 116 montre deux possibilités de liaison .

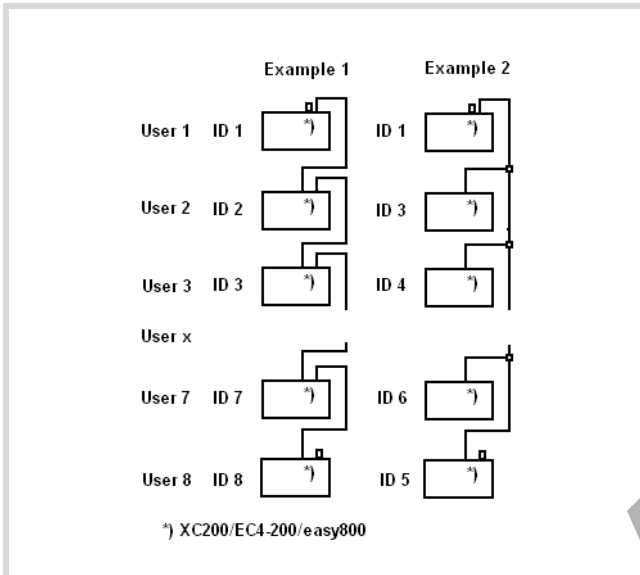


Figure 116 : Possibilités de liaison des automates

Exemple (Exemple) 1 : emplacement géographique égal à easyNet Exemple (Exemple) 2 : emplacement géographique non égal à easyNet

Le participant correspondant à l'emplacement physique 1 reçoit toujours le n° easyNet1. C'est à ce participant que sont raccordés les 7 autres participants, via easyNet.

Du fait que l'EC4-200 possède (comme easy800) deux connexions easyNet avec des câbles de signaux supplémentaires (SEL_IN/OUT), il peut être intégré dans le réseau easyNet de la même manière qu'un appareil easy800.

Tableau 18 : Connexion des signaux entre l'EC4-200 et easy800

EC4-200	<->	easy800
ECAN_H		ECAN_H
ECAN_L		ECAN_L
GND		GND
SEL_IN (ID _n)		SEL_OUT (ID _{n-1})
SEL_OUT (ID _n)		SEL_IN (ID _{n+1})
n = 1...8		

18 Programmation via easyNet (Routage)

On désigne par "routage" (Routing, en anglais) la possibilité d'établir une liaison en ligne entre un appareil de programmation (PC) et un participant de destination quelconque (apte au routage) au sein d'un réseau easyNet, sans que l'appareil de programmation ait besoin d'être relié directement au participant de destination. L'appareil de programmation peut être relié à un autre participant au sein du réseau. La liaison de routage vous permet d'effectuer des actions qui sont également disponibles dans le cadre d'une liaison en ligne directe entre un appareil de programmation et un participant :

- Transfert d'un programme vers l'appareil
- Modifications en ligne
- Test du programme (Débogage)

Le routage offre un avantage : il vous permet d'accéder, à partir d'un participant relié à l'appareil de programmation, à tous les participants de easyNet aptes au routage. L'exploitation de participants décentralisés s'en trouve facilitée.

La transmission de données des liaisons de routage est toutefois beaucoup plus lente que celle des liaisons directes (série ou TCP/IP). Ceci est par exemple particulièrement visible au niveau des temps d'actualisation (qui s'avèrent plus longs) d'éléments de visualisation (ou de variables) ou au niveau des vitesses de téléchargement (qui s'avèrent plus lentes).

Le routage ne peut être exécuté que si les conditions préalables suivantes sont remplies :

- Le participant procédant au routage ET le participant de destination doivent tous deux gérer le routage.
- Les deux participants doivent être reliés à l'aide du bus.
- Les participants doivent disposer de la même vitesse de transmission bus active.

Routage via EC4-200

Les possibilités de routage sont indiquées dans figure 117 et figure 120.

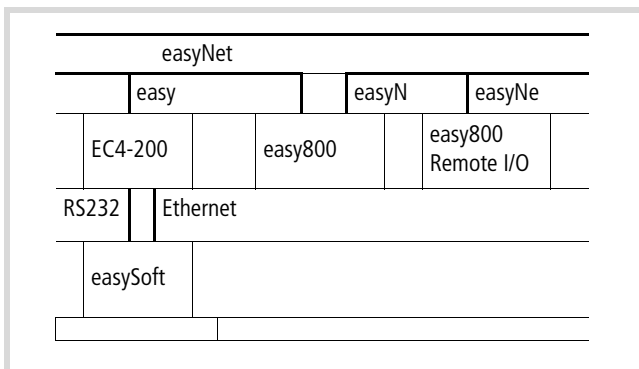


Figure 117 : Routage via l'EC4-200 (Id1) vers easy 800

Comme le montre la figure 117, vous pouvez accéder à easy800 (Id2) via l' EC4-200. Pour cela, le PC peut être couplé à l'aide du logiciel de programmation easySoft à l' EC4-200 lorsque l'EC4-200 dispose déjà d'une configuration dans laquelle il a reçu un n° easyNet-Id (→ figure 114).

Comment créer une liaison entre le PC et l'EC4-200 :

- ▶ Dans le menu «Communication → Liaison» de easySoft, sélectionnez l'interface COM... Les automates du type EC4P-222-... offrent un choix supplémentaire : celui de l'interface Ethernet.
 - Ethernet : sous le titre "Profils Ethernet", cliquez sur le bouton "Editer" et indiquez l'adresse IP du EC4-200 (exemple : 192.168.119.60, port = 10001).
 - COM... : sélectionnez une interface et la vitesse de transmission.
- ▶ Cliquez sur le bouton "En ligne". La liaison entre le PC et le participant raccordé sera alors établie.
- ▶ Sous le titre "Appareil", sélectionnez un participant easy avec lequel vous souhaiteriez communiquer.

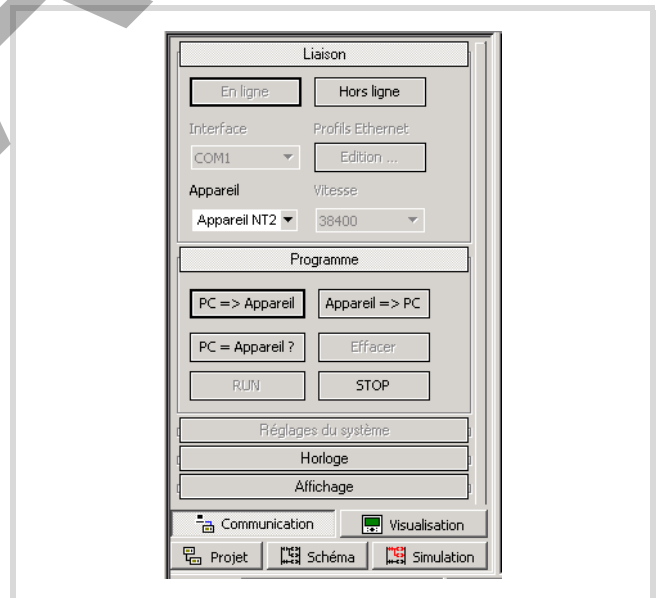


Figure 118 : Transfert d'un programme en mode En ligne

Vous pouvez à présent exécuter les fonctions suivantes dans les champs de commande opérateur :

Champ	Fonction
Horloge	Activation de l'horloge locale, synchronisation des horloges au sein du réseau.
Programme	Commutation entre RUN et STOP
Affichage	Visualisation des états des variables easyNet et des informations relatives aux appareils

Si vous avez sélectionné un appareil easy800 en tant que participant de destination, vous pouvez exécuter les fonctions suivantes dans le champ "Programme" :

- Transfert d'un programme
- Modifications en ligne



Attention !

Tout accès en communication via easyNet augmente la charge du bus (jusqu'à 15 % d'augmentation).

Vous pouvez modifier les réglages suivants :

Vitesse de transmission	Dans le menu "Projet", au niveau du synoptique du réseau
Retard du bus (Busdelay)	
REMOTE RUN	En fonction du participant choisi dans l'onglet "Paramètres de communication" → Configurateur NET
Send IO	
NET-easyNet	

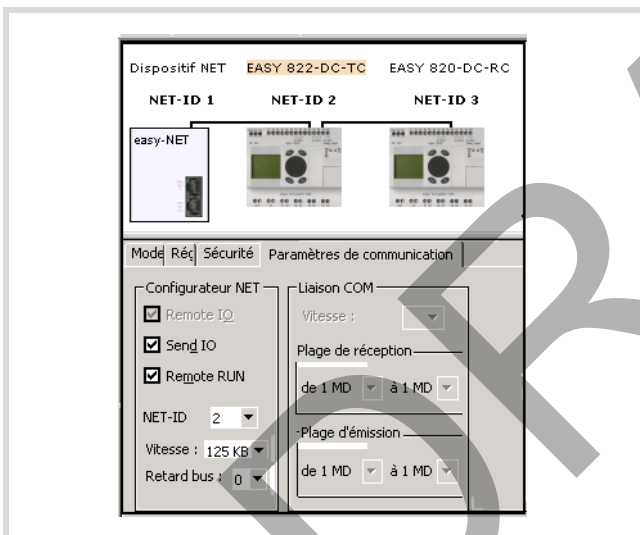


Figure 119 : Paramètres de communication de easy800 (Id2)

Comment transférer les nouveaux réglages vers le participant :

- ▶ Dans le menu «Communication → Programme», cliquez sur le bouton "PC => appareil".
- ▶ Vous pouvez lire les réglages à partir du participant, en cliquant sur le bouton "Appareil => PC".

Le routage est également possible si, comme indiqué sur la figure 120, vous configurez un appareil easy800 avec easyNet1 et l' EC4-200 avec easyNet2. Si vous choisissez ensuite l' EC4-200 comme participant de destination, vous pouvez exécuter à l'aide du participant les actions suivantes, après appel du menu Communication de easySoft :

- Démarrage/Arrêt
- Réglage de l'heure
- Visualisation des formations relatives aux appareils.

Le PC équipé de easySoft peut être couplé à tout automate configuré pour easyNet.

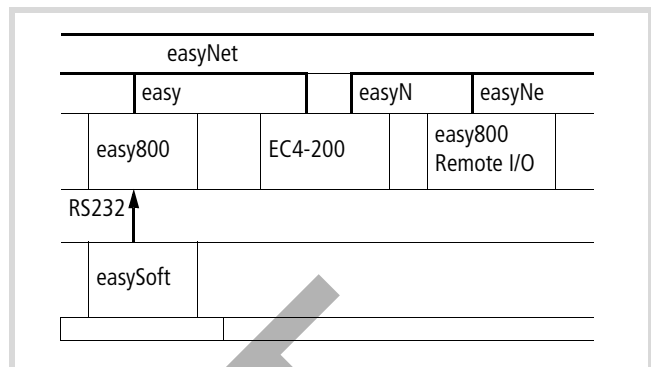


Figure 120 : Routage via easy800 (Id1) sur EC4-200/easy800

19 Modules de couplage réseau pour EC4-200

Les modules de couplage réseau EASY205-ASI, EASY221-CO, EASY204-DP et EASY222-DN vous permettent de raccorder l'EC4-200 en tant qu'esclave aux réseaux ASI, CAN, PROFIBUS-DP ou DeviceNet (→ tableau 19). L'automate peut par ailleurs être utilisé comme participant dans un réseau easyNet.

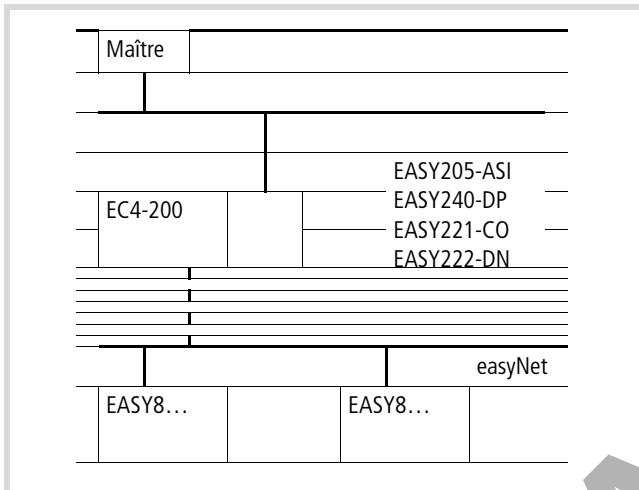


Figure 121 : EC4-200 avec module de couplage réseau

Le type de l'échange de données entre le maître et les modules de couplage réseau est indiqué dans le tableau 19.

Tableau 19 : Synoptique des modules de couplage réseau

Module de couplage réseau	Réseau	Echange de données
EASY205-ASI	ASi+	Cyclique
EASY204-DP	PROFIBUS-DP	Cyclique + acyclique
EASY221-CO	CANopen	Cyclique + acyclique
EASY222-DN	DeviceNet	Cyclique + acyclique

→ Les modules de couplage réseau sont décrits de manière détaillée dans les manuels relatifs aux appareils easy800 (→ tableau 20). Du fait que les automates EC4-200 se comportent comme les appareils easy800, ces manuels valent également pour les EC4-200 utilisés en association avec les modules de couplage réseau cités. Les paragraphes qui suivent ne présentent par suite que les divergences ou spécificités de fonctionnement de ces différents modules de couplage réseau.

Tableau 20 : Manuels relatifs aux modules de couplage réseau

Référence	Manuel (AWB)
EASY204-DP	AWB2528-1401GB
EASY221-CO	AWB2528-1479GB
EASY222-DN	AWB2528-1427GB

EASY205-ASI

Echange de données cyclique

Le maître envoie 8 bits au module de couplage réseau EASY205-ASI raccordé à l'EC4-200 : 4 bits correspondant aux données de sortie et 4 bits correspondant aux paramètres. Il reçoit de l'EC4-200 4 bits correspondant aux données d'entrée.

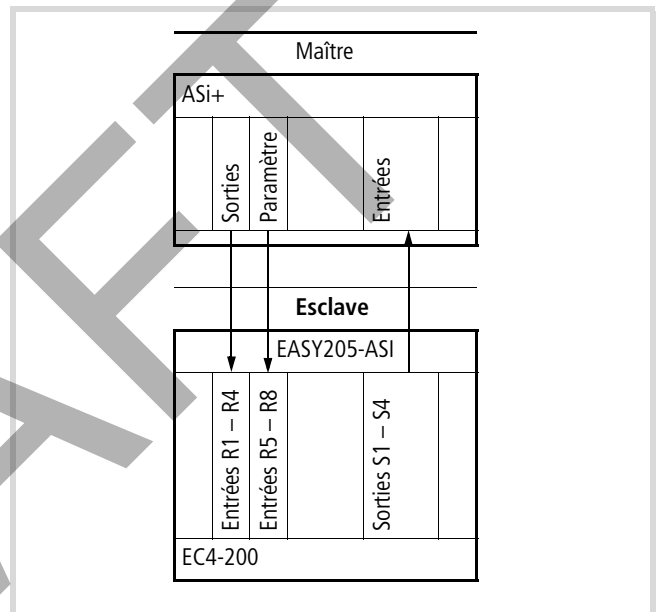


Figure 122 : Echange de données cyclique au niveau de EASY205-ASI

Tableau 21 : Données d'entrée/sortie de l'EC4-200

Maître → EC4		EC4 → Maître			
Sorties du maître	Q0 → R1	Entrées de l'EC4	Sorties de l'EC4	S1 → I0	Entrées du maître
	Q1 → R2			S2 → I1	
	Q2 → R3			S3 → I2	
	Q3 → R4			S4 → I3	
Paramètres du maître	P0 → R5	Entrées de l'EC4			
	P1 → R6				
	P2 → R7				
	P3 → R8				

Configuration

La configuration s'opère dans la Configuration de l'automate du logiciel de programmation easySoft-CoDeSys. Le module de couplage réseau est entré en tant que module d'extension dans l'arborescence de la configuration. Il possède des canaux d'entrée et de sortie prédéfinis (R1...R8, S1...S4) destinés au transfert cyclique des données.

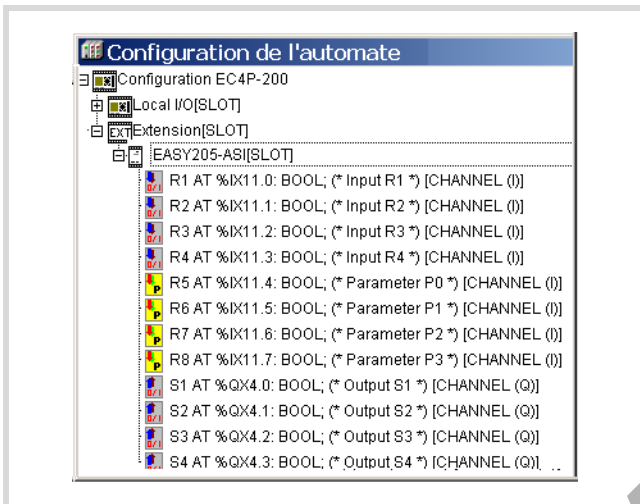


Figure 123 : Configuration de EASY205-ASI

Paramétrage de l'adresse des participants

Le module de couplage EASY205-ASI reçoit son adresse de participant via un appareil de programmation externe.

EASY221-CO, EASY204-DP, EASY222-DN

Le processus d'échange de données entre les modules de couplage réseau EASY et un maître est décrit de manière détaillée dans d'autres manuels : → tableau 20.

Echange de données cyclique

Le principe de l'échange de données cyclique est identique pour les modules de couplage réseau EASY204-DP, EASY221-CO, EASY222-DN.

Le maître échange avec les modules de couplage réseau reliés à l'EC4-200 des données de 3 octets dans chaque sens.

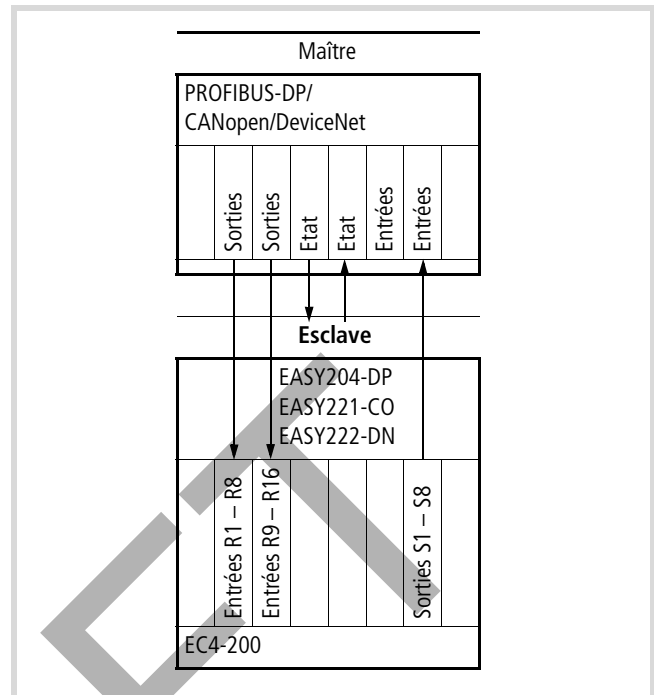


Figure 124 : Echange de données cyclique entre le maître et un module de couplage réseau EASY204-DP, EASY221-CO ou EASY222-DN

Du point de vue maître, ces données sont écrites dans l'EC4-200.

Octet	Signification ¹⁾
0	Etat (exemple : RUN/HALT)
1	R9...R16 (entrées)
2	R1...R8 (entrées)

1) La signification des bits (lorsque l'un des bits de l'octet 0 indique par exemple l'état RUN/HALT de l'automate) est décrite dans d'autres manuels : → tableau 20.

Du point de vue maître, ces données sont lues dans l'EC4-200.

Octet	Signification ¹⁾
0	Etat (exemple : RUN/HALT)
1	S1...S8 (sorties)
2	Non affecté

1) La signification des bits (lorsque l'un des bits de l'octet 0 indique par exemple l'état RUN/HALT de l'automate) est décrite dans d'autres manuels : → tableau 20.

Configuration

La configuration s'opère dans la Configuration de l'automate du logiciel de programmation easySoft-CoDeSys. Un module de couplage réseau est entré en tant que module d'extension dans l'arborescence de la configuration. Il possède des canaux d'entrée et de sortie prédéfinis (R1...R16, S1...S8) qui permettent le transfert cyclique des données.

Paramétrage de l'adresse des participants

Le paramétrage de l'adresse de participant d'un module de couplage réseau s'opère dans une boîte de dialogue spécifique au paramétrage, dans la Configuration de l'automate. L'adresse est attribuée au module par transfert du programme ou par chargement du projet de démarrage.

L'adresse paramétrée dans la Configuration de l'automate peut être écrasée par une saisie dans le fichier Startup.ini. La saisie correspondante dans le fichier Startup.ini est :

```
EXTENSION_SLAVE_ADDRESS = <Adresse>
```

La figure 125 montre (en prenant l'exemple du EASY204-DP) à quel emplacement a lieu la saisie de l'adresse.

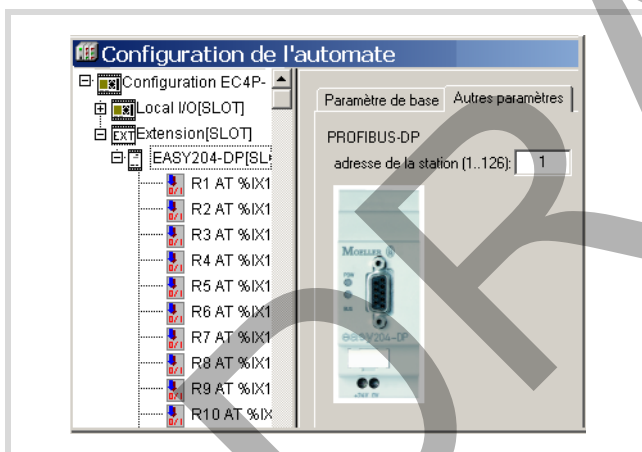


Figure 125 : Saisie d'une adresse

→ Remarque relative aux EASY204-DP :
L'adresse du bus n'est modifiable que lorsque la communication avec le maître n'est pas active.

Lorsqu'une adresse valable a été attribuée à un module, ce dernier la mémorise de manière interne et la réutilise à chaque redémarrage. Si vous paramétrez une nouvelle adresse dans la Configuration de l'automate et que vous procédez à un transfert de programme, cette adresse ne sera transmise que lorsqu'aucune communication n'aura lieu avec le maître durant le transfert (par retrait du câble bus DP, par exemple).

Lorsque vous chargez le programme du PC vers l'automate, ce dernier vérifie si l'adresse actuellement utilisée par l'automate coïncide avec l'adresse configurée. Toute divergence entraîne l'apparition d'un message d'avertissement.

Echange de données acyclique

L'échange de données acyclique permet d'accéder à des objets définis de l'EC4-200. On entend ici par "objets" un sous-ensemble des objets gérés par easy800/MFD.

Les objets indiqués dans le tableau ci-dessous sont gérés par l'EC4-200 et peuvent être adressés par un maître au niveau d'un réseau CAN, PROFIBUS-DP ou DeviceNet.

Tableau 22 : Objets de l'EC4-200

Nom de l'objet	Type d'accès (R = lecture/ W = écriture)
Mode de fonctionnement ¹⁾	R/W
Identification (uniquement pour EASY204-DP)	R
Entrées I1...I16	R
Entrées analogiques I7, I8, I11, I12 ²⁾	R
Entrées R1...R16 ¹⁾	R
Sorties Q1...Q8	R
Sortie analogique QA1	R
Sorties S1...S8 ¹⁾	R
Diagnostic local easyNet1-easyNet163)	R
Entrées des participants réseau IW1...IW8 ³⁾	R
Entrées des participants réseau RW1...RW8 ³⁾	R
Sorties des participants réseau QW1...QW8 ³⁾	R
Sorties des participants réseau SW1...SW8 ³⁾	R
Données de réception des participants réseau RNW1...RNW8 ³⁾	R
Données d'émission des participants réseau SNW1...SNW8 ³⁾	R
Mémoires internes de type bit M1...M96 ⁴⁾	R/W
Mémoires internes de type octet MB1...MB96 ⁴⁾	R/W
Mémoires internes de type mot MW1...MW96 ⁴⁾	R/W
Mémoires internes de type double-mot MD1...MD96 ⁴⁾	R/W
Données de 8 octets (MD67 - MD68) ⁴⁾	R/W
Données de 16 octets (MD69...MD72) ⁴⁾	R/W
Données de 32 octets (MD73...MD80) ⁴⁾	R/W
Données de 64 octets (MD81...MD96) ⁴⁾	R/W

1) Dans le cas de PROFIBUS-DP, uniquement pour maîtres de classe 2

2) IA1...IA4 dans les manuels d'utilisation → page 97

3) On entend par "participants réseau" les participants du réseau easyNet.

4) Représentation des mémoires internes des appareils easy800/MFD sur l'EC4P-200 selon tableau 23.

L'accès à d'autres objets easy800/MFD entraîne un message d'erreur.

→ Remarques valables uniquement pour EASY204-DP : Les données et mémoires internes au format mot et double-mot sont transférées au format Motorola. Aucun échange d'octets n'a lieu !

Adresses de départ pour les entrées/sorties et les mémoires internes

Les adresses de départ destinées aux plages d'adresses des entrées et des sorties peuvent être paramétrées par vos soins dans la Configuration de l'automate. La plage des mémoires internes figure dans le tableau 23.

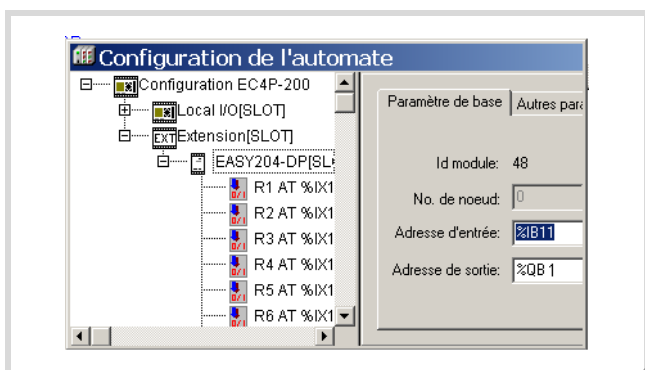


Figure 126 : Paramétrage des plages d'adresses

La configuration et le paramétrage de l'adresse des participants ont déjà été décrits au paragraphe "Echange de données cyclique".

Tableau 23 : Représentation de la plage des mémoires internes de EASY800 sur l'EC4-200 (les valeurs de l'EC4-200 sont indiquées entre parenthèses)

Bits	96–89 (11.7–11.0)	88–81 (10.7–10.0)	80–73 (9.7–9.0)	72–65 (8.7–8.0)	64–57 (7.7–7.0)	56–49 (6.7–6.0)	48–41 (5.7–5.0)	40–33 (4.7–4.0)	32–25 (3.7–3.0)	24–17 (2.7–2.0)	16–9 (1.7–1.0)	8–1 (0.0–0.7)
Octet	12 (11)	11 (10)	10 (9)	9 (8)	8 (7)	7 (6)	6 (5)	5 (4)	4 (3)	3 (2)	2 (1)	1 (0)
Mot	6 (10)		5 (8)		4 (6)		3 (4)		2 (2)		1 (0)	
Double-mot	3 (8)			2 (4)				1 (0)				
Octet	24 (23)	23 (22)	22 (21)	21 (20)	20 (19)	19 (18)	18 (17)	17 (16)	16 (15)	15 (14)	14 (13)	13 (12)
Mot	12 (22)		11 (20)		10 (18)		9 (16)		8 (14)		7 (12)	
Double-mot	6 (20)				5 (16)				4 (12)			
Octet	36 (35)	35 (34)	34 (33)	33 (32)	32 (31)	31 (30)	30 (29)	29 (28)	28 (27)	27 (26)	26 (25)	25 (24)
Mot	18 (34)		17 (32)		16 (30)		15 (28)		14 (26)		13 (24)	
Double-mot	9 (32)				8 (28)				7 (24)			
Octet	48 (47)	47 (46)	46 (45)	45 (44)	44 (43)	43 (42)	42 (41)	41 (40)	40 (39)	39 (38)	38 (37)	37 (36)
Mot	24 (46)		23 (44)		22 (42)		21 (40)		20 (38)		19 (36)	
Double-mot	12 (44)				11 (40)				10 (36)			
Octet	60 (59)	59 (58)	58 (57)	57 (56)	56 (55)	55 (54)	54 (53)	53 (52)	52 (51)	51 (50)	50 (49)	49 (48)
Mot	30 (58)		29 (56)		28 (54)		27 (52)		26 (50)		25 (48)	
Double-mot	15 (56)				14 (52)				13 (48)			
Octet	72 (71)	71 (70)	70 (69)	69 (68)	68 (67)	67 (66)	66 (65)	65 (64)	64 (63)	63 (62)	62 (61)	61 (60)
Mot	36 (70)		35 (68)		34 (66)		33 (64)		32 (62)		31 (60)	

Double-mot	18 (68)				17 (64)				16 (60)			
Octet	84 (83)	83 (82)	82 (81)	81 (80)	80 (79)	79 (78)	78 (77)	77 (76)	76 (75)	75 (74)	74 (73)	73 (72)
Mot	42 (82)		41 (80)		40 (78)		39 (76)		38 (74)		37 (72)	
Double-mot	21 (80)				20 (76)				19 (72)			
Octet	96 (95)	95 (94)	94 (93)	93 (92)	92 (91)	91 (90)	90 (89)	89 (88)	88 (87)	87 (86)	86 (85)	85 (84)
Mot	48 (94)		47 (92)		46 (90)		45 (88)		44 (86)		43 (84)	
Double-mot	24 (92)				23 (88)				22 (84)			
Mot	54 (106)		53 (104)		52 (102)		51 (100)		50 (98)		49 (96)	
Double-mot	27 (104)				26 (100)				25 (96)			
Mot	60 (118)		59 (116)		58 (114)		57 (112)		56 (110)		55 (108)	
Double-mot	30 (116)				29 (112)				28 (108)			
Mot	66 (130)		65 (128)		64 (126)		63 (124)		62 (122)		61 (120)	
Double-mot	33 (128)				32 (124)				31 (120)			
Mot	72 (142)		71 (140)		70 (138)		69 (136)		68 (134)		67 (132)	
Double-mot	36 (140)				35 (136)				34 (132)			
Mot	78 (154)		77 (152)		76 (150)		75 (148)		74 (146)		73 (144)	
Double-mot	39 (152)				38 (148)				37 (144)			
Mot	84 (166)		83 (164)		82 (162)		81 (160)		80 (158)		79 (156)	
Double-mot	42 (164)				41 (160)				40 (156)			
Mot	90 (178)		89 (176)		88 (174)		87 (172)		86 (170)		85 (168)	
Double-mot	45 (176)				44 (172)				43 (168)			
Mot	96 (190)		95 (188)		94 (186)		93 (184)		92 (182)		91 (180)	
Double-mot	48 (188)				47 (184)				46 (180)			
Double-mot	51 (200)				50 (196)				49 (192)			
Double-mot	54 (212)				53 (208)				52 (204)			
Double-mot	57 (224)				56 (220)				55 (216)			
Double-mot	60 (236)				59 (232)				58 (228)			
Double-mot	63 (248)				62 (244)				61 (240)			
Double-mot	66 (260)				65 (256)				64 (252)			
Double-mot	69 (272)				68 (268)				67 (264)			
Double-mot	72 (284)				71 (280)				70 (276)			
Double-mot	75 (296)				74 (292)				73 (288)			
Double-mot	78 (308)				77 (304)				76 (300)			
Double-mot	81 (320)				80 (316)				79 (312)			
Double-mot	84 (332)				83 (328)				82 (324)			
Double-mot	87 (344)				86 (340)				85 (336)			
Double-mot	90 (356)				89 (352)				88 (348)			
Double-mot	93 (368)				92 (364)				91 (360)			
Double-mot	96 (380)				95 (376)				94 (372)			

DRAFT

Annexe

Réseau CAN/easyNet

Accessoires

- Connecteur RJ45, réf. : EASY-NT-RJ45 (8 broches)

→ Les câbles préfabriqués présentent des connecteurs RJ45 aux deux extrémités.

Tableau 24 : Câbles préfabriqués

Longueur du câble cm	Référence
30	EASY-NT-30
80	EASY-NT-80
150	EASY-NT-150

- Câble non préfabriqué, réf. : EASY-NT-CAB (100 m ; $4 \times 0,18 \text{ mm}^2$)
- Pince à sertir pour connecteur RJ45, réf. : EASY-RJ45-TOOL.
- Résistance de terminaison de bus, réf. : EASY-NT-R
Connecteur RJ45 avec résistance de terminaison de bus intégrée 120Ω

Longueurs et sections des câbles

Pour garantir le fonctionnement correct du réseau, il est nécessaire que les longueurs, les sections et la résistivité des câbles soient conformes aux indications du tableau ci-dessous.

Longueur des câbles m	Résistivité des câbles $\text{m}\Omega/\text{m}$	Section mm^2	AWG
jusqu'à 40	≤ 140	0.13	26
jusqu'à 175	≤ 70	0.25 ... 0.34	23, 22
jusqu'à 250	≤ 60	0.34 ... 0.5	22, 21, 20
jusqu'à 400	≤ 40	0.5 ... 0.6	20, 19
jusqu'à 600	≤ 26	0.75 ... 0.8	18
jusqu'à 1000	≤ 16	1.5	16

L'impédance caractéristique des câbles utilisés doit être de 120Ω .

→ Pour toute information complémentaire relative aux longueurs des câbles CAN et des raccordements CAN, reportez-vous à la norme ISO 11898.

Calcul de la longueur d'un câble lorsque la résistance de ce câble est connue

Si la résistance du câble par unité de longueur (résistance linéique R' en Ω/m) est connue, la résistance totale R_L du $\cos \varphi$ du câble ne doit pas dépasser les valeurs suivantes. R_L dépend des vitesses de transmission choisies :

Vitesse de transmission kBaud	Résistivité du câble R_L Ω
10 à 125	≤ 30
250	≤ 25
500	≤ 12

l_{\max} = longueur maximale du câble, en m

R_L = résistance totale du câble, en Ω

R' = résistance du câble par unité de longueur, en Ω/m

$$l_{\max} = \frac{R_L}{R'}$$

Calcul de la section lorsque la longueur de ce câble est connue

On calcule la section minimale adaptée à l'extension maximale connue du réseau.

l = longueur du câble, en m

S_{\min} = section minimale du câble, en mm^2

ρ_{cu} = résistivité spécifique du cuivre, dans le cas où l'on ne dispose d'aucune autre indication : $0,018 \Omega\text{mm}^2/\text{m}$

$$S_{\min} = \frac{l \times \rho_{\text{cu}}}{12,4}$$

→ Lorsque le résultat de l'opération ne donne aucune section normalisée, choisissez la section immédiatement supérieure.

Calcul de la longueur du câble lorsque la section est connue

On calcule la longueur maximale du câble à partir d'une section de câble connue :

l_{\max} = longueur du câble, en m

S = section du câble, en mm^2

ρ_{cu} = résistivité spécifique du cuivre, dans le cas où l'on ne dispose d'aucune autre indication : $0,018 \Omega\text{mm}^2/\text{m}$

$$l_{\max} = \frac{S \times 12,4}{\rho_{\text{cu}}}$$

Exemple de programme pour le démarrage/l'arrêt (START/STOP) de l'automate à l'aide d'un interrupteur externe

Pour le lancement, vous avez besoin de la fonction SysStartPlcProgram et pour l'arrêt, de la fonction SysStopPlcProgram, toutes deux issues de la bibliothèque SysLibPlcCtrl.lib.

Dans ce cas, le comportement au démarrage de l'automate doit être positionné sur WARMSTART dans le configurateur de l'automate, sous <Autres paramètres → Réglages>.

Fonctionnement

Un enregistrement de la fonction "FuncCalledWhenPlcInStop" sur l'événement ("Event") "EVENT_TASKCODE_NOT_CALLED" a lieu par le biais de l'UOP "StartPrg" (qui est appelée une seule fois à chaque démarrage de l'automate). Cet enregistrement provoque l'appel de la fonction "FuncCalledWhenPlcInStop" via l'événement ("Event") "EVENT_TASKCODE_NOT_CALLED" lorsque l'automate est à l'état STOP. L'état de l'entrée est surveillé à l'aide de la fonction "StartStopFunction" et la fonction destinée au démarrage ou à l'arrêt de l'automate est appelée en cas de changement d'état.

Du fait que l'UOP "StartPrg" est appelée une seule fois, aucune sortie ni aucun paramètre ne doit être activé(e) dans cette UOP. Vous devez programmer les programmes spécifiques à l'utilisateur dans des UOP distinctes.

- ▶ Activez l'événement système "Start" et dénommez l' "UOP appelée " Startprg.

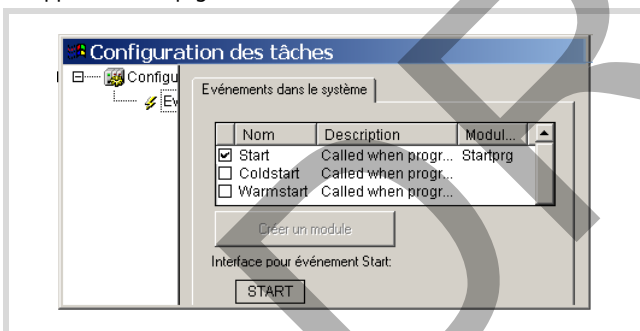


Figure 127 : Activation d'un événement système

- ▶ Dans le dossier MODULES, ouvrez une nouvelle UOP dénommée "Startprg" et programmez la fonction "SysCallbackRegister" (qui "présente" les fonctions Start/Stop au système d'exploitation).

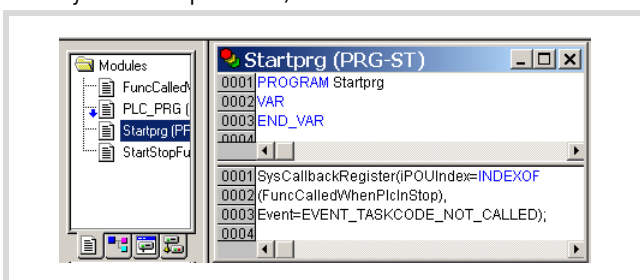


Figure 128 : Fonction "Startprg"

- ▶ Déclarez les variables globales suivantes.

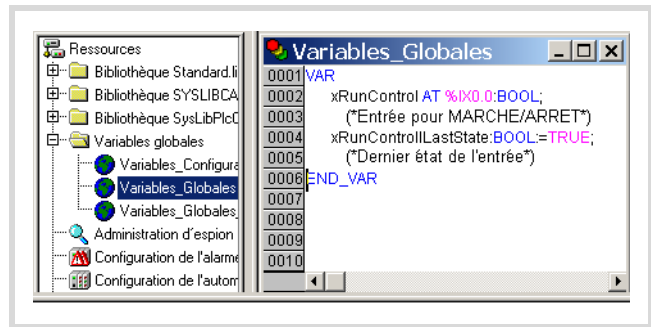


Figure 129 : Déclaration de variables globales

- ▶ Saisissez le programme pour PLC_PRG conformément à la figure 130. Il est important que le programme utilisateur ou les appels d'UOP soient insérés selon la figure 130.

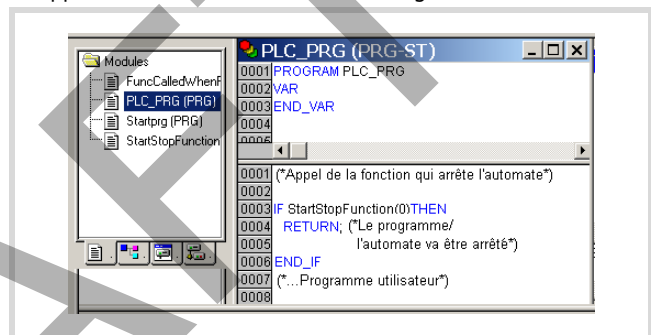


Figure 130 : Scrutation START/STOP

- ▶ Indiquez les fonctions "FuncCalledWhenPlcInStop" et "StartStopFunction".

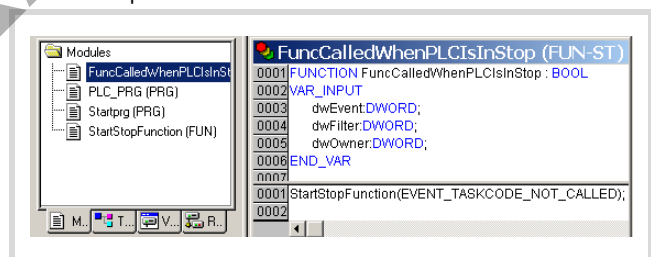


Figure 131 : Appel de la fonction FuncCalledWhenPlcInStop

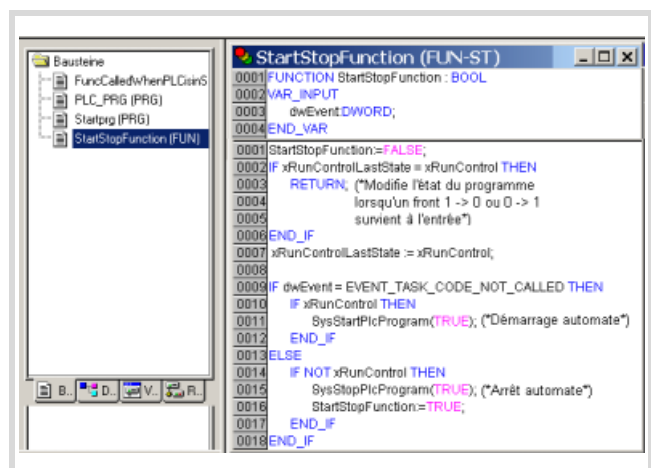


Figure 132 : Fonction chargée de la surveillance de l'entrée

Câble de liaison EASY800-PC-CAB

Connecteur femelle 9 broches sur le câble (connecteur Terminal/PC)

Broche	Plage de signaux
2	RxD
3	TxD
4	DTR
5	GND
7	RTS

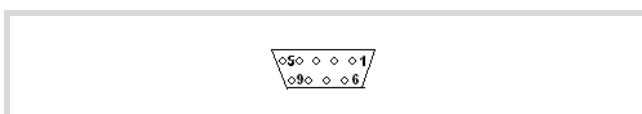


Figure 133 : Connecteur femelle 9 broches

→ Pour le bon fonctionnement du câble, le signal RTS doit impérativement être activé du fait que la tension au niveau du câble RTS alimente les éléments situés dans le connecteur du câble.

Encombres et poids

Dimensions (L x H x P)	
[mm]	107.5 x 90 x 72
avec adaptateur pour module mémoire	107.5 x 90 x 79
[inches]	4.23 x 3.54 x 2.84
avec adaptateur pour module mémoire	4.23 x 3.54 x 3.11
Pas modulaires d'encombrement (PE)	6
Poids	
[g]	320
[lb]	0.705
Montage	
Encliquetage sur profilé chapeau DIN 50022, 35 mm ou fixation par vis à l'aide de 3 pattes de montage ZB4-101-GF1	

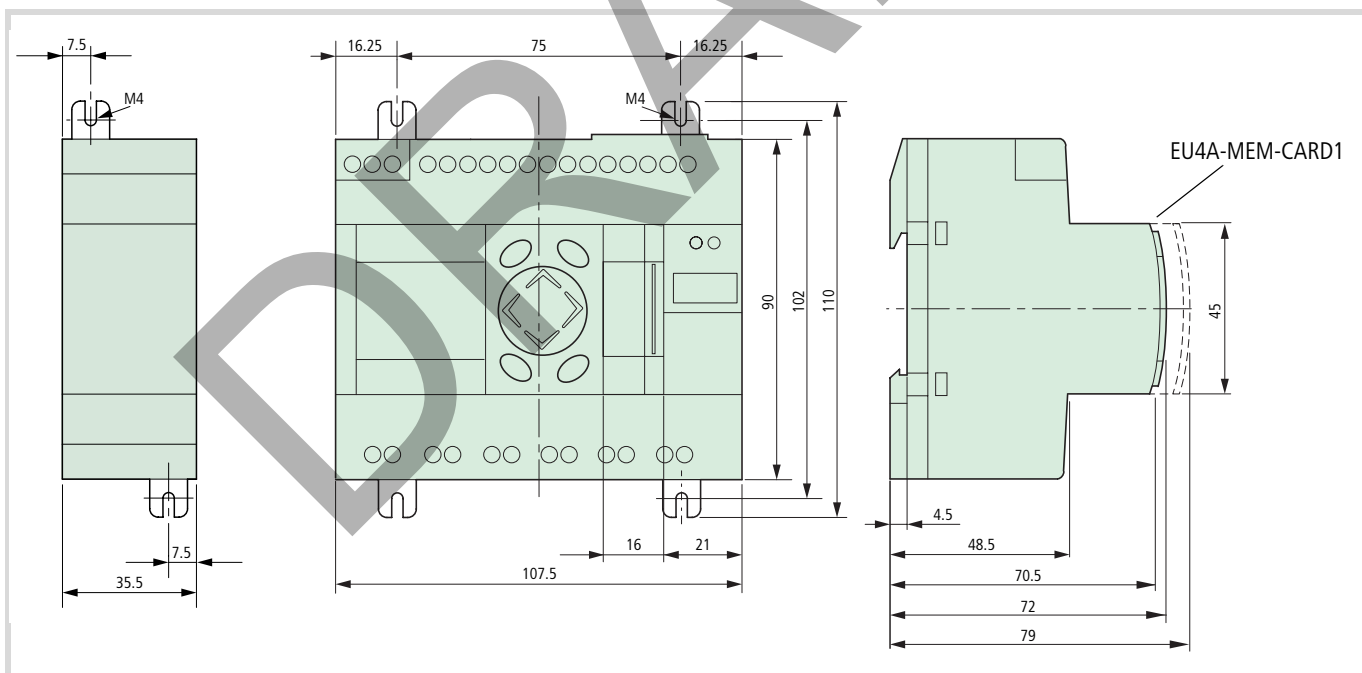


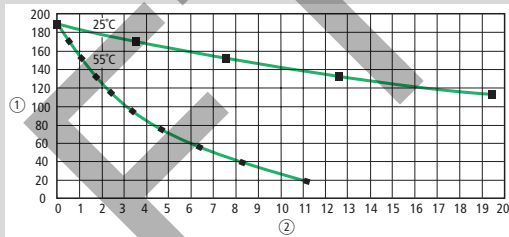
Figure 134 : Dimensions en mm (indications en inches → tableau 25)

Tableau 25 : Dimensions en inches

mm	inches	mm	inches
4.5	0.177	79	3.11
16.25	0.64	90	3.54
48.5	1.91	102	4.01
70.5	2.78	107.5	4.23
72	2.83	110	4.33
75	2.95		

Caractéristiques techniques

Conditions d'environnement climatiques (froid selon IEC 60068-2-1, chaleur sèche selon IEC 60068-2-2)			
Température d'emploi, montage horizontal ou vertical	°C, (°F)	-25 à 55, (-13 à 131)	
Condensation		Eviter la condensation par des mesures appropriées	
Afficheur à cristaux liquides (fiabilité de la lecture)	°C, (°F)	0 à 55, (32 à 131)	
Température de stockage/transport	°C, (°F)	-40 à 70, (-40 à 158)	
Humidité relative de l'air (IEC 60068-2-30), sans condensation	%	5 à 95	
Pression atmosphérique (lors du fonctionnement)	hPa	795 à 1080	
Conditions d'environnement mécaniques			
Degré de protection IEC/EN 60529		IP20	
Vibrations (IEC/EN 60068-2-6)			
Amplitude constante de 3,5 mm	Hz	5 à 9	
Accélération constante de 1 g	Hz	9 à 150	
Chocs (IEC/EN 60068-2-27) Forme demi-sinusoidale, 15 g/11 ms	Chocs	18	
Chute et culbute (IEC/EN 60068-2-31)	Hauteur de chute	mm	50
Chute libre, appareil emballé (IEC/EN 60068-2-32)		m	1
Position de montage			horizontale, verticale
Compatibilité électromagnétique (CEM)			
Décharges électrostatiques (ESD), (IEC/EN 61000-4-2, niveau 3)			
Décharge dans l'air	kV	8	
Décharge au contact	kV	6	
Immunité aux champs électromagnétiques rayonnés aux fréquences radioélectriques (RFI), (IEC/EN 61000-4-3)	V/m	10	
Antiparasitage / Classe de valeurs limites			EN 55011, EN 55022 / Classe B
Immunité aux transitoires électriques rapides en salves (Burst), (IEC/EN 61000-4-4, niveau 3)			
Câbles d'alimentation	kV	2	
Câbles de signaux	kV	2	
Ondes de choc (Surge) (IEC/EN 61000-4-5, niveau 2)	kV	0,5 symétrique, 1 asymétrique	
Perturbations conduites (IEC/EN 61000-4-6)	V	10	
Rigidité diélectrique			
Dimensionnement des distances d'isolement et des lignes de fuite			EN 50178
Rigidité diélectrique			EN 50178
Catégorie de surtension/Degré de pollution			II/2
Outils et sections raccordables			
Conducteurs à âme massive, section minimale à maximale	mm ²	0.2 à 4	
	AWG	22 à 12	
Conducteurs souples avec embout, section minimale à maximale	mm ²	0.2 à 2.5	
	AWG	22 à 12	

Câblage en usine	AWG	30
Tournevis pour vis à tête fendue, largeur de la lame	mm	3,5 × 0,8
	inch	0,14 × 0,03
Couple de serrage	Nm	0.6
UC (unité centrale)		
Données concernant la mémoire		
Code programme	Koctets	256
Données programme	Koctets	14 segments de 16 Koctets
Mémoires internes/Entrées/Sorties/Données rémanentes	Koctets	16/4/4/8
Temps de cycle pour 1 Kinstructions		< 0.3
Sauvegarde par piles/Précision de l'horloge temps réel		
Durée de sauvegarde de l'horloge		
Précision de l'horloge temps réel		
par jour	s/jour	±5
par an	h/an	±0,5
Interfaces		
Interface de programmation		
Technique de raccordement		RJ45, 8 broches
RS 232 (sans câbles de commande)		
Port de l'automate		COM 1
Séparation galvanique		aucune
Mode Programmation		
Vitesse de transmission des données		4.8, 9.6, 19.2, 38.4, 57.6
Format des caractères		8 bits de données, aucune parité, 1 bit d'arrêt
Mode Transparent		
Vitesse de transmission des données		0.3, 0.6, 1.2, 2.4, 4.8, 9.6, 19.2, 38.4, 57.6
Format des caractères		8E1, 801, 8N1, 8N2, 7E2, 7O2, 7N2, 7E1
Nombre d'octets d'émission dans un bloc		190
Nombre d'octets de réception dans un bloc		190
Ethernet		
Vitesse de transmission des données	Mbits/s	10
Séparation galvanique		Oui
Interface multifonctions (RS232) sans câbles de commande		
Mode Transparent		
Port de l'automate		COM 2
Séparation galvanique		Oui, dans le câble easy800-PC-CAB
Technique de raccordement		Câble easy800-PC-CAB
Vitesse de transmission des données	Kbits/s	9.6, 19.2

① = durée de sauvegarde, en heures
 ② = durée de fonctionnement, en années

CAN(open)/easyNet		
Vitesse de transmission des données	Kbits/s	10, 20, 50, 100, 125, 250, 500 Valeur par défaut : 125
Séparation galvanique par rapport aux entrées/aux sorties/à l'alimentation		Oui
Résistance de terminaison de bus		externe : 120 Ω ou connecteur EASY-NT-R (incluant une résistance de terminaison de bus de 120 Ω)
Technique de raccordement		2 x RJ45, 8 broches
Mode CAN(open) :		
– Participant	Nombre	max.126
– Type de PDO		asynchrone, cyclique, acyclique
– Profil de l'appareil		selon DS301 V4
Mode easyNet		
– Participants	Nombre	max. 8
Tension d'alimentation		
Tension assignée		
Valeur assignée	V DC, (%)	24, (-15, +20)
Plage admissible	V DC	20,4 à 28,8
Ondulation résiduelle	%	≤ 5
Courant d'entrée sous 24 V DC, en moyenne	mA	140
Tolérance aux microcoupures, IEC/EN 61131-2	ms	10
Puissance dissipée sous 24 V DC, en moyenne	W	3.4
Entrées		
Entrées tout-ou-rien		
Nombre		12
Entrées utilisables pour des signaux analogiques		I 7,8,11,12
Entrées utilisables pour des signaux d'impulsion (compteurs rapides)		I 1,2,3,4
Entrées destinées à la génération d'interruptions		I 1,2,3,4
Visualisation d'état		Afficheur à cristaux liquides (LCD)
Séparation galvanique		
par rapport à la tension d'alimentation, à l'interface PC		Non
entre les différentes entrées TOR		Non
par rapport aux sorties, à l'interface CAN		Oui
Tension assignée		
Valeur assignée	V DC	24
avec signal à « 0 »		
I1 à I6 et I9 à I10	V DC	< 5
I7, I8, I11, I12	V DC	< 8
avec signal à « 1 »		
I1 à I6 et I9 à I10	V DC	> 15
I7, I8, I11, I12	V DC	> 8
Courant d'entrée avec signal à « 1 » (sous 24 V DC)		
I1 à I6, I9 à I10	mA	3.3
I7, I8, I11, I12	mA	2.2

Temps de réponse pour le passage de « 0 » à « 1 »		
I1 à I4	ms	0.02
I5 à I12	ms	0.25
Temps de réponse pour le passage de « 1 » à « 0 »		
I1 à I4	ms	0.02
I5 à I12	ms	0.25
Longueur du câble (non blindé)	m	100
Autres fonctions des entrées		
Entrées pour signaux analogiques		
Nombre		4 (I7,I8,I11,I12)
Plage de signaux	V DC	0 à 10
Résolution analogique	V	0.01
Résolution tout-ou-rien	Bits	10
	Valeur	0 à 1023
Impédance d'entrée	k Ω	11.2
Précision de la valeur réelle		
deux appareils	%	± 3
au sein d'un appareil	%	± 2
Courant d'entrée	mA	< 1
Longueur du câble (blindé)	m	30
Entrées pour compteurs rapides		
		I1, I2
Nombre/Plage de valeurs	Bits	2 x 16 bits (I1, I2) 1 x 32 bits (I1)
Fréquence maximale	kHz	50
Sens de comptage (possibilité de commutation par voie logicielle)		
		Incrémentation/Décrémentation
Longueur du câble (blindé)	m	20
Forme des impulsions		carrée
Rapport impulsions-pauses		1:1
Compteur incrémental		
		I1, I2, I3, I4
Nombre		1
Plage de valeurs	Bits	32 bits
Fréquence maximale	kHz	40
Longueur du câble (blindé)	m	20
Forme des impulsions		carrée
Entrées de comptage		I1, I2 = entrées de comptage I3 = impulsion de référence I4 = fenêtre de référence
Décalage des signaux		90°
Rapport impulsions-pauses		01:01
Entrées destinées à la génération d'interruptions		
		I1, I2, I3, I4
Fréquence maximale	kHz	3

Sorties à relais		
Nombre de sorties		6
Mise en parallèle de sorties pour une augmentation de puissance		non admissible
Protection d'une sortie à relais		
Par disjoncteur de protection ligne B16	A	16
ou par fusible (lent)	A	8
Séparation galvanique		
Séparation sûre	V AC	300
Isolation de base	V AC	600
Longévité mécanique	Manceuvres	10 × 10
Circuits des relais		
Courant thermique conventionnel	A	8
Recommandés pour les charges ci-contre, sous 12 V AC/DC	mA	> 500
Protection contre les courts-circuits, $\cos \varphi = 1$ Caractéristique B (B16) sous 600 A	A	16
Protection contre les courts-circuits, $\cos \varphi = 0,5$ à $0,7$ Caractéristique B (B16) sous 900 A	A	16
Tension assignée de tenue aux chocs U_{imp} entre contact et bobine	kV	6
Tension assignée d'isolement U_i		
Tension assignée d'emploi U_e	V AC	250
Séparation sûre selon EN 50178 entre bobine et contact	V AC	300
Séparation sûre selon EN 50178 entre deux contacts	V AC	300
Pouvoir de fermeture, IEC 60947		
AC-15 250 V AC, 3 A (600 man./h)	Manceuvres	300 000
DC-13 L/R ≤ 150 ms 24 V DC, 1 A (500 man./h)	Manceuvres	200 000
Pouvoir de coupure, IEC 60947		
AC-15 250 V AC, 3 A (600 man./h)	Manceuvres	300 000
DC-13 L/R F 150 ms 24 V DC, 1 A (500 man./h)	Manceuvres	200 000
Charge des lampes à incandescence		
1 000 W sous 230/240 V AC	Manceuvres	25 000
500 W sous 115/120 V AC	Manceuvres	25 000
Charge des tubes fluorescents 10 × 58 W sous 230/240 V AC		
Tubes fluorescents - avec ballast - avec compensation individuelle - non compensés	Manceuvres	25 000

Fréquence de commutation des relais		
Nombre de manœuvres (mécaniques)	Manœuvres	10 millions (10^7)
Fréquence de commutation (mécanique)	Hz	10
Charge ohmique (charge des lampes à incandescence, par exemple)	Hz	2
Charge inductive	Hz	0.5
Sorties à transistors		
Nombre de sorties		8
Tension assignée d'emploi U_e	V DC	24
Plage admissible	V DC	20,4 à 28,8
Ondulation résiduelle	%	≤ 5
Courant d'alimentation		
avec signal à « 0 », en moyenne/maximal	mA	18/32
avec signal à « 1 », en moyenne/maximal	mA	24/44
Protection contre l'inversion de polarité		Oui
▽Attention ! L'application d'une tension aux sorties en cas d'inversion de polarité entraîne un court-circuit.		
Séparation galvanique		Oui
Courant assigné I_e avec signal à « 1 », maximal	A	0.5
Charge des lampes sans R_V	W	5
Courant résiduel avec signal à « 0 », par canal	mA	< 0.1
Tension de sortie maximale		
avec signal à « 0 », avec charge externe, $10 M\Omega$	V	2.5
avec signal à « 1 », $I_e = 0.5 A$		$U = U_e - 1 V$
Protection contre les courts-circuits (thermique), groupe Q1 à Q4 /groupe Q5 à Q8 : évaluation effectuée via les entrées de diagnostic I16 (pour Q1 à Q4), I17 (pour Q5 à Q8)		Oui
▽Attention ! Dans le programme, positionnez le groupe des sorties à 0 pour éviter toute surcharge de la sortie.		
Courant de déclenchement sur court-circuit pour $R_a \leq 10 m\Omega$ (en fonction du nombre de canaux actifs et de leur charge)	A	$0,7 \leq I_e \leq 2$
Courant de court-circuit total max.		A
Courant de court-circuit de crête		A
Coupure thermique		Oui
Fréquence de commutation maximale en cas de charge ohmique constante $RL = 100 k\Omega$ (en fonction du programme et de la charge)	Manœuvres/h	40000

Possibilité de mise en parallèle des sorties en cas de charge ohmique ; en cas de charge inductive avec circuit de protection externe (→ paragraphe « Raccordement des sorties à transistors », page 25) ; combinaison au sein d'un groupe	Oui
Groupe 1 : Q1 à Q4	
Groupe 2 : Q5 à Q8	
Nombre maximal de sorties	4
Courant total maximum A	2
▽ Attention ! Les sorties doivent être commandées simultanément et pendant des durées identiques.	
Affichage d'état des sorties	Afficheur à cristaux liquides (LCD)

Charge inductive (sans circuit de protection externe)

Explications d'ordre général :

$T_{0,95}$ = temps en millisecondes, jusqu'à obtention de 95 % du courant statique

$$T_{0,95} \approx 3 \times T_{0,65} = 3 \times \frac{L}{R}$$

Catégories d'emploi des groupes Q1 à Q4, Q5 à Q8

$T_{0,95} = 1 \text{ ms}$ $R = 48 \Omega$ $L = 16 \text{ mH}$	Facteur de simultanéité, par groupe ; g =		0.25
	Facteur de marche relatif	%	100
	Fréquence de commutation maximale $f = 0,5 \text{ Hz}$	Manœuvres/h	1500
	Facteur de marche maximal FM = 50 %		
DC13 $T_{0,95} = 72 \text{ ms}$ $R = 48 \Omega$ $L = 1.15 \text{ H}$	Facteur de simultanéité g =		0.25
	Facteur de marche relatif	%	100
	Fréquence de commutation maximale $f = 0,5 \text{ Hz}$	Manœuvres/h	1500
	Facteur de marche maximal FM = 50 %		

Autres charges inductives :

$T_{0,95} = 15 \text{ ms}$ $R = 48 \Omega$ $L = 0.24 \text{ H}$	Facteur de simultanéité g =		0.25
	Facteur de marche relatif	%	100
	Fréquence de commutation maximale $f = 0,5 \text{ Hz}$	Manœuvres/h	1500
	Facteur de marche maximal FM = 50 %		

Charge inductive, avec circuit de protection externe pour chaque charge (→ paragraphe « Raccordement des sorties à transistors », page 25)

Facteur de simultanéité g =		1
Facteur de marche relatif	%	100
Fréquence de commutation maximale Facteur de marche maximal	Manœuvres/h	En fonction du circuit de protection

Sortie analogique		
Nombre		1
Séparation galvanique		
par rapport à la tension d'alimentation		non
par rapport aux entrées TOR		non
par rapport aux sorties TOR		oui
par rapport au réseau easy-NET		oui
Type de sortie		Tension CC
Plage de signal	V DC	0 à 10
Courant de sortie maximal	mA	10
Résistance de charge	k Ω	1
Protection contre les courts-circuits et les surcharges		oui
Résolution analogique	V	0,01
Résolution tout-ou-rien	Bit	10
	Valeur	0 à 1023
Temps de réponse	us	100
Précision =		
(–25 à 55 °C), selon la plage	%	2
(25 °C), selon la plage	%	1
Temps de conversion		à chaque cycle de l'UC

Jeux de caractères

Latin 1, « Europe de l'Ouest »

Code	Signification	Code	Signification	Code	Signification	Code	Signification
0	Espace	64	@	128	€	192	À
1	À	65	Á	129		193	Á
2	Â	66	B	130		194	Â
3	Ã	67	C	131		195	Ã
4	Ä	68	D	132		196	Ä
5	Å	69	E	133		197	Å
6	Æ	70	F	134		198	Æ
7	Ç	71	G	135		199	Ç
8	È	72	H	136		200	È
9	É	73	I	137		201	É
10	Ê	74	J	138		202	Ê
11	Ë	75	K	139		203	Ë
12	Ï	76	L	140		204	Ï
13	Ñ	77	M	141		205	Ñ
14	Ñ	78	N	142	¢	206	Ñ
15	Ò	79	O	143		207	Ò
16	Ó	80	P	144		208	Ó
17	Ô	81	Q	145		209	Ô
18	Õ	82	R	146		210	Õ
19	Ş	83	S	147		211	Ş
20	Ť	84	T	148		212	Ť
21	Û	85	U	149		213	Û
22	Ü	86	V	150		214	Ü
23	Ý	87	W	151		215	Ý
24	ÿ	88	X	152		216	ÿ
25	ÿ	89	Y	153		217	ÿ
26	z	90	Z	154		218	z
27	+	91	[155	¢	219	+
28	+	92	\	156		220	+
29	+	93]	157		221	+
30	+	94	^	158		222	+
31	␣ (curseur)	95	_	159		223	␣
32	Espace	96	`	160		224	␣
33	!	97	a	161		225	!
34	"	98	b	162		226	"
35	#	99	c	163		227	#
36	\$	100	d	164		228	\$
37	%	101	e	165		229	%
38	&	102	f	166		230	&
39	'	103	g	167		231	'
40	<	104	h	168	¢ (Curseur)	232	<
41	>	105	i	169		233	>
42	*	106	j	170		234	*
43	+	107	k	171		235	+
44	,	108	l	172	=	236	,
45	-	109	m	173		237	-
46	.	110	n	174		238	.
47	/	111	o	175		239	/
48	0	112	p	176		240	0
49	1	113	q	177		241	1
50	2	114	r	178		242	2
51	3	115	s	179		243	3
52	4	116	t	180		244	4
53	5	117	u	181		245	5
54	6	118	v	182	¢	246	6
55	7	119	w	183		247	7
56	8	120	x	184		248	8
57	9	121	y	185		249	9
58	:	122	z	186		250	:
59	;	123	{	187		251	;
60	<	124		188		252	<
61	=	125	}	189		253	=
62	>	126	~	190		254	>
63	?	127		191		255	?

Jeux de caractères Latin 2, "Europe centrale" (pour le polonais, le hongrois et le tchèque)

Code	Signification	Code	Signification	Code	Signification	Code	Signification
0	Espace	64	Ā	128	Ě	192	Ř
1	Ā	65	Ą	129	Ħ	193	Š
2	Ē	66	Ā	130	Ɔ	194	Š
3	Ĉ	67	Ā	131	Ɔ	195	Š
4	Ċ	68	Ā	132	„	196	Š
5	Ė	69	E	133	„	197	Ĺ
6	Ɔ	70	F	134	„	198	Č
7	Ĝ	71	G	135	„	199	Ĉ
8	H	72	H	136	□	200	Č
9	I	73	I	137	„	201	É
10	J	74	J	138	Š	202	Ě
11	K	75	K	139	<	203	Ě
12	L	76	L	140	š	204	Ě
13	M	77	M	141	ř	205	í
14	N	78	N	142	ž	206	í
15	Ń	79	Ń	143	ž	207	Ń
16	Ŕ	80	P	144	Ť	208	Ŕ
17	Ŗ	81	Ŗ	145	'	209	Ŗ
18	Ŕ	82	R	146	'	210	Ŗ
19	S	83	S	147	„	211	Ŕ
20	Ť	84	T	148	„	212	Ŕ
21	Ů	85	U	149	.	213	Ů
22	Ű	86	U	150	■	214	Ű
23	Ű	87	W	151	—	215	Ű
24	X	88	X	152	z	216	Ŕ
25	Y	89	Y	153	z	217	Ů
26	Z	90	Z	154	z	218	Ů
27	†	91	Ł	155	>	219	Ů
28	‡	92	˘	156	š	220	Ů
29	→	93	Ǯ	157	ř	221	Ů
30	+	94	^	158	ž	222	Ů
31	␣ (curseur)	95	˘	159	ž	223	Ů
32	Espace	96	˘	160	√	224	Ů
33	!	97	a	161	„	225	š
34	”	98	b	162	„	226	š
35	#	99	c	163	ł	227	š
36	\$	100	d	164	„	228	š
37	%	101	e	165	ř	229	í
38	&	102	f	166	ř	230	č
39	'	103	g	167	„	231	č
40	<	104	h	168	■ (curseur)	232	č
41	>	105	i	169	ŕ	233	č
42	*	106	j	170	š	234	č
43	+	107	k	171	„	235	č
44	,	108	l	172	„	236	č
45	-	109	m	173	„	237	í
46	.	110	n	174	„	238	í
47	/	111	o	175	ž	239	d'
48	0	112	p	176	o	240	ř
49	1	113	q	177	±	241	ř
50	2	114	r	178	„	242	ř
51	3	115	s	179	ž	243	č
52	4	116	t	180	˘	244	č
53	5	117	u	181	μ	245	č
54	6	118	v	182	ŕ	246	č
55	7	119	w	183	ŕ	247	ž
56	8	120	x	184	„	248	ř
57	9	121	y	185	ř	249	č
58	#	122	z	186	ř	250	č
59	ž	123	˘	187	„	251	č
60	<	124	ı	188	ł	252	č
61	=	125	˘	189	„	253	č
62	>	126	˘	190	ř	254	ř
63	?	127	£	191	ž	255	„

Jeux de caractères "Cyrillique" (pour le russe)

Code	Signification	Code	Signification	Code	Signification	Code	Signification
0	Espace	64	Ё	128	Ё	192	А
1	А	65	А	129	І	193	В
2	Б	66	В	130	Ѡ	194	В
3	С	67	С	131	І	195	Г
4	Д	68	Д	132	"	196	Д
5	Е	69	Е	133		197	Е
6	Ғ	70	Ғ	134	з	198	Ж
7	Г	71	Г	135	з	199	З
8	Н	72	Н	136	□	200	И
9	І	73	І	137	з	201	Й
10	Ј	74	Ј	138	Ѧ	202	К
11	К	75	К	139	<	203	Л
12	Л	76	Л	140	Ѧ	204	М
13	М	77	М	141	К	205	Н
14	Н	78	Н	142	Ѧ	206	О
15	О	79	О	143	Ѧ	207	П
16	Р	80	Р	144	Ѧ	208	Р
17	Q	81	Q	145	'	209	С
18	Р	82	Р	146	'	210	Т
19	С	83	С	147	"	211	У
20	Т	84	Т	148	"	212	Ф
21	U	85	U	149	.	213	Х
22	U	86	U	150	■	214	Ц
23	W	87	W	151	—	215	Ч
24	X	88	X	152	№	216	Ш
25	Y	89	Y	153	h	217	Щ
26	Z	90	Z	154	Ѧ	218	Ъ
27	□	91	□	155	>	219	Ы
28	↓	92	↓	156	Ѧ	220	Ь
29	→	93	→	157	К	221	Э
30	←	94	←	158	Ѧ	222	Ф
31	␣ (curseur)	95	—	159	Ѧ	223	Я
32	Espace	96	^	160	√	224	а
33	!	97	а	161	∅	225	б
34	"	98	б	162	∅	226	в
35	#	99	с	163	Ј	227	г
36	\$	100	д	164	Ѧ	228	д
37	%	101	е	165	Г	229	е
38	&	102	ф	166	!	230	ж
39	'	103	г	167	Ё	231	з
40	<	104	h	168	■ (curseur)	232	и
41	>	105	i	169	⊖	233	й
42	*	106	ј	170	Є	234	к
43	+	107	к	171	≤	235	л
44	,	108	l	172		236	м
45	-	109	m	173		237	н
46	.	110	n	174	h	238	о
47	/	111	o	175	İ	239	п
48	0	112	p	176	°	240	р
49	1	113	q	177	±	241	с
50	2	114	r	178	I	242	т
51	3	115	s	179	i	243	у
52	4	116	t	180	r	244	ф
53	5	117	u	181	μ	245	х
54	6	118	v	182	⊖	246	ц
55	7	119	w	183	⊖	247	ч
56	8	120	x	184	è	248	ш
57	9	121	y	185	Ѧ	249	щ
58	:	122	z	186	e	250	ъ
59	;	123	↓	187	≥	251	ы
60	<	124	!	188	Ј	252	ь
61	=	125	>	189	S	253	э
62	>	126	~	190	S	254	ю
63	?	127	Ј	191	i	255	я

Index des mots clés

A	Accès aux données, sur module mémoire	13	Charge du bus, bus de terrain CANopen	60, 62	
	Accès aux E/S, direct	53	Code d'erreur	54	
	Activation des valeurs initiales	45	CoDeSys Gateway Server (serveur pour passerelle CoDeSys)	70	
	Actualisation du système d'exploitation	41	Codeur incrémental	23	
	Adaptateur pour module mémoire	27	COLDSTART	49	
	Adressage d'un automate raccordé à un bus de terrain CANopen	70	Commandes navigateur	59	
	Adresse IP	64	Communication		
	Scrutation/Modification	66	Canal	64	
	Adresse participant		Communication au sein du réseau		
	EASY204-DP, EASY221-CO, EASY222-DN	99	easyNet	87	
	Affichage		Communication avec un automate de destination	71	
	Entrées/sorties des appareils d'extension	40	Comportement à la mise sous tension	41, 67	
	Entrées/sorties locales	39	Comportement au démarrage	41, 67	
	Affichage (Display)	75	Réglage dans XSoft	43	
	Afficheur déporté	75	Compteur incrémental	47	
	Afficheur externe	75	Compteurs	23, 45	
	Afficheur LCD	38	Interruptions	50	
	Afficheur MFD	75	Compteurs rapides	45	
	Afficheur, interactif	75	Compteurs rapides, entrées	12	
	Afficheurs multifonctions	86	Conception, EC4-200	11	
	Appareils d'extension	17	Configuration		
			Easy 205-ASI	98	
			EASY204-DP, EASY221-CO, EASY222-DN	99	
			Entrées/sorties	39	
			MFD4 raccordé à easyNet	92	
			Configuration, XIO-EXT121-1	39	
			Connexion PC	27	
			Counter	45	
			Coupeure de la tension d'alimentation	43	
			Coupeure/Interruption de la tension d'alimentation	43	
			Court-circuit	25	
B	Bibliothèques		D	Définition/Modification de la vitesse de transmission	63
	CANUser.lib, CANUser_Master.lib	7	Démarrage	43	
	EC_File.lib	13	Démarrage/Arrêt via easyNet	96	
	EC_SysLibCom.lib	73	Durée de période	50	
	EC_Util.lib	52, 62	Durée de sauvegarde par pile	14	
	EC_Visu2.lib	62			
	Installation	61	E	EASY-LINK	17, 28
	SysLibRTC	14	easyNet	87	
	Blocs fonctionnels	61, 77	Interface	16, 27	
	Compteurs 16 bits	47	Echange de données, modules de couplage réseau		
	Compteurs 32 bits	46	acyclique	99	
	Disp_DisplayElement	78	cyclique	97, 98	
	Disp_DisplayPage	76, 79	Emission/réception des données utilisateur (via easyNet)	88	
	Disp_GetDisplayInfo	78	Encombres, poids	105	
	Synoptique	76	Entrées		
	Bus, topologie (easyNet)	94	opérandes symboliques	17	
			Type et nombre	11	
			Entrées de diagnostic	12, 17	
			Etablissement liaison PC – EC4-200	63	
			Etude	21	
C	Câble de liaison EASY800-PC-CAB	105			
	CAN				
	Interface	16			
	Liaison	27			
	Paramètres d'un appareil	71			
	Paramètres d'un maître	71			
	Réglages concernant le routage	71			
	Canal de communication	64			
	canload, commande navigateur	60			
	Capacités mémoire	41			
	Capteur (20 mA)				
	Raccordement	22			
	Charge du bus	87			

Exemples	
Accès à un programme d'automate	72
Affichage de textes et valeurs	81
Détection de valeurs analogiques	21
Elaboration d'un programme	80
Fichier STARTUP.INI pour EC4-200	67
Paramétrage du Node-Id, Vitesse de transmission	71
Programme avec appel de fonction	51
Traitement des interruptions	52
Transfert de blocs de données bit	90
Extension centralisée	28
Extension décentralisée	28
Extensions	
Raccordement	28
F	
factoryset	44
Fixation par vis	19
Fonctionnement	61
Fonctions	77
CAN_BUSLOAD	62
DisableInterrupt	52
du mode Transparent	73
EnableInterrupt	52
FileOpen	13
FileRead	13
GetDisplayInfo	12
NET_UPDATE	88
ReadBitDirect	53
Synoptique	76
TimerInterruptEnable	50
Fonctions de test	45
Forçage	45
Forçage de variables et d'entrées/sorties	45
Fonctions	
Disp_EnableDisplay	77
Disp_RegisterVariable	77, 82
NET_GET	88
G	
Générateur d'impulsions	23
H	
Horloge système, sauvegarde par piles	14
Horloge temps réel	14
I	
Indication de la date (principe PUT-GET)	91
Informations appareils, visualisation (easyNet)	96
Installation	21
Interface	
CAN	16
Définition des paramètres de communication	63
EASY-LINK	17
easyNet	16
multifonctions	15
Interruption	52
Interruption de la tension d'alimentation	43
L	
Langue des menus	
Modification	37
Liaison PC – MFD4	
Réglage dans easySoft	95
Liaison TCP/IP (lors du routage)	69
Logiciel de programmation	9
Longueur des câbles	103
M	
Marche	41
Menu	
Guidage par menus	29
Saisie de valeurs	29
Structure	31
Menu principal	
Synoptique	31
Menu spécial	
Synoptique	32
Mise en service	45
MMC	13
Mode Cycle unique	45
Mode Pas à pas	45
Mode transparent	14, 73
Modification de la fonction des dossiers	39
Modification des paramètres	37
Module mémoire	13, 27
Modules de couplage réseau	
Raccordement	28
Montage	
Platine de montage	19
sur profilé chapeau	19
Mot de passe	
Activation	35
Effacement	36
Incorrect	36
Modification	36
Oublié	36
Réglage	35
Suppression de la protection	36
N	
N° de nœud (Node-Id)	70
Navigateur PLC-Browser	59
NET-easyNet	92
No Analog Output (aucune sortie analogique)	39
No Counter (aucun compteur)	39
No Keys (aucune touche de fonction ni de direction)	39
Numéro de nœud (Node-Id)	70
P	
Paramétrage d'un canal	72
Paramétrage, MFD4 dans easySoft-CoDeSys	93
Paramètres de communication	63
Paramètres de communication, réglage	
MFD4 raccordé à easyNet	92
Pattes de montage	19
Plage de mémoires internes, représentation de easy800 sur EC4-200	100
Point d'arrêt (Breakpoint)	45

Possibilités de diagnostic	72	Réponse PING	66
Potentiomètre d'entrée de consignes		Représentation du curseur	29
Raccordement	22	Réseau	
Principe PUT-GET	91	Modules de couplage	97
Programmation via le réseau CAN (routage)	69	Raccordement à easyNet	103
Programme		Réseau easyNet	87
Arrêt	44	Résistances de terminaison de bus (easyNet)	94
Démarrage	43	Routage	
Elaboration, Généralités sur le traitement	80	Conditions préalables	69
Routine	44	Procédure	71
Téléchargement	95	via easyNet	95
Traitement	44	Routine d'application	51
Projet de démarrage (Bootproject)			
Effacement	55	S	
Génération et transfert	55	Sauvegarde par piles	43
Projet de démarrage (bootproject)	41	Sections des câbles	103
Protection des lignes	21	Send I/O	92
		Serveur pour passerelle CoDeSys	70
R		Servovalve	
Raccordement		Raccordement	26
Bouton-poussoir, interrupteur	21	setrtc, commande navigateur	60
Capteur 20 mA	22	Signaux d'entrée/sortie	47
Codeur incrémental	23	Sonde de température	
Compteurs rapides	23	Raccordement	22
Contacteurs, relais	24	Sortie analogique	
Déecteur de proximité	21	Raccordement	26
Entrées analogiques	21	Sorties	
Entrées tout-ou-rien	21	Opérandes symboliques	17
Extensions	20, 28	Raccordement	24
Générateur d'impulsions	23	Type et nombre	13
Modules de couplage réseau	20, 28	Sorties à relais	39
Potentiomètre d'entrée de consignes	22	Sorties à transistors	
Réseau easyNet	103	Modification du type de sortie dans la configuration	39
Servovalve	26	Source d'interruption	53
Sonde température	22	START, événement système	49
Sortie analogique	26	STARTUP.INI	67
Sorties	24	STOP	44, 49
Sorties à relais	24	Surcharge	25
Sorties à transistors	25	Surveillance de courts-circuits	13, 17
Tension d'alimentation	21	Surveillance du temps de cycle	44
Raccordement des entrées analogiques	21	Synoptique des entrées/sorties	17
Raccordement des entrées TOR	21	Synoptique des références	
Raccordement des sorties à relais	24	Automate	9
Raccordement des sorties à transistors	25	Système	
Référencement	48	Evénements	44
Réglage de l'heure		Paramétrage	67
via easyNet	96	Temps	44
Réglage de l'heure	37		
Réglage du contraste de l'afficheur LCD	38	T	
Réglage du jour de la semaine	37	Taille des blocs pour le transfert des données	69
Réglage du rétro-éclairage de l'afficheur LCD	38	Téléchargement, système d'exploitation	56
Réglage Usine	38	Téléchargement/Actualisation du système d'exploitation	56
Réglages usine		Temps d'accès	
Rétablissement	44	Augmentation	46
Relais Outputs	39	Tension d'alimentation	
Remise à zéro (Reset)	44	Raccordement	21
Remote I/O	87	Timer-Interrupt	50
Remote RUN	92		

Topologie (easyNet)94
Topologie linéaire (easyNet)94
Touches12
Touches de commande29
Touches de direction, entrées12
Touches de fonction, entrées12
Transmission de données via easyNet88
<hr/>	
U Utilisation29
<hr/>	
V Variables	
Comportement après une RAZ (Reset)45
Comportement au démarrage43
Variables rémanentes43
Visualisation d'état dans XSoft45
Visualisation d'état au niveau des DEL13
Visualisation des informations relatives aux appareils (easy- Net)96
Vitesse87
Vue d'ensemble des signaux, entrées/sorties47
<hr/>	
W WARMSTART49

DRAFT