

Gaël Maiano

Rapport de stage :

Développement d'une

Supervision

a

UIMM

LA FABRIQUE
DE L'AVENIR

Table des matières

Introduction	3
II - La démarche de développement de l'application	10
<u>II.1 Etablir d'abord la communication entre les six postes (dont celui de supervision)</u>	11
<u>II.2 – Etablir ensuite les tables de données dans les programmes des API et dans l'IHM de supervision.</u>	14
<u>II.3 - Créer les écrans et les programmes nécessaires à leur utilisation, au sein des API ou de l'IHM.</u>	15
III Les écrans	15
<u>III.1 L'écran Etat des postes</u>	16
<u>III.1.2 Champ « Etats Palettisation »</u>	16
<u>III.2 L'écran des alarmes</u>	23
<u>III.2.1 Alarmes analogiques</u>	23
<u>III.2.2 Alarmes de bit</u>	24
<u>III.3 Ecran Gestion des utilisateurs</u>	29
<u>III.4 Ecran de Courbes</u>	31
<u>III.4.1 Courbe « Stock Mélange »</u>	31
<u>III.4.2 Courbe du réservoir d'injection du poste Remplissage</u>	33
<u>III. 5 Ecrans de recettes</u>	35
<u>III.5.1 Première recette Mélange 2 en « INT ».</u>	36
<u>III.5.2 Deuxième recette Mélange 2 en « Real », « Int » et « Time »</u>	37
Conclusion	40
Annexe 1	42
<u>A.1.1 Champ « Etats Mélange »</u>	42
<u>A 1.2 « Etats Remplissage »</u>	44
<u>A 1.3 « Etats Bouchage »</u>	46
<u>A 1.4 « Etats Etiquetage »</u>	49
Annexe 2	52
<u>A 2.2 Alarmes de bits du Poste Etiquetage</u>	57

Remerciements

Je tiens à remercier, à l'occasion de ce rapport, la Région Bretagne, sans laquelle je n'aurais pu participer ni à cette formation, ni à ce stage.

Je remercie Monsieur Laurent GRAVOUIL, qui, dans une situation sanitaire très particulière, a permis l'organisation de mon stage au sein du Pôle Formation de l'UIMM de Lorient qu'il dirige, et tous les formateurs de ce pôle qui ont contribué à ma formation à Lorient.

Je remercie aussi le Pôle Formation de Quimper qui m'a accueilli durant ce stage à l'occasion d'un rendez-vous de suivi de projet avec Monsieur Le Poulichet.

Et je le remercie tout particulièrement, lui, mon formateur, Monsieur Le Poulichet, pour l'aide donnée durant le stage et toute la formation.

Et bien sûr, je remercie aussi mes camarades de formation, pour leur soutien et leur gentillesse tout au long du stage.

Introduction

Dans un cadre de restrictions dues à l'épidémie de la COVID, j'ai effectué mon stage au CFAI de Bretagne, au sein du Pôle Formation de l'UIMM de Lorient du 15 février au 2 avril 2021.

1 L'entreprise

Pour se développer et permettre l'innovation, les entreprises ont constamment besoin de nouvelles compétences. Pour répondre à ce besoin, trois organismes de formations : AFPI Bretagne, DIAFOR Organisation, et CFAI Bretagne se sont associés pour former le Pôle Formation de l'UIMM Bretagne. Les entreprises industrielles y dispose des ressources dont elles ont besoin pour adapter la mise en œuvre de leur stratégie, améliorer leur compétitivité, mieux anticiper l'évolution des métiers et être accompagnées dans l'évaluation, l'acquisition et l'adaptation des compétences de leurs salariés.

Mais le Pôle Formation de l'UIMM de Bretagne, c'est aussi pour les territoires, des moyens de communication vers leurs interlocuteurs mis à leur disposition, des possibilités d'emploi pour les apprentis, les demandeurs d'emploi, et les salariés en leur offrant de nouvelles qualification professionnelles.

L'UIMM Bretagne est un acteur reconnu qui contribue à l'attractivité de la branche industrielle. C'est :

en formation continue

7 000 stagiaires

400 000 heures de formations

+ de 2000 groupes en formation continue par an sur 6 sites de formation

en alternance :

1 700 alternants

33 diplômes de Bac pro, BTS, DUT, Licence et ingénieur

13 sites de formation

2 L'organisation du stage

Dans la situation sanitaire très particulière que nous avons connu tout au long de cette formation, c'est encore le Pôle Formation de l'UIMM de Lorient et Monsieur Laurent GRAVOUIL son directeur, qui ont permis l'existence et l'organisation de mon stage et c'est lui qui a endossé les fonctions de tuteur et maître de stage.

Monsieur Sébastien LE POULICHEZ, formateur référent, a assuré l'encadrement

Des rendez-vous de suivi du stage, réalisé largement en autonomie, ont eu lieu en semaine 3 et en semaine 5 et 6.

Comme matériel, je disposais d'un ordinateur sur lequel était installé TIA portal ET WinCC RT Advanced.

3 Le cahier des charges

Durant de notre formation à Lorient, nous avons étudié le fonctionnement d'une ligne de production qui, en cinq postes, constitue un mélange liquide dont sont remplis des bidons de 2, 3 ou 5 litres, qui sont ensuite bouchés, étiquetés et mis en palettes de six, neuf ou douze bidons.

Nous disposions d'un outil pédagogique appelé « Ligne Cyber » qui attribuait deux valises à chacun des cinq postes (API et IHM). Dans l'une des valises il y a avait un automate Siemens, un IHM Siemens, un automate Schneider, dans l'autre valise un Raspberry et un écran PC.

Le Raspberry et l'écran servent à afficher une animation en Flash qui simule le comportement des éléments de la ligne. Les fonctionnements dépendent du programme inscrit dans l'automate Siemens. L'automate Schneider est redondant à l'automate Siemens mais la présence des deux types d'automate permet de passer de l'un à l'autre lors des démonstrations pédagogiques.

La programmation de quatre des cinq postes (Remplissage, Bouchage, Etiquetage et Palettisation), initialement majoritairement en Grafset, a fait l'objet d'un TP : par groupe de deux ou trois stagiaires, une nouvelle programmation des API en Ladder ou en SCL puis des IHM en Win CC nous était demandée.

Le cahier des charges de mon stage de printemps était de développer une supervision sur le résultat de cet exercice, programmes API et IHM, dans leur état d'avancement. Le cinquième poste (Mélange), programmé par les formateurs, a été ajouté le 9 mars pour la cohérence du projet avec une ligne complète.

4 La documentation

La documentation était constituée principalement par celle accompagnant la « Ligne Cyber ».

Pour compléter mes connaissances, j'ai utilisé des ressources « Internet », en particulier:

- en anglais

20 vidéos sur l'utilisation de WinCC dans les IHM Siemens de Kelly Anton

https://www.youtube.com/watch?v=vUb_coL1yy0&list=PLXgzIIGmsGirF1sZp0VojRPreub1E9ckW

ainsi que les vidéos sur les alarmes et les recettes de « Hegamurl » :

https://www.youtube.com/watch?v=nMGtGfOX_2U&list=RDCMUC1P4ACs0hsr7AWcl-mWKQbQ&index=1

et en français de Hamadi ben Sassi :

<https://www.youtube.com/watch?v=iv8ReT0Wnxg&list=RDCMUC1P4ACs0hsr7AWcl-mWKQbQ&index=2>

5 Le candidat

Gaël MAIANO 07 62 21 13 49 gaelmaiano@gmail.com

Diplômes

- Bac S

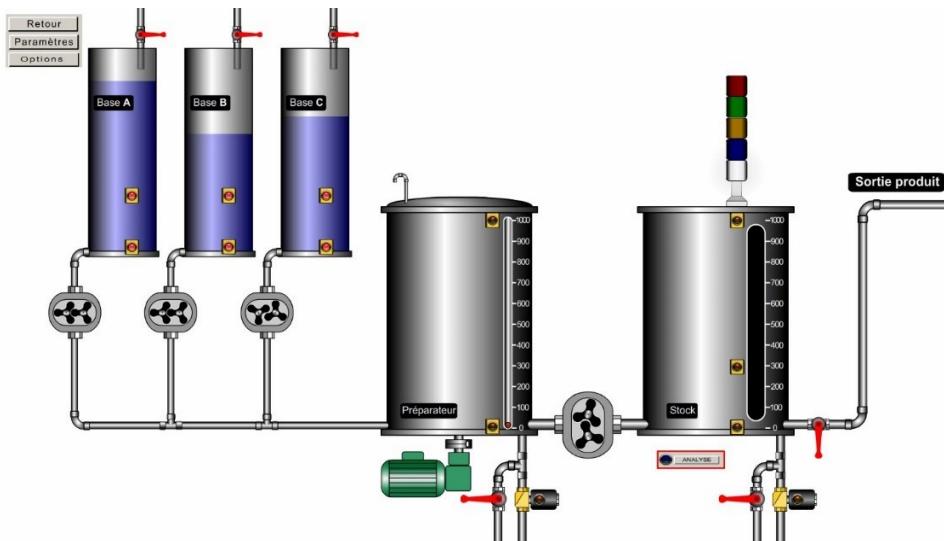
- Master en Géographie de l'Université de Nantes (M2 Sociétés Aménagement Gouvernance des territoires)

- BTS SIO SISR en un an obtenu au Greta de Morlaix (informatique des système et réseaux).

I - Présentation de la Ligne de Production Virtuelle

Le développement de la supervision était basée sur la Ligne de Production Virtuelle, composée de cinq postes dont le but est de produire un mélange, le mettre en bidon et ranger ces bidons sur des palettes.

I.1 Le poste mélange



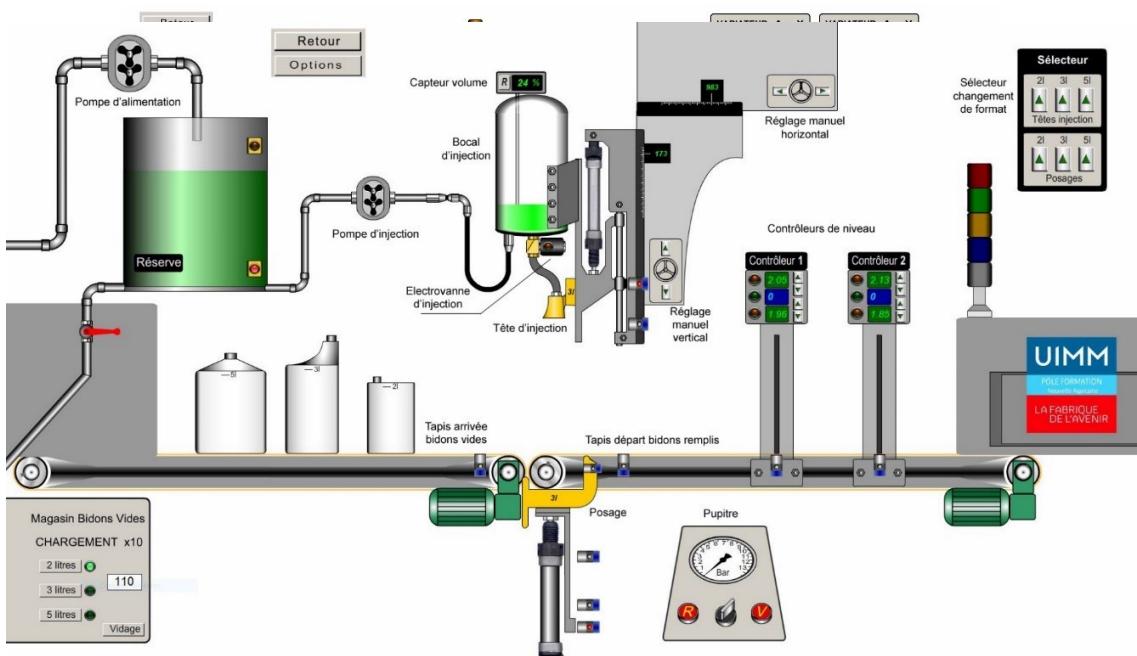
Le poste de mélange se compose de trois réservoirs de produits de base A, B et C, de trois pompes volumétriques identiques aspirant dans chacun des réservoirs A, B et C et refoulant dans un réservoir préparateur de 1000l. Un agitateur est fixé en position basse du préparateur.

Une pompe de transfert volumétrique de grand débit aspire le contenu du préparateur pour le refouler dans un réservoir de stock de 1000l également.

Les réservoirs préparateur et stock possèdent une vanne manuelle et une électrovanne de purge. Trois vannes manuelles permettent le réapprovisionnement des produits de base.

Une vanne est placée sur le tuyau de sortie produit vers la machine de remplissage.

I.2 Le poste remplissage



Le poste de remplissage remplit des bidons deux litres, trois litres et cinq litres. Ces bidons sont approvisionnés, par 10, de la capacité voulue, dans le « Magasin de Bidons Vides ».

Les bidons sont amenés un par un par le tapis de gauche et placé dans son posage.

Le bidon est alors levé avec le vérin vertical et son posage en même temps que la tête d'injection descend.

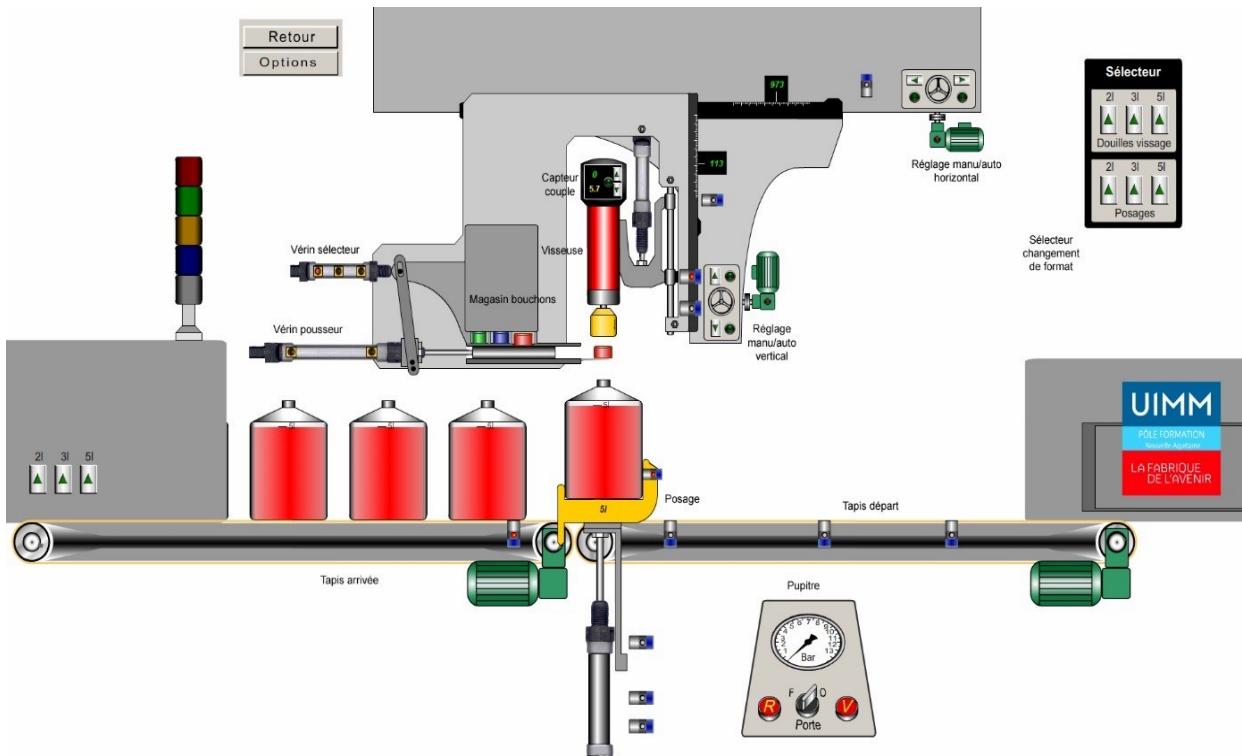
La réserve est alimentée en produit par la pompe d'alimentation. La pompe d'injection charge le bocal d'injection.

Une fois le remplissage effectué, la tête d'injection remonte, le posage descend et le bidon repose sur le tapis de départ qui déplace le bidon vers la sortie.

Pour le changement de format de bidon, il faut arrêter la machine pour choisir le type de posage et de tête d'injection.

Un pupitre permet d'ouvrir la porte, de vider totalement la machine (« V ») et de reseter le système en cas de collision (« R »).

I.3 Le poste Bouchage



Le poste de bouchage peut visser trois types de bouchons en fonction du format des bidons. Ces bouchons sont présents dans un magasin vertical à trois colonnes. Le réapprovisionnement est transparent et automatique. Un vérin sélecteur déplace un système à guillotine ouvrant une des trois colonnes du magasin. Un vérin pousseur place le bouchon juste en dessous la douille de vissage.

Les bidons sont amenés un par un par le tapis de gauche jusqu'au tapis suivant. Grâce à ce dernier le bidon est placé dans son posage.

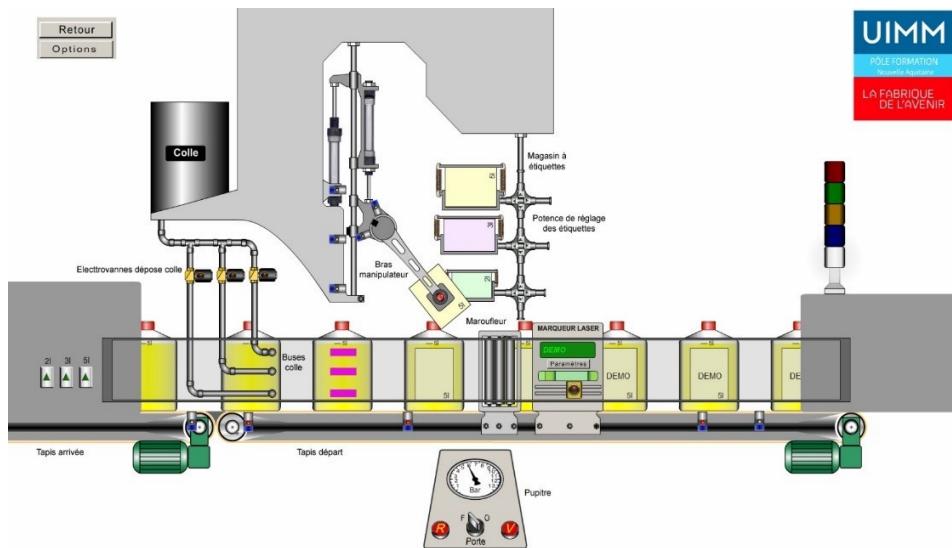
Le bidon est alors levé avec le vérin vertical et son posage. Puis le second vérin descend la visseuse en marche. la douille de vissage visse le bouchon et le couple de serrage

augmente jusqu'à la valeur souhaitée. A ce moment-là, la visseuse remonte et le bidon redescend pour être acheminé par le tapis de départ.

Pour le changement de format il faut arrêter la machine ouvrir la porte, ce qui donne accès au menu « Sélecteur » afin de choisir le type de posage et de douille de vissage.

Le système de réglage en vertical et horizontal de la machine est manuel et/ou automatique. En automatique deux moteurs électriques avec comptage d'impulsions peuvent positionner la visseuse. Une prise d'origine est cependant nécessaire au démarrage de la machine.

I.4 Le poste Etiquetage



Le poste de marquage dépose de la colle sur les bidons, pose la bonne étiquette, maroufle puis marque avec un laser un texte sur l'étiquette.

Les bidons sont amenés un par un par le tapis de gauche jusqu'au tapis suivant. Ce second tapis moins rapide ne s'arrête pas et les quatre opérations se font sans blocage des bidons.

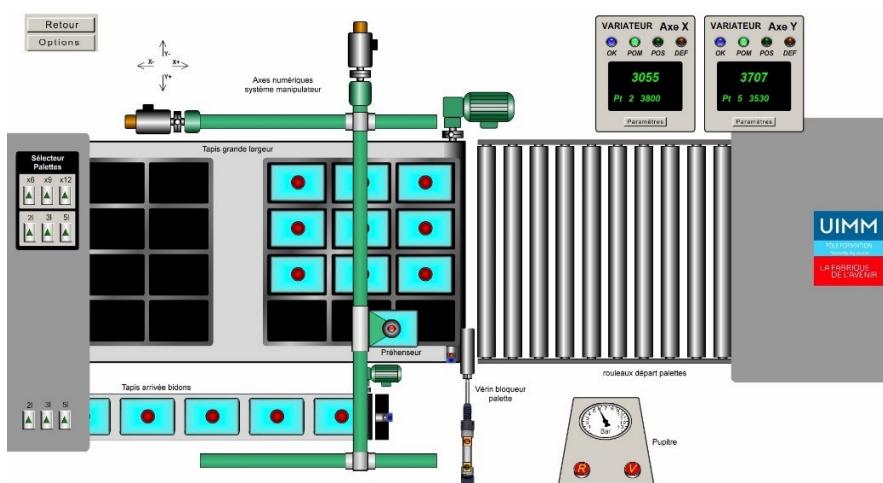
La réserve de colle est permanente. Trois électrovannes en s'ouvrant font déposer de la colle par les trois buses sur les bidons au format 3l et 5l. Pour les bidons de 2l, seules les deux buses inférieures doivent être alimentées.

Le bras manipulateur saisit la bonne étiquette dans le magasin puis la dépose sur le bidon.

Le maroufleur à rouleaux colle l'étiquette au passage du bidon.

Le marqueur laser inscrit sur l'étiquette le texte sélectionné. Huit textes différents peuvent être inscrits. Le nombre de caractères par texte est limité à sept.

I.5 Le poste Palettisation



Le poste de palettisation prend les bidons un par un sur le tapis d'arrivée grâce au système manipulateur et les dépose dans les logements des palettes.

Les palettes sont en réapprovisionnement permanent. Un sélecteur de palettes accessible à la souris permet de choisir le type de palettes. Il existe trois tailles de palettes par type de bidons : palette de 6, de 9 et 12 emplacements.

La palette pour être chargée doit être en butée sur le vérin bloqueur.

Les variateurs donnent à tout moment la position et l'état de l'axe numérique correspondant. Un bouton « paramètres » permet de régler les différents points de prise et dépose.

II - La démarche de développement de l'application

La supervision consiste à **contrôler** et à **piloter** en **temps réel** le processus de fabrication automatisé à partir d'un poste central. Elle permet de visualiser la bonne marche de l'installation et d'être alerté immédiatement en cas de défaut ou d'alarme.

Deux grandes étapes sont nécessaires pour développer une application de supervision :

- La configuration de données
- L'édition des écrans et l'écriture des programmes complémentaires.

Pour développer mon application, j'ai :

établi la communication entre les cinq postes et celui de supervision,
formé les tables de données dans les programmes des API et dans l'IHM de supervision,
créé les écrans et les programmes nécessaires à leur utilisation, au sein des API ou de l'IHM.

Parallèlement, j'ai recherché de la documentation, hormis celle spécifique à la ligne Cyber

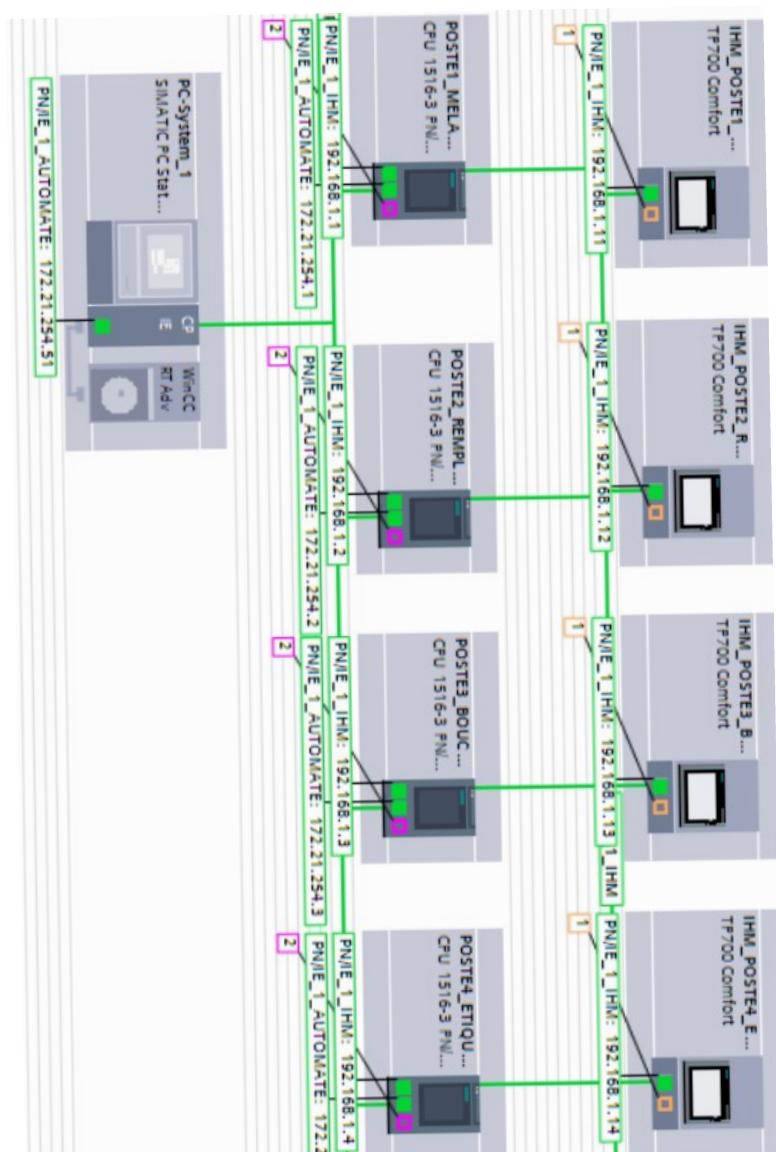
II.1 Etablir d'abord la communication entre les six postes (dont celui de supervision)

Le premier travail est d'établir des réseaux de communication entre les cinq API, les cinq IHM Opérateur et la station de supervision. Apparemment simple, cette communication n'a fonctionné correctement qu'à la fin du stage, malgré l'aide apportée par M. LE POULICHEZ en semaine 3 pour reconfigurer les réseaux.

Finalement, j'ai supprimé la liaison entre le PC et les PLC et entre le PC et les HMI puis réinstallé une nouvelle station PC appelé PC system-1. J'ai donc reprogrammé sur ce nouveau poste tout le travail déjà effectué.

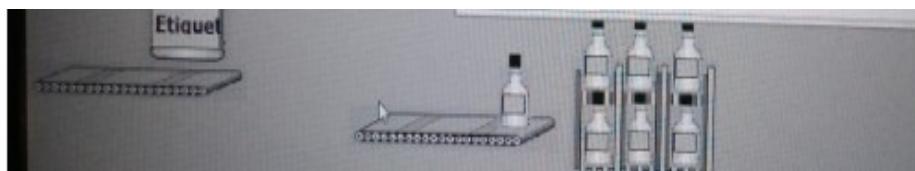
Le réseau est constitué en Profinet et la liaison PLC/HMI est précisée dans TIA en utilisant le menu Appareils & Réseaux / onglet Liaisons en tant que « liaison IHM ». Deux sous-réseaux PNIE sont créés, un pour les IHM et un pour les API. Les deux sous-réseaux sont sur le masque 255.255 .255.0

Le réseau en Profinet



Une fois les réseaux de nouveau installés, j'ai d'abord testé la communication entre l'IHM de supervision et chacun des API :

Pour cela, j'ai ouvert une vue dans l'IHM et j'y ai placé 5 champs E/S correspondant à chacun des postes. Dans chacun des API, j'ai créé un DB de test qui contient une variable test de type INT. Chaque champ de test de la vue IHM est lié à chacune de ces variables.



...CyberV7 > POSTE2_REMPLISSAGE [CPU 1516-3 PN/DP] > Blocs de programme > db_test_remplissage [DB1]

	Nom	Type de données	Valeur de départ	Valeur de visualisati.	Rémanence	Accessible ...	Ecritu...	Visible da...
1	Static				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2	test	Int	0	1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

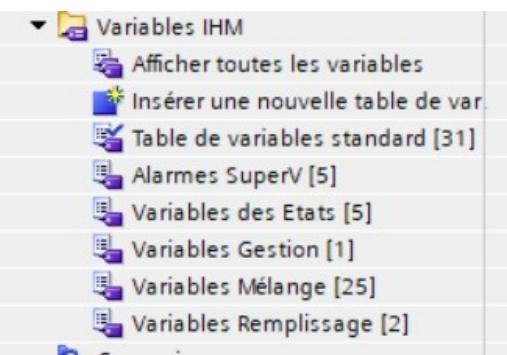
La simulation permet de vérifier que la communication est bien établie : seuls les deux premiers postes sont connectés ici avec les variables Test des DB d'abord à 0 puis forcées à 1 parce que TIA ne permet pas de lancer les cinq simulations simultanément.



II.2 – Etablir ensuite les tables de données dans les programmes des API et dans l'IHM de supervision.

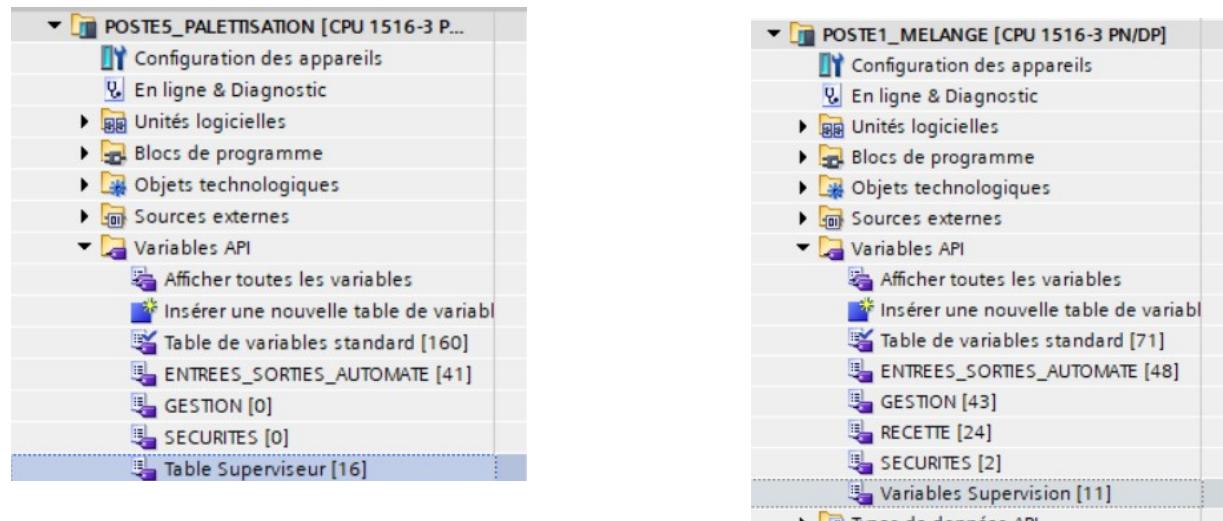
La recherche de variables dans complexe selon les postes et variables répondaient parfois aux ETATS_EN_COURS du poste Méthode pour y répondre par l'établissement d'exemple absence de variables d'

Ce travail, supposé préalable, a



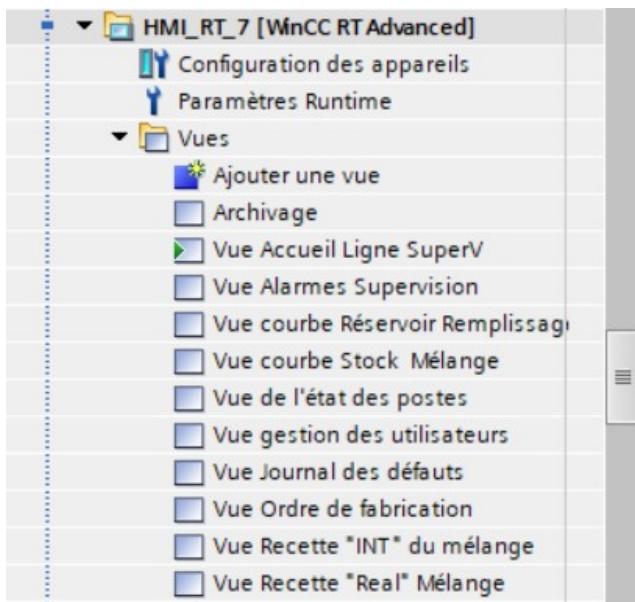
é plus ou moins configurées dans la supervision. Ces variables peuvent être créées ou configurées à l'aide de la copie, ... (par les API).

Les tables ne sont pas détaillées dans ce compte rendu pour présenter les variables. Elles sont souvent visibles dans les sous programmes présentés. J'ai créé des tables de variables dans les API et dans l'IHM.



II.3 - Créer les écrans et les programmes nécessaires à leur utilisation, au sein des API ou de l'IHM.

Ces écrans sont constitués à partir de vues créées dans l'IHM. Leur arborescence est structurée par leurs fonctions (Accueil, Etats, recettes...). Certains portent sur les cinq postes (Etats par exemple) mais d'autres n'ont été développés que pour un seul poste (recettes, courbes...).



Les écrans font l'objet d'une présentation détaillée qui suit.

III Les écrans

J'ai constitué les écrans de supervision :

Accueil : rappelle schématiquement la ligne de production et permet la navigation d'un écran à l'autre

- Etats des postes : états en temps réel de chacun des cinq postes (production, arrêt, maintenance ...).

Alarmes : alarmes en temps réel transmises par chacun des cinq postes.

Gestion des utilisateurs (personnels habilités, mot de passe, autorisation).

Courbes, deux écrans dont un présentant deux courbes.

Recettes : deux vues de recettes, l'une constituée à partir d'une banque de donnée existante dans le programme Mélange, l'autre constituant son propre DB, plus réaliste.

III.1 L'écran Etat des postes

III.1.1 Présentation de la démarche de travail

Cet écran reporte la situation des marches/arrêts, maintenance, fonctionnement en automatique.... de chacun des cinq postes.

Il est présenté en premier car il a représenté le plus gros travail : la programmation de chacun des cinq API ne comportait pas toujours de variables reportant cette situation et aucune liste d'état commune n'existe, probablement parce que la conception d'une supervision n'avait pas été prévue lors de l'établissement du projet de la ligne de production.

Pour que cet écran de contrôle ait un sens, il fallait au moins établir une liste pour chacun des postes.

La démarche conduite est détaillée pour le poste Palettisation et pour éviter les répétitions la présentation de quatre autres postes est renvoyée en annexe 1 mais peut être traitée à la demande.

La démarche consiste à :

rechercher les éléments dans les programmes des API et de l'IHM et voir la cohérence des deux programmations,

établir des sous-programmes de transfert des variables TOR représentant les différents états dans une variable INT, utilisée par l'IHM comme variable de report

créer dans l'IHM, l'affichage dans un champ de la vue

s'assurer du bon résultat par la simulation.

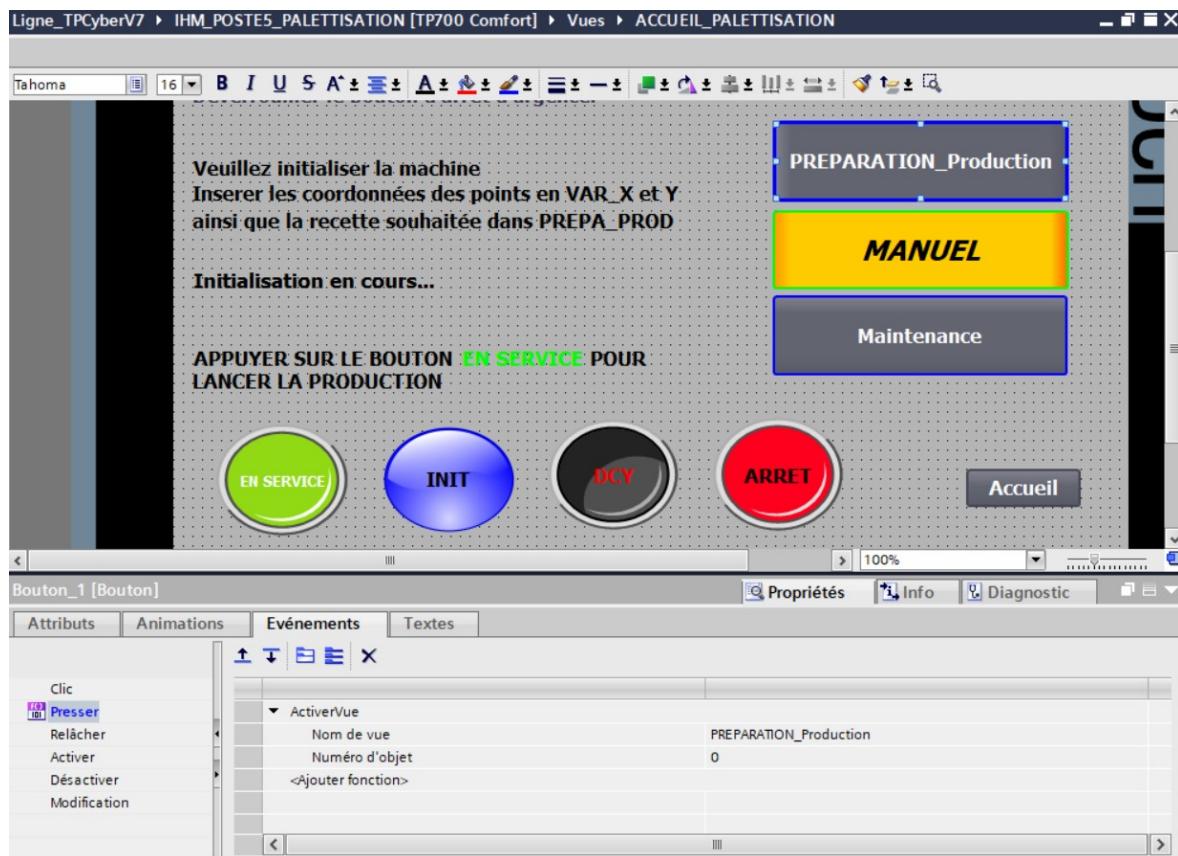
III.1.2 Champ « Etats Palettisation »

Le poste Palettisation est le cinquième poste mais il est présenté dès à présent parce qu'il a demandé un travail assez complet.

III.1.2.1 Recherche d'éléments

L'IHM du poste Palettisation présente une situation partielle de ce qui pourrait être l'état du poste (boutons Préparation production, Maintenance, Manuel). Mais seul le bouton Préparation production est lié à une variable de l'IHM.

Les boutons En_SERVICE, INIT , DCY et ARRET sont eux aussi liés à des variables de l'IHM et ces variables mènent à quelques variables TOR de l'API (M_BP_INIT en %M7.0, IHM_BP_INIT en %M160.0 et BP_INIT en %I3.2, BP_ARRET en %I3.4).



Ces variables sont utilisées par 2 blocs de l'API : FB411 (mise en marche) et FB 408 (gestion des voyants) qui ne décrivent que partiellement l'état du poste.

L'outil recherche de TIA portal n'apporte pas d'information supplémentaire.

Enfin, dans l'IHM Opérateur, des listes existent mais elles ne sont pas utilisables comme liste d'états du poste.

Listes de textes			
... Nom	Sélection	Commentaire	
CHOIX_LITRES	Plage (... - ...)		
CHOIX_PALETTIES	Plage (... - ...)		
Defauts production	Plage (... - ...)		
Etat Cuve	Plage (... - ...)		
Etat Machine	Plage (... - ...)		
Etat Pressostat	Plage (... - ...)		
Etat production	Plage (... - ...) ▾		
Liste de textes_PRODUCTION	Numéro de bit (0 - 31)		
MESSAGE ARRET URGENCE	Bit (0, 1)	ETAT ARRET D'URGENCE	
Noms_Vues	Numéro de bit (0 - 31)		
<ajouter>			

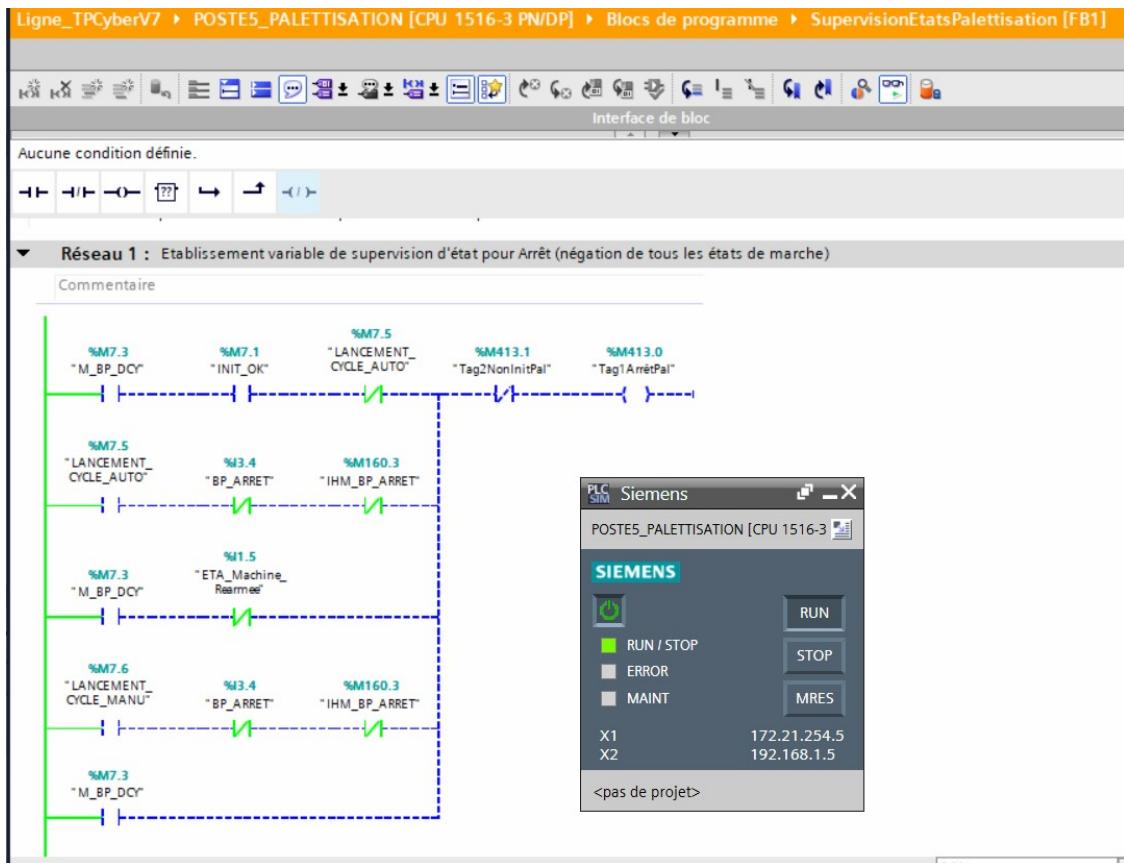
Entrées dans la liste de textes		
... Par dé..	Valeur	Texte
0	Arrêt	
1	Production en cours	
2	Arrêt demandé, fin du cycle...	
3	Arrêt obtenu. Appui sur DCY pour reprise.	
4	Arrêt de production en cours...	
5		
6		
7		
8		
<ajouter>		

III.1.2.2 Etablissement dans l'API du sous-programme de transfert

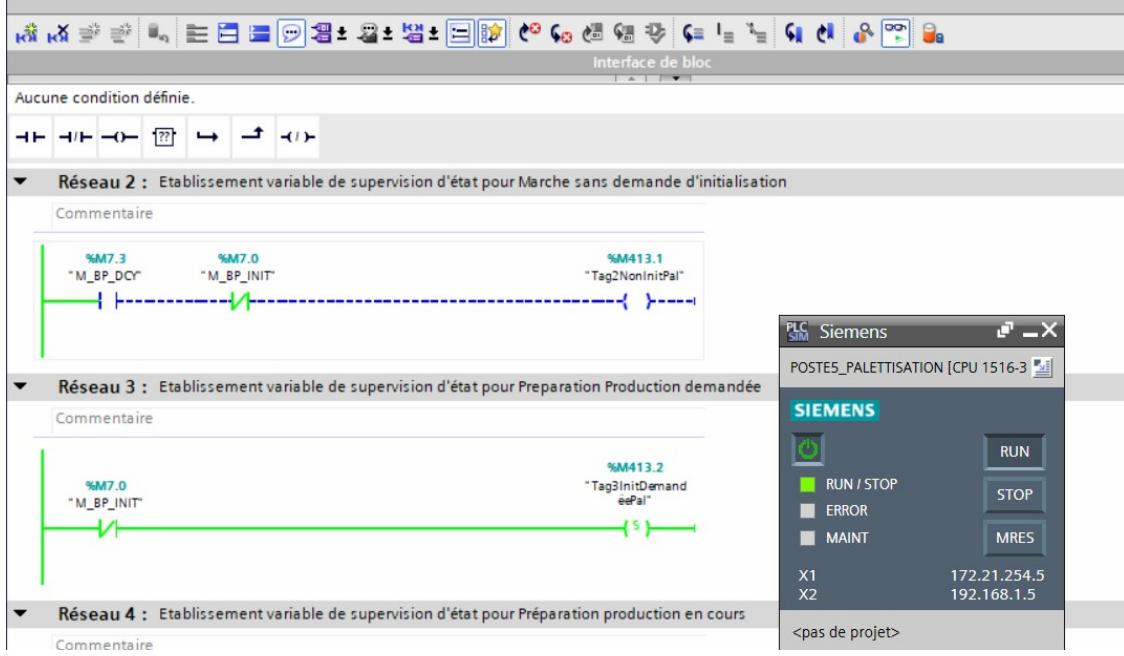
J'ai donc créé le sous-programme FB[1] SupervisionEtatsPalletisation dans le programme de l'API qui définit et place les états sur des variables TOR.

Ces variables sont adressées sur un mot que l'on sait libre grâce à la table d'affectation, ici sur l'octet MB413 du MW412.

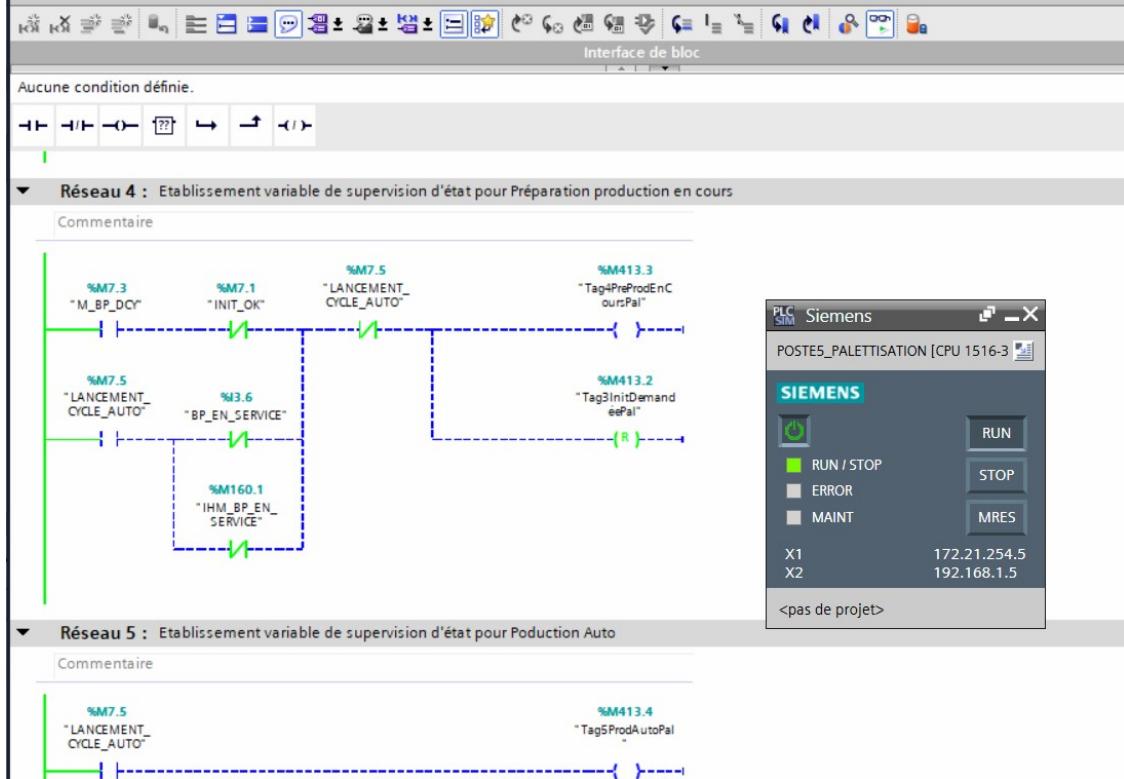
Quelques uns de ses réseaux sont présentés ici lors d'une simulation où la variable "Tag3InitDemandéePal" est forcée à 1 par l'inversion de l'entrée du réseau (M_BP_INIT en contact non passant).



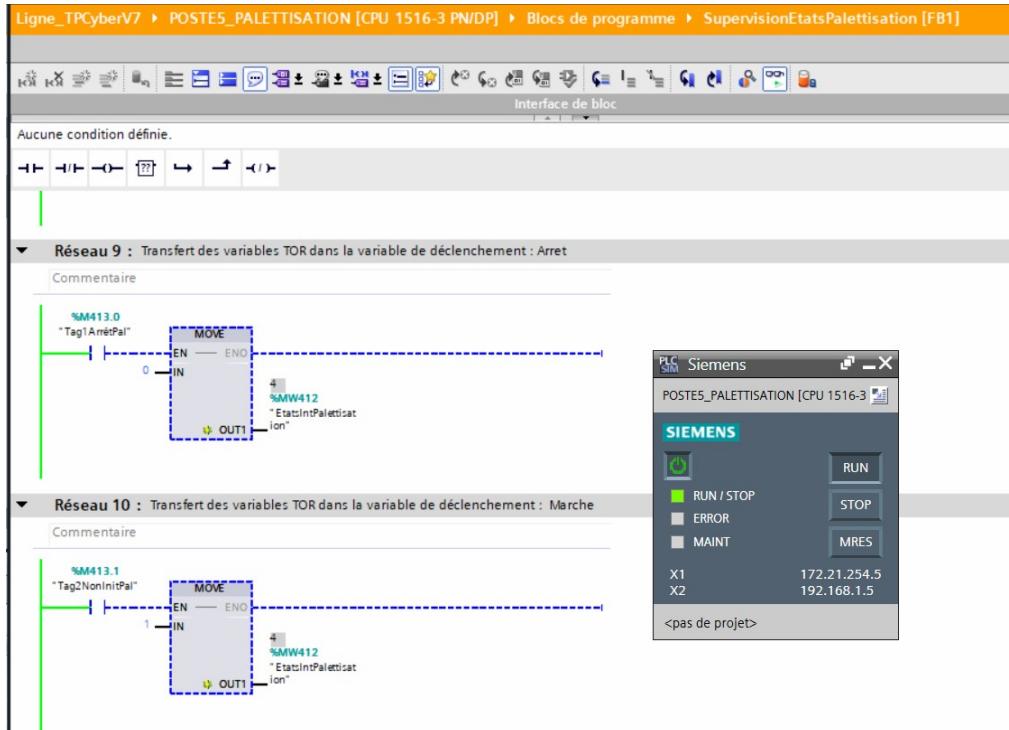
Ligne_TPCyberV7 > POSTES_PALETTISATION [CPU 1516-3 PN/DP] > Blocs de programme > SupervisionEtatsPalettisation [FB1]

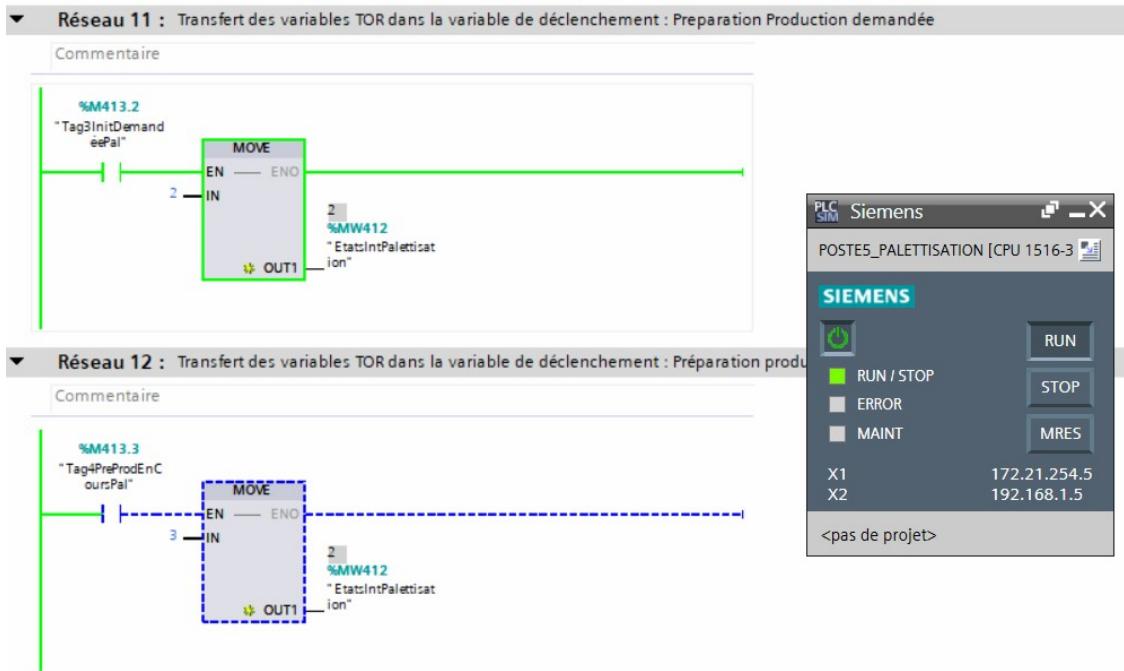


Ligne_TPCyberV7 > POSTES_PALETTISATION [CPU 1516-3 PN/DP] > Blocs de programme > SupervisionEtatsPalettisation [FB1]



Un Move dans la FB[1] transfère l'état décrit par une constante de 1 à 7 dans la variable %MW412 type Int “EtatsIntPalettisation”. Il est activé par la variable TOR correspondante à l'état. La variable %MW412 est utilisée dans l'IHM pour provoquer l'affichage correspondant dans la liste des états du poste.





III.1.2.3 Etablissement du champ d'affichage dans la vue de l'IHM

Dans l'IHM du superviseur, j'ai créé une liste de 8 états, reliés variable %MW412.

Navigateur du projet

Appareils

Ligne_TPCyberV7 > PC-System_1 [SIMATIC PC station] > HMI_RT_7 [WinCC RT Advanced] > Listes de textes et de graphiques

Listes de textes

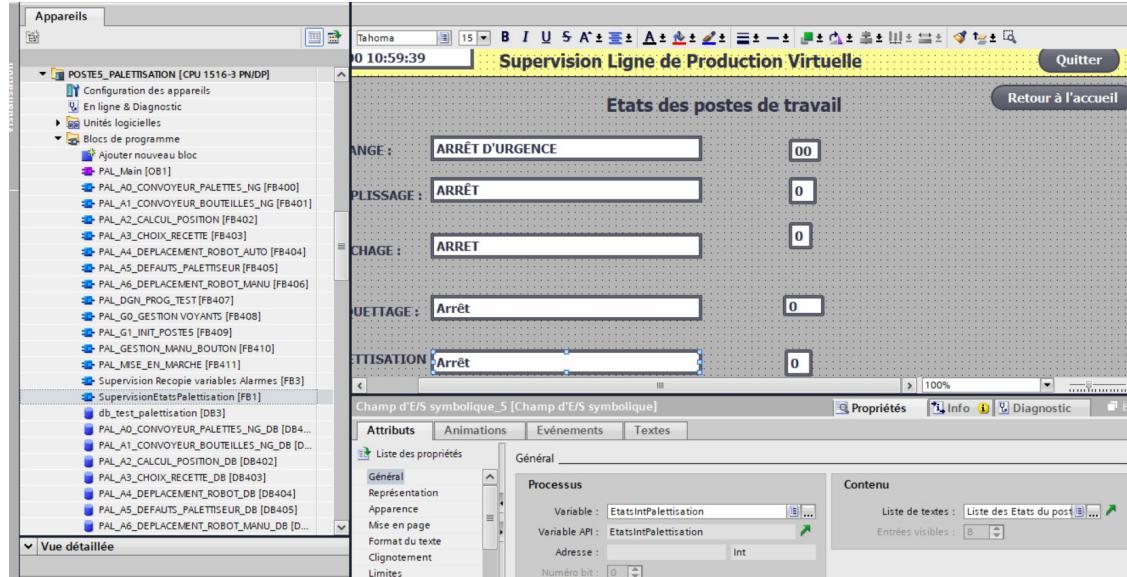
... Nom	Sélection	Commentaire
Liste des Etats du poste Bouchage	Plage (... - ...)	
Liste des Etats du poste Etiquetage	Plage (... - ...)	
Liste des Etats du poste Mélange	Plage (... - ...)	Semblable à la liste * Etat machin
Liste des Etats du poste Palettisation	Plage (... - ...)	
Liste des Etats du poste Remplissage	Plage (... - ...)	
<ajouter>		

Entrées dans la liste de textes

... Par déf.	Valeur	Texte
1	0	Arrêt
2	1	Marche / Machine non initialisée
3	2	Préparation Production demandée
4	3	Préparation Production en cours
5	4	Production Auto
6	5	Production Manuelle
7	6	Maintenance
	7	Le poste Palettisation est en arrêt d'urgence
<ajouter>		

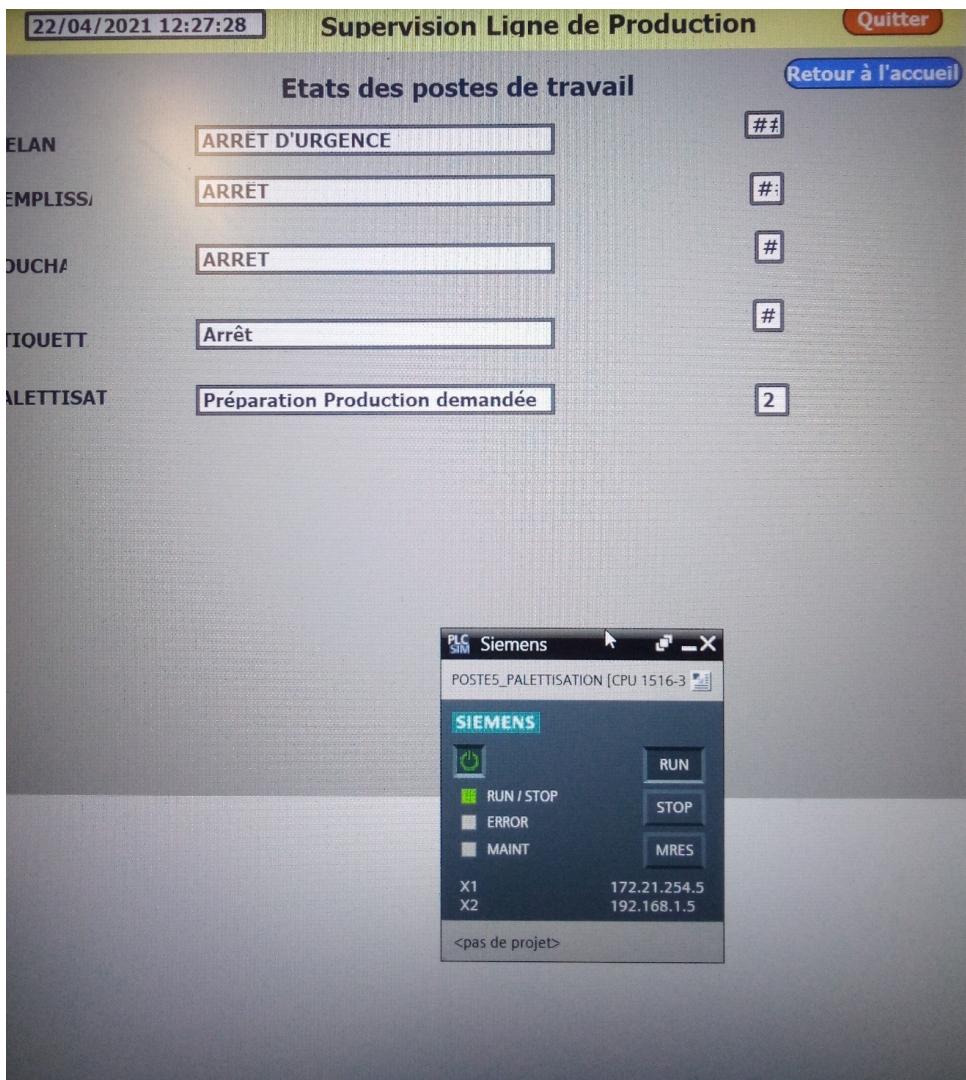
Un champ E/S symbolique lié à cette liste est inséré dans la vue Etat des postes. La visibilité est dynamisée sur une plage 0 à 7.

A des fins de tests, j'ai ajouté un champ E/S, lui aussi lié à la variable %MW412 dont il renvoie la valeur.



III.1.1.4 Vue en simulation

L'écran présente le champ correspondant au poste Palettisation en « Préparation demandée » qui correspond à la valeur 2 de la variable MW412 (cf le champ E/S de test).



III.2 L'écran des alarmes

Deux types de fonction d'alarmes peuvent être utilisées pour prévenir de défauts survenant dans la ligne de production :

- les alarmes qui portent sur des variables analogiques, en définissant par exemple sur la valeur de ces variables des seuils d'alerte ou à ne pas dépasser,
- et des alarmes qui portent sur des variables TOR, faciles à concevoir avec des capteurs binaires mais qui doivent être rassemblées sur un mot de deux octets pour pouvoir être traitées en tant qu'alarmes.

III.2.1 Alarmes analogiques

Aucune alarme analogique n'a été prévue pour l'ensemble de la ligne.

Néanmoins, la variable « quantitelitre » %MW18 de l'API Remplissage, exprimant le pourcentage de remplissage du réservoir d'injection de ce poste, se prête bien à la programmation d'une alarme analogique. Je l'ai utilisée pour créer deux seuils :

- un seuil haut de 95 %, programmé comme « Warning » (pas d'acquittement requis) dans l'écran d'alarmes de l'IHM de supervision,
- et un seuil bas de 5%, retenu pour déclencher une « error » et ainsi provoquer l'ouverture de la page « PopUp » sur toutes les vues de l'IHM du superviseur. Le défaut nécessitera l'acquittement par le superviseur et ne disparaîtra que lorsque le niveau sera ramené au-dessus de ce seuil.

Ligne_TPCyberV7 > PC-System_1 [SIMATIC PC station] > HMI_RT_7 [WinCC RT Advanced] > Alarmes IHM							
Alarmes de bit		Alarmes analogiques		Alarmes de l'API		Alarmes système	
ID	Nom	Texte d'alarme	Classe d'alar...	Variable de d...	Valeur limite	Mode limite	Journal
1	ReservoirPleinRempl...	Le réservoir d'injection Remplissage Warnings	Warnings	Quantitép... 95	95	Supérieure à	<input type="checkbox"/>
2	ReservoirVdeRempl...	Le réservoir d'injection Remplissage Errors	Errors	Quantitép... 5	5	Inférieure à	<input type="checkbox"/>
<ajouter>							

III.2.2 Alarmes de bit

III.2.2.1 Présentation de la démarche de travail

La programmation d'alarmes de bits porte aussi sur les cinq postes, donc pour cet écran, comme pour celui des états, ma démarche n'est présentée ici que sur deux postes les programmations concernant les trois postes sont renvoyées en annexe 2 mais peuvent être traitée à la demande.

Pour déclencher la fonction alarme et ainsi demander leur acquittement par un opérateur ou être mémorisés automatiquement dans un journal d'archivage, les défauts signalés par des variables TOR doivent être rassemblés sur un mot de deux octets. Cette fonction n'ayant pas été requise au départ du projet, il n'y avait pas de ligne directrice qui aurait orienté la programmation dans ce sens au niveau des API et chaque poste a donc constitué un cas particulier pour la programmation d'alarmes de bits.

D'une façon assez semblable à la définition de l'écran Etats des postes, il faut :
rechercher les éléments dans les programmes des API et de l'IHM,

établir des sous-programmes de transfert des défauts TOR dans une variable type Word, utilisée par l'IHM comme variable de déclenchement

établir dans l'IHM, le déclenchement de la fonction Alarme et ses affichages
s'assurer du bon résultat par la simulation.

III.2.2.2 Alarmes de bits du poste Remplissage

III.2.2.2.1 Vue Recherche d'éléments

La vue Production de l'IHM Remplissage montre que plusieurs défauts y sont présents sous la forme de variables TOR mais aucune alarme n'y a pas été programmée (ni variable, ni fonction, ni vue des alarmes, ni vue globale).

Ligne_TPCyberV7 > IHM_POSTE2_REMPLISSAGE [TP700 Comfort] > Alarmes IHM										
Alarmes de bit Alarmes analogiques Alarmes de l'API Alarmes système Classes d'alarmes Groupes d'alarmes										
Alarmes de bit										
ID	Nom	Texte d'alarme	Classe d'alar...	Variable de d...	Bit de ..	Adresse de dé...	Variable d'ac...	Bit d'a...	Adresse d'acq...	Journal
1	Alarme de bit_1		Warnings	<aucune v...>	0	<aucune vari...>	0	0		

Dans l'API, les défauts ont été placés sur le mot MW50 jusqu'au bit M51.3 mais le mot MW50 n'a pas été attribué à une variable. Il peut donc être utilisé comme variable de déclenchement (VarAlarmesRemplissage de type Word) pour renvoyer les alarmes de bit à l'IHM.

III.2.2.2.2 Etablissement du sous-programme de transfert dans l'API

Sur les 4 bits restés libres du MW50, j'ai placé trois défauts supplémentaires (dont l'ARU) pour enrichir la programmation des alarmes de ce poste. Le bit M51.6 restera non utilisé.

La FB[1] Supervision Alarmes Rempl (placée dans l'OB1) est programmée pour transférer les variables des trois dans l'octet MB51.

III.2.2.2.3 Etablissement de l'alarme dans l'IHM

Une table de variables Alarmes est ouverte dans l'IHM de supervision. La variable Alarmes Remplissage y est liée à la variable VarAlarmesRemplissage (%MW50).

Ligne_TPCyberV7 ▶ PC-System_1 [SIMATIC PC station] ▶ HMI_RT_7 [WinCC RT Advanced] ▶ Variables IHM ▶ Alarmes SuperV [5]					
Alarmes SuperV					
Nom	Type de données	Connexion	Nom API	Variable API	Adresse
Alarmes Bouchage	Int	HMI_Liaison_8	POSTE3_BOUCHAGE	DEFAUT	
Alarmes Etiquetage	Int	HMI_Liaison_9	POSTE4_ETIQUETAGE	Liste_défauts	
Alarmes Mélange	Word	HMI_Liaison_6	POSTE1_MELANGE	Var_AlarmMelange	
Alarmes Palettisation	Int	HMI_Liaison_10	POSTE5_PALETTISATION	DEFAUTS_ALARMS	
Alarmes Remplissage	Word	HMI_Liaison...	POSTE2_REMPLISSAGE	VarAlarmesRemplissag...	
<ajouter>					

Alarmes de bit		Alarmes analogiques		Variables d'archive	
ID	Nom	Texte d'alarme	Classe d'alar...	Variable de d...	Bit de ...
9	AALNiveauBasRéserv...	Le niveau de la cuve réserve est bas	Warnings	Alarmes Rem...	12
10	ALARURemplissage	Poste Remplissage en arrêt d'urgent	Errors	Alarmes Rem...	15
11	AlBonbonneRempli...	Le bocal d'injection de Remplissage	Warnings	Alarmes Rem...	13
12	AlCuveRemplissage	Défaut Alimentation de la cuve Rése	Warnings	Alarmes Rem...	0
13	AlCuveVanneRempl...	Défaut vanne de la cuve Remplissag	Warnings	Alarmes Rem...	1
14	AlPompCuveRempli...	Défaut de pompe cuve Remplissage	Warnings	Alarmes Rem...	2
15	AlBocalRemplissage	Défaut au bocal d'injection Rempli	Warnings	Alarmes Rem...	3
16	AlVérinInjectionRe...	Défaut du vérin d'injection Remplissat	Warnings	Alarmes Rem...	4
17	AlAlimBidonRempli...	Défaut d'alimentation des bidons Re	Errors	Alarmes Rem...	5
18	AlArrivBidonRempli...	Défaut arrivée bidon Remplissage	Warnings	Alarmes Rem...	6
19	AlSabotRemplissage	Défaut au sabot du bidon Remplissa	Warnings	Alarmes Rem...	7
20	AlEvacuationRempli...	Défaut à l'évacuation Remplissage	Warnings	Alarmes Rem...	8
21	AlEvacBourrageRe...	Défaut Evacuation Bourrage Remplis	Warnings	Alarmes Rem...	9
22	AlEvacBidonRemp...	Défaut d'évacuation du bidon Rempl Errors		Alarmes Rem...	10
23	AlLevageRemplissa...	Défaut au vérin de levage Remplissa	Warnings	Alarmes Rem...	11
<ajouter>					

La variable, sélectionnée dans la table, permet de remplir directement ses bits d'alarmes.

La fonction « vue des alarmes » est alors glissée sur la page d'alarmes de l'IHM de supervision à partir du menu Accessoires/Contrôle. Son activation se fera alors automatiquement si l'un des bits d'alarme passe à 1.

L'utilisation de la vue globale permet l'affichage en PopUp de la fenêtre en cas de déclenchement d'une alarme.

III.2.2.2.4 Vues en simulation

La page de l'IHM de supervision est activée en simulation avec le défaut « bonbonne de l'injecteur vide » forcé.



III.2.2.3 Alarmes de bits du poste Mélange

Comme initialement aucune alarme n'était prévue pour ce poste, j'ai utilisé les variables signalant les niveaux bas et haut du stock et les niveaux bas des produits A, B et C et l'arrêt d'urgence, pour signaler des alarmes à l'IHM.

Le MW140 était libre dans la table d'affectation. J'ai renvoyé les variables TOR sur ce mot par un sous-programme FB [6] Supervision Alarmes, simple, inséré dans l'OB1 du poste Mélange et placée une variable « Var_AlMélange » de type Word sur ce mot (%MW140).

La variable Alarmes Mélange est créée dans la table de variables Alarmes de l'IHM de supervision et liée à la variable Var_AlMélange %MW140. Elle rassemble les bits d'alarme, introduits dans la fenêtre Alarmes de bit.

Ligne_TPCyberV7 > PC-System_1 [SIMATIC PC station] > HMI_RT_7 [WinCC RT Advanced] > Variables IHM > Alarmes SuperV [5]

J'ai programmé certains de ces bits d'alarmes en « errors », d'autres en « warnings » pour activer, ou non, un acquittement et provoquer, ou non, l'ouverture de la page pop-up d'alarme.

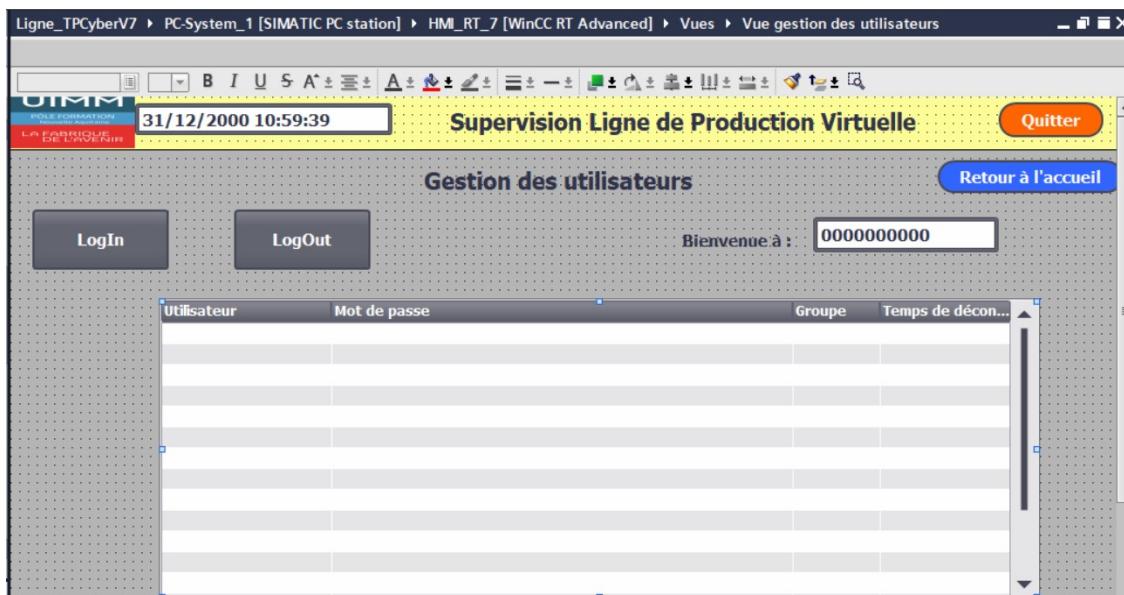
La vue de l'écran accueil montre ici la fenêtre d'alarme activée en PopUp par les défauts ARU de mélange et deux défaut de Remplissage.



III.3 Ecran Gestion des utilisateurs

Une page est créée dans l'IHM de supervision pour enregistrer les utilisateurs autorisés, leur donner des droits d'accès et mémoriser leurs connexions.

Cette page est reliée par un bouton à la page d'accueil, à partir de laquelle on peut aussi y accéder. Le bouton LogIn permet à l'utilisateur de s'identifier et d'entrer son mot de passe. Celui du LogOut fait quitter l'accès. Un champ E/S affiche l'utilisateur se connectant.



Pour créer cette page, j'ai utilisé la fonction « Vue des utilisateurs » prise dans le menu contrôle, ainsi que celle de « Gestion des utilisateurs » prise dans le navigateur du projet.

Utilisateurs						
	Nom	Mot de passe	Fermeture de session a..	Durée de la session	Numéro	Commentaire
Administrateur	*****		<input checked="" type="checkbox"/>	5	1	L'utilisateur 'Administrateur' ..
Admin	*****		<input checked="" type="checkbox"/>	5	2	
Utilisateur1	*****		<input checked="" type="checkbox"/>	5	3	
Technicien1	*****		<input checked="" type="checkbox"/>	5	4	
<ajouter>						

Groupes						
	Membre de	Nom	Numéro	Nom d'affichage	Vieillissement d..	Commentaire
		Groupe administrateurs	1	Groupe administrateurs	<input type="checkbox"/>	Le groupe 'Administrateurs' ...
		Utilisateurs	2	Utilisateurs	<input type="checkbox"/>	Le groupe 'Utilisateurs' a le d..
		Maintenance	3	Maintenance	<input type="checkbox"/>	Le groupe Maintenance n'a q..
<ajouter>						

Pour l'exemple, les utilisateurs Admin, Utilisateur1 et Technicien1 respectivement munis des codes d'accès 5555,1111 et 3333 (ces codes de faible sécurité seraient changés dans le cadre d'une utilisation réelle) sont mis dans trois groupes différents (Administrateurs, Utilisateurs et Maintenance), lesquels sont dotés d'autorisations d'accès plus ou moins larges : toutes pour les administrateurs, ... uniquement commande pour « Maintenance ».

Groupes					
	Nom	Numéro	Nom d'affichage	Vieillissement d..	Commentaire
	Groupe administrateurs	1	Groupe administrateurs	<input type="checkbox"/>	Le groupe 'Administrateurs' a tous les droits au début.
	Utilisateurs	2	Utilisateurs	<input type="checkbox"/>	Le groupe 'Utilisateurs' a le droit 'Commande' au début.
	Maintenance	3	Maintenance	<input type="checkbox"/>	Le groupe Maintenance n'a que le droit 'Commande'.
<ajouter>					

Autorisations					
Actif	Nom	Nom d'affichage	Numéro	Commentaire	
<input type="checkbox"/>	Gestion des utilisateurs	Gestion des utilisateurs	1	Autorisation 'Gestion utilisat...	
<input type="checkbox"/>	Contrôle	Contrôle	2	Autorisation 'Contrôle'.	
<input checked="" type="checkbox"/>	Commande	Commande	3	Autorisation 'Commande'.	
<input type="checkbox"/>	Autorisation_gestion_utili...	Autorisation gestion utilisateurs	4		
<ajouter>					

Les boutons LogIn LogOut sont animés par l'action du « Planificateur de tâche » pris dans le navigateur de projet, la tâche étant celle de « Changement d'utilisateur » sélectionnée dans la colonne « Déclenchement » et exécutée par la propriété Fonction Système/Gestion des utilisateurs/ LireNomUtilisateur et reliée à la variable « Utilisateur » créée dans la TAV de l'IHM en type WString.

25/04/2021 16:34:15 Supervision Ligne de Production Vi [Quitter](#)

Gestion des utilisateurs

[Retour à l'accueil](#)

[LogIn](#) [LogOut](#)

Bienvenue Admin

Utilisateur	Mot de passe	Groupe	Temps de d...
Admin	*****	Groupe ...	5
Administrateur	*****	Groupe ...	5
PLC User	*****	non aut...	5
Technicien1	*****	Mainten...	5
Utilisateur1	*****	Utilisate...	5

25/04/2021 16:35:12 Supervision Ligne de Production Vi [Quitter](#)

Gestion des utilisateurs

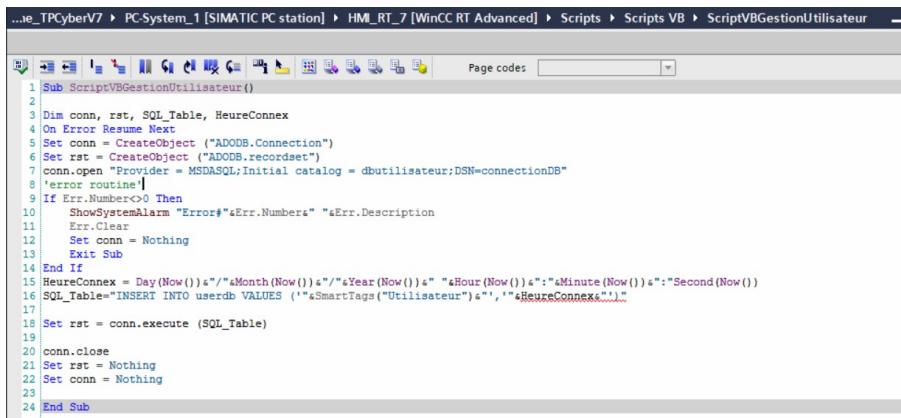
[Retour à l'accueil](#)

[LogIn](#) [LogOut](#)

Bienvenue Technicien

Utilisateur	Mot de passe	Groupe	Temps de d...
Technicien1	*****	Mainten...	5

L'enregistrement des connexions/déconnexions des utilisateurs est réalisé ici par l'utilisation d'un VBscript placé dans le Planificateur de tâche, qui appelle un fichier de base de données dans lequel ces connexions sont inscrites sous la forme date et heures.



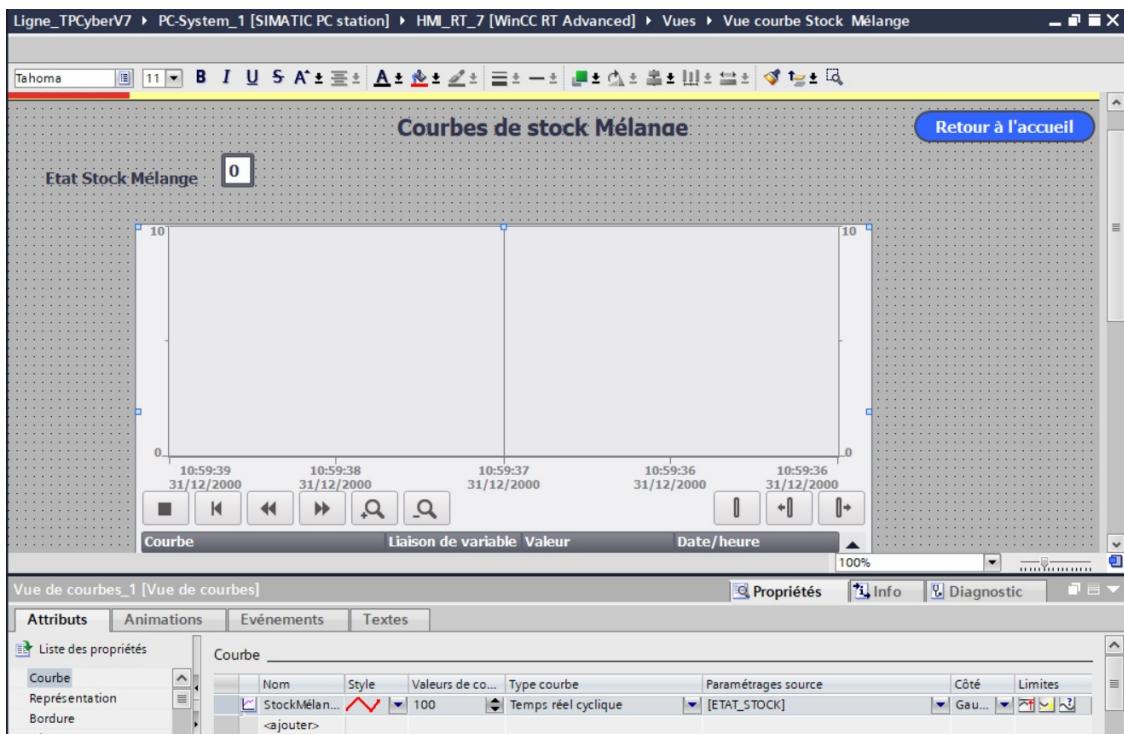
```
...ie_TPCyberV7 > PC-System_1 [SIMATIC PC station] > HMI_RT_7 [WinCC RT Advanced] > Scripts > Scripts VB > ScriptVBGestionUtilisateur
Sub ScriptVBGestionUtilisateur()
    Dim conn, rst, SQL_Table, HeureConnex
    On Error Resume Next
    Set conn = CreateObject ("ADODB.Connection")
    Set rst = CreateObject ("ADODB.recordset")
    conn.open "Provider = MSQASQL;Initial catalog = dbutilisateur;DSN=connectionDB"
    'error routine'
    If Err.Number<>0 Then
        ShowSystemAlarm "Error#" & Err.Number & Err.Description
        Err.Clear
        Set conn = Nothing
        Exit Sub
    End If
    HeureConnex = Day(Now()) & "/" & Month(Now()) & "/" & Year(Now()) & " " & Hour(Now()) & ":" & Minute(Now()) & ":" & Second(Now())
    SQL_Table="INSERT INTO userdb VALUES ('" & SmartTags("Utilisateur") & "','" & HeureConnex & "')"
    Set rst = conn.execute (SQL_Table)
    conn.close
    Set rst = Nothing
    Set conn = Nothing
End Sub
```

III.4 Ecran de Courbes

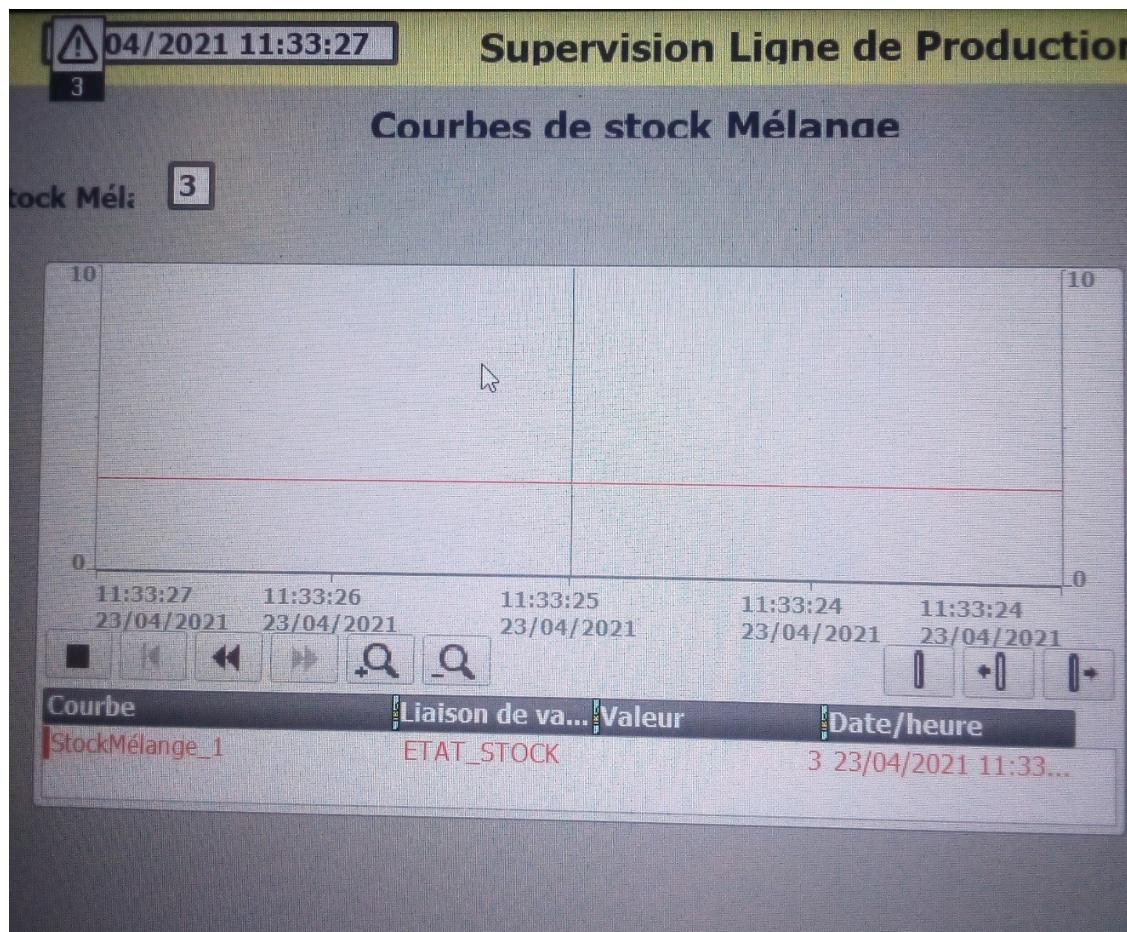
Des courbes sont établies pour le poste mélange et pour le poste remplissage.

III.4.1 Courbe « Stock Mélange »

La courbe est établie sur l'écran « Vue Stock mélange » et animée par la variable ETAT_STOCK %MW8 du poste Mélange. Cette variable est un type INT programmée pour prendre les valeurs de 0 à 4.

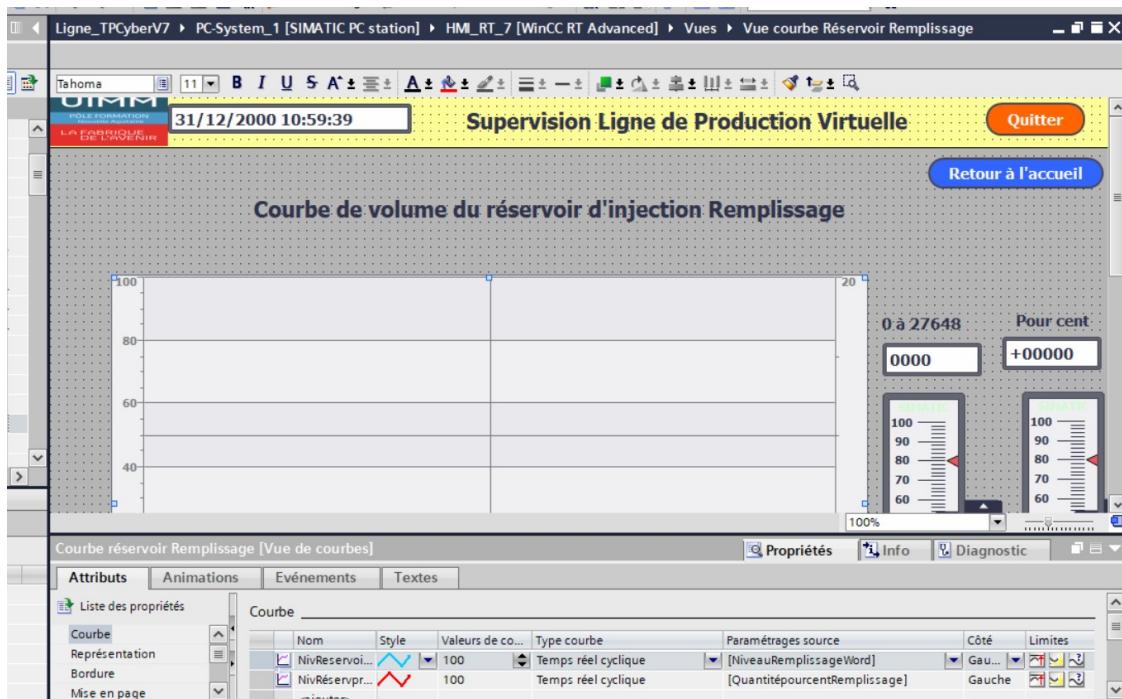


La simulation est établie ici avec une valeur de stock égale à 3 (stock assez plein).

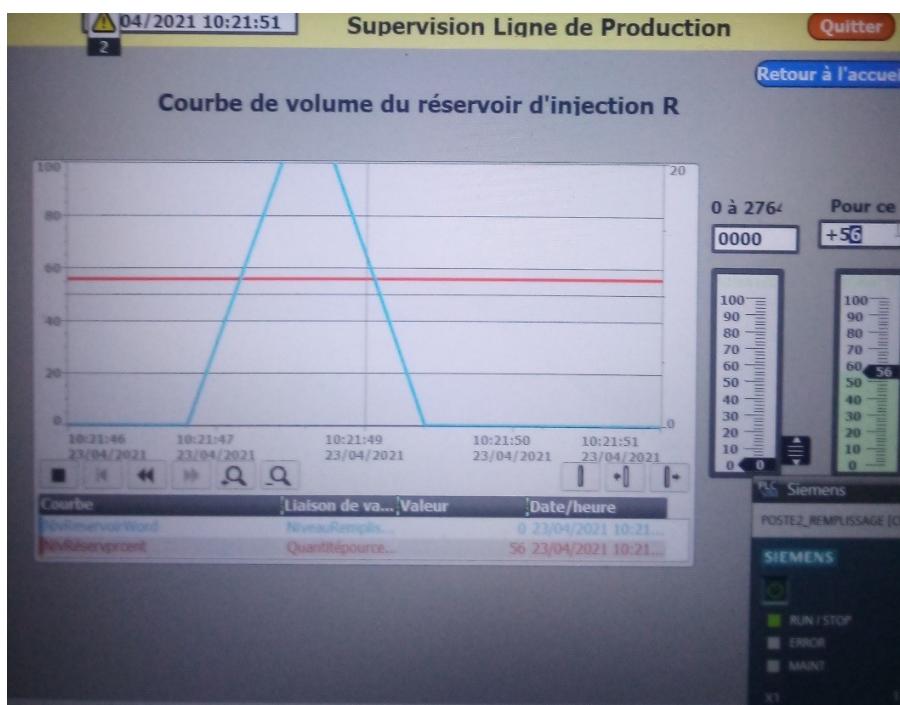


III.4.2 Courbe du réservoir d'injection du poste Remplissage

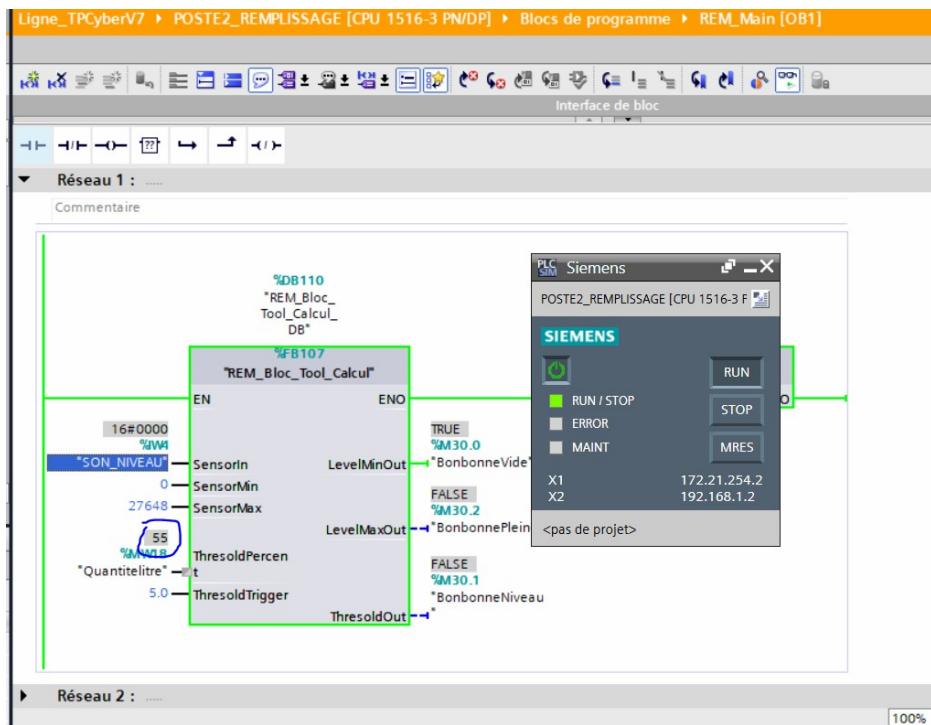
Deux variables sont disponibles dans la table de variables du programme de l'API Remplissage. L'une, "Son_Niveau" %IW4 sur un Word exprime le niveau de la sonde de la bonbonne d'injection, l'autre "QuantitéLitres" sur un INT en %MW18 est le pourcentage de remplissage.



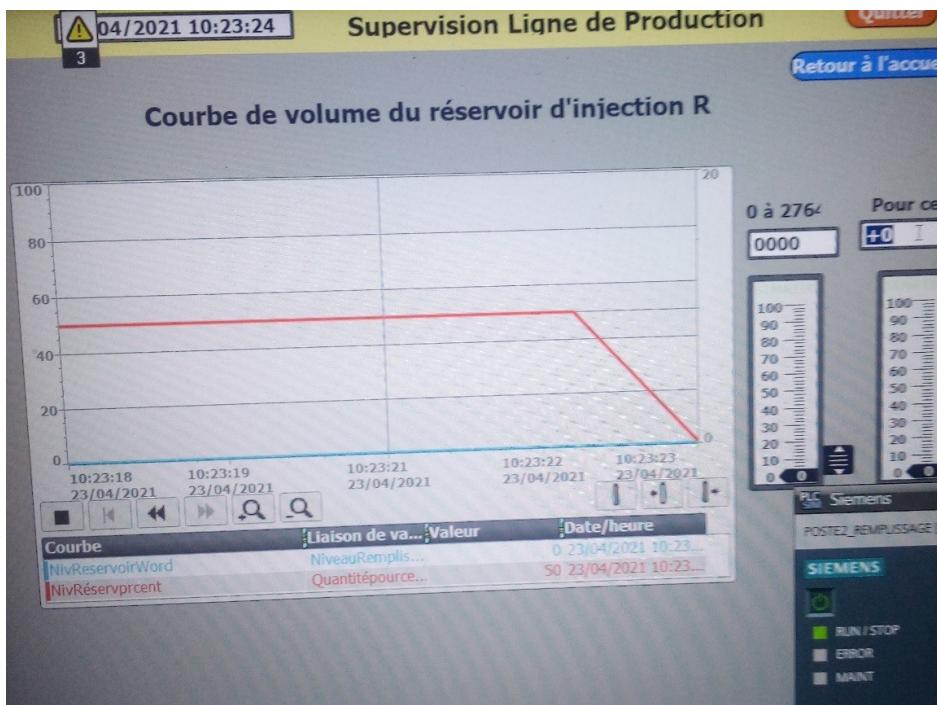
Les courbes de l'écran « Vue courbes réservoir Remplissage » sont établies sur ces deux variables et la vue est munie de deux curseurs et de deux champs E/S liés à ces deux variables pour mieux voir le bon fonctionnement en simulation.



La simulation est activée ici avec une valeur de 55 dans la variable « quantitélitre »,



avant de retomber à zéro



III. 5 Ecrans de recettes

J'ai établi deux écrans de recettes :

- la « Vue Recette "INT" du mélange ». Dans le programme de l'API Mélange, le DB [101] a été créé en convertissant toutes ses variables utilisées en type INT. Cette première recette « INT » utilise ces données.
 - la « Vue Recette "Real" Mélange » qui, elle, utilise les variables originales que j'ai placées dans le DB[13].

III.5.1 Première recette Mélange 2 en « INT ».

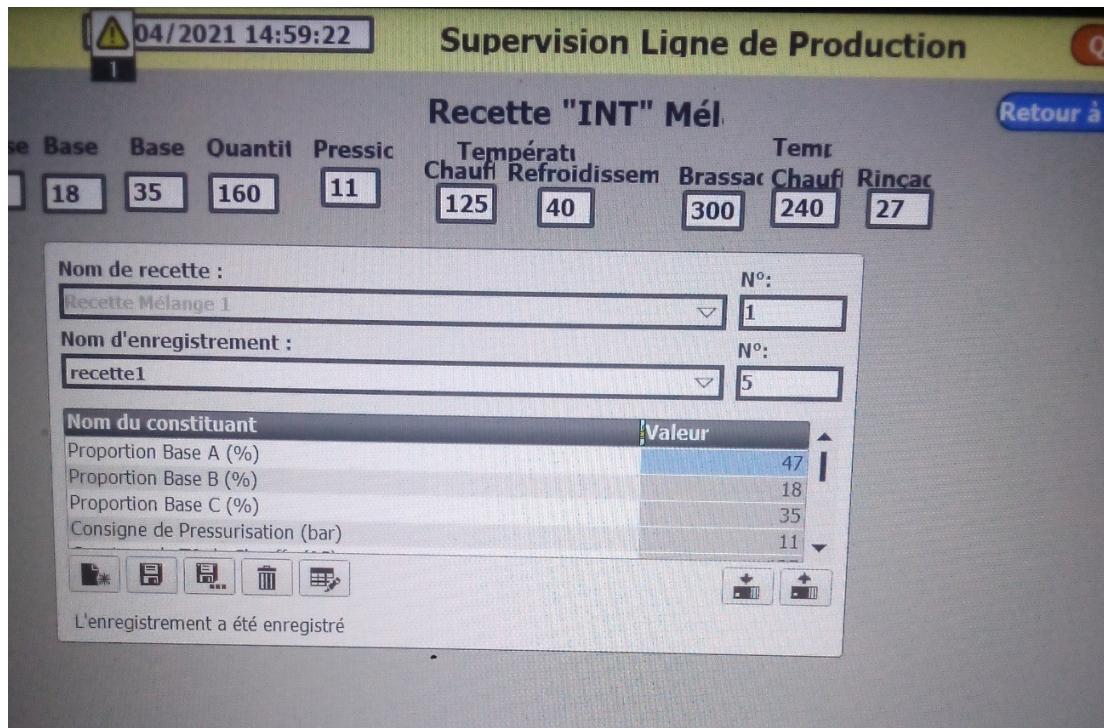
Les éléments à paramétrier pour la recette sont dans ce cas dans le DB[101].

		Conserver les valeurs actuelles		Instantané	Copier les instantanés dans les valeurs de départ						
Bloc de données_1 (instantané créé : 26/03/2021 09:33:02)											
	Nom	Type de données	Décalage	Valeur de départ	Rémanence	Accessible ...	Écritu...	Visible da...	Valeur de ...		
1	Static										
2	BaseA	Int	0.0	0							
3	BaseB	Int	2.0	0							
4	BaseC	Int	4.0	0							
5	Pression	Int	6.0	0							
6	Temperature_Chauffe	Int	8.0	0							
7	Température_Refroidi...	Int	10.0	0							
8	Temps_Brassage	Int	12.0	0							
9	Temps_Chauffe	Int	14.0	0							
10	Temps_Rincage	Int	16.0	0							
11	reserve	Int	18.0	0							
12	QuantiteProduit	Int	20.0	0							

La recette est insérée dans un écran « Recette » en ouvrant la fonction Recette du menu du projet IHM, nommée, puis les « éléments » à paramétrer sont introduits.

Ensuite, la recette est glissée sur la vue IHM correspondante en utilisant son rappel dans la fenêtre « vue détaillée ». Ceci place automatiquement dans l'écran la fenêtre de recette dans laquelle les valeurs des paramètres peuvent être introduits par l'opérateur.

J'ai ajouté sur l'écran des champs E/S de test pour visualiser les valeurs paramétrées dans la recette en mode simulation.



Bloc de données_1 (instantané créé : 26/03/2021 09:33:02)								
Nom	Type de données	Décalage	Valeur de départ	Valeur de visualisati.	Rémanence	Accessible ...	Ecritu...	...
1 Static								
2 BaseA	Int	0.0	0	47		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
3 BaseB	Int	2.0	0	18		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
4 BaseC	Int	4.0	0	35		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
5 Pression	Int	6.0	0	11		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
6 Temperature_Chauffe	Int	8.0	0	125		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
7 Température_Refroidissem...	Int	10.0	0	40		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
8 Temps_Brassage	Int	12.0	0	300		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
9 Temps_Chauffe	Int	14.0	0	240		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
10 Temps_Rincage	Int	16.0	0	27		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
11 reserve	Int	18.0	0	53		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
12 QuantiteProduit	Int	20.0	0	160		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

III.5.2 Deuxième recette Mélange 2 en « Real », « Int » et « Time »

Comme les variables utilisées dans la recette sont en fait de type soit Real soit Time, j'ai constitué un nouveau DB prenant l'ensemble des variables de la table « recette » dans leur type.

DBSuperVRecettesdemélange										
	Nom	Type de données	Valeur de départ	Rémanence	Accessible ...	Ecritu...	Visible da...	Valeur de ..	Surveilla...	Commentaire
1	Static									
2	SVBaseA	Real	0.0		<input checked="" type="checkbox"/>					
3	SVBaseB	Real	0.0		<input checked="" type="checkbox"/>					
4	SVBaseC	Real	0.0		<input checked="" type="checkbox"/>					
5	SVNbrImpulsionsA	Int	0		<input checked="" type="checkbox"/>					
6	SVNbrImpulsionsB	Int	0		<input checked="" type="checkbox"/>					
7	SVNbrImpulsionsC	Int	0		<input checked="" type="checkbox"/>					
8	SVPressionRéel	Real	0.0		<input checked="" type="checkbox"/>					
9	SVQuantitéProduitRéel	Real	0.0		<input checked="" type="checkbox"/>					
10	SVTempéraChau... Réel	Real	0.0		<input checked="" type="checkbox"/>					
11	SVTempéraRefroidissem... Réel	Real	0.0		<input checked="" type="checkbox"/>					
12	SVMargeTempéraRefr... Réel	Real	0.0		<input checked="" type="checkbox"/>					
13	SVTempsRéalBrassage	Real	0.0		<input checked="" type="checkbox"/>					
14	SVTimeTempsBrassage	Time	T#0ms		<input checked="" type="checkbox"/>					
15	SVTempsRéalChau... Réel	Real	0.0		<input checked="" type="checkbox"/>					
16	SVTimeTempsChau... Réel	Time	T#0ms		<input checked="" type="checkbox"/>					
17	SVTimeTpsRestantBras... Réel	Time	T#0ms		<input checked="" type="checkbox"/>					
18	SVTimeTpsRestantCha... Réel	Time	T#0ms		<input checked="" type="checkbox"/>					
19	SVTimeTpsRestantRinc... Réel	Time	T#0ms		<input checked="" type="checkbox"/>					
20	SVTempsRéalRincage	Real	0.0		<input checked="" type="checkbox"/>					
21	SVTimeTempsRincage	Time	T#0ms		<input checked="" type="checkbox"/>					
22	SVTestRecetteVideRéel	Real	0.0		<input checked="" type="checkbox"/>					
23	SVTRHeures	Int	0		<input checked="" type="checkbox"/>					
24	SVTRMinutes	Int	0		<input checked="" type="checkbox"/>					
25	SVTRSecondes	Int	0		<input checked="" type="checkbox"/>					

Une fonction SCL (la FB[7] SVchargeDB13) envoyée dans l'OB[1]) les transfert dans ce DB[13]).

```
Ligne_TPCyberV7 ▶ POSTE1_MELANGE [CPU 1516-3 PN/DP] ▶ Blocs de programme ▶ SVChargeDB13 [FB7]

Interface de bloc

IF... CASE... FOR... WHILE... (*...) REGION
  1 "DBSuperVRecettesdemélange".SVBaseA := "BaseA";
  2 "DBSuperVRecettesdemélange".SVBaseB := "BaseB";
  3 "DBSuperVRecettesdemélange".SVBaseC := "BaseC";
  4 "DBSuperVRecettesdemélange".SVNbrImpulsionsA := "Nombre Impulsions A";
  5 "DBSuperVRecettesdemélange".SVNbrImpulsionsB:= "Nombre Impulsions B";
  6 "DBSuperVRecettesdemélange".SVNbrImpulsionsC := "Nombre Impulsions C";
  7 "DBSuperVRecettesdemélange".SVPressionRéel := "Pression";
  8 "DBSuperVRecettesdemélange".SVQuantitéProduitRéel := "QuantiteProduit";
  9 "DBSuperVRecettesdemélange".SVTempéraChau... Réel := "Temperature_Chau... Réel";
 10 "DBSuperVRecettesdemélange".SVTempéraRefroidissementRéel := "Température_...
 11 "DBSuperVRecettesdemélange".SVTempéraRefroidissementRéel := "Temp... Réel";
```

Dans cette deuxième recette, les « éléments » sont remplis a partir du DB[13].

Cette recette est glissée de la fenêtre vue détaillée sur la vue IHM Vue Recette "Real" Mélange sur laquelle des champs supplémentaires sont insérés comme précédemment pour une meilleure visualisation de la simulation.

III.6 Ecran Accueil

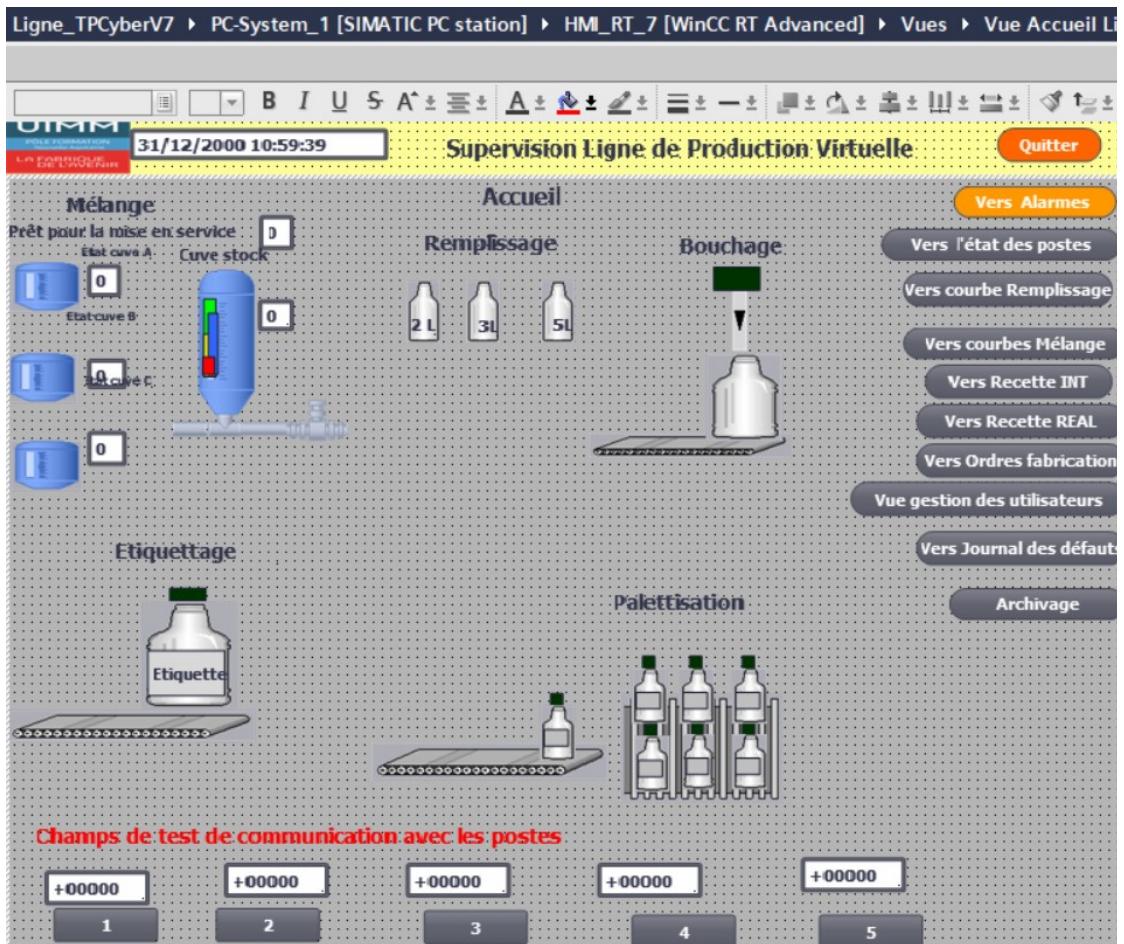
Cet écran est celui qui est utilisé à l'ouverture de la supervision.

Il présente les cinq postes dans leur ordre de fonctionnement de façon schématisée et permet d'atteindre les différents écrans par des boutons. Ces boutons sont ajoutés à la conception de chaque nouvel écran. Accueil est le tronc de l'arborescence : on y revient pour atteindre un écran différent. L'arborescence est restée très simple, à un seul niveau, compte tenu des fonctions qui ont été développées dans cette supervision.

Dans l'écran Accueil, j'ai placé les champs de test qui m'ont permis de m'assurer que la communication avec chacun des postes était bien établie.

De plus, dans la représentation du poste Mélange, j'ai introduit :

- un champ d'entrée de texte qui affiche son état et un champ E/S qui donne la valeur correspondante de la variable INT,
- des champs E/S qui indique les niveaux par les valeurs des variables INT des cuves « Produits A, B, C »
- des barres de niveau qui montrent le remplissage de la cuve stock en fonction de la valeur de sa variable INT de l'API.



Conclusion

Malgré l'épidémie de la COVID, j'ai effectué ce stage de sept semaines grâce à l'organisation que l'UIMM de Lorient a rapidement mis en place pour faire face à une situation de pénurie de stages dans les entreprises. La contrepartie a été pour moi de réaliser seul le développement de cette application en autonomie, c'est-à-dire sans l'aide de collègues de travail plus anciens que l'on peut questionner quand une difficulté semble incontournable.

Mais cette carte blanche présentait aussi beaucoup d'intérêt parce que j'ai dû mieux comprendre, appliquer, et surtout compléter les notions apprises lors des premiers mois de la formation, que ce soit la programmation et le fonctionnement des API et ou celles de programmation des IHM, et ceci tout en faisant avancer ce projet de supervision,

réelle opportunité pour moi, dans la mesure où cette dernière est à la croisée de l'informatique et de l'automatisme.

Cela m'a aussi permis de découvrir que la supervision avait des espaces d'application bien plus larges que celle de notre « Ligne de Production Cyber » et ses cinq postes et quelle était en elle-même une spécialisation.

La supervision est un outil très puissant de contrôle et de gestion centralisés des lignes de production automatisées : le travail ébauché ici avec Wincc Advance est poussé plus loin avec Wincc Pro dans les entreprises, et permet par exemple d'accéder à distance aux lignes de production en utilisant des outils tels Web UX ou Web Navigator ou plus simplement réunir dans un projet des équipements de constructeurs différents en utilisant OPC.

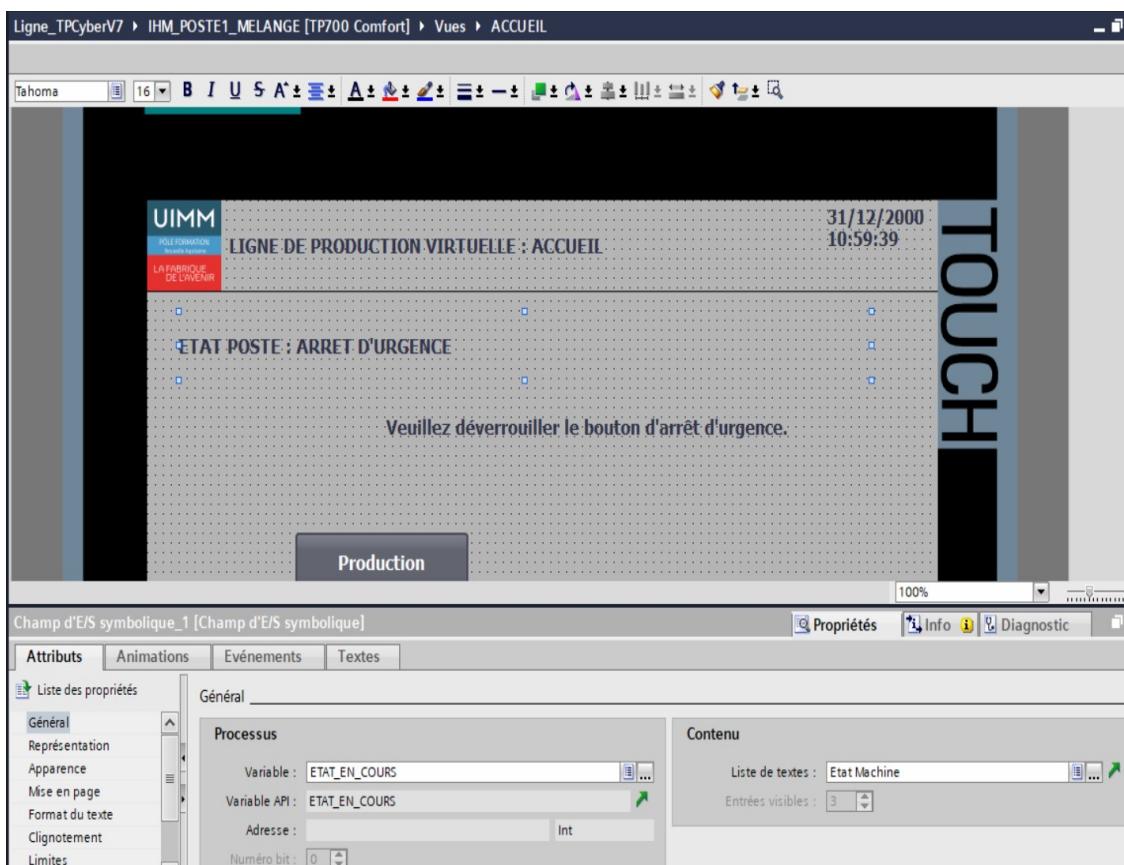
Annexe 1

Réalisation des champs Etats Mélange, Remplissage, Bouchage et Etiquetage » de la vue Etats des postes.

A.1.1 Champ « Etats Mélange »

A.1.1.1 Recherche d'éléments

Les états sont programmés dans l'API, la liste existe dans l'IHM et est reliée à la variable d'activation (ETAT_EN_COURS %MW3)



Remarque : placer l'arrêt d'urgence sur la valeur 0 de la liste conduit à un affichage erroné lorsque l'API est à l'arrêt. Pour l'éviter, j'ai établi les listes en supervision en plaçant arrêt sur la valeur 0.

Liste de texte de 12 Etats de Mélange

Ligne_TPCyberV7 ▶ IHM_POSTE1_MELANGE [TP700 Comfort] ▶ Listes de textes et de graphiques			
Listes de textes			
...	Nom ▲	Sélection	Commentaire
1_	Etat Cuve	Plage (... - ...)	
1_	Etat Machine	Plage (... - ...) ▾	
1_	Etat Pressostat	Plage (... - ...)	
1_	Etat production	Plage (... - ...)	
1_	Noms_Vues	Plage (... - ...)	
	<ajouter>		
Entrées dans la liste de textes			
...	Par dé..	Valeur ▲	Texte
1_	0		ETAT POSTE : ARRET D'URGENCE
1_	1		ETAT POSTE : PREPARATION POUR REMISE EN ROUTE
1_	2		ETAT POSTE : ARRET DANS L'ETAT INITIAL
1_	3		ETAT POSTE : MISE P.O. DANS L'ETAT INITIAL
1_	4		ETAT POSTE : MARCHE DE PREPARATION
1_	5		ETAT POSTE : PRODUCTION
1_	6		ETAT POSTE : MARCHE DE CLOTURE
1_	7		ETAT POSTE : PRODUCTION TOUT DE MEME
1_	8		ETAT POSTE : ARRET DEMANDE FIN DE CYCLE
1_	9		ETAT POSTE : MARCHE DE VERIFICATION DANS LE DE
1_	10		ETAT POSTE : MARCHE DE VERIFICATION DANS L'ORD
1_	11		ETAT POSTE : MARCHES DE TEST
	<ajouter>		

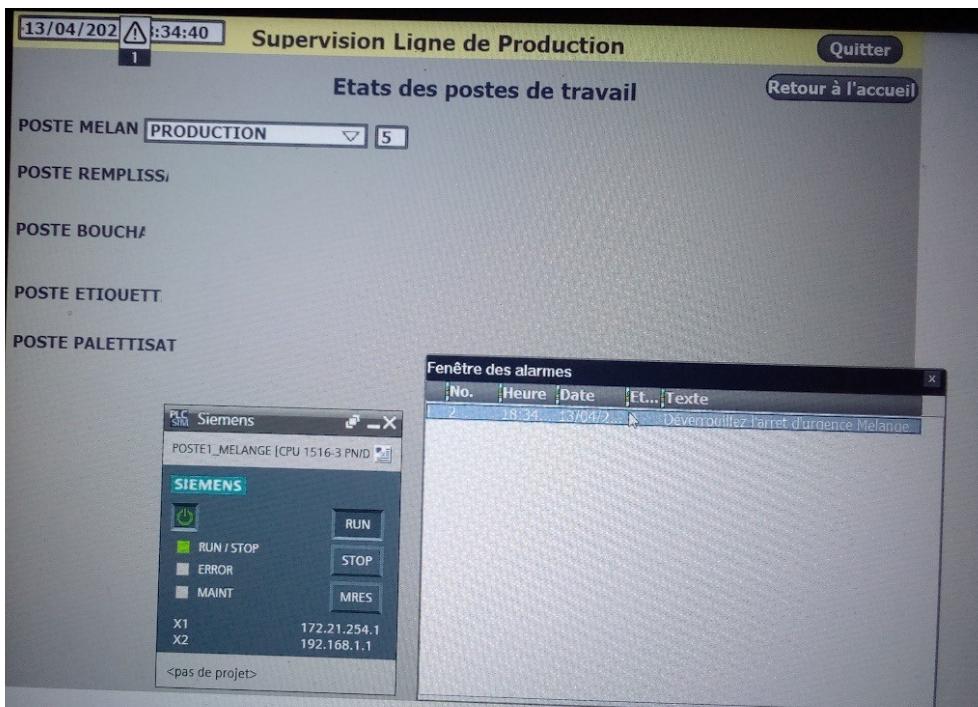
A.1.1.2 Etablissement du champ d'affichage dans la vue de l'IHM

Dans l' IHM de supervision, la liste de texte recopie celle de l'IHM Mélange et elle est reliée à la variable "ETAT_EN_COURS" dans un champ E/S symbolique sur la vue. La visibilité est dynamisée sur une plage de 0 à 11.

A des fins de contrôle et de tests, j'ai ajouté un champ E/S lié à la variable ETAT_EN_COURS.

A.1.1.3 Vue en simulation

En simulation, le champ affiche 'Production' pour une valeur de la variable forcée à 5.



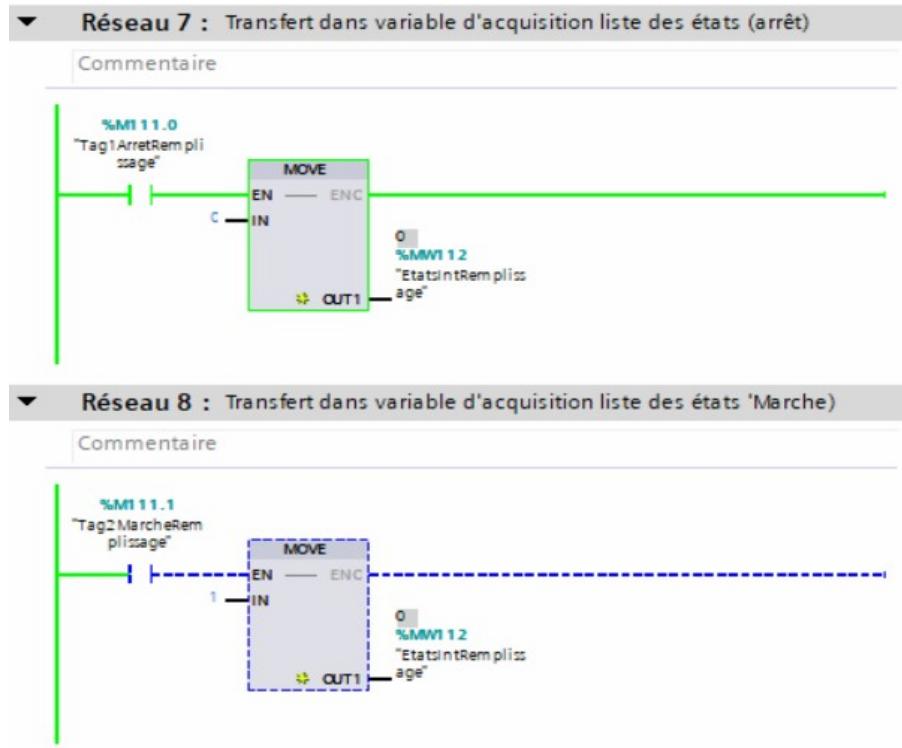
A 1.2 « Etats Remplissage »

A 1.2.1 Recherche d'éléments

La table d'affectation montre que la mémoire est disponible à partir du %MW110. Les Bits de %M111.0 à %M111.5 sont pris pour les variables TOR Arrêt, Marche, Production, Maintenance, Préparation mise en route et Arrêt d'urgence.

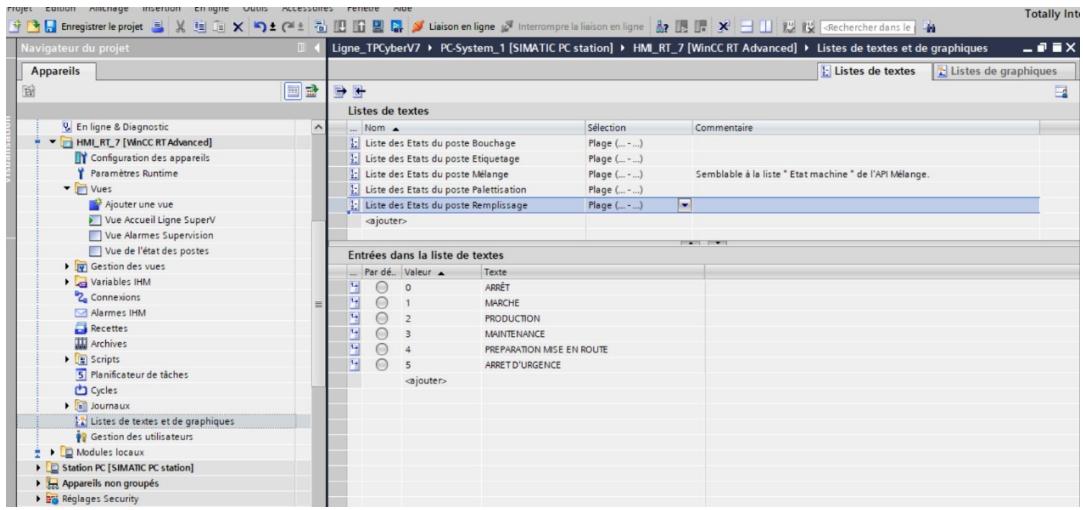
Le sous-programme FB [2] Supervision Etats établit les états traduits par les variables TOR et les transfère dans la variable type INT EtatsIntRemplissage %MW112.

La variable %MW112 est chargée par le MOVE des réseaux 7, 8, 9, 10, 11 et 12, positionnés ici sur l'état Arrêt (0) et déclenche l'affichage programmé dans l'IHM par la liste introduite dans la fonction liste de texte.



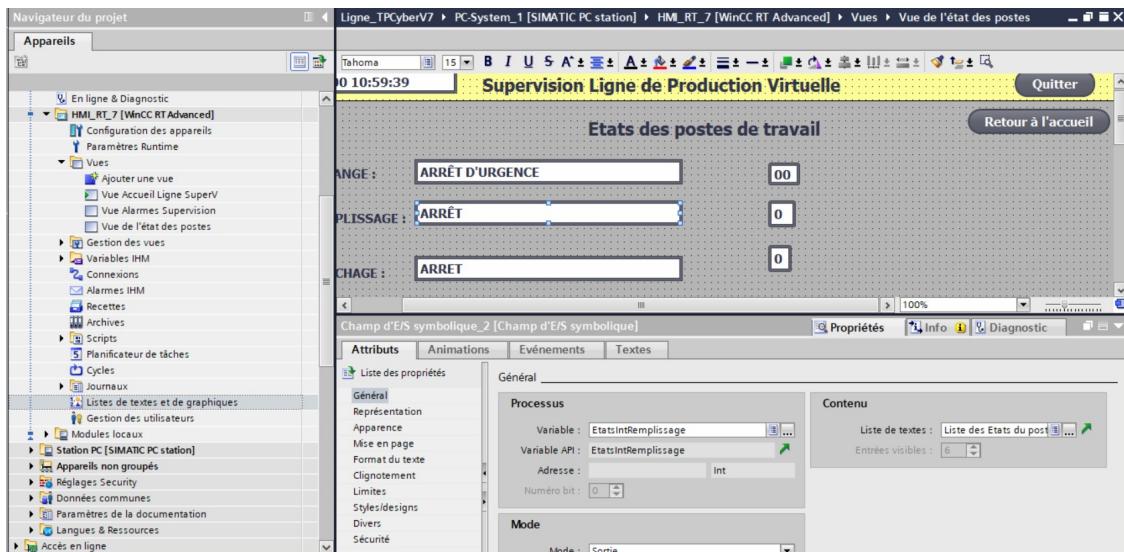
A 1.2.2 Etablissement dans l'API du sous-programme de transfert

Comme précédemment, on crée une liste des états du poste Remplissage dans notre IHM,

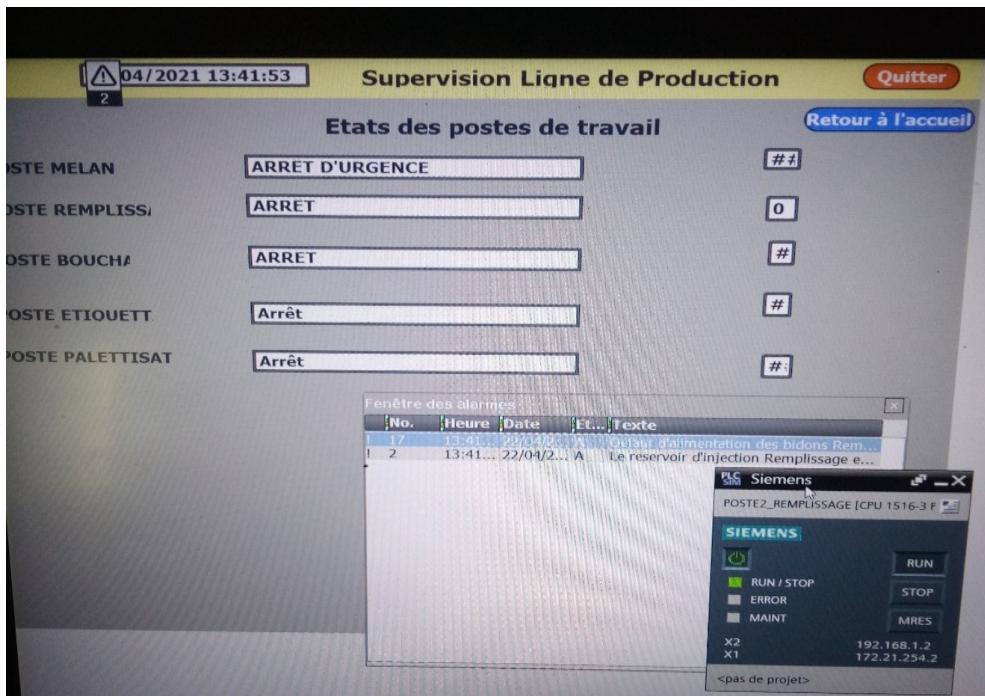


et dans la vue IHM de supervision Etats des postes de travail, on place un champ E/S symbolique lié à cette liste et un champ E/S supplémentaire lié à la variable (INT) de déclenchement %MW112 EtatsIntRemplissage pour test et contrôle.

La visibilité est dynamisée sur une plage de 0 à 5.



La vue en simulation montre que le champ Remplissage est activé sur Arrêt (le champ E/S est à 0 et non #).



A 1.3 « Etats Bouchage »

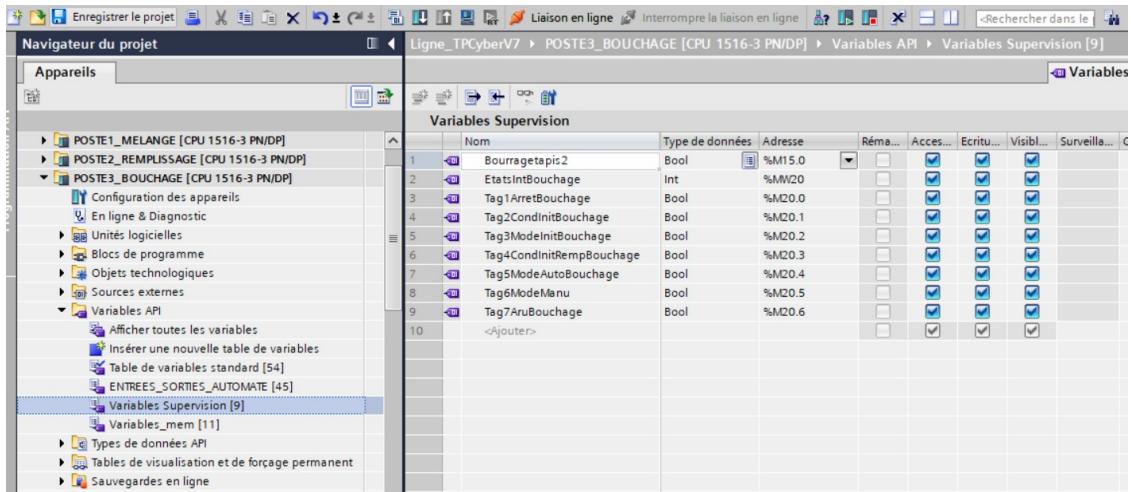
A 1.3.1 Recherche d'éléments

Dans L'IHM Bouchage il n'y a pas de programmation d'état du poste et pas de variable dédiée qui renvoie à l'API.

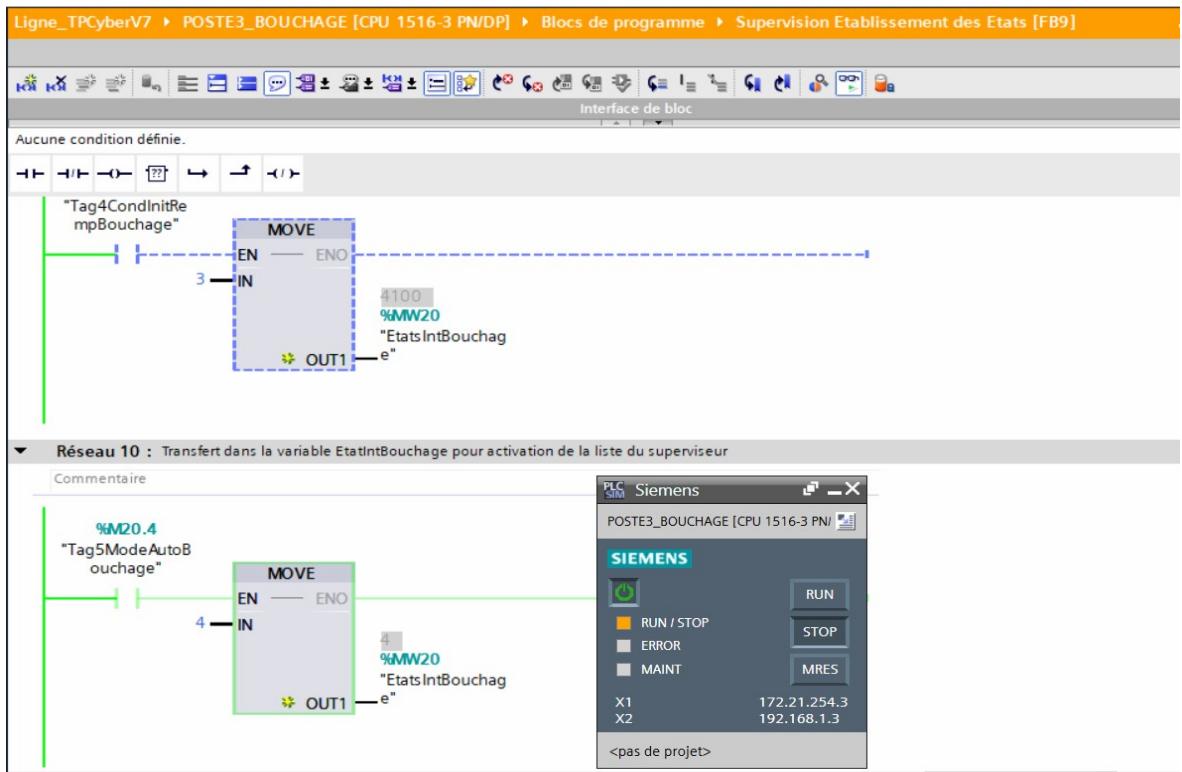
Dans le Programme, le Réseau 1 du Bloc OB1, reprend les variables "Mode_auto" en %M3.1, "Mode_manuel" en %M3.2, "Mode.Initialisation" en %M3.4, ARU en %I3.7 et Arret en %I3.4 mais il y a aussi d'autres variables sur le MW3.

A 1.2.1 Etablissement dans l'API du sous-programme de transfert

Un sous-programme de transfert, simple, est constitué comme pour les postes précédents et après une vérification dans la table d'affectation, les variables TOR sont placées de %MW20.0 à %MW20.6.



Le mot MW20 est utilisé pour la variable INT EtatsIntBouchage et le transfert dans cette variable est fait par les réseaux 6 à 12 de la FB [9], vus en simulation dans l'état « Mode Auto » (constante =4).



A 1.3.3 Etablissement du champ d'affichage dans la vue IHM.

Comme précédemment, je crée une liste des états du poste Bouchage dans notre IHM.

Vigie du projet

Ligne_TPCyberV7 > PC-System_1 [SIMATIC PC station] > HMI_RT_7 [WinCC RT Advanced] > Listes de textes et de graphiques

Appareils

Listes de textes

Nom	Sélection	Commentaire
Liste des Etats du poste Bouchage	Plage (... - ...)	
Liste des Etats du poste Etiquetage	Plage (... - ...)	
Liste des Etats du poste Mélange	Plage (... - ...)	Semblable à la liste "Etat machin"
Liste des Etats du poste Palettisation	Plage (... - ...)	
Liste des Etats du poste Remplissage	Plage (... - ...)	
<ajouter>		

Entrées dans la liste de textes

Par dé..	Valeur	Texte
0	ARRET	Etablissement des conditions d'initialisation
1		Mode INITIALISATION
2		Conditions après initialisation remplies
3		Mode AUTO
4		Mode MANUEL
5		Poste Bouchage en Arrêt d'Urgence
6		<ajouter>

Liste des Etats du poste Bouchage [Liste de textes]

Propriétés Info Diagnostic

La visibilité est dynamisée sur une plage de 0 à 7.

Ligne_TPCyberV7 > PC-System_1 [SIMATIC PC station] > HMI_RT_7 [WinCC RT Advanced] > Vues > Vue de l'état des postes

CHAGE : ARRET

ETIQUETTAGE : Arrêt

TTISATION : Arrêt

Attributs Animations Evénements Textes

Propriétés Info Diagnostic

Champ d'E/S symbolique_3 [Champ d'E/S symbolique]

Général

Processus

- Variable : EtatsIntBouchage
- Variable API : EtatsIntBouchage
- Adresse : Int
- Numéro bit : 0

Contenu

- Liste de textes : Liste des Etats du post
- Entrées visibles : 7

Mode

- Mode : Sortie

A 1.4 « Etats Etiquetage »

A 1.4.1 Recherche d'éléments

Dans la vue Accueil de l'IHM, le champ d'état est positionné sur Arrêt d'urgence. Il est lié à une liste qui recopie identiquement celle du poste Mélange, effectivement présente dans les listes de texte de cet IHM.

Malheureusement, ce champ n'est pas relié à une variable de l'API.

Listes de textes			
...	Nom	Sélection	Commentaire
1-	Defauts production	Plage (... - ...)	
1-	Etat Cuve	Plage (... - ...)	
1-	Etat Machine	Plage (... - ...) ▾	
1-	Etat Pressostat	Plage (... - ...)	
1-	Etat production	Plage (... - ...)	
1-	ListeMarquage	Plage (... - ...)	
1-	Noms_Vues	Plage (... - ...)	
	<ajouter>		

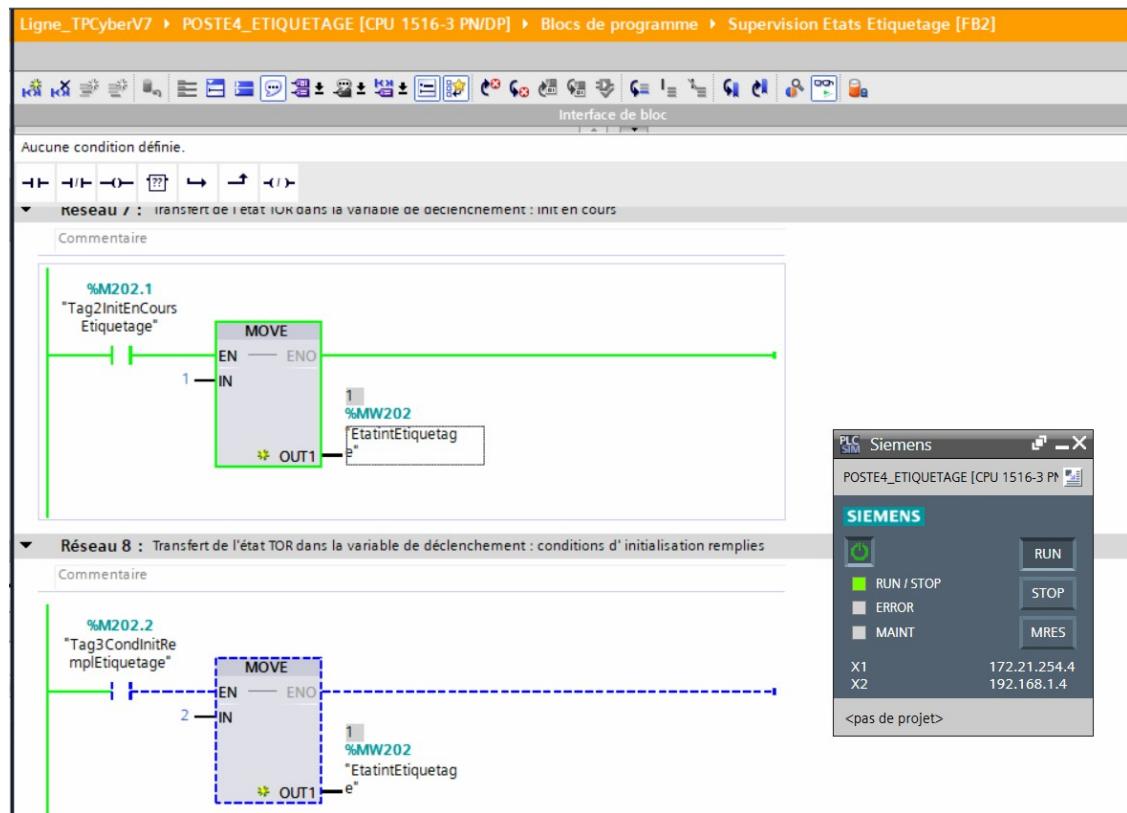
Entrées dans la liste de textes		
...	Par dé..	Valeur
1-	0	ETAT POSTE : ARRET D'URGENCE
1-	1	ETAT POSTE : PREPARATION POUR REMISE EN ROUTE
1-	2	ETAT POSTE : ARRET DANS L'ETAT INITIAL
1-	3	ETAT POSTE : MISE P.O. DANS L'ETAT INITIAL
1-	4	ETAT POSTE : MARCHE DE PREPARATION
1-	5	ETAT POSTE : PRODUCTION
1-	6	ETAT POSTE : MARCHE DE CLOTURE
1-	7	ETAT POSTE : PRODUCTION TOUT DE MEME
1-	8	ETAT POSTE : ARRET DEMANDE FIN DE CYCLE
1-	9	ETAT POSTE : MARCHE DE VERIFICATION DANS LE DE
1-	10	ETAT POSTE : MARCHE DE VERIFICATION DANS L'ORD
1-	11	ETAT POSTE : MARCHES DE TEST
	<ajouter>	

A 1.4.2 Etablissement dans l'API du sous-programme de transfert

Il faut donc comme précédemment constituer un sous-programme de transfert, ce qui est réalisé ici à partir de 5 variables de l'API : MEM_BP_ARRET en %M90.1, INIT_EN_COURS en %M85.1, CONDITIONS INITIALES en %M1.2, BIT DE MARCHE en %M84.0 et ARU en %I3.7

Le sous-programme FB2 placé dans l'OB 1 de l'API réalise la composition des variables TOR et les place sur le mot MW202.

Les réseaux 7 et 8 sont présentés ici en simulation avec la variable correspondant à « Initialisation en cours » forcée à 1 en plaçant une bobine inverse en % M202.1 dans le réseau 2.



A 1.4.3 Etablissement du champ d'affichage dans la vue IHM.

Une liste de 5 états est placée dans le champ de l'état du poste Etiquetage de la vue de l'IHM de Supervision.

Ligne_TPCyberV7 > PC-System_1 [SIMATIC PC station] > HMI_RT_7 [WinCC RT Advanced] > Listes de textes et de graphiques

... Nom	Sélection	Commentaire
Liste des Etats du poste Bouchage	Plage (... - ...)	
Liste des Etats du poste Etiquetage	Plage (... - ...)	Semblable à la liste "Etat machin"
Liste des Etats du poste Mélange	Plage (... - ...)	
Liste des Etats du poste Palettisation	Plage (... - ...)	
Liste des Etats du poste Remplissage	Plage (... - ...)	
<ajouter>		

Entrées dans la liste de textes

... Par dé..	Valeur	Texte
0	Arrêt	
1	Initialisation en cours	
2	Conditions initiales remplies	
3	Production en service	
4	Le poste Etiquetage est en arrêt d'urgence	

Ligne_TPCyberV7 > PC-System_1 [SIMATIC PC station] > HMI_RT_7 [WinCC RT Advanced] > Vues > Vue de l'état des postes

Champ d'E/S symbolique_4 [Champ d'E/S symbolique]

Attributs Animations Événements Textes

Général

Processus

- Variable : EtatIntEtiquetage
- Variable API : EtatIntEtiquetage
- Adresse : Int
- Numéro bit : 0

Mode

- Mode : Sortie

Contenu

- Liste de textes : Liste des Etats du poste Etiquetage
- Entrées visibles : 5

La liste est liée à la variable EtatInt Etiquetage.

En simulation, l'écran affiche bien « Initialisation en cours » correspondant à la valeur forcée dans l'API (valeur 1 dans le champ de contrôle).

22/04/2021 14:05:35

Supervision Ligne de Production

Quitter

Etats des postes de travail

Retour à l'accueil

POSTE MELAN	ARRÊT D'URGENCE	#:
POSTE REMPLISSAGE	ARRÊT	#:
POSTE BOUCHAGE	ARRET	#:
POSTE ETIQUETTE	Initialisation en cours	1
POSTE PALETTISAT	Arrêt	#:

PLC Siemens

POSTE4_ETIQUETAGE [CPU 1516-3 P]

SIEMENS

RUN / STOP

ERROR

MAINT

RUN

STOP

MRES

X1 172.21.254.4

X2 192.168.1.4

<pas de projet>

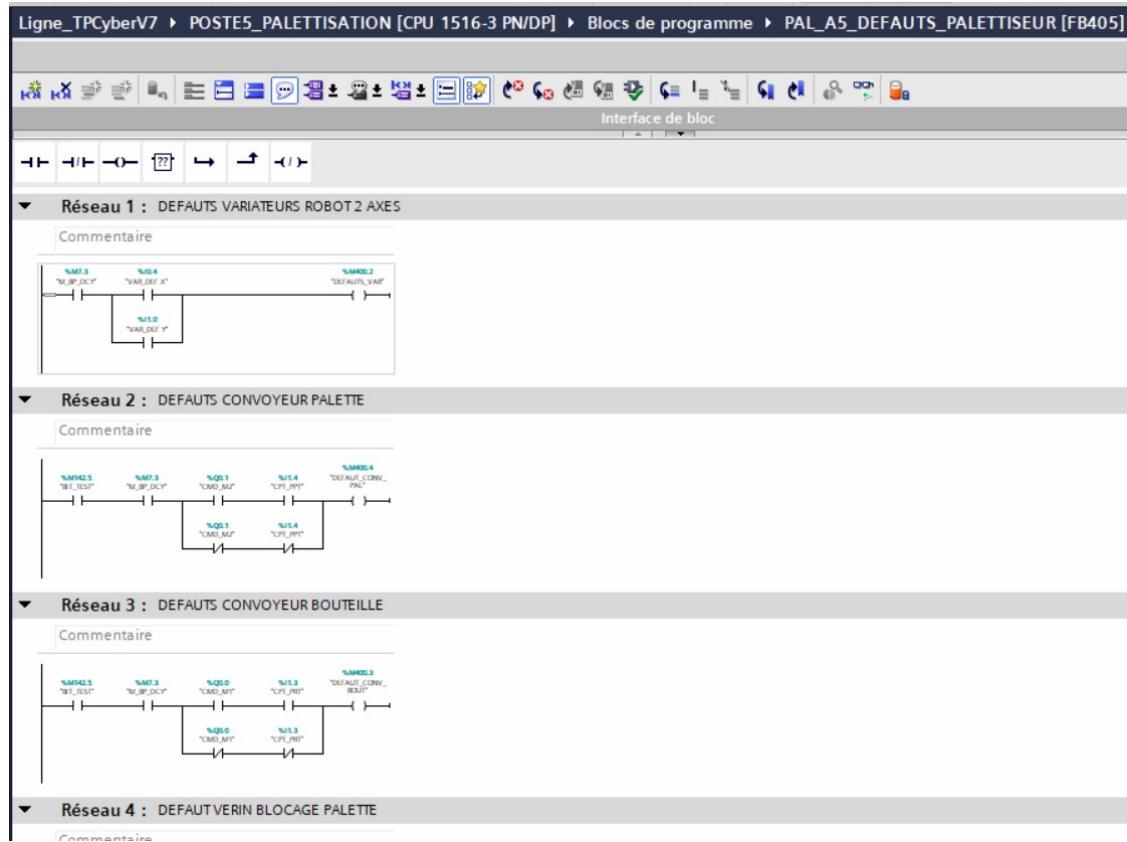
Annexe 2

Réalisation des alarmes « Palettisation », « Bouchage » et « Etiquetage ».

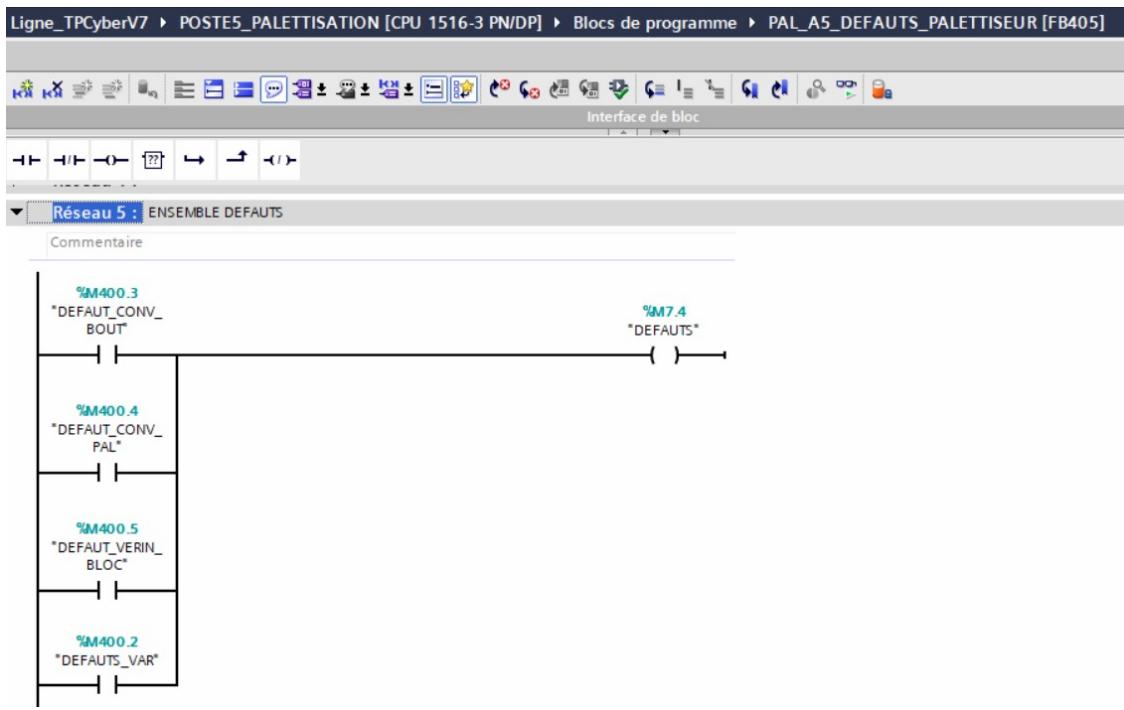
A.2.1.1 Alarmes de bits du poste Palettisation

A.2.1.1.1 Recherche d'éléments

Dans l'API, des défauts sont construits par le sous-programme Pal_A5_DEFAUTSPALETTEUR renvoyé dans l'OB[1] sur des variables TOR. Ces variables sont rangées dans le MW400, préparant son utilisation comme variable de déclenchement d'alarmes.



Tous ces défauts sont aussi ramenés sur une variable **TOR** en %M7.4 pour traduire globalement que le poste rencontre un/des défauts.



Dans l'IHM, neuf bits d'alarme sont reportés, déclenchés par la variable DEFAUTS (1). Deux fenêtres sont prévues dans la vue globale, l'une avec la colonne texte, l'autre sans.

Ligne_TPCyberV7 > IHM_POSTE5_PALETTISATION [TP700 Comfort] > Alarmes IHM								
	Alarmes de bit	Alarmes analogiques	Alarmes de l'API	Alarmes système	Classes d'alarmes			
Alarmes de bit								
ID	Nom	Texte d'alarme	Classe d'alarme	Variable de déclenchement	Bit de déclenchement	Adresse de déclenchement	Vari...	
2	CONV_ARRIVE_PAL...	MOTEUR OU CAPTEUR ARRIVE PALETTE HS	No Acknowle...	DEFAUTS(1)	4	DEFAUTS_AL...	<8 U	
3	CAPTEUR PRT	LE CAPTEUR ESTHS	Warnings	DEFAUTS(1)	7	DEFAUTS_AL...	<8 U	
4	CAPTEUR PPT	LA CAPTEUR PPT ESTHS	Warnings	DEFAUTS(1)	6	DEFAUTS_AL...	<8 U	
5	DEFAUT_VERIN	DEFAUT SORTI VERIN	No Acknowle...	DEFAUTS(1)	5	DEFAUTS_AL...	<8 U	
6	PREHENSEUR	PREHENSEUR HS	No Acknowle...	DEFAUTS(1)	8	DEFAUTS_AL...	<8 U	
8	DEFAUT_ARU		Warnings	DEFAUTS(1)	1	DEFAUTS_AL...	<8 U	
7	CONV_BOUTEILLE	MOTEUR OU CAPTEUR TAPIS BOUTEILLE HS	No Acknowle...	DEFAUTS(1)	3	DEFAUTS_AL...	<8 U	
9	DEFAUT_ALIMEN		No Acknowle...	DEFAUTS(1)	11	DEFAUTS_AL...	<8 U	
10	VAR_DEF_X	VARIATEUR X OU Y DDEFAUTS	Warnings	DEFAUTS(1)	2	DEFAUTS_AL...	<8 U	
<ajouter>								

La variable DEFAUTS (1) est liée à la variable « DEFAUTS_ALARMS » de l'API (%MW200). Mais cette variable de l'API, vide dans la table d'affectation (il n'y a pas de bits d'alarme dedans), n'est reliée à rien. De plus, les 4 défauts programmés sur le M400 ne sont pas pris en compte. Cette programmation n'est pas exploitable pour la supervision.

A.2.1.1.2 Etablissement de l'alarme dans l'IHM de supervision

Pour programmer l'IHM de supervision, je renvoie l'arrêt d'urgence et les 4 défauts programmés dans l'API sur une variable de déclenchement VarAlarmePalettisation en %MW402 (programmation des 4 réseaux simples de la FB[2] placée dans l'OB[1]).

Ces défauts sont repris en alarmes de bits dans la fonction Alarmes de l'IHM de supervision en y liant la variable IHM et la variable API.

The screenshot shows the WinCC RT Advanced software interface with the following details:

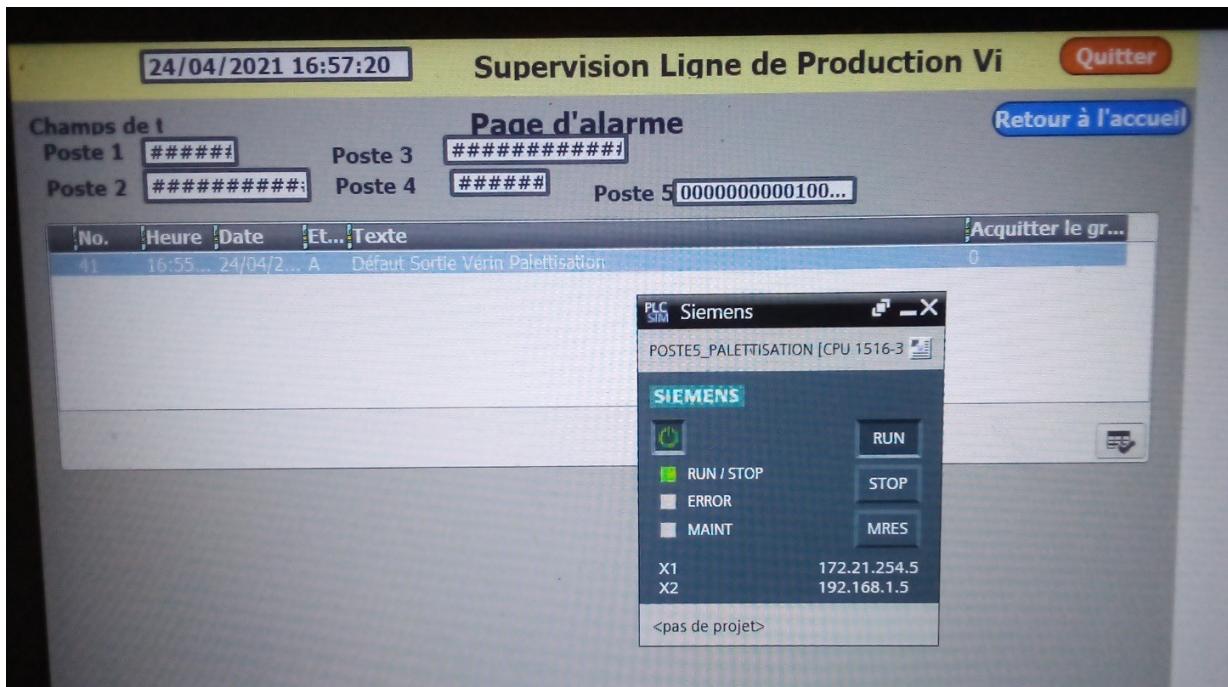
- Top Bar:** Ligne_TPCyberV7 > PC-System_1 [SIMATIC PC station] > HMI_RT_7 [WinCC RT Advanced] > Variables IHM > Alarms SuperV [5]
- Alarms SuperV Table:**

Nom	Type de données	Connexion	Nom API	Variable API	Adresse
Alarmes Bouchage	Int	HMI_Liaison_8	POSTE3_BOUCHAGE	DEFAULT	
Alarmes Etiquetage	Int	HMI_Liaison_9	POSTE4_ETIQUETAGE	Liste_défauts	
Alarmes Mélange	Word	HMI_Liaison_6	POSTE1_MELANGE	Var_AlarmMelange	
Alarmes Remplissage	Word	HMI_Liaison_7	POSTE2_REMPLISSAGE	VarAlarmesRemplissage	
Alarmes Palettisation	Word	HMI_Liaison... []	POSTE5_PALETTISATION	VarAlarmePalettisation []	
<ajouter>					
- Alarms de bit Table:**

ID	Nom	Texte d'alarme	Classe d'... Errors	Variable de déclenc... Alarmes Palettisation 1	Bit ... VarAlarmePalettisation...<aucune vari...	Adresse de déclenche... VarAlarmePalettisation...<aucune vari...	Variable d'ac... <aucune vari...
38	ARU Palettisation	ARU Palettisation	Errors	Alarmes Palettisation 1	VarAlarmePalettisation...<aucune vari...	VarAlarmePalettisation...<aucune vari...	
40	DéfautvariateurRobotPal	Défaut variateur robot Palettisation	Errors	Alarmes Palettisation 2	VarAlarmePalettisation...<aucune vari...	VarAlarmePalettisation...<aucune vari...	
41	DéfautConvBouteillePal	Défaut convoyeur bouteille Palettisation	Errors	Alarmes Palettisation 3	VarAlarmePalettisation...<aucune vari...	VarAlarmePalettisation...<aucune vari...	
42	DéfautConvPalettePal	Défaut convoyeur palette Palettisation	Errors	Alarmes Palettisation 4	VarAlarmePalettisation...<aucune vari...	VarAlarmePalettisation...<aucune vari...	
43	DéfautVerinBlocagePal	Défaut vérin blocage Palettisation	Errors	Alarmes Palettis... []	5 []	VarAlarmePalettisation...<aucune vari... []	
44							

A.2.1.1.3 Ecran en simulation

La simulation est activée pour un défaut “Sortie vérin de blocage” des palettes qui correspond au cinquième bit forcé à 1 de la variable, visible dans le champ E/S de test.



A.2.2 Alarmes de bits du poste Bouchage

A 2.2.1 Recherche d'éléments

Dans le poste Bouchage, la programmation d'alarmes a été faite sur huit défauts (TOR) qui ont été adressée sur un mot (MW14) mais un défaut (bourrage tapis) n'a pas été programmé.

Pour la supervision, l'arrêt d'urgence est recopié en M14.7 et le bourrage tapis en M15.0.

La programmation de l'IHM de supervision est ensuite effectuée sur la variable de déclenchement %MW14.

Dans l'IHM Opérateur, trois fenêtres d'alarmes (alarmes système, en attente et non acquittées) figurent en page globale.

Remarque : à la simulation, le défaut « vérin poussoir » s'affiche en continu.

2.1.2 Etablissement de l'alarme dans l'IHM de supervision

Le réseau 1 de la FB [8] recopie le défaut « bourrage-tapis » sur le bit n°8 du mot MW14 (%M15.0).

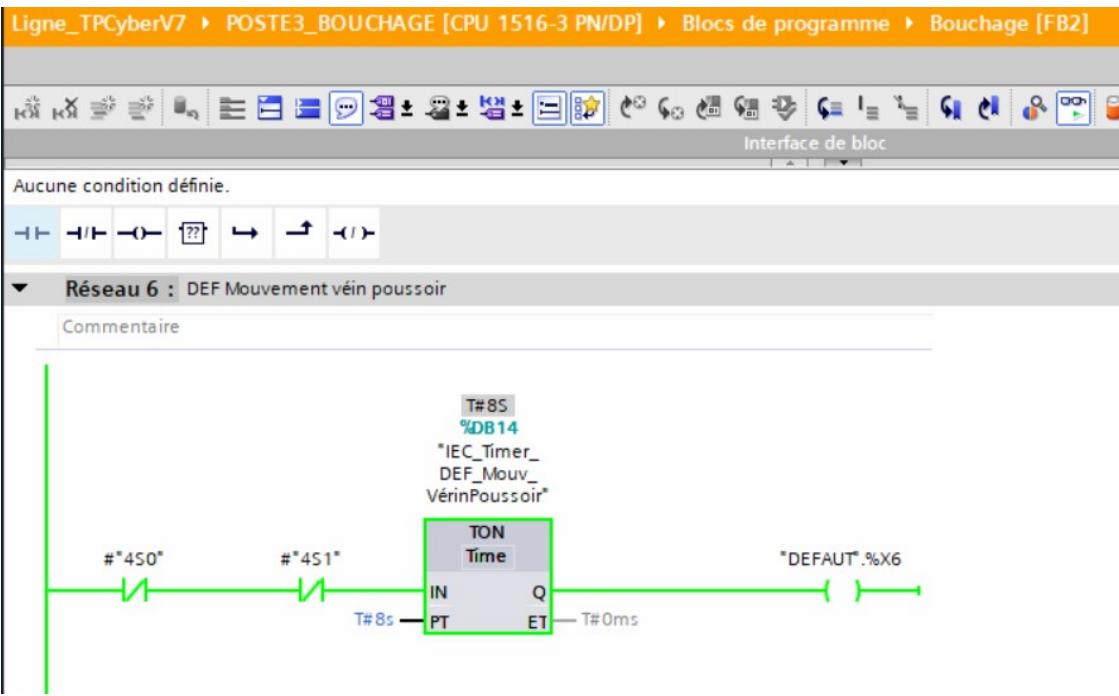
La variable Alarmes Bouchage de l'IHM de supervision est liée au MW14 du programme de l'API et reprend donc maintenant 9 défauts, tous programmés en « Errors ».

Ligne_TPCyberV7 > PC-System_1 [SIMATIC PC station] > HMI_RT_7 [WinCC RT Advanced] > Variables IHM > Alarmes SuperV [5]						
Alarmes SuperV						
Nom	Type de données	Connexion	Nom API	Variable API	Adresse	...
Alarmes Bouchage	Int	HMI_Liaison...	POSTE3_BOUCHAGE	DEFAUT
Alarmes Etiquetage	Int	HMI_Liaison_9	POSTE4_ETIQUETAGE	Liste_défa...
Alarmes Mélange	Word	HMI_Liaison_6	POSTE1_MELANGE	Var_AlarmMelange
Alarmes Palettisation	Int	HMI_Liaison_10	POSTE5_PALETTISATION	DEFAUTS_ALARMS
Alarmes Remplissage	Word	HMI_Liaison_7	POSTE2_REMPLISSAGE	VarAlarmesRemplissage
<ajouter>						

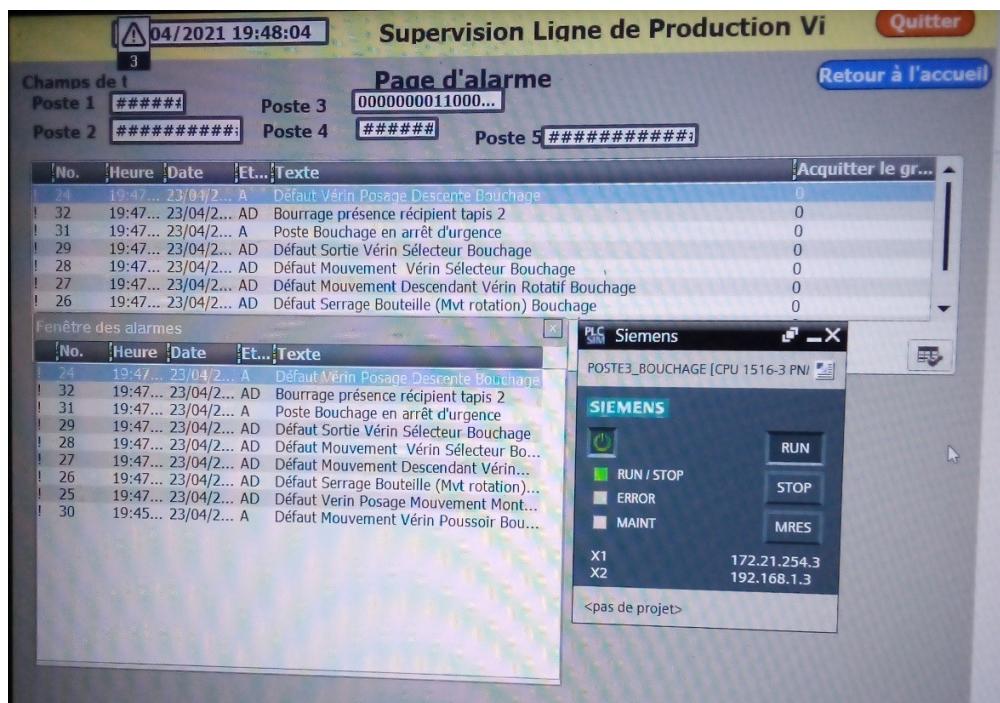
Alarmes de bit							Alarmes analogiques	
ID	Nom	Texte d'alarme	Classe d'alar...	Variable de d...	Bit de ...	Adresse de déclenchement	Variable d'ac...	Bit d'a...
24	Alarme de bit_1	Défaut Vérin Posage Descente Boucl Errors	Alarmes Bouc...	0	DEFAUTx0	<aucune vari...	0	
25	Alarme de bit_2	Défaut Vérin Posage Mouvement Mo Errors	Alarmes Bouc...	1	DEFAUTx1	<aucune vari...	0	
26	Alarme de bit_3	Défaut Serrage Bouteille (Mot rotatio Errors	Alarmes Bouc...	2	DEFAUTx2	<aucune vari...	0	
27	Alarme de bit_4	Défaut Mouvement Descendant Vén Errors	Alarmes Bouc...	3	DEFAUTx3	<aucune vari...	0	
28	Alarme de bit_5	Défaut Mouvement Vén Sélecteur I Errors	Alarmes Bouc...	4	DEFAUTx4	<aucune vari...	0	
29	Alarme de bit_6	Défaut Sortie Vérin Sélecteur Boucha Errors	Alarmes Bouc...	5	DEFAUTx5	<aucune vari...	0	
30	Alarme de bit_7	Défaut Mouvement Vén Poussoir Bc Errors	Alarmes Bouc...	6	DEFAUTx6	<aucune vari...	0	
31	Alarme de bit_8	Poste Bouchage en arrêt d'urgence Errors	Alarmes Bouc...	7	DEFAUTx7	<aucune vari...	0	
32	Alarme de bit_9	Bourrage présence récipient tapis 2 Errors	Alarmes Bouc...	8	DEFAUTx8	<aucune vari...	0	
<ajouter>								

La simulation est activée en forçant la variable bourrage tapis 2 dans le réseau 1 de la FB[8].

La page d'alarme affiche bien cette alarme. Elle affiche aussi celle du mouvement vérin car cette alarme est due à la programmation initiale de la FB[2] du poste Bouchage, comme le montre la vue de la simulation.



Sur l'écran de la page d'alarmes en simulation, le champ de test permet de forcer les alarmes et on voit leur acquittement par l'API (sauf les deux défauts présents à son niveau).



A 2.2 Alarmes de bits du Poste Etiquetage

A.2.2.1 Recherche d'éléments

La table de variables montre 5 défauts dont :

- 4 variables TOR « Défaut_marquage » %MW200.1, « Défaut_verins » %MW200.2, « Défaut Alimentation Bidons » en %M200.3, et « ALARME BOURRAGE » en %M200.4. Ces variables qui traduisent des défauts sont créées par le bloc ETI_Defauts FB[300].
- et un mot MW200 « Liste_défaits » sur lequel ils sont rangés.

L'arrêt d'urgence n'a pas été renvoyé sur le MW200.

L'IHM Opérateur reprend le MW200 en variable de déclenchement et les défauts en bits d'alarme et une fenêtre d'alarme est prévue en vue globale. Mais il n'y a pas de d'écran d'alarmes, et, forcées à la simulation, les sorties TOR n'ont pas déclenché pas d'alarmes sur le pupitre opérateur.

A.2.2.2 Etablissement de l'alarme dans l'IHM de supervision

Pour la supervision, j'ajoute l'arrêt d'urgence sur le mot MW200 en transférant dans le programme de l'API, l'arrêt d'urgence (ARU %I3.7) à l'adresse %M200.7 par un sous-programme simple FB[1] placé dans l'OB1 de l'API. Dans l'IHM, la variable de déclenchement d'alarme est liée à cette variable.

Ligne_TPCyberV7 > PC-System_1 [SIMATIC PC station] > HMI_RT_7 [WinCC RT Advanced] > Variables IHM > Alarmes SuperV [5]

Alarmes SuperV

Nom	Type de données	Connexion	Nom API	Variable API	Adresse
Alarmes Bouchage	Int	HMI_Liaison_8	POSTE3_BOUCHAGE	DEFAUT	
Alarmes Etiquetage	Int	HMI_Liaison_...	POSTE4_ETIQUETAGE	Liste_défauts	
Alarmes Mélange	Word	HMI_Liaison_6	POSTE1_MELANGE	Var_AlarmMelange	
Alarmes Palettisation	Int	HMI_Liaison_10	POSTE5_PALETTISATION	DEFAUTS_ALARMS	
Alarmes Remplissage	Word	HMI_Liaison_7	POSTE2_REMPLISSAGE	VarAlarmesRemplissage	
<ajouter>					

Alarmes de bit		Alarmes analogiques		Variables d'archive			
ID	Nom	Texte d'alarme	Classe d'alar...	Variable de décle...	Bit d...	Adresse de déclen...	Variable d'ac...
33	defaut_marquage	Défaut Marquage Etiquetage	Warnings	Alarmes Etiquetage 2	Liste_défauts.x2	<aucune vari...	
34	defaut_verin	Défaut Vérin Etiquetage	Warnings	Alarmes Etiquetage 3	Liste_défauts.x3	<aucune vari...	
35	defaut_bidon	Défaut Alimentation Bidon Etiquetage	Warnings	Alarmes Etiquetage 4	Liste_défauts.x4	<aucune vari...	
36	bourrage	Défaut Bourrage Etiquetage	Errors	Alarmes Etiquetage 5	Liste_défauts.x5	<aucune vari...	
37	AlarmeARUEtiqueta...	Poste Etiquetage en arrêt d'urgence	Errors	Alarmes Etiquetage 7	Liste_défauts.x7	<aucune vari...	
<ajouter>							

Les quatre premiers défauts sont programmés en « warnings », les deux suivants en « Errors ».