

DIFFUSION :	<input checked="" type="checkbox"/>	CONFIDENTIELLE	<input type="checkbox"/>	RESTREINTE	<input type="checkbox"/>	CONTROLEE	<input type="checkbox"/>	NON CONTROLEE	<input type="checkbox"/>
SERVICE EMETTEUR									
DIFFUSION INTERNE					DIFFUSION EXTERNE				
<p><u>Automatisme</u></p> <p>Notice</p> <p>Programmes</p> <p>Généraux</p>									
IND.	DATES	ETABLI	VERIFIE	APPROUVE					
					Conception du système				
	DATE	NOM	VISA	Document de Spécification et de Conception du Système Automatisme					
ETABLI	12/01/2016	J.LAURAIN							
VERIFIE									
APPROUVE				Direction du Patrimoine Immobilier					
REF. DPI :									

Document de spécification et de

Conception du Système (DSCS)

Page d'évolution

Indice	Pages	date	Raison de l'évolution

SOMMAIRE DETAILLE

1. OBJET DU DOCUMENT	5
1.1 INTRODUCTION	5
1.2 PRE REQUIS	5
1.1. GLOSSAIRE	5
2. DESCRIPTION DE LA METHODE	6
2.1 NIVEAU ACTIONNEURS / MESURES / REGULATEURS	6
2.2 NIVEAU GRAFCETS	6
2.3 NIVEAU GRAFCETS MAITRE	6
2.4 EXEMPLE DE MISE EN OEUVRE DE LA METHODE	7
2.4.1 Documentation	7
3. DESCRIPTION DETAILLEES	8
3.1 L'AUTOMATE	8
3.1.1 Organisation mémoire automate	8
3.2 ACTIONNEURS	10
3.2.1 AUTOMATISME	11
3.3 REGULATEURS	13
3.3.1 AUTOMATISME	14
3.4 MESURES	15
3.5 ALARMES	16
3.5.1 AUTOMATISME	16
3.6 GRAFCETS	17
3.6.1 AUTOMATISME	17
4. STRUCTURE DU PROGRAMME	18
4.1 ROLE DES PROGRAMMES (PRG) DE BASES	18
4.1.1 « PLC_PRG »	18
4.1.2 « P01_TRAITEMENTS_PERMANENTS »	19
4.1.3 « P02_COMMUNICATIONS »	20
4.1.4 « P03_MESURES »	20
4.1.5 « P04_DIVERS »	20
4.1.6 « P05_GRAFCETS »	20
4.2 STRUCTURATION DES GRAFCETS	21
4.3 VARIABLES SYSTEME	22
5. PROGRAMMER AVEC LES PG	23
5.1 INTRODUCTION	23
5.2 L'ETUDE	23
5.3 REGLES DE BASES A SUIVRE	24
5.3.1 Types de langages	24
5.3.2 Commentaires.....	24
5.3.3 Règle de nommage.....	25
Exemple : _01234BP01_xNomVariable	26
5.3.4 Où déclarer les variables ?	27
5.3.5 Gestion des versions	27
5.4 DECLARATION DES VARIABLES EN SEMI-AUTOMATIQUE	28
5.4.1 Cartographie et liste des Entrées/Sorties	28
5.4.2 Actionneurs, mesures et régulateurs.....	28
5.4.3 Divers Bits, Mots et Réels.....	29
5.4.4 Grafcets	29
5.5 PROGRAMMATION STRUCTURE GRAFCETS	30
5.6 PROGRAMMATION ECHAINEMENT DES ETAPES D'UN GRAFCET	32

Document de spécification et de Conception du Système (DSCS)

5.7	PROGRAMMATION « DIVERS » D'UN GRAFCET	33
5.8	PROGRAMMATION DES ACTIONNEURS	34
5.9	PROGRAMMATION DES REGULATEURS	35
5.10	PROGRAMMATION DES ALARMES	36
5.11	PROGRAMMATION D'UNE ETAPE GRAFCET	37
5.11.1	Action « _Entree »	37
5.11.2	Action « Divers »	37
5.11.3	Action « Ecriture_CDE_Actionneurs »	38
5.11.4	Action « Sortie_ »	38
5.11.5	Action « Transitions »	39
5.12	PROGRAMMATION DES MESURES	40
5.13	RECOPIE DES ENTREES AUTOMATE VERS LES VARIABLES SUPERVISION	41
5.14	ALARME DEMANDE D'INTERVENTION	42
5.15	MISE EN PLACE D'UNE COMMUNICATION MODBUS TCP UDP	43
5.15.1	Activation de la communication	43
5.15.2	Création d'un message	44
5.16	MISE EN PLACE D'UNE COMMUNICATION MODBUS RTU ASCII	46

I. Objet du document

I.1 INTRODUCTION

Le but de ce document est de décrire la méthode d'analyse, le fonctionnement des **Programmes Généraux** et les principes de réalisation de programme d'automatismes à partir de cette méthode.

Elle est basée sur une programmation automate développée sous CoDeSys 2.3.

I.2 PRE REQUIS

La lecture de ce document nécessite :

- La maîtrise de CoDeSys 2.3.

I.1. GLOSSAIRE

- | | |
|-------------|---|
| ▪ GTC | : G estion T echnique C entralisée |
| ▪ Abyla | : Logiciel de gestion du patrimoine |
| ▪ WAGO | : Matériel d'automatisme |
| ▪ Orchestra | : Plateforme de supervision Wonderware |
| ▪ BdD | : B ase de D onnées |

2. Description de la méthode

C'est une méthode d'organisation de la programmation de l'automate. Elle est structurée en 3 niveaux :

3	GRAFCETS MAÎTRE
2	GRAFCETS
1	ACTIONNEURS / MESURES / REGULATEURS

2.1 NIVEAU ACTIONNEURS / MESURES / REGULATEURS

La méthode banalise la programmation des actionneurs (*vannes, moteurs, régulateurs ...*) par l'utilisation de modèles d'objets. Par principe, tout actionneur appartient à un grafcet et un seul.

La méthode banalise également la programmation des mesures par l'utilisation d'un modèle d'objet. Contrairement aux actionneurs toutes les mesures sont globales et ne sont pas affectées à un grafcet.

2.2 NIVEAU GRAFCETS

Un grafcet est défini par une « partie physique » de l'installation à programmer. Cette partie forme un ensemble « cohérent » (Exemple : Une chaudière, un départ sous-station, une CTA, ...). Elle est composée d'un morceau de circuit ou d'installation (équipement, conduite, bac...) et d'organes de commande (Actionneurs / Régulateurs).

Un grafset est un moteur d'exécution capable d'exécuter une suite d'étapes (*enchaînement séquentiel*).

Une étape est définie comme une phase élémentaire du process : Démarrage, Vidange, Transfert... Elle est affectée à un grafcet et peut commander n'importe quels actionneurs de celui-ci.

Une étape agit sur l'installation au travers des actionneurs/Régulateurs en leur donnant des commandes. Elle reçoit des informations venant des capteurs ou de l'opérateur.

2.3 NIVEAU GRAFCETS MAITRE

Un grafcet maître n'est pas nécessairement défini par une « partie physique » de l'installation. Il est là pour piloter d'autres grafcets qui lui sont esclaves (*Exemple : Cascade de chaudières*).

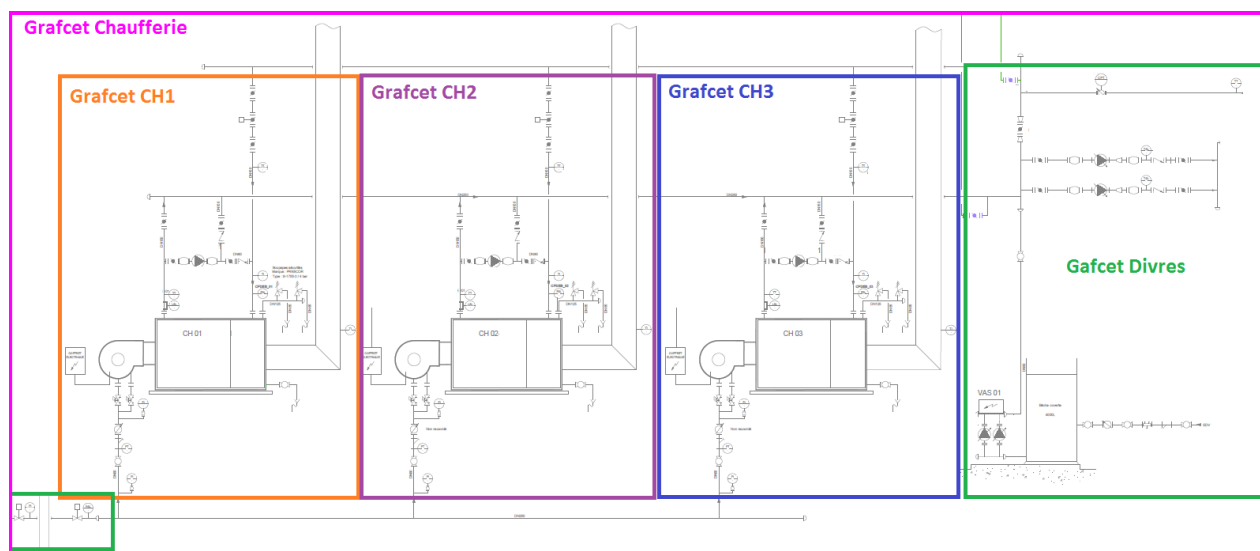
Dans le cas d'installation en série, le grafset maître peut piloter son installation et demander à l'installation qui lui succède de démarrer à son tour.

Un grafcet maître peut-être l'esclave d'un autre grafcet maître. Il n'y a pas de limitation.

2.4 EXEMPLE DE MISE EN OEUVRE DE LA METHODE

Dans le principe, le découpage en grafcet d'une installation se fait en regroupant les Actionneurs permettant de réaliser une fonction.

En ayant une parfaite connaissance du process que l'on cherche à découper, des simplifications peuvent être apportées au découpage en grafcet.



Ce process est une chaufferie constituée de 3 chaudières et de quelques actionneurs périphériques ne pouvant pas être rattachés à l'une ou l'autre des chaudières.

L'installation est donc découpée en 3 grafquets pour les chaudières et un grafcet « Divers » pour les autres actionneurs (*car chaque actionneur doit appartenir à un grafcet*).

La chaufferie doit gérer une cascade de mise en marche des trois chaudières.

Un Grafcet Chauffage (Grafcet Maître) est donc ajouté pour cette gestion. Ce grafcet ne pilote aucun actionneur, mais effectue uniquement la gestion de ses 4 grafquets esclaves.

Ainsi, ce découpage en grafquets va nous permettre de :

- Programmer trois chaudières (avec des fonctionnements pouvant être différents les uns des autres) sans avoir à penser au fonctionnement de l'ensemble de la chaufferie.
- Programmer la gestion de la chaufferie (Cascade chaudières) sans avoir à réfléchir au détail du fonctionnement d'une chaudière. En quelque sorte, les chaudières sont les actionneurs du grafcet chaufferie.
- Faciliter la lecture du programme et donc sa mise en service et son dépannage.

Le grafcet « divers » fait exception. Il regroupe les actionneurs périphériques à la chaufferie n'ayant pas de séquentiel propre mais fonctionnant lorsque la chaufferie est en marche.

2.4.1 Documentation

Le fichier EXCEL « PG_Découpage mémoire API_xxxx_xx_xx_v\$.xlsm » permet de documenter le découpage et la programmation selon la méthode décrite ci-dessus. Il permet également de générer l'ensemble des variables du programme automate et de la GTC.

3. Description détaillées

3.1 L'AUTOMATE

Afin de faciliter les échanges avec la plateforme Archestra (GTC), les objets dans les automates doivent être organisés en tables structurées. Il existe donc une ou plusieurs tables par type d'objet. Chaque élément a dans ses tables, un numéro d'ordre (*index*) saisie dans le fichier de découpage mémoire qui permet de calculer l'adresse de chaque élément.

Le fichier Excel « PG_Découpage mémoire API_xxxx_xx_xx_v\$.xslm » sert de base de travail entre les différents supports de l'application (Base de Données ABYLA – Automates – Supervision) et permet l'homogénéisation des Mnémoniques entre les différentes applications.

On y retrouve différents onglets :

- Découpage mémoire de l'automate : Contient les Zones d'adressage de l'automate, dont les adresses de début sont à renseigner dans la fiche Automate.
- Actionneurs : Contient la liste des Actionneurs et leurs informations (Index, Type, Adresse)
- Mesures : Contient la liste des Mesures et leurs informations (Index, Type, Adresses)
- Régulateurs : Contient la liste des Régulateurs et leurs informations (Index, Type, Adresses)
- Divers : Contient la liste des Divers Bits, Mots et Réels ainsi que leurs informations (Index, Type, Adresses)
- Grafcets : Contient la liste des grafcets et leurs informations (Index, Nom, Adresses)

3.1.1 Organisation mémoire automate

Le nombre d'objets maximum d'un automate est défini de la manière suivante :

- | | |
|---|--------------|
| ▪ Les Actionneurs | → 100 objets |
| ▪ Les Mesures | → 100 objets |
| ▪ Les Régulateurs | → 50 objets |
| ▪ Les Bits divers | → 320 objets |
| ▪ Les Mots divers | → 200 objets |
| ▪ Les Réels divers (ou variables 32 bits) | → 200 objets |

Tous les automates ont le découpage mémoire suivant :













Longueur (Mots)				@ Registre	@ Modbus	Type
256	de IW	0	Entrées Physiques	0	400001	
	à	255		255	400256	BOOL
256	de IW	256	Entrées Réseau	768	400769	
	à	511		1023	401024	BOOL
256	de QW	0	Sorties Physiques	512	400513	
	à	255		767	400768	BOOL
256	de QW	256	Sorties Réseau	256	400257	
	à	511		511	400512	BOOL
100	de MW	0	Actionneurs Etat	12288	412289	
	à	99		12387	412388	INT
100	de MW	100	Mesures Etats	12388	412389	
	à	199		12487	412488	INT
200	de MW	200	Mesures	12488	412489	
	à	399		12687	412688	REAL
400	de MW	400	Mesures Infos	12688	412689	
	à	799		13087	413088	REAL
50	de	800	Régulateurs Etat	13088	413089	
	à	849		13137	413138	INT
1000	de	850	Régulateurs Infos	13138	413139	
	à	1849		14137	414138	REAL
20	de	1850	Supervision Bits divers	14138	414139	
	à	1869		14157	414158	BOOL
200	de	1870	Supervision Mots divers	14158	414159	
	à	2069		14357	414358	INT
800		2070	Supervision Réels divers	14358	414359	
	à	2869		15157	415158	REAL
200		2870	Graficets Infos	15158	415159	
	à	3069		15357	415358	INT
1026		3070	Reserve	15358	415359	
	à	4095		16383	416384	

Dans la supervision l'adressage des informations se fait automatiquement en fonction de l'index et des adresses de début de chaque zone mémoire de l'automate.

Exemple : pour l'actionneur 9, ces informations se trouvent à l'adresse : 412289 + 9 ➔ 412298

3.2 ACTIONNEURS

Un organe de commande est défini comme étant un actionneur pilotable par un automate programmable. Cet actionneur peut avoir ou non des retours d'état câblés sur l'automate.

Type	Fermé/Arrêt	Ouvert/Marche	En défaut	En manu	Verrouillé	Diagnostic
	Gris	Vert	Clignotant Rouge/Gris ou vert	Point rouge sur fond gris ou vert	Point jaune sur fond gris ou vert	
Vanne						oui
Pompe/Moteur						oui
Vanne modulante		 <div>En vert Dès ouverture > 0%</div>				non Généralement associé à un régulateur

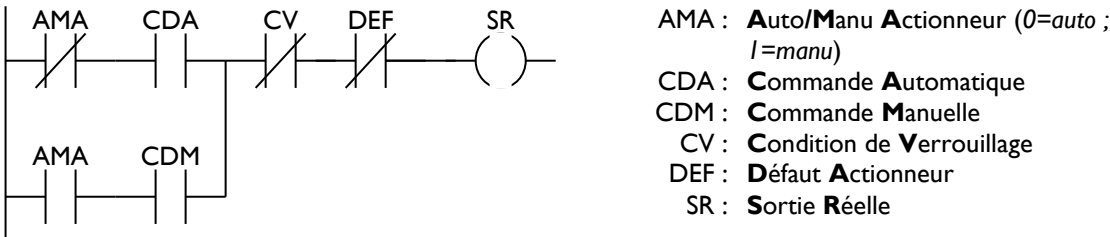
C'est le mot d'état de l'organe qui permet l'animation de toutes ces informations. La structure de ce mot est la suivante :

MOT D'ETAT ACTIONNEUR	.0	SR	Sortie Réelle de la commande actionneur
	.1	XO	Retour commandé
	.2	XF	Retour non commandé
	.3	DEF	Défaut actionneur
	.4	AMA	Auto/Manu Actionneur : FALSE = Auto ; TRUE = Manu
	.5	CDM	Commande Manu de l'actionneur
	.6	OKM	Autorisation de passer l'actionneur en Manu
	.7	CV	Condition de verrouillage. TRUE = Verrouillé
	.8		
	.9		
	.10		
	.11	DEF_Ext	Défaut Actionneur Externe (Exemple : Défaut Thermique)
	.12	Forcage_CV	Demande de forçage des conditions de verrouillage OK
	.13	LOC	Commutateur local de l'actionneur
	.14	ACQ	Acquittement alarme
	.15	Simulation	Mode de simulation

3.2.1 AUTOMATISME

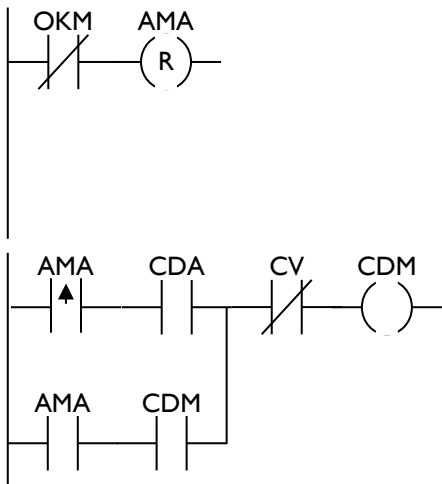
3.2.1.1 La commande des organes

Chaque organe peut être commandé soit automatiquement par le grafcet, soit par l'opérateur via la supervision. Chaque commande ne peut se faire que dans le mode qui lui est associé. L'équation de sortie d'un actionneur est la suivante :



La commande programme n'est active que lorsque actionneur se trouve en mode Automatique.
 La commande depuis la supervision n'est possible que lorsque actionneur se trouve en mode Manuel.
 Le passage en mode automatique ou manuel se fait individuellement par actionneur. Le bit d'autorisation de passage en manuel (OKM) permet d'interdire de manière permanente ou temporaire, le passage en manuel d'un actionneur.

Les équations des bits manu actionneur (AMA) et de commande manuelle (CDM) sont les suivantes :



Lors du passage en mode Manuel d'un actionneur, celui-ci garde la position qu'il avait en mode Automatique. Lorsque l'opérateur repasse un organe en mode Automatique, celui-ci reprend une position correspondant à la commande automatique.

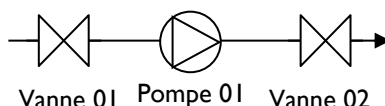
3.2.1.2 Le verrouillage des actionneurs

Qu'un actionneur soit en mode Automatique ou Manuel, il ne peut être commandé que si les conditions de verrouillage (CV) ne sont pas vérifiées. C'est une sécurité sur la marche d'un actionneur.

En mode automatique, lors de la disparition d'une condition de verrouillage, si la commande automatique est encore active, l'organe sera à nouveau commandé.

En mode manuel, lors de l'apparition d'une condition de verrouillage, la commande manuelle est remise à zéro. Même si les conditions sont à nouveaux remplies, l'actionneur ne sera plus commandé sans intervention de l'opérateur.

Exemple :



La pompe n'aura le droit d'être commandée que si les vannes 01 et 02 sont ouvertes. Le verrouillage de la pompe pourra tenir compte des informations de fermeture des vannes 01 et 02. Ces deux retours de fin de course pourront être mis en entrée du bloc de gestion Actionneur sur l'entrée CV.

3.2.1.3 La gestion des défauts Actionneurs

Le programme de gestion des défauts détecte toutes les incohérences entre les retours d'informations et la commande réelle des actionneurs.

Pour générer le défaut, l'incohérence doit être présente durant un temps supérieur au temps de surveillance.

Le défaut d'un actionneur est maintenu et nécessite un acquittement par l'opérateur. Dans ce cas, le bit de défaut peut être intégré dans l'équation de sortie de l'organe.

L'apparition d'un défaut sur un actionneur peut provoquer l'apparition d'un défaut bloquant (*DEF_Bloquant*) ou non bloquant (*DEF_NBloquant*) sur son grafcet d'appartenance. Ce comportement est défini lors de l'analyse et de la programmation des défauts dans les grafquets.

3.2.1.4 Actionneur en position

Un bit actionneur en position (*EP*) permet de dire si celui-ci est en position par rapport à la commande passée par le grafcet (*OrdreMarche*).

3.3

REGULATEURS

Un régulateur est défini comme étant un actionneur ayant une commande modulable (*Analogique*) pilotable par un automate programmable. Dans une régulation cascade la sortie d'un régulateur peut-être la consigne de l'autre.

Type	Mode Auto	Mode Manu	Mode Track	Diagnostic
	Gris	Bleu	Gris clair	
Régulateur	<div>A</div> <div>100%</div>	<div>M</div> <div>100%</div>	<div>T</div> <div>100%</div>	oui

C'est le mot d'état du régulateur qui permet le passage d'informations binaire à la supervision. La structure de ce mot est la suivante :

MOT D'ETAT REGULATEUR	.0	Réglage_PID	Mode Réglage du PID. TRUE = Réglage
	.1		
	.2		
	.3		
	.4		
	.5		
	.6		
	.7		
	.8		
	.9		
	.10		
	.11		
	.12		
	.13		
	.14		
	.15	TRUE	Toujours = True. Indication utilisation Programmes généraux pour la supervision pour une compatibilité avec les anciens programmes

Document de spécification et de Conception du Système (DSCS)

Une structure régulateur permet l'affichage et le passage d'information (Type REEL) entre l'automate et la supervision. La structure est la suivante :

REGULATEUR	Mode	Mode de régulation : 0 = Auto ; 1 = Manu ; 2 = Track
	Consigne	Consigne de régulation
	Mesure	Mesure à réguler
	Sortie	Sortie 0-100% du régulateur
	SortieOP	Sortie demandée par l'opérateur pour le mode Manu
	SortieTrack	Sortie demandée par l'automate pour le mode Track
	Reglage_P	Valeur de l'action proportionnelle
	Reglage_I	Valeur de l'action Intégrale
	Reglage_D	Valeur de l'action Dérivée

3.3.1 AUTOMATISME

Chaque régulateur peut être commandé soit automatiquement par le grafcet, soit par l'opérateur par l'intermédiaire de la supervision.

En Automatique :

- Le régulateur régule selon la consigne qui lui est définie.
- La consigne est définie par l'automate dans le programme.

En Manuel :

- Le régulateur fixe sa sortie avec la valeur demandée par l'opérateur (*SortieTrack*)

En Track :

- Le régulateur fixe sa sortie avec la valeur demandée par l'automate (*SortieTrack*)
- Lorsque le régulateur est dans ce mode, l'opérateur n'a plus la possibilité de changer de mode. C'est l'automate qui a la main sur le régulateur.

Le mode track peut être utilisé pour pré-positionner la sortie régulateur lors d'un démarrage ou pour avoir une sortie fixe (exemple : 100% lors d'une cascade chaudières).

Document de spécification et de Conception du Système (DSCS)

3.4 MESURES

Une mesure permet la lecture des données process analogique d'une installation (*Température, débit, ...*).

Type	Affichage	Diagnostic
	Bleu	
Mesure	18.5 °C	oui

C'est le mot d'état mesure qui permet le passage d'informations binaire à la supervision. La structure de ce mot est la suivante :

MOT D'ETAT MESURE	.0		
	.1		
	.2		
	.3		
	.4		
	.5		
	.6		
	.7		
	.8		
	.9		
	.10		
	.11		
	.12		
	.13		
	.14		
	.15	Simulation	Mode de simulation

Une Tables des mesures contient la liste de toutes les mesures de l'automate. Les Mesures sont du type REEL.

Une structure Infos Mesure permet le passage d'information (Type REEL) entre l'automate et la supervision. La structure est la suivante :

Info Mesure	EchMin	Echelle Mini
	EchMax	Echelle Maxi

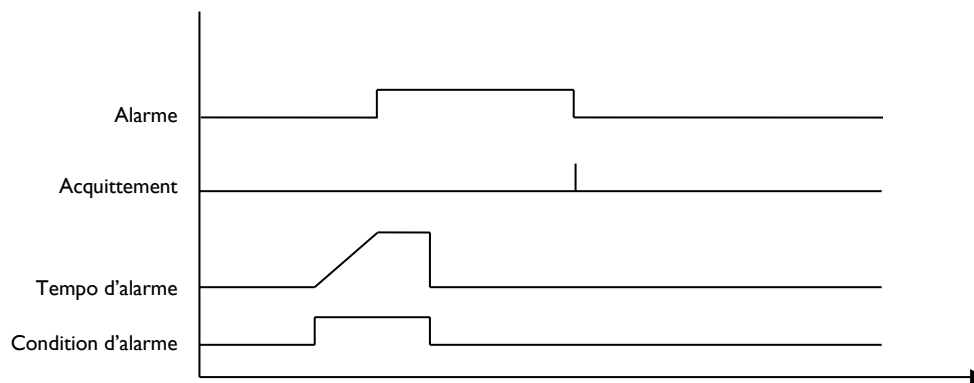
3.5 ALARMES

Une Alarme permet la surveillance d'un état afin de signaler un dysfonctionnement.

3.5.1 AUTOMATISME

Chaque alarme est calculée par un grafcet.

L'alarme est active lorsque les conditions d'alarme sont présentes durant un temps minimum.
Lorsque les conditions d'alarme disparaissent, l'alarme reste active jusqu'à l'acquittement par l'opérateur.



3.6 GRAFCETS

Un grafcet est un moteur d'exécution permettant le pilotage d'un groupe d'actionneurs regroupés en un ensemble cohérent afin d'assurer une fonction process :

- CTA
- Chaudière
- Départ sous-station
- ...

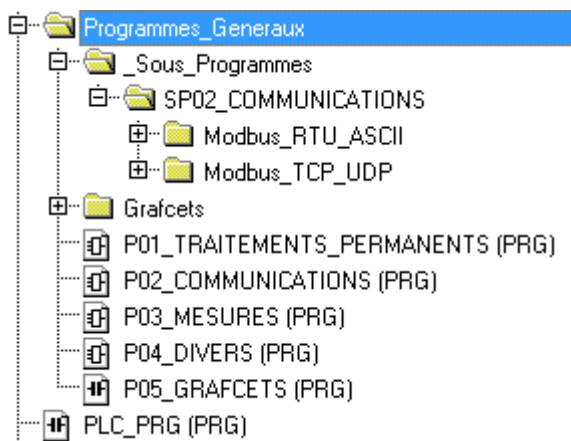
Une structure Grafcet permet de connaître l'état du grafcet. Une bonne connaissance de cette structure peut permettre de simplifier le code de certaine étape grafcet. La structure est la suivante :

Structure Grafcet	NumG7	Index du grafcet
	EtapeEC	Numéro de l'étape en cours de scrutation
	EtapeSUIV	Etape suivante à exécuter (<i>≠ -1 uniquement lorsque CT_OK = TRUE</i>)
	CT_OK	TRUE = Conditions de transition valide
	E_Entree	A TRUE au premier cycle d'exécution d'une étape du Grafcet
	E_Sortie	A TRUE au dernier cycle d'exécution d'une étape du Grafcet
	EP	Grafcet en position. TRUE = Tous les actionneurs du grafcet sont dans la position demandée par l'étape en cours.
	EP_M	Indique le EP du grafcet maître d'un grafcet esclave
	EP_E	A TRUE lorsque tous les grafcets esclaves d'un grafcet maître sont en position.
	AMA	TRUE = Au moins un des actionneurs du grafcet est en manu
	DEF_Bloquant	Présence d'au moins un défaut bloquant dans le grafcet
	DEF_NBloquant	Présence d'au moins un défaut non bloquant dans le grafcet
	DEF_Bloquant_M	Présence d'au moins un défaut bloquant dans le grafcet maître
	DEF_NBloquant_M	Présence d'au moins un défaut non bloquant dans le grafcet maître
	DEF_Bloquant_E	Présence d'au moins un défaut bloquant dans les grafcets esclaves
	DEF_NBloquant_E	Présence d'au moins un défaut non bloquant dans les grafcets esclaves

3.6.1 AUTOMATISME

Cette rubrique est décrite dans le chapitre « Structuration des Grafcets ».

4. Structure du Programme



De base les **Programmes Généraux** contiennent un certain nombre de fichiers programmes PRG. Ces programmes permettent d'organiser l'exécution des tâches de façon toujours identique et de donner à l'utilisateur un squelette de programmes directement opérationnel.

4.1 ROLE DES PROGRAMMES (PRG) DE BASES

4.1.1 « PLC_PRG »

L'utilisateur ne doit effectuer **AUCUNE** modification dans ce programme.

Ce programme appelle les autres programmes selon l'ordre suivant :

- 1 – P01_TRAITEMENTS_PERMANENTS
- 2 – P02_COMMUNICATIONS
- 3 – P03_MESURES
- 4 – P04_DIVERS
- 5 – P05_GRAFCETS

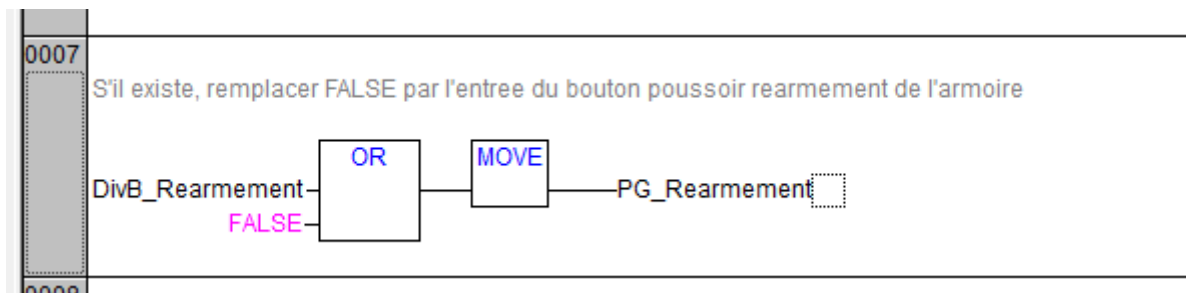
4.1.2 « P01_TRAITEMENTS_PERMANENTS »

Ce programme effectue la gestion des traitements permanents suivant :

- Date courante (Variable structurée : PG_DateCourante)
- Tops système (Variable structurée : PG_TOPS)
- Réarmement automate (Variable : PG_Rearmement)
- Défaut communication avec la supervision (Variable : PG_Defaut_Com_Supervision)
- L'appel éventuel des sous-programmes créés par l'automaticien

Dans ce programme l'automaticien doit modifier uniquement :

- Le réseau 7 si une entrée Réarmement existe dans les entrées de l'automate. Il doit l'ajouter à la place de la constante « FALSE » ci-dessous.



- A partir du réseau 9 l'automaticien peut ajouter l'appel de ces sous-programmes permanents (Exemple : Gestion Alarme anti-intrusion)



Les autres réseaux ne doivent pas être modifiés.

4.1.3 « P02_COMMUNICATIONS »

Ce programme effectue l'appel des sous-programmes de gestion des communications.
En standard ce programme traite les protocoles suivant :

- Modbus RTU ASCII
- Modbus TCP UDP

Aucune programmation n'est à faire pour l'utilisation de l'un de ces deux protocoles de communication.

Si un autre protocole doit être utilisé, le sous-programme de gestion de ce protocole sera à ajouter ici.

4.1.4 « P03_MESURES »

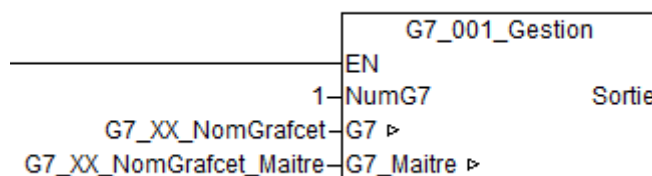
Les mesures n'étant pas affectées à un grafcet. C'est dans ce programme quelles sont programmées.

4.1.5 « P04_DIVERS »

C'est dans ce programme que sont faites les recopies des entrées automate vers les divers bits à destination de la supervision.

4.1.6 « P05_GRAFCETS »

Ce programme effectue l'appel des différents grafcet créés lors de la programmation. C'est lors de cet appel qu'est défini le lien maître/esclave entre les grafkets.

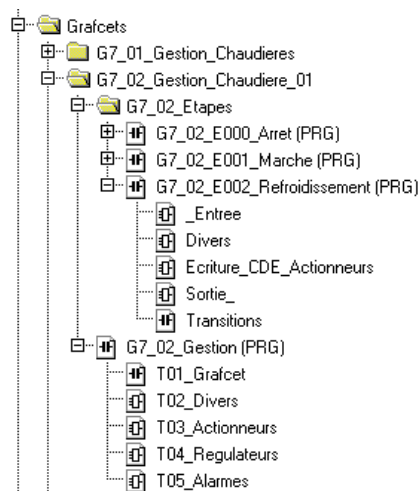


Avec :

- NumG7 : Index du Grafcet
- G7 : Variable structurée du grafcet
- G7_Maitre : Variable structurée du grafcet maître

Pour plus de détail sur les grafket, voir le chapitre « Structure Grafkets »

4.2 STRUCTURATION DES GRAFCETS



Les Grafcets sont programmés dans le dossier « Programmes_Generaux/Grafcets/ » du projet CoDeSys.

Un grafcet est constitué de deux types de programmes (PRG) :

- G7_xx_Gestion
 - Ce programme est le moteur d'exécution du grafcet. Il effectue l'enchaînement des étapes.
 - Effectue la mise à jour des informations de la structure Grafcet
 - Il est constitué de 5 « Actions » (T01 à T05)
 - Le PRG ainsi que ses 5 actions sont exécutés à chaque cycle automate dans l'ordre des numéros d'actions
 - Uniquement les « Actions » (T01 à T05) peuvent être modifiées lors de la programmation
 - T01_Grafcet : Ecriture de l'enchaînement des étapes du grafcet
 - T02_Divers : Gestion des traitements permanents liés au grafcet (*Exemple : Calcul d'états de fonctionnement*)
 - T03_Actionneurs : Programmation des actionneurs affectés au grafcet
 - T04_Regulateurs : Programmation des régulateurs affectés au grafcet
 - T05_Alarmes : Calcul des alarmes process du grafcet
- G7_xx_Eyyy_NomEtape
 - Ce programme est le moteur d'exécution d'une étape du grafcet.
 - Il est constitué de 5 « Actions »
 - Uniquement les « Actions » peuvent être modifiées lors de la programmation
 - _Entree : Action à effectuer en entrée d'étape (*Exemple : Initialisation*)
 - Divers : Traitements permanents de l'étape
 - Ecriture_CDE_Actionneurs : Programmation des commandes actionneurs
 - Sortie_ : Action à effectuer en sortie d'étape (*Exemple : RAZ*)
 - Transitions : Test des conditions de transition de l'étape
 - Le PRG est exécuté uniquement lorsque l'étape est active dans le grafcet
 - Les « Actions » sont exécutées dans l'ordre et de la façon suivante :
 - _Entrée : Uniquement lors de la première scrutation de l'étape
 - Divers : A chaque cycle lorsque l'étape est active
 - Ecriture_CDE_Actionneurs : A chaque cycle lorsque l'étape est active
 - Sortie_ : Uniquement lors de la dernière scrutation de l'étape
 - Transitions : A chaque cycle lorsque l'étape est active

4.3 VARIABLES SYSTEME

Les Programmes Généraux contiennent un certain nombre de variables génériques pouvant être utilisées lors du développement. Leurs rôles est de faciliter le développement :

- PG_DT_Systeme : Date courante de l'automate avec gestion du passage à l'heure d'été.
- PG_DateCourante : Structure avec les informations de la date courante
 - o dwLowMSecs
 - o dwHighMsec
 - o Year
 - o Month
 - o Day
 - o Hour
 - o Minute
 - o Second
 - o Milliseconds
 - o DayOfWeek
- PG_TOPS : Structure de bits TOP. Chaque top n'est à TRUE qu'un cycle. Il n'est pas utile de faire une détection de front
 - o de_100ms
 - o de_500ms
 - o de_1s
 - o de_6s
 - o de_10s
 - o de_30s
 - o de_1min
 - o de_10min
 - o de_30min
 - o de_1h
 - o de_24h
 - o a_24h
- PG_BitPoubelle : Peut-être utilisé comme bit de travail si sa valeur n'est pas à conserver entre 2 cycles.
- PG_Rearmement : Bit de réarmement venant de la supervision
- PG_Default_Com_Supervision : Bit à TRUE lorsque la communication entre l'automate et la GTC est perdu.
- PG_Simulation : Permet le passage en simulation des mesures et des actionneurs d'un automate dans le but de faire des tests plateforme sans avoir l'installation physique.
- INST_WORD_TO_BIT : A utiliser lors de l'utilisation du FB « WORD_TO_BIT ». Ce FB ne mémorise aucune donnée entre 2 cycles. Il n'est donc pas nécessaire de créer de multiple instance de ce FB.

5. Programmer avec les PG

5.1 INTRODUCTION

Les **Programmes Généraux (PG)** sont une méthode de programmation permettant de standardiser les programmes afin qu'ils soient compréhensibles par n'importe quel automaticien maîtrisant ce standard.

Ce standard nécessite de la rigueur lors de l'étude et du codage. Mais il doit éviter les dérives de programmation propre à l'imagination de chacun.

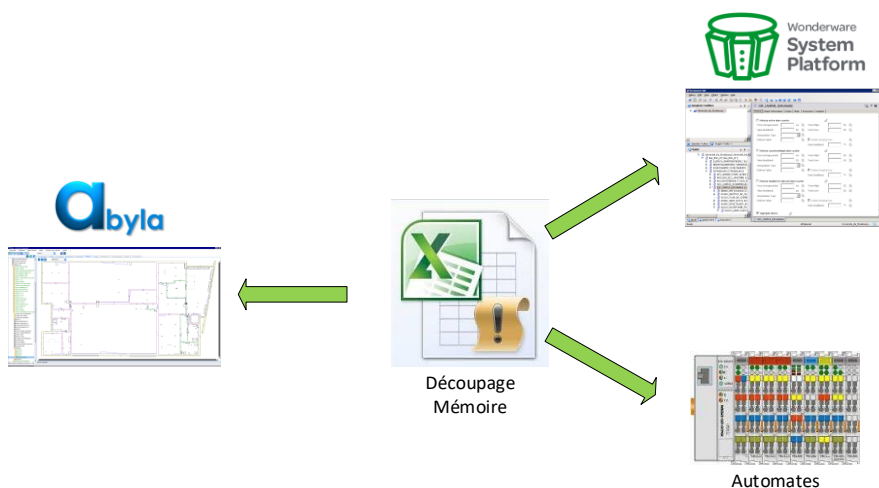
En standard, les Programmes Généraux traitent les aspects suivant :

- Mot de vie avec la supervision
- Réarmement depuis la supervision
- Traitement des Mesures
- La communication Modbus TCP/UDP
- La communication Modbus RTU
- Le traitement des Graficets

5.2 L'ETUDE

La première phase est l'étude. Elle consiste à renseigner le fichier de découpage mémoire « PG_Découpage mémoire API_XXXX_XX_XX_v\$.xlsm ».

Cette phase est **primordiale**. **Aucune programmation ne doit être faite avant que ce fichier ne soit renseigné.** Le découpage mémoire permet de générer la quasi-totalité des variables automate et la totalité des variables de supervision. Le gain de temps lié au développement passe par une phase d'étude aboutie.



5.3 REGLES DE BASES A SUIVRE

5.3.1 Types de langages

L'utilisation du langage « FBD » (function block diagram) est à privilégier lors des développements.

Lorsque cela le nécessite, il sera possible d'utiliser les langages « LD » (ladder diagram) ou « ST » (Structured Text). Les autres langages sont à proscrire.

5.3.2 Commentaires

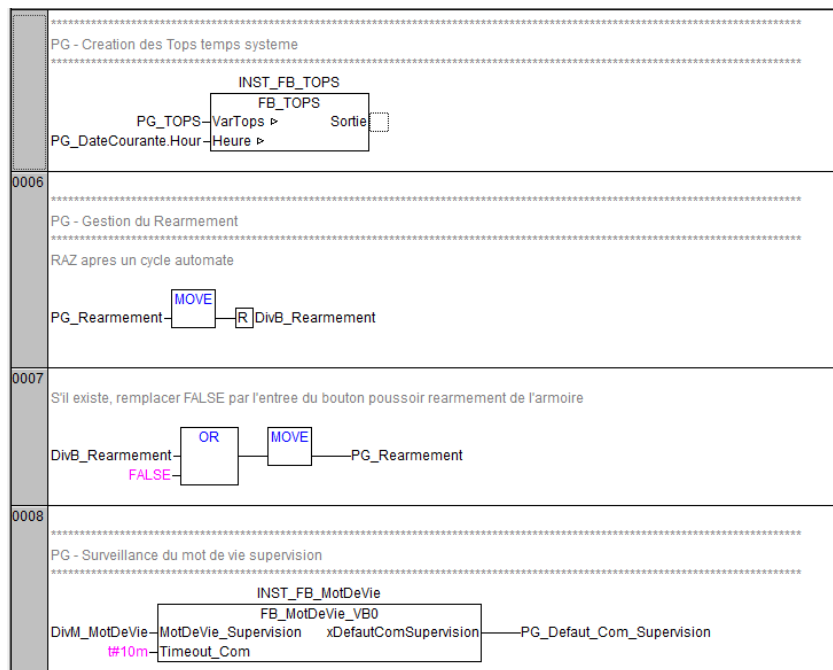
- Toute variable doit être commentée :
 - Inputs
 - Outputs
 - Registres
 - Variables locales
 - Variables globales.
 - Variables de FB/FC

Example :

SV_ACT_Etat	AT %MW0 : K_ACT_TAB_Etat;	(* Actionneur - Mots d'etat *)
SV_MES_Etat	AT %MW100 : K_MES_TAB_Etat;	(* Mesure - Mots d'etat *)
SV_MES_Infos	AT %MW400 : K_MES_TAB_Infos;	(* Mesure - Infos *)
SV_REG_Etat	AT %MW800 : K_REG_TAB_Etat;	(* Regulateur - Mots d'etat *)
SV_REG_Infos	AT %MW850 : K_REG_TAB_Infos;	(* Regulateur - Infos *)

- Les réseaux doivent être commentés au maximum

Example :



5.3.3 Règle de nommage

5.3.3.1 Fichiers Programmes

Afin de donner de la lisibilité au code, les fichiers programmes doivent respecter les règles de nommage suivantes :

Type	Emplacement	Règle de nommage (Exemple)	Explication
Sous-Programme (PRG)	Programmes_Generaux/_Sous_Programmes/	SP01_NomSousProgramme	« SP01_ » est une partie fixe indiquant que le sous-programme est appelé à partir du programme « P01_TAITEMENTS_PERMANENTS ». La suite correspond au nom du sous-programme.
Dossier	Programmes_Generaux/_Sous_Programmes/SP02_COMMUNICATIONS/	Modbus_RTU_ASCII	Le nom du Dossier correspond au nom du Protocol de communication. L'exemple correspond à du Modbus_RTU_ASCII
Sous-Programme (PRG)	Programmes_Generaux/_Sous_Programmes/SP02_COMMUNICATIONS/Nom du protocol	SP02_MSTR01_COMM_Protocol	« SP02_ » est une partie fixe indiquant que le sous-programme est appelé à partir du programme « P02_COMMUNICATIONS ». « MSTR01_COMM_ » Signifie MASTER01 de communication, avec le 01 pouvant être incrémenté dans le cas de l'utilisation de plusieurs cartes de communication. La suite correspond au nom du Protocol.
Action	Programmes_Generaux/_Sous_Programmes/SP02_COMMUNICATIONS/Nom du protocol/SP02_MSTR01_COMM_NomProtocol.xxx	MSG_xx_LibelleTrame	« MSG_xx_ » indique le numéro du message traité par le programme de gestion du Protocol. La suite est libre et doit décrire le rôle de la trame de communication.
dossier	Programmes_Generaux/Grafcets/	G7_xx_LibelleG7	« G7_ » indique qu'il s'agit d'un Grafcet. « xx_ » est l'index du grafcet dans le fichier découpage mémoire. La suite est libre et correspond au libellé du grafcet.
dossier	Programmes_Generaux/Grafcets/G7_01_LibelleG7/	G7_xx_Etapes	« G7_ » indique qu'il s'agit d'un Grafcet. « xx_ » est l'index du grafcet dans le fichier découpage mémoire. Et "Etapes" indique qu'il s'agit du dossier ou seront stockées les étapes du grafcet.
Sous-Programme (PRG)	Programmes_Generaux/Grafcets/G7_01_LibelleG7/G7_xx_Etapes/	G7_xx_Eyyy_Arret	« G7_ » indique qu'il s'agit d'un élément de grafcet. « xx_ » correspond à l'index du grafcet dans le fichier découpage mémoire. « Eyyy_ » correspond au numéro de l'étape du grafcet. La suite est le libellé de l'étape.
Sous-Programme (PRG)	Programmes_Generaux/Grafcets/G7_01_LibelleG7/	G7_xx_Gestion	« xx » est l'index du grafcet dans le découpage mémoire.
FB	PRG_FC_FB/Outils/...	FB_NomDuFB_Va0	« FB_ » pour indiquer le type de bloc. « _Va0 » = version du bloc.
FC	PRG_FC_FB/Outils/...	FC_NomDuFC_Va0	« FC_ » pour indiquer le type de bloc. « _Va0 » = version du bloc.

Document de spécification et de Conception du Système (DSCS)

5.3.3.1 Variables

La plupart des variables automate sont générées à partir du fichier de Découpage Mémoire. Cependant et afin de donner de la lisibilité au code, les variables déclarées manuellement doivent respecter les règles de nommage suivantes :

Type	Emplacement	Règle de nommage (Exemple)	Explication
Input TOR	Configuration de l'Automate	IN_NomVariable	
Output TOR	Configuration de l'Automate	OUT_NomVariable	
Input ANA	Configuration de l'Automate	INA_NomVariable	
Output ANA	Configuration de l'Automate	OUTA_NomVariable	
Variable de communication en Réception	Variables globales/VG_COMMUNICATION	DialRxx_NomVariable	« xx » correspond au numéro du message utilisé pour la communication.
Variable de communication en Emission	Variables globales/VG_COMMUNICATION	DialExx_NomVariable	« xx » correspond au numéro du message utilisé pour la communication.
Déclaration d'instance de FB	Dans le fichier programme ou est utilisé le FB	INST_NomDuFB	
Divers Bit	Variables globales/ VG_Divers_BIT_MOT_REAL	DivB_NomVariable	
Divers Mot	Variables globales/ VG_Divers_BIT_MOT_REAL	DivM_NomVariable	
Divers Réel	Variables globales/ VG_Divers_BIT_MOT_REAL	DivR_NomVariable	
Déclaration de grafcet	Variables globales/ VG_Grafcets	G7_xx_LibelleGrafcet	« xx » correspond à l'index du grafcet dans le fichier découpage mémoire.
Variable débutant par un numérique		_0123456789_NomVariable	CoDeSys ne supportant pas les mnémoniques débutants par un numérique, Ceux-ci devront débiter par « _ ».

Pour les variables interne à l'automate et non adressées, la zone « NomVariable » doit indiquer le type de donnée :

Type	Particule à introduire devant le nom
BOOL	x
REAL	r
DATE AND TIME	dt
WORD	w
DOUBLE WORD	dw
TIME	t
INT	i
DINT	di
UDINT	ui

Exemple : 01234BP01 xNomVariable

5.3.4 Où déclarer les variables ?

Toutes les variables adressées doivent être déclarées dans le fichier « Variables globales » correspondant au type de variables :

- VG_ACT_MES_REG : Variables Actionneurs, mesures et régulateur
- VG_Divers_BIT_MOT_REAL : Variables Divers Bit, Mot et Réel
- VG_Grafcets : Structures Grafcets

Toutes les variables non adressées doivent être déclarées au plus près de leurs utilisations :

- VG_COMMUNICATIONS : Variables de communication
- VG_Automate : Variables globales utilisé dans tout l'automate et sans lien avec un grafcet en particulier.
- Fichier PRG grafcet : Variables globale au grafcet/Etapes du grafcet.
- Fichier PRG étape : Variables uniquement utilisées pas une étape.

5.3.5 Gestion des versions

Lorsqu'une modification de code est apportée au programme une nouvelle version doit être créée.

- Modifier le fichier PRG « versions » afin d'assurer le suivi des modifications.
- Enregistrer le fichier avec le nouvel indice de version.

5.4 DECLARATION DES VARIABLES EN SEMI-AUTOMATIQUE

Le fichier Découpage Mémoire génère la plupart des variables automate. Ce chapitre décrit les variables concernées et la méthode d'intégration des variables générées dans le projet automate.

Cette rubrique n'est utilisable que lorsque l'étude et le fichier découpage mémoire sont terminés.

5.4.1 Cartographie et liste des Entrées/Sorties

(Ce chapitre n'est valable que pour les automates WAGO. Une procédure similaire pourra être créée pour d'autres marques.)

Nous allons voir ici comment créer la cartographie ainsi que la déclaration des Entrées / Sorties du projet à partir du découpage mémoire.

Pour cela il faut utiliser les deux onglets « Liste_des_E_S » et « Aide_Wago_Carto_XML » du fichier découpage mémoire.

Etape	Action	Résultat																																																																																																																																																																																																																																																																																																																																																																																																																																																
1	Dans CoDeSys : Exportez la configuration automate Vide .																																																																																																																																																																																																																																																																																																																																																																																																																																																	
2	Ouvrir l'export avec le bloc-notes	Une structure XML se trouve dans le fichier.																																																																																																																																																																																																																																																																																																																																																																																																																																																
3	Dans l'onglet « Liste_des_E_S ». Sélectionner une carte (Toutes les colonnes) et copier la sélection																																																																																																																																																																																																																																																																																																																																																																																																																																																	
4	Aller sur l'onglet « Aide_Wago_Carto_XML » et coller la sélection sur le modèle de carte correspondant	<div>Sur la partie droite dans la zone verte des données se sont actualisées</div> <div><table><tr><th></th><th>A</th><th>B</th><th>C</th><th>D</th><th>E</th><th>F</th><th>G</th><th>H</th><th>I</th><th>J</th><th>K</th><th>L</th><th>M</th><th>N</th><th>O</th></tr><tr><th></th><th></th><th>N° Carte</th><th>Référence</th><th>Signal</th><th>Affectation Automate</th><th>Prefixe</th><th>Mnémonique</th><th>Description</th><th></th><th></th><th></th><th></th><th></th><th></th><th></th></tr><tr><td>20</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>21</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>22</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>23</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>24</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>25</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>26</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>27</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>28</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>29</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>30</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>31</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>32</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>33</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>34</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>35</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>36</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>37</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>38</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>39</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>40</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>41</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>42</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>43</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>44</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr></table></div>		A	B	C	D	E	F	G	H	I	J	K	L	M	N	O			N° Carte	Référence	Signal	Affectation Automate	Prefixe	Mnémonique	Description								20																21																22																23																24																25																26																27																28																29																30																31																32																33																34																35																36																37																38																39																40																41																42																43																44															
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O																																																																																																																																																																																																																																																																																																																																																																																																																																			
		N° Carte	Référence	Signal	Affectation Automate	Prefixe	Mnémonique	Description																																																																																																																																																																																																																																																																																																																																																																																																																																										
20																																																																																																																																																																																																																																																																																																																																																																																																																																																		
21																																																																																																																																																																																																																																																																																																																																																																																																																																																		
22																																																																																																																																																																																																																																																																																																																																																																																																																																																		
23																																																																																																																																																																																																																																																																																																																																																																																																																																																		
24																																																																																																																																																																																																																																																																																																																																																																																																																																																		
25																																																																																																																																																																																																																																																																																																																																																																																																																																																		
26																																																																																																																																																																																																																																																																																																																																																																																																																																																		
27																																																																																																																																																																																																																																																																																																																																																																																																																																																		
28																																																																																																																																																																																																																																																																																																																																																																																																																																																		
29																																																																																																																																																																																																																																																																																																																																																																																																																																																		
30																																																																																																																																																																																																																																																																																																																																																																																																																																																		
31																																																																																																																																																																																																																																																																																																																																																																																																																																																		
32																																																																																																																																																																																																																																																																																																																																																																																																																																																		
33																																																																																																																																																																																																																																																																																																																																																																																																																																																		
34																																																																																																																																																																																																																																																																																																																																																																																																																																																		
35																																																																																																																																																																																																																																																																																																																																																																																																																																																		
36																																																																																																																																																																																																																																																																																																																																																																																																																																																		
37																																																																																																																																																																																																																																																																																																																																																																																																																																																		
38																																																																																																																																																																																																																																																																																																																																																																																																																																																		
39																																																																																																																																																																																																																																																																																																																																																																																																																																																		
40																																																																																																																																																																																																																																																																																																																																																																																																																																																		
41																																																																																																																																																																																																																																																																																																																																																																																																																																																		
42																																																																																																																																																																																																																																																																																																																																																																																																																																																		
43																																																																																																																																																																																																																																																																																																																																																																																																																																																		
44																																																																																																																																																																																																																																																																																																																																																																																																																																																		
5	Copier/Coller la zone verte du fichier Excel dans le Bloc-notes Comme expliqué en haut de la feuille Excel.																																																																																																																																																																																																																																																																																																																																																																																																																																																	
6	Reproduite les étapes 3 à 5 pour l'ensemble des cartes.																																																																																																																																																																																																																																																																																																																																																																																																																																																	
7	Enregistrer le fichier Bloc-notes																																																																																																																																																																																																																																																																																																																																																																																																																																																	
8	Importer le fichier ainsi obtenu dans CoDeSys																																																																																																																																																																																																																																																																																																																																																																																																																																																	
9	Ajouter éventuellement les cartes manquantes à la cartographie																																																																																																																																																																																																																																																																																																																																																																																																																																																	

5.4.2 Actionneurs, mesures et régulateurs

Document de spécification et de Conception du Système (DSCS)

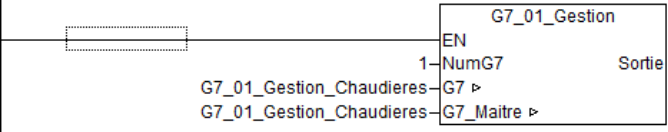
5.5 PROGRAMMATION STRUCTURE GRAFCETS

Nous allons voir ici comment créer la structure programme d'un grafcet.

Pour cela il faut utiliser l'onglet « Grafquets » du fichier découpage mémoire ainsi que l'analyse fonctionnelle des grafquets.

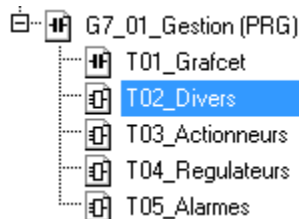
Etape	Action	Résultat
I	La structure grafcet est la suivante	
2	Pour créer un nouveau grafcet : Créer les dossiers « G7_xx_NomGrafcet » et « G7_xx_Etapes »	
3	Aller dans le dossier « PRG_FC_FB/Outils/Construction_ Grafcet/G7_XX_NomGrafcet/ » Et faire clic droit/copier objet sur le PRG « G7_XX_Gestion » Remplacer XX par l'index du grafcet	Le fichier PRG est créé en bas du projet.
4	Glisser le fichier PRG dans le dossier « G7_xx_NomGrafcet/ »	
5	Aller dans le dossier « PRG_FC_FB/Outils/Construction_ Grafcet/G7_XX_NomGrafcet/ G7_XX_Etapes/ » Et faire clic droit/copier objet sur le PRG « G7_XX_EYYY_NomEtape » Remplacer XX par l'index du grafcet, YYY par le numéro d'étape et NomEtape par son libellé.	Le fichier PRG est créé en bas du projet.
6	Glisser le fichier PRG dans le dossier « G7_xx_NomGrafcet/ G7_XX_EYYY_NomEtape/ »	
7	Pour chaque étape du grafcet reproduire les points 5 et 6.	

Document de spécification et de Conception du Système (DSCS)

8	Chaque PRG « G7_XX_Gestion » doit être appelés dans le PRG « P05_GRAFCETS »	<div data-bbox="608 277 655 544">0002</div> <div data-bbox="683 309 1045 331">Grafcet 01 - Gestion Cascade des chaudières</div> 
9	Si le grafcet à un Maître, la structure G7 du maître doit être mise sur l'entrée « G7_Maitre » de l'appel. Sinon c'est la structure du G7 qui est mise également sur cette entrée.	

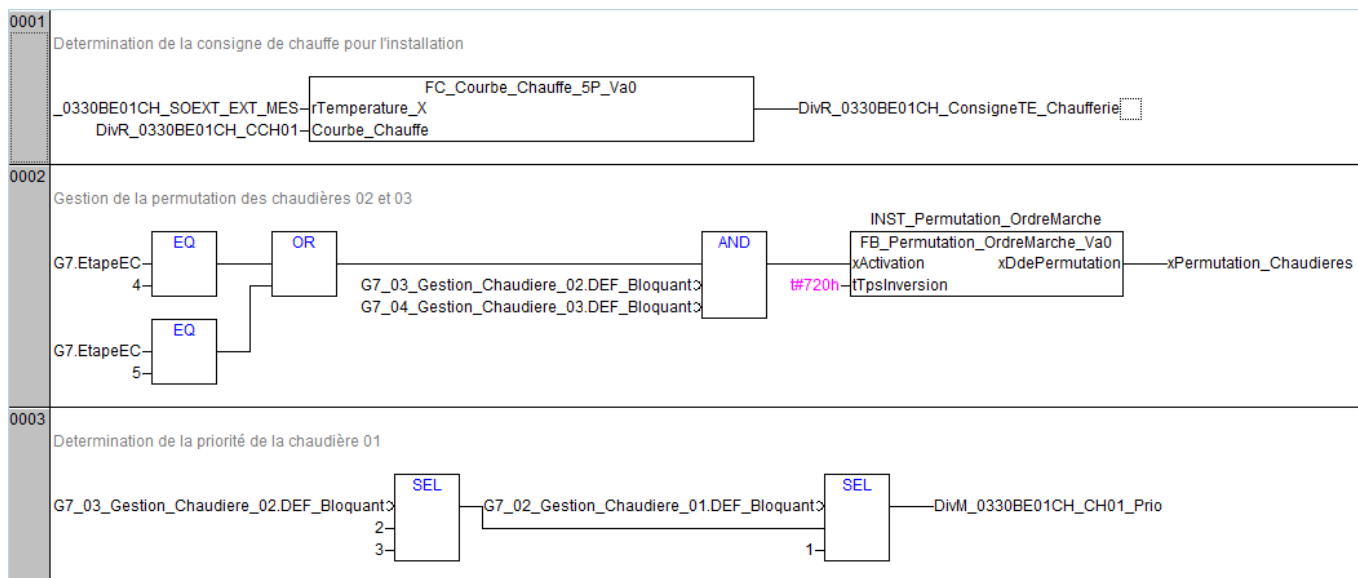
5.7 PROGRAMMATION « DIVERS » D'UN GRAFCET

Les actions permanentes non liées à une étape précise d'un grafcet sont à programmer dans le PRG « G7_xx_Gestion » action « T02 Divers ».



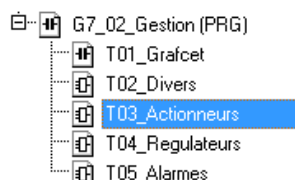
Les informations programmées ici sont du type :

- Calcul de consigne
- Gestion des priorités chaudière
- ...

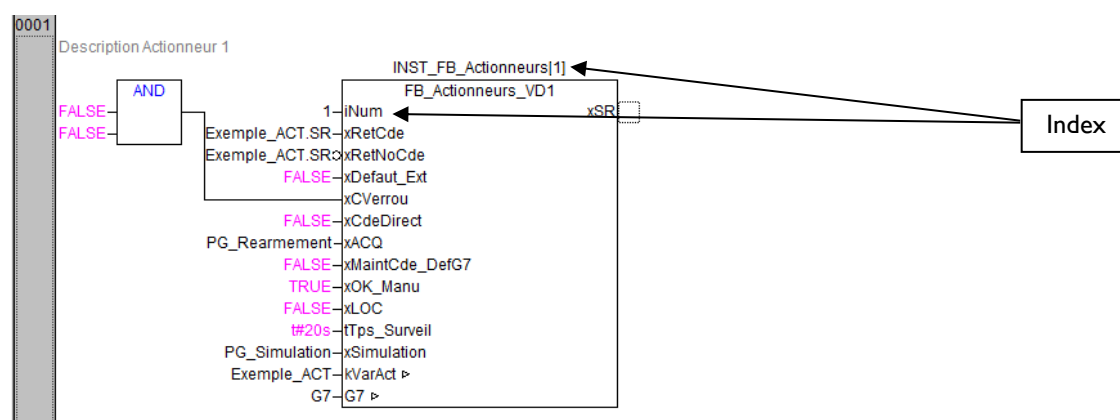


5.8 PROGRAMMATION DES ACTIONNEURS

Les actionneurs sont programmés au niveau des grafjets d'appartenances dans le PRG « G7_xx_Gestion » action « T03_Actionneurs » :



Chaque actionneur est programmé de la façon suivante :



- Une description doit être saisie.
- INST_FB_Actionneurs[x] avec x = index de l'actionneur.

Entrées :

- iNum : Index de l'actionneur
- xRetCde : Retour commandé (Entrée automate → Fin de course ouvert ou Retour marche)
- xRetNoCde : Retour non commandé (Entrée automate → Fin de course fermé ou Pas Retour marche)
- xDefault_Ext : Défaut externe (Défaut thermique, ...)
- xCVerrou : Condition de verrouillage (Exemple pour une pompe : Vanne amont et aval ouvertes)
- xCdeDirect : Commande direct. (Commande l'actionneur directement depuis le bloc.
Ex : pompe de relevage sur niveau haut)
- xACQ : Demande d'acquiescement du défaut actionneur (toujours = PG_Rearmement)
- xMaintCde_DefG7 : Indique si l'actionneur a sa commande maintenue en cas de défaut bloquant de son grafjet d'appartenance. (Exemple : maintien d'actionneur de mise en sécurité)
- xOK_Manu : TRUE = autorise le passage en Manu de l'actionneur depuis la supervision
- xLOC : Mode local. (Entrée automate → Commutateur local d'un moteur).
Evite la génération du défaut lorsque l'actionneur est en mode local.
- tTps_Surveil : Temps de surveillance avant apparition du défaut.
- xSimulation : TRUE = Actionneur en simulation. xRetCde et xRetNCde sont simulés par l'automate (Mettre PG_Simulation. Permet la simulation de l'automate lors des tests plateforme)
- kVarAct : Variable Actionneur.
- G7 : Structure Grafjet du Grafjet d'appartenance (Toujours « G7 »).

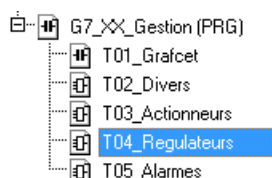
Sorties :

- xSR : Sortie réelle de l'actionneur (Sortie automate)

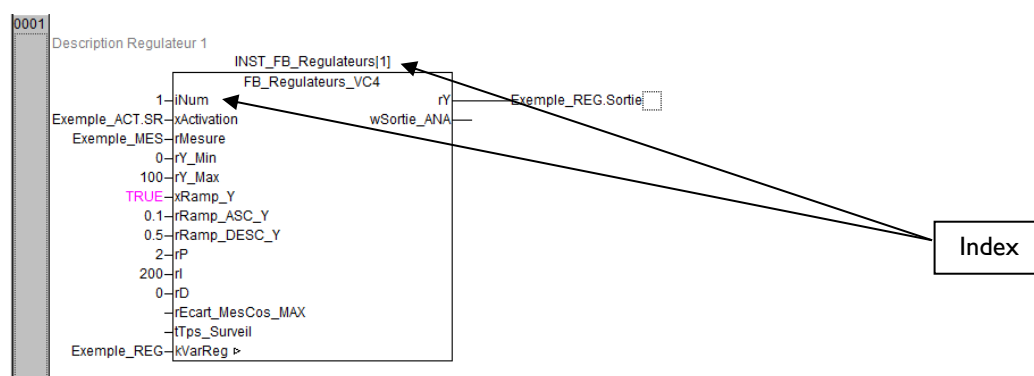
xRetCde et xRetNoCde sont obligatoires. Si l'une ou l'autre de ces informations n'existe pas, il faut la simuler à partir d'une des autres informations RetNoCde ou même le SR. Si l'entrée n'est pas renseignée, un défaut actionneur sera généré automatiquement par le bloc actionneur.

5.9 PROGRAMMATION DES REGULATEURS

Les régulateurs sont programmés au niveau des grafjets d'appartenances dans le PRG « G7_xx_Gestion » action « T04_Regulateurs » :



Chaque régulateur est programmé de la façon suivante :



- Une description doit être saisie.
- INST_FB_Regulateurs[x] avec x = index du régulateur.

Entrées :

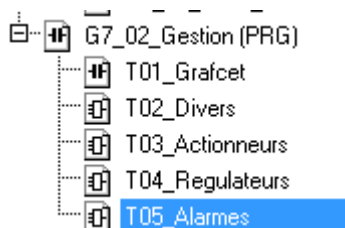
- iNum : Index du régulateur
- xActivation : Activation de la régulation. (Exemple : la commande de la pompe lors d'une régulation de débit. Si la pompe est à l'arrêt la régulation ne doit pas être active.)
- rMesure : Mesure à utiliser pour la régulation (Variable de la zone mesures)
- rY_Min : Talon mini de la sortie régulateur (0-100%)
- rY_Max : Valeur maxi de la sortie régulateur (0-100%)
- xRamp_Y : TRUE pour activer l'utilisation d'une rampe sur la sortie régulateur
- rRamp_ASC_Y : Valeur du nombre d'incrément par seconde sur la sortie lorsque la rampe est activée.
- rRamp_DESC_Y : Valeur du nombre de décrétement par seconde sur la sortie lorsque la rampe est activée.
- rP : Action proportionnelle
- rI : Action intégrale
- rD : Action dérivée
- rEcart_MesCos_MAX : Ecart mesure/consigne max avant affichage d'une alerte sur le GTC
- tTps_Surveil : Temps de surveillance de la dérive mesure/consigne avant affichage de l'alerte sur la GTC
- kVarReg : Structure Régulateur interne à l'automate

Sorties :

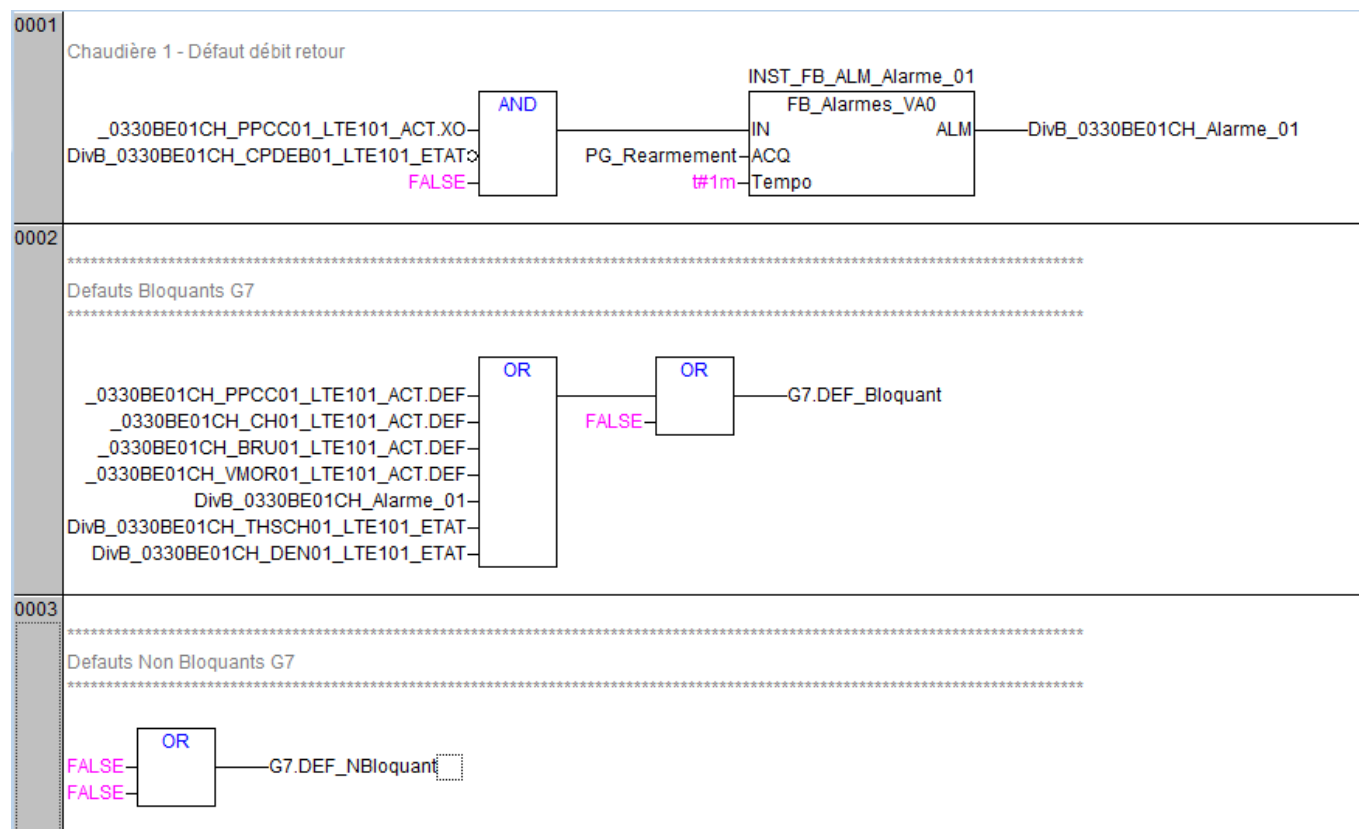
- rY : Valeur de la sortie régulateur. (Mettre *StructureRegulateur.Sortie* pour afficher la valeur en visu dynamique)
- wSortie_ANA : Sortie automate. (A utiliser lorsque le régulateur doit piloter une sortie analogique « 0-32767 points »)

5.10 PROGRAMMATION DES ALARMES

Les Alarmes process sont programmées au niveau des grafjets dans le PRG « G7_xx_Gestion » action « T05_Alarmes » :



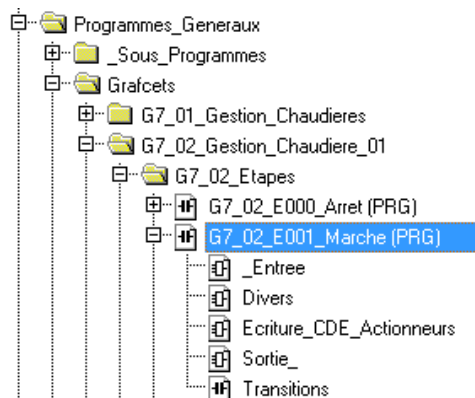
Les alarmes sont programmées de la façon suivante :



- L'utilisation du « FB_Alarmes_VA0 » permet :
 - o De temporiser l'apparition de l'alarme
 - o L'auto-maintien de l'alarme jusqu'au réarmement
- Si une alarme ne doit pas être auto-maintenue, il ne faut pas utiliser « FB_Alarmes_VA0 ».
- Les alarmes doivent être classées soit en défaut bloquant ou en défaut non bloquant
(Cela permet de remonter au niveau de la structure grafjet l'information regroupée sur le type d'alarme en cours)

5.11 PROGRAMMATION D'UNE ETAPE GRAFCET

Les Etapes sont programmées dans le dossier « /G7_xx_Etapes/ » et sont constituées d'actions :



Le PRG « G7_xx_NomEtape » ne doit pas être modifié par l'automaticien. Uniquement les actions sont à programmer.

5.11.1 Action « _Entree »

S'exécute au premier cycle d'exécution lors de l'activation de l'étape.

Ici doit être codée toute action d'initialisation de variable permettant la bonne exécution de l'étape.

Exemple :

- Initialisation d'une valeur pour un décompte
- RAZ de Bit de flags
- Forçage d'un régulateur en mode Track
- ...

5.11.2 Action « Divers »

S'exécute en permanence lorsque l'étape est active.

Ici doit être codé tout traitement permanent lors de l'exécution de l'étape.

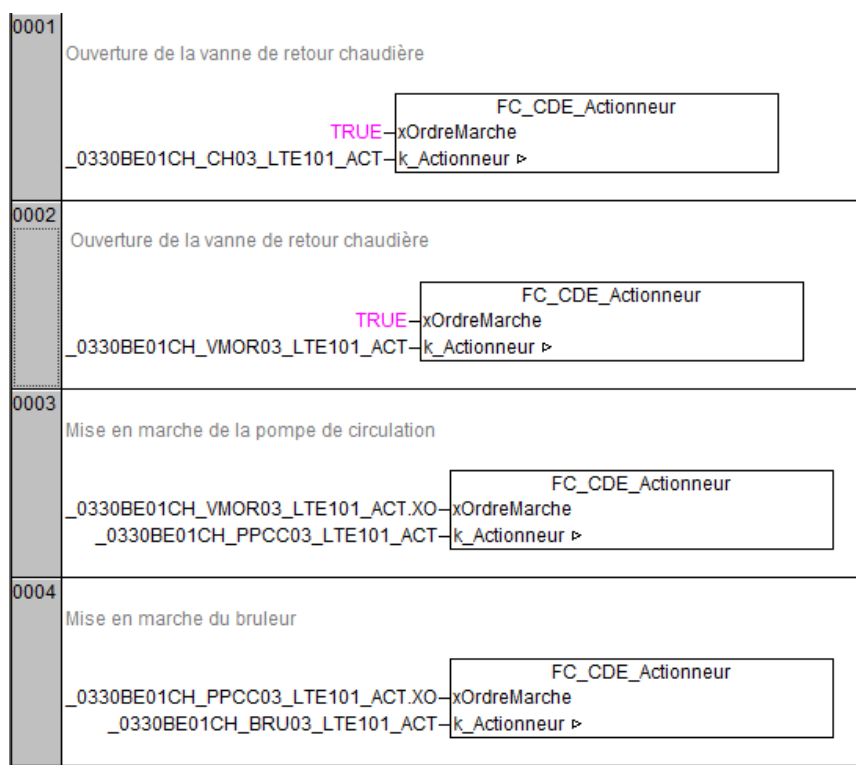
Exemple :

- Décompte d'un volume ou d'une temporisation
- Calcul consigne régulateur
- ...

5.11.3 Action « Ecriture_CDE_Actionneurs »

S'exécute en permanence lorsque l'étape est active.

Ici doivent être codées toutes les commandes des actionneurs utilisées par l'étape.



La commande des actionneurs doit être faite en utilisant le « FC_CDE_Actionneur ».

5.11.4 Action « Sortie_ »

S'exécute au dernier cycle d'exécution lorsque la transition de l'étape est active.

Ici doit être codée toute action d'initialisation de variable lors de la sortie d'étape

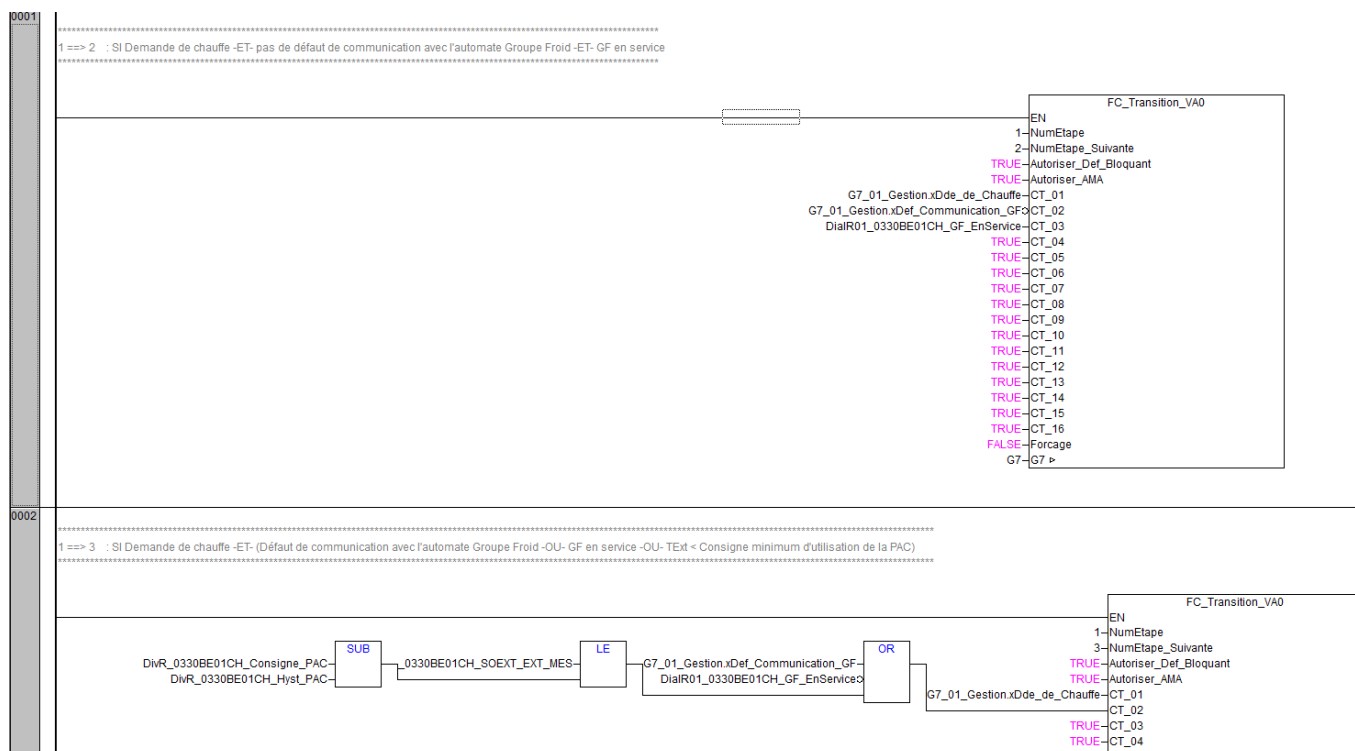
Exemple :

- RAZ d'une valeur pour un décompte
- Passage du mode Track à Auto pour un régulateur
- ...

5.11.5 Action « Transitions »

S'exécute en permanence lorsque l'étape est active.

Ici doivent être codées toute les transitions d'une étape en utilisant le bloc « FC_Transition_VA0 ».

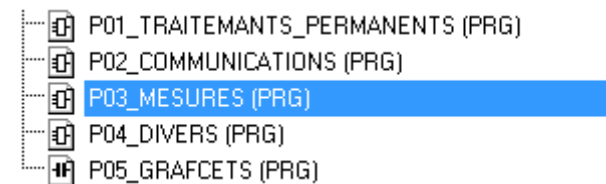


Description des entrées du bloc transition :

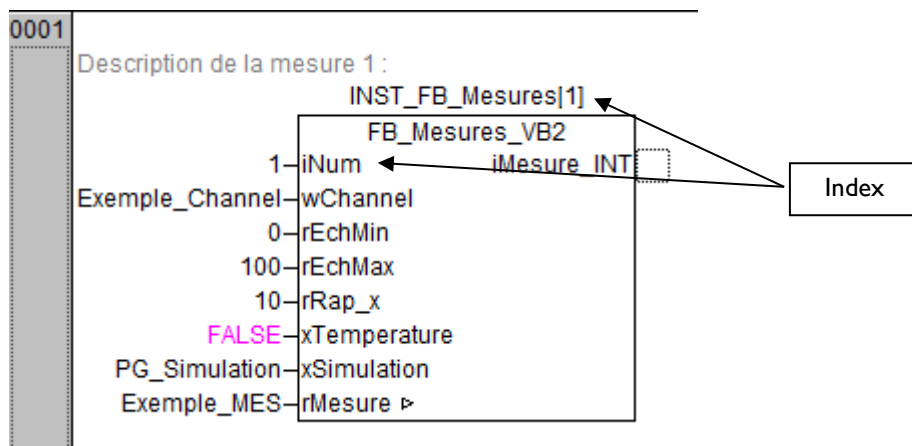
- NumEtap : Numéro de l'étape
- NumEtap_Suivante : Numéro de l'étape à activer lorsque la transition est active
- Autoriser_Def_Bloquant : TRUE = autorise la transition lorsque le grafcet est en défaut bloquant ; FALSE = Bloque la transition si le grafcet est en défaut bloquant.
- Autoriser_AMA : FALSE = Bloque la transition si le grafcet a au moins l'un de ses actionneur en manu.
- CT_01 à CT_16 : Conditions de transition. Pour que la transition soit active, il faut que toutes les entrées « CT_xx » soient vraies (TRUE). Les conditions en « OR » devront être programmées sur une entrée unique du bloc transition.
- Forçage : Force la transition même si toutes les CT_xx ne sont pas vraies.
- G7 : Toujours = G7. Structure générique contenant les informations du grafcet.

5.12 PROGRAMMATION DES MESURES

Les mesures sont programmées au niveau du programme principal :



Chaque mesure est programmée de la façon suivante :



- Une description doit être saisie.
- `INST_FB_Mesures[x]` avec x = index de la mesure.

Entrées :

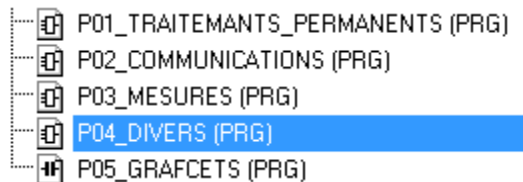
- `iNum` : Index de la mesure.
- `wChannel` : Entrée analogique (Entrée automate) Utiliser le Bloc « `FB_WORD_Lissage_Moy` » sur cette entrée si la mesure doit être moyennée
- `rEchMin` : Valeur d'échelle minimum de la mesure.
- `rEchMax` : Valeur d'échelle maximum de la mesure.
- `rRap_x` : Rapport de conversion pour la calcul de `iMesure_INT` ($iMesure_INT = Mesure * rRap_x$)
- `xTemperature` : TRUE si la mesure est une température.
- `xSimulation` : Permet de simuler la mesure (Par défaut mettre `PG_Simulation` qui permet de simuler l'automate en test plateforme)
- `rMesure` : Variable de la zone mémoire mesure. (Valeur remontée à la supervision)

Sortie :

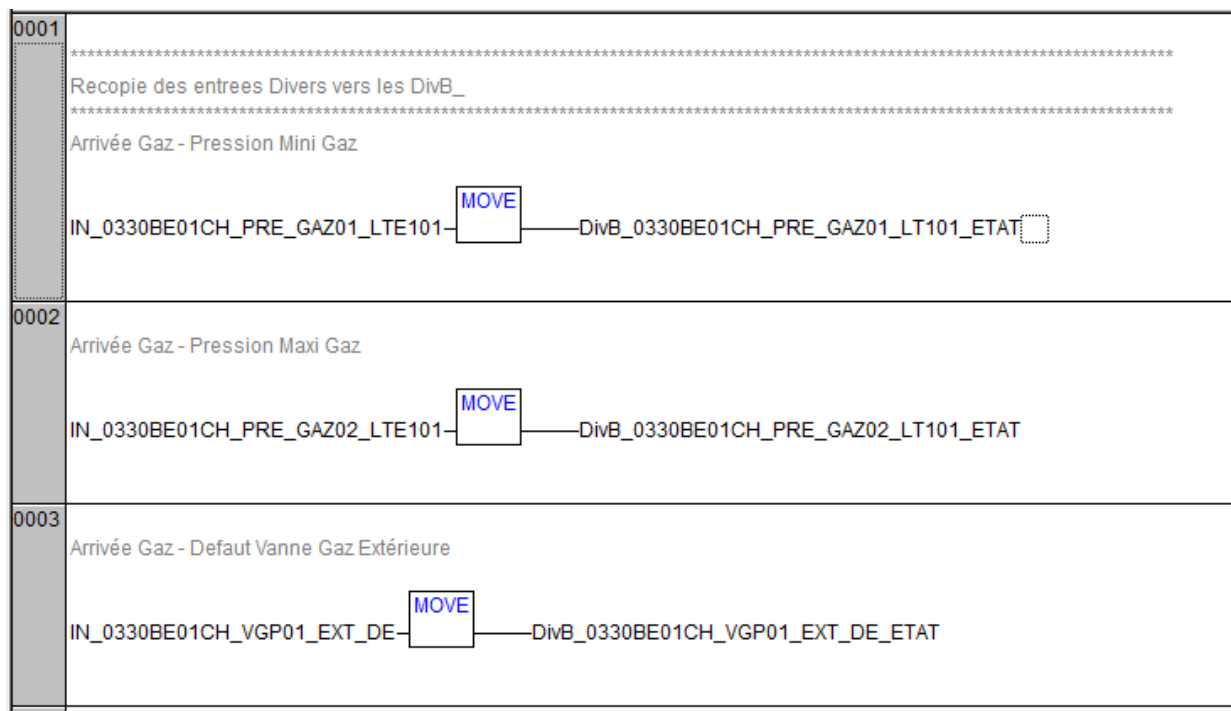
- `iMesure_INT` : Valeur de la mesure en Réel convertie en entier ($iMesure_INT = Mesure * rRap_x$). (*Facultatif*)

5.13 RECOPIE DES ENTREES AUTOMATE VERS LES VARIABLES SUPERVISION

Certaines entrées automate sont à recopier vers les divers bits à destination de la supervision. Cela doit être programmé au niveau du programme principal :



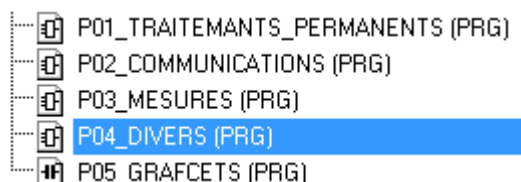
Example :



5.14 ALARME DEMANDE D'INTERVENTION

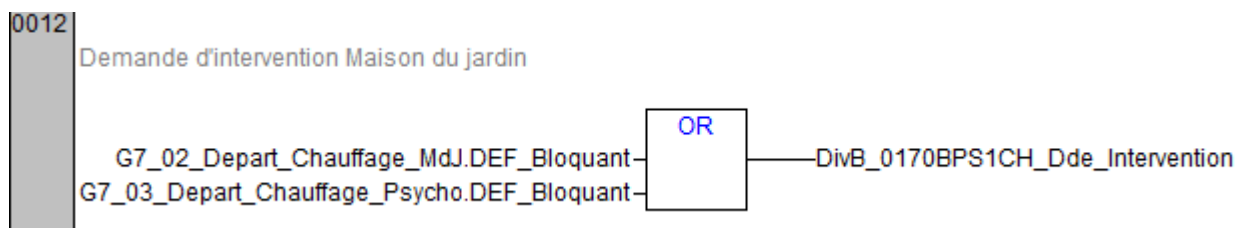
Sur chaque installation une alarme du type demande d'intervention doit être créer. Elle permet via la GTC d'envoyer un SMS aux personnels de maintenance lorsque l'installation ne rend plus le service pour lequel est faite. Exemple : CTA sans ventilation, un départ sur une sous-station de chauffage dont les deux pompes sont hors service, ect...

La demande d'intervention se programme généralement dans le P04_DIVERS :



IL est le plus souvent la synthèse des défauts bloquants des grafçets de l'automate.

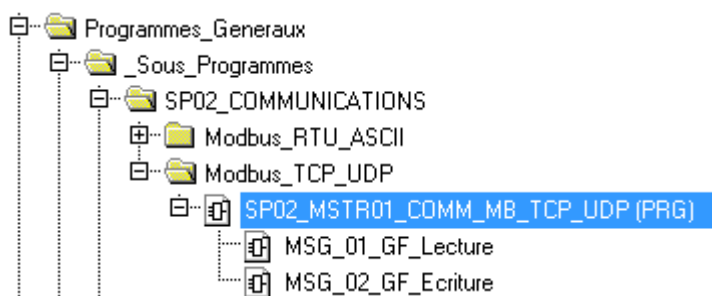
Example :



Cette variable se déclare au dans le fichier Excel découpage mémoire dans l'onglet « Divers », rubrique « Divers BITS » en utilisant le type « Alarme Dde Intervention ».

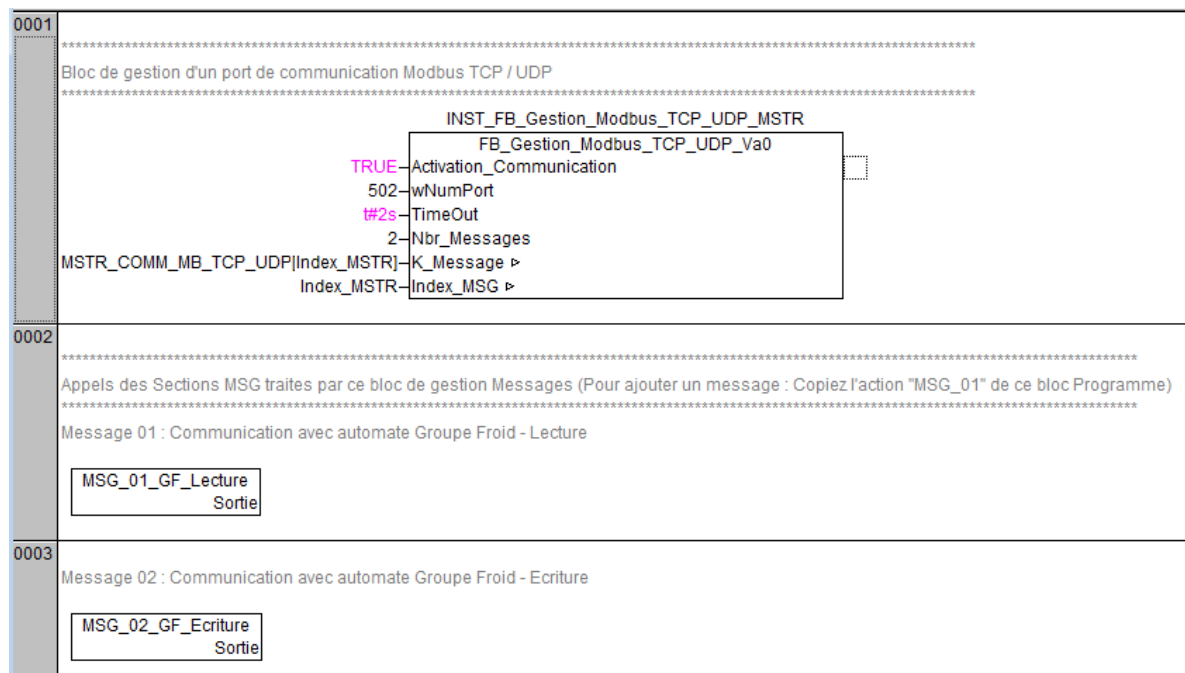
5.15 MISE EN PLACE D'UNE COMMUNICATION MODBUS TCP UDP

La communication Modbus TCP ou Modbus UDP est implémentée directement dans les **Programmes Généraux**.
Par défaut elle est inhibée :



5.15.1 Activation de la communication

Pour activer la communication Modbus TCP UDP, il suffit de passer l'entrée « Activation_Communication » à TRUE et de renseigner le bon nombre de messages (*Trames*).

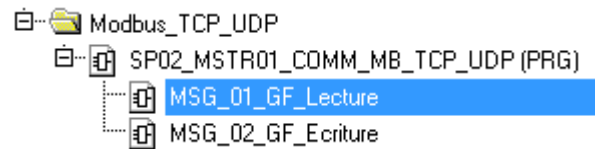


Les messages sont programmés dans les actions du PRG « SP02_MSTR01_COMM_MB_TCP_UDP ». De base 16 messages peuvent être traités par ce bloc. Les messages sont traités les uns après les autres et non en parallèle. Si plus de message devaient être traités La structure « MSTR_COMM_MB_TCP_UDP[16] peut être augmentée. Si des messages doivent être effectués en parallèle, un PRG « SP02_MSTR02_COMM_MB_TCP_UDP » devra être créé et appelé dans le PRG « P02 COMMUNICATIONS ».

Pour être actif chaque message doit être appelé dans le PRG « SP02_MSTR01_COMM_MB_TCP_UDP » (Voir réseau 2 et 3 ci-dessus).

5.15.2 Création d'un message

Les messages sont créés au travers des actions du PRG « SP02_MSTR01_COMM_MB_TCP_UDP ».

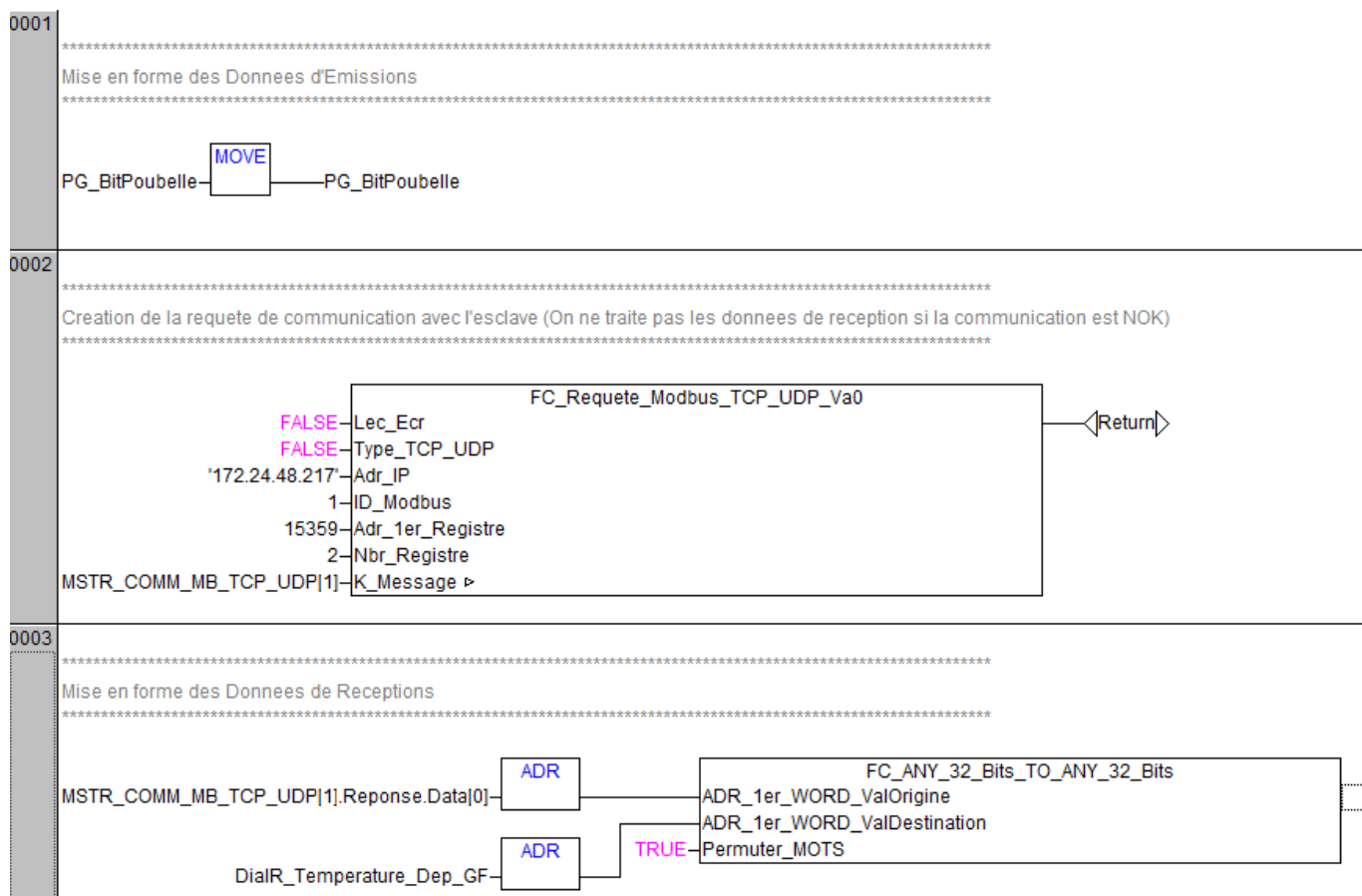


Pour ajouter un message :

- Clic droit sur un message existant
- Copier objet
- Donner le nom du message

La programmation d'un message se fait de la façon suivante :

- Pour un message en lecture :

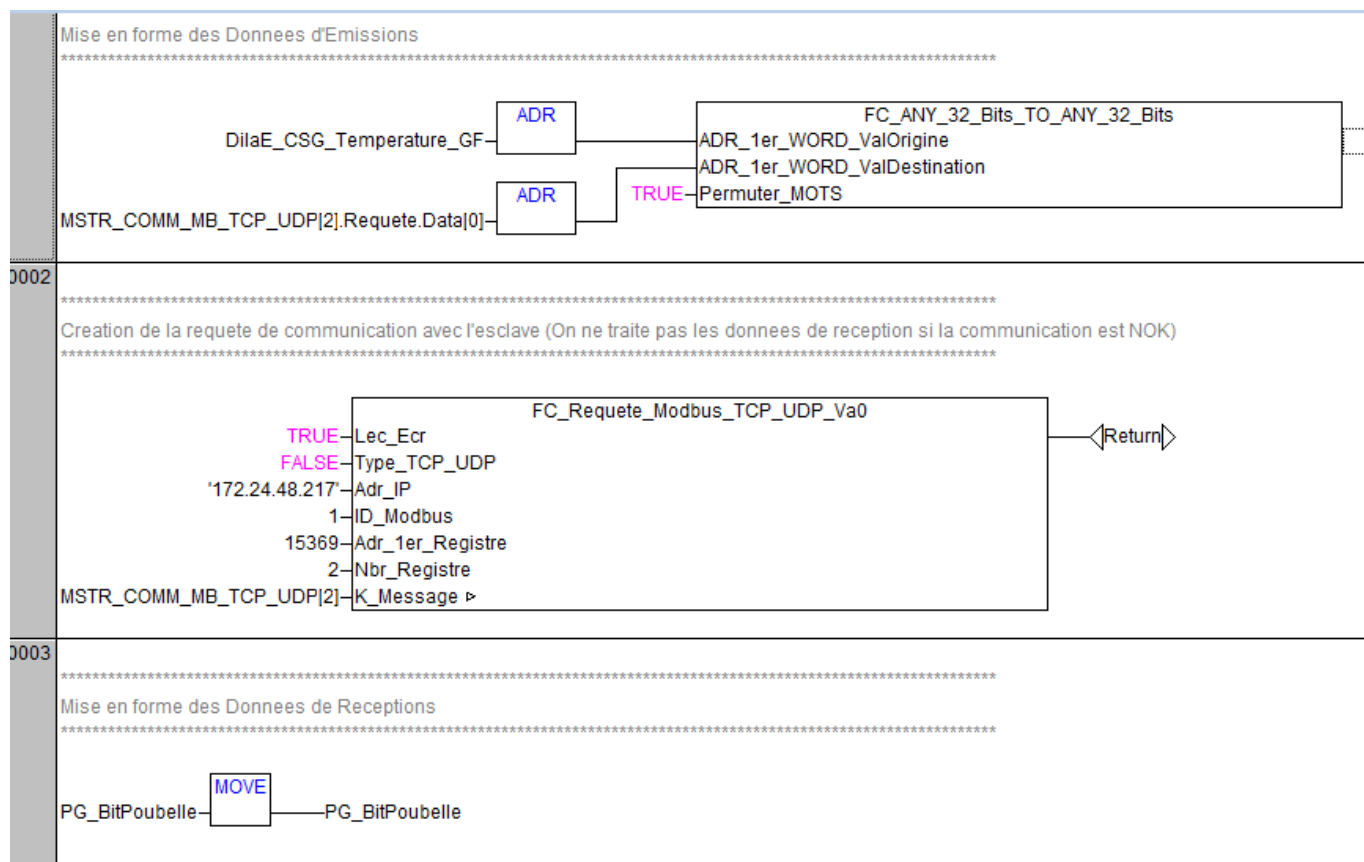


Les données lues (Word) se trouvent dans la structure MSTR_COMM_MB_TCP_UDP.Reponse.Data[0-124]

Lors de la lecture de variables 32 bits, les FC « FC_ANY_32_Bits_TO_ANY_32_Bits » doivent être utilisés pour la reconstruction de la valeur. Ce FC permet également la permutation des deux mots en cas d'inversion entre le mot de poids fort et le mot de poids faible.

Document de spécification et de Conception du Système (DSCS)

- Pour un message en écriture :

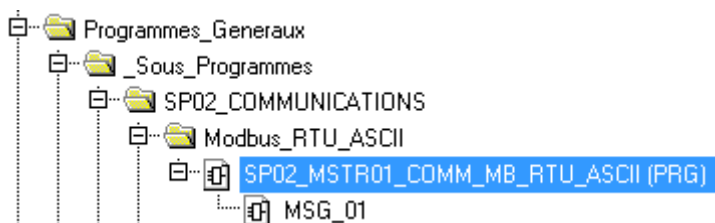


Les données à écrire (Word) doivent être mises dans la structure MSTR_COMM_MB_TCP_UDP.Requete.Data[0-124]

Lors de l'écriture de variables 32 bits les FC « FC_ANY_32_Bits_TO_ANY_32_Bits » doivent être utilisés pour le découpage de la variable 32 bits en deux mots de 16 bits dans la structure Data[]. Ce FC permet également la permutation des deux mots en cas d'inversion entre le mot de poids fort et le mot de poids faible.

5.16 MISE EN PLACE D'UNE COMMUNICATION MODBUS RTU ASCII

La communication Modbus RTU ou Modbus ASCII est implémentée directement dans les Programmes Généraux.
Par défaut elle est inhibée :



Son fonctionnement est identique à la communication Modbus TCP_UDP. (Voir ce chapitre pour plus d'informations)

Seul le paramétrage de la carte RS485 s'ajoute.

