

الجمهورية الجزائرية الديمقراطية الشعبية  
République Algérienne démocratique et populaire

وزارة التعليم العالي والبحث العلمي  
Ministère de l'enseignement supérieur et de la recherche scientifique

جامعة سعد حطاب البلدية  
Université SAAD DAHLAB de BLIDA

كلية التكنولوجيا  
Faculté de Technologie

قسم الإلكترونيك  
Département d'Électronique



## Mémoire de Projet de Fin d'Études

présenté par

Bounab Issam

&

Achlaf Hakim

Pour l'obtention du diplôme de Master en Électronique option Signaux en  
Ingénierie des Système et Informatique Industrielle (SISII)

---

Thème

---

# Automatisation du process N.E.P dans l'entreprise Trèfle

---

Proposé par : BENNILA Norddine

Année Universitaire 2014-2015

## Remerciements

---

Au nom du tout puissant et son salut sur le prophète  
Louange à dieu le miséricordieux de nous illuminer le chemin du savoir et  
Nous avoir Aidés

Nous tenons à exprimer nos vifs remerciements à notre promoteur  
Mr. BENNILA NORDDINE pour nous avoir encadrés durant notre projet de fin  
D'études et nous conseillé tout le long de notre travail.

Nous remercions chaleureusement les travailleurs de l'entreprise Trefle , qui nous  
ont ouvert leurs portes et donner l'opportunité de réaliser ce projet et plus  
particulièrement  
Mr HADJ SAID LARBI, et à tous ceux qui nous ont aidé durant notre  
Stage pratique.

Nos sincères remerciements à notre Ami Misseraoui Ishak qui nous ont conseillé et  
Éclairé sur notre travail.

Nous remercions les membres du jury qui nous ont fait l'honneur  
D'examiner notre modeste travail.

## Dédicace

Je dédie ce travail à  
Mes très chers parents,  
Mes très chères sœurs,  
Mon très cher frère,  
En particulier ma grand-mère  
Mon binôme qui m'a supporté toute ces années,  
Tous mes amis et collègues d'études,  
Et à tous ceux qui me sont chers.

ISSAM

Je dédie ce travail à  
Mes très chers parents,  
Mes très chères sœurs,  
Mes très chers frères,  
Mon binôme qui m'a supporté toute ces années,  
Tous mes amis et collègues d'études,  
Et à tous ceux qui me sont chers.

HAKIM

---

**ملخص:** البحث الذي قمنا به خلال الفترة من أجل تحسين محطة تنظيف المستخدمة من طرف الشركة ترافل، ولهذا استخدمنا البرنامج STEP 7 المتوفرة من قبل شركة سيمنز، STEP 7 يحتوي على جهاز محاكاة PIC سيمنس. وقد استعملناه في مشروعنا لمحاكاة برامج واستخدمنا للاتصالات في الغياب الحقيقي 'API'، أما بالنسبة للمراقبة استعملنا برنامج WinCC واجهة الرجل/آلة.

كلمات المفاتيح: محطة التنظيف، المبرمج-ليأبي، نظامالتواصل بيناتسانوالة

---

**Résumé :** Les recherches que nous avons menées durant la période de réalisation de cette thèse ont pour but d'améliorer la station de nettoyage en place N.E.P utilisé dans la société trèfle, pour cela on a utilisé le logiciel de programmation STEP7 fourni par le concepteursiemens. Le STEP7 contient un simulateur d'automate SIEMENS. Nous l'avons utilisé dans notre projet pour simuler des programmes et de faire la communication vu l'absence de l'automate réel « API » et Pour la supervision nous avons utilisé le logiciel WinCC flexible en utilisant l'interface Homme/Machine « HMI ».

**Mots clés :** N.E.P; API; HMI.

---

**Abstract:** The research we have carried out during the period of realization of this thesis is to improve the cleaning station CIP up clover used in society, for it was used STEP 7 programming software provided by Siemens home. The STEP 7 contains a SIEMENS PLC simulator. We used it in our project to simulate programs and to communication in the absence of the actual controller "PLC" for supervision and we used the WinCC software using interface Man / Machine "HMI".

**Keywords :** CIP, PLC, HMI.

---

## Listes des acronymes et abréviations

**NEP** : Nettoyage En Place.

**IHM**: Interface Homme Machine.

**API** : Automate Programmable Industrielle.

**GRAF CET** : Graphe Fonctionnel de Commande Etape Transition.

**LIST** : Le langage à instruction (machine).

**CONT** : Le langage à contact.

**LOG** : Le langage graphique.

**MPI** : L'interface multipoint.

**PROFIBUS** : Bus de terrain propriétaire.

**PT100** : Sonde de température à résistance platine.

**P** : Action Proportionnelle.

**I** : Action intégral.

**D** : Action Dérivé.

**Ta** : Constante du temps.

**Tu**: Temps de retard.

**UHT** : Upérisation Haute Température.

**Sidel** : Société industrielle des emballages légers.

**PET** : Polyéthylène téréphtalate.

**SPA** : Société par action.

# Table des matières

## Automatisation du process N.E.P dans l'entreprise Trèfle.

<b>Introduction générale</b> .....	1
<b>Chapitre 1</b> Présentation de l'entreprise Trèfle .....	3
1.1    Présentation :.....	3
1.2.    Mission et gamme de produit Trèfle :.....	5
1.2.1.    Mission et activités de Trèfle : .....	5
1.2.2.    La gamme de produits :.....	6
1.3.    Organisation :.....	6
1.4.    Laproductionjournalière : .....	7
1.5.    Equipement de l'entreprise Trèfle : .....	8
1.5.1.    Machines de conditionnement : .....	8
1.5.2.    Les cuves : .....	9
1.5.3.    STATION DE NETTOYAGE (NEP) :.....	10
1.6.    Conclusion :.....	11
<b>Chapitre 2</b> Principe de fonctionnement de nettoyage en place.....	12
1.3    Organisation.....	7
c.1    Vanne en position "fermée" :.....	18
c.2    Vanne en position "ouverte" :.....	19
3.2.4    Traitement du programme automate:.....	34
d.1    Bloc système pour fonctions standard et fonctions système:.....	39
D.2    Editeur mnémoniques:.....	40
d.2    Différents types de variables contenues dans STEP7:.....	41
D.4    Langage de programmation:.....	42
D.5    Simulateur S7-PLCSIM: .....	43
3.5.1    CPU 313C-2DP : .....	53
3.5.2    . Les modules.....	54
D.1    L'appel des blocs de sous-programme: .....	74
D.2    Etape de démarrage cycle EC : .....	75
D.3    D'autre opération qu'on a utilisée dans notre programme :.....	76
a    . Introduction .....	79
Méthode de Ziegler-Nichols : .....	80
d.1    . Méthode de la réponse indicielle :.....	80

D.2	Méthode du point critique :	81
e.1	. Méthode de réglabilité.....	82
4.2	Supervision sous WINCC FLEXIBLE :	89
4.2.2	Intégration du projet STEP7 :	90
<b>Chapitre 3</b>	<b>Généralités sur STEP7 / WinCC flexible et les étapes de programmation</b> .....	<b>32</b>
1.3	Organisation.....	7
c.1	Vanne en position "fermée" :	18
c.2	Vanne en position "ouverte" :	19
3.2.4	Traitement du programme automate:.....	34
d.1	Bloc système pour fonctions standard et fonctions système:.....	39
D.2	Editeur mnémoniques:.....	40
d.2	Différents types de variables contenues dans STEP7:.....	41
D.4	Langage de programmation:.....	42
D.5	Simulateur S7-PLCSIM: .....	43
3.5.1	CPU 313C-2DP :	53
3.5.2	. Les modules.....	54
D.1	L'appel des blocs de sous-programme: .....	74
D.2	Etape de démarrage cycle EC :	75
D.3	D'autres opérations qu'on a utilisées dans notre programme :	76
a	. Introduction .....	79
	Méthode de Ziegler-Nichols :	80
d.1	. Méthode de la réponse indicielle :	80
D.2	Méthode du point critique :	81
e.1	. Méthode de réglabilité.....	82
4.2	Supervision sous WINCC FLEXIBLE :	89
4.2.2	Intégration du projet STEP7 :	90
1.3	Organisation.....	7
c.1	Vanne en position "fermée" :	18
c.2	Vanne en position "ouverte" :	19
3.2.4	Traitement du programme automate:.....	34
d.1	Bloc système pour fonctions standard et fonctions système:.....	39
D.2	Editeur mnémoniques:.....	40
d.2	Différents types de variables contenues dans STEP7:.....	41
D.4	Langage de programmation:.....	42

D.5 Simulateur S7-PLCSIM: .....	43
3.5.1 CPU 313C-2DP : .....	53
3.5.2 . Les modules.....	54
D.1 L'appeldesblocsde sous-programme: .....	74
D.2 Etape de démarrage cycle EC : .....	75
D.3 D'autre opération qu'on a utilisée dans notre programme :.....	76
a . Introduction .....	79
Méthode de Ziegler-Nichols : .....	80
d.1 . Méthode de la réponse indicielle :.....	80
D.2 Méthode du point critique : .....	81
e.1 . Méthodede réglabilité.....	82
4.2 Supervision sous WINCC FLEXIBLE : .....	89
4.2.2 Intégration du projet STEP7 : .....	90
<b>Chapitre 4</b> Interface de supervision de la station NEP.....	89
1.3 Organisation .....	7
c.1 Vanne en position "fermée" : .....	18
c.2 Vanne en position "ouverte" : .....	19
3.2.4 Traitement du programme automate:.....	34
d.1 Bloc système pour fonctions standard et fonctions système:.....	39
D.2 Editeur mnémoniques:.....	40
d.2 Différents types de variables contenues dans STEP7:.....	41
D.4 Langage de programmation:.....	42
D.5 Simulateur S7-PLCSIM: .....	43
3.5.1 CPU 313C-2DP : .....	53
3.5.2 . Les modules.....	54
D.1 L'appeldesblocsde sous-programme: .....	74
D.2 Etape de démarrage cycle EC : .....	75
D.3 D'autre opération qu'on a utilisée dans notre programme :.....	76
a . Introduction .....	79
Méthode de Ziegler-Nichols : .....	80
d.1 . Méthode de la réponse indicielle :.....	80
D.2 Méthode du point critique : .....	81
e.1 . Méthodede réglabilité.....	82
4.2 Supervision sous WINCC FLEXIBLE : .....	89
4.2.2 Intégration du projet STEP7 : .....	90

<b>Conclusion générale</b> .....	108
<b>Annexes</b> .....	110
<b>Bibliographie</b> .....	111

## Liste des figures et tableaux

<i>Figure 1.7 : La position de l'entreprise sur Google</i> .....	4
<i>Figure 1.1:Logo de l'entreprise Trèfle</i> .....	4
<i>Figure 1.2: L'usine Trèfle</i> .....	5
<i>Tableau1.1: Gamme de produits Trèfle</i> .....	6
<i>Figure 1.3:Organigramme de l'entreprise Trèfle</i> .....	7
<i>Figure 1.4: Production journalière de Trèfle par type de conditionnement</i> .....	8
<i>Figure 1.5: Machine Arcil (IZIA4)</i> .....	9
<i>Figure 1.6: Les cuves de la station NEP</i> .....	10
<i>Figure 2.1: Turbine de pulvérisation</i> .....	14
<i>Figure 2.2:vanne d'arrêt</i> .....	16
<i>Figure 2.3:vanne de dérivation</i> .....	16
<i>Figure 2.4:Vanne Delta SW4</i> .....	17
<i>Figure 2.5:Vanne Delta DA3+</i> .....	18
<i>Figure 2.6:Vanne en position « fermée »</i> .....	19
<i>Figure 2.7:Vanne en position « ouverte »</i> .....	20
<i>Figure 2.8:Vanne modulante</i> .....	21
<i>Figure 2.9:Capteur de température PT100</i> .....	22
<i>Figure 2.10: Capteur de niveau</i> .....	23
<i>Figure 2.11:Schéma d'un Débitmètre Promass 83</i> .....	23
<i>Figure 2.12: Débitmètre Promass E</i> .....	24
<i>Figure 2.13: Conductivimètre</i> .....	25
<i>Figure 2.14: Clapet anti-retour</i> .....	25
<i>Figure 2.15:L'échangeur tubulaire</i> .....	26
<i>Figure 2.16:Le régulateur de température (EUROTHERM 3216)</i> .....	27



Figure 2.17: Variateur de vitesse. ....	27
Figure 2.18: Schéma d'un Purgeur de vapeur. ....	28
Figure 2.19: Purgeur de vapeur. ....	28
Figure 2.20: Pompe de pression. ....	29
Figure 2.21: représentation graphique de la station de nettoyage. ....	30
Figure 2.22: fenêtre de commande. ....	30
Figure 3.1: Architecture d'un API. ....	34
Figure 3.2: TRAITEMENT API. ....	35
Figure 3.3: Logiciel STEP7. ....	36
Figure 3.4: Configuration du matériel. ....	37
Figure 3.5: Structure du programme STEP7. ....	38
Figure 3.6: Fenêtre de programmation. ....	39
Figure 3.7: table de Mnémoniques. ....	41
Tableau 3.1 : Types de variables utilisés dans Step7. ....	41
Figure 3.8: langages de programmation STEP 7 . ....	42
Figure 3.9: langage graph. ....	43
Figure 3.10: Interface de simulation PLCSIM. ....	44
Figure 3.11: création d'un projet avec l'assistant du WinCC flexible. ....	46
Figure 3.12: la liaison entre WinCC flexible et l'automate. ....	47
Figure 3.13: bibliothèques de WinCC flexible pour réaliser un projet. ....	47
Figure 3.14: Vue d'ensemble du progiciel WinCC flexible. ....	49
Figure 3.15: table des variables. ....	50
Figure 3.16: Intégration du projet Step7 dans WinCC flexible. ....	51
Figure 3.17: Liaison avec l'automate S7300. ....	51
Figure 3.18: Organigramme de l'application. ....	55
Figure 3.19: Le SIMATIC Manager. ....	56
Figure 3.20: configuration de matériels. ....	57
Figure 3.21: configuration de la sortie analogique. ....	57
Figure 3.22: adresses du module DI16/DO16. ....	58
Figure 3.23: adresses du 1er module AI8xRTD. ....	59
Figure 3.24: adresses du 2eme module AI8xRTD. ....	59
Figure 3.25: adresses du 3eme module AI8xRTD. ....	60
Figure 3.26: Bloc d'organisation et blocs de fonctions et blocs fonctionnels et blocs de données. ....	63
Figure 3.27: quelques mnémoniques d'entrées et de sorties. ....	65
Figure 3.28: S7-PLCSIM1 en Mode Stop. ....	67
Figure 3.29: simple GRAFCET de Préparation CUVE EAU PROPRE (EP). ....	67
Figure 3.30: le Bloc FB22. ....	68
Figure 3.31: Conversion de l'étape X1 de GRAFCET en langage Contact. ....	68
Figure 3.32: Conversion de l'étape X2 de GRAFCET en langage Contact. ....	69
Figure 3.33: le Bloc Fonction "SCALE" FC105. ....	70
Figure 3.34: Chargement de fonction FC105. ....	71
Figure 3.35: Fonction de comparaison. ....	71
Figure 3.36: fonction de comparaison. ....	72
Figure 3.37: Fonction de comparaison. ....	72
Figure 3.38: fonction de comparaison. ....	73
Figure 3.39: Alarme de niveau de la Cuve EP. ....	73
Figure 3.40: pas d'alarme le niveau est supérieur à 14%. ....	74
Figure 3.41: appel de FB22 dans OB1. ....	74
Figure 3.42: appel de bloc FB1. ....	75

Figure 3.43: réseau 1 de bloc FB1. ....	76
Figure 3.44:l'opération ROUND et l'opération DI_R. ....	77
Figure 3.45:Pause de la LIGNE 3. ....	77
Figure 3.46:l'appel des blocs FB et FC dans OB1. ....	78
Figure 3.47: temporisation sous forme de retard à la montée. ....	79
Tableau 3.2: Paramètres PID obtenus à partir d'une réponse indicielle (ZNt). ....	80
Tableau 3.3: Paramètres PID obtenus à partir du point critique (ZNf). ....	81
Figure 3.48.Echangeur. ....	81
Figure 3.49 : Schéma fonctionnel de l'installation. ....	82
Tableau 3.4. Les mesures des températures en C° par rapport au temps.....	83
Tableau 3.5. Les mesures des températures en % par rapport au temps. ....	83
Figure 3.50. Réponse indicielle du système enregistré expérimentalement. ....	84
Figure 3.51:utilisation de bloc FB41. ....	87
Figure 3.52:opération de transfert.....	87
Figure 4.1 : le choix de pupitre. ....	90
Figure 4.2 : Intégration du projet Step7 dans WinCC flexible. ....	90
Figure 4.3 : Liaison avec l'automate S7300.....	91
Figure 4.4 : Exemple des variables utilisés dans WINCC flexible. ....	92
Figure 4.5:vue principale de la station NEP. ....	93
Figure 4.6 : vue opérateur.....	93
Figure 4.7 : vue ingénieur.....	94
Figure 4.8 vue des alarmes. ....	94
Figure 4.9.vue des capteurs. ....	95
Figure 4.10.vue des pompes. ....	96
Figure 4.11.vue des vannes. ....	96
Figure 4.12 : état des cuves.....	97
Figure 4.13.l'etat actuel de la station. ....	98
Figure 4.14.vue temps d'envoi. ....	99
Figure 4.15 : vue station NEP. ....	100
Figure 4.16.vue des machines et cuves pour la ligne 1. ....	100
Figure 4.17.vue source. ....	101
Figure 4.18.vue chauffage. ....	101
Figure 4.19.choisir le régulateur. ....	102
Figure 4.20.Vue régulation de température. ....	102
Figure 4.21.choisir nettoyage ou préparation. ....	103
Figure 4.22.choisir la cuve a préparée. ....	103
Figure 4.23.préparer la cuve eau chaude.....	104
Figure 4.24.préparation d'eau chaude en cours. ....	104
Figure 4.25.le matériels à nettoyer ou nettoyage de station NEP. ....	105
Figure 4.26.le matériels à nettoyer. ....	105
Figure 4.27.nettoyer la cuve 3 avec le cycle court à chaud.....	106
Figure 4.28. La ligne 2 en marche. ....	106





# Introduction générale

---

Le développement massif de techniques de l'automatisme a permis le passage de la machine automatisée à celui de systèmes automatisés de production, qui gèrent l'alimentation en énergie et qui permettent d'avoir une meilleure qualité des produits en plus de la sécurité et de la flexibilité des processus, mais cela entraîne un accroissement des besoins, en particulier la manipulation d'un grand nombre de variables et la gestion de véritables flux de communication.

Un automate programmable est un système électronique, destiné à être utilisé dans un environnement industriel, il utilise une mémoire programmable pour le stockage interne des instructions orientées utilisateur aux fins de mise en œuvre de fonctions spécifiques, telles que des fonctions de logique, de mise en séquence, de temporisation, de comptage et de calcul arithmétique, pour commander au moyen d'entrées et de sorties divers types de machines ou de processus.

La station NEP fait partie de l'équipement des stations d'entreprise Trèfle, elle est contrôlée par l'automate programmable APV, l'inconvénient de logiciel « APV » est l'absence de disponibilité de ce logiciel qui permet de simuler des solutions programmables proposées par les ingénieurs pour modifier ou améliorer le déroulement des différentes séquences, pour cela on a essayé d'utiliser le logiciel de programmation STEP7 fourni par le constructeur SIEMENS, dans le but d'éliminer les problématiques suivantes :

- Remplacer les boutons de choix de cycles et lignes dans l'armoire par un pupitre digital.
- Remplacer le régulateur externe par un autre qui est interne dans les modules

d'API.

- L'ajout des nouvelles alarmes dans le but de renforcer la sécurité.
- Introduire un programme qui sert à afficher le temps restant de nettoyage.
- Introduire un programme qui sert à mettre la station NEP en mode Pause et de reprendre le fonctionnement si il y a une fuite ou des dégâts dans les matériels.

Le STEP 7 contient un simulateur d'automate SIEMENS tel que le

S7-300. Nous l'avons utilisé dans notre projet pour simuler des programmes et faire la communication vu l'absence de l'automate réel. Pour la supervision nous avons utilisé le logiciel WinCC flexible.

Notre travail est réparti en 4 chapitres, le premier décrit et localise d'une manière générale l'entreprise Trefle et ces activités au niveau de marché algérien, le deuxième chapitre contient la présentation et la description de la station NEP et leurs composants ainsi les méthodes de nettoyage et de préparation utilisées avec le choix des cycles, le troisième chapitre présente le logiciel de programmation STEP 7 et de supervision WinCC flexible et aussi les étapes qu'on a utilisées pour créer notre programme et dans le dernier chapitre on présente l'application développée.

# Chapitre 1 Présentation de l'entreprise Trèfle

---

## 1.1 Présentation :

Trèfle est une **entreprise agroalimentaire** algérienne ; (Voir la [figure 1.1](#)),

Dont le siège social est implanté sur **la zone industrielle, site 1, Ben Boulaid, Blida.** (Voir [figure 1.7](#)),

Depuis sa création **en 1983** par **Mr TLEMSANI**, Trèfle n'a pas cessé d'évoluer son entreprise, il emploie **598** personnes,

Elle acquiert le statut **d'EURL** (Entreprise Unipersonnelle à Responsabilité Limitée)

(Voir la [figure 1.2](#)),

S'est lancé dans la production du yaourt brassé avec une capacité de **3500 pots/heure.**

**En 1990**, L'entreprise **Trèfle** avait eu l'acquisition d'une nouvelle conditionneuse de capacité **6500 pots/heure**, en utilisant le même process. Puis, la même année, L'entreprise **Trèfle** avait eu l'acquisition d'une chaîne de fromagerie (pâte molle et pâte pressée).

Après une période de stagnation due à la situation économique et sociale en Algérie, L'entreprise **Trèfle** a acquis **en Avril 1998** sa première ligne de conditionnement en yaourt étuvé de capacité de production **12 500 pots/heure.**

**Au mois de septembre** de la même année, **l'entreprise** avait eu l'acquisition d'une deuxième ligne de conditionnement en crème dessert et yaourt aux fruits

**En 2000**, L'entreprise **Trèfle** a acquis la troisième ligne de conditionnement en yaourt étuvé de capacité **12500 pots/heure.**

C'est **en 2001** qu'il y a eu lancement du nouveau complexe, avec transfert des équipements initiaux et avait eu l'acquisition d'une quatrième ligne de production en yaourt étuvé, de capacité **40 000 pots/heure**, le tout est alimenté par un atelier moderne de process APV, entièrement automatisé, portant la capacité totale de production à **77 500 pots/heure.**

**En 2002**, L'entreprise **Trèfle** a renforcé l'unité par deux nouvelles lignes de conditionnement pour la production du yaourt brassé et des fromages frais ainsi qu'une ligne SIDEL pour les produits frais et **UHT** (Upérisation Haute Température) en bouteilles avec une capacité de **120.000 bouteilles/jour.**

Puis, **en Décembre 2003**, L'entreprise **Trèfle** avait eu l'acquisition d'une septième ligne de conditionnement de capacité **40 000 pots/heure** en yaourt étuvé et crème

dessert.

**L'entreprise Trèfle** n'a cessé de se développer pour répondre à la demande, en lançant, en **septembre 2004**, l'acquisition d'une nouvelle unité de conditionnement en **bouteilles PET**, de produits frais, de capacité **22 000 bouteilles/heure**.

**En 2007**, L'entreprise **Trèfle** acquiert un statut de **SPA**, sa capacité de production est de **200 000L/jour**.



Figure 1.7 : La position de l'entreprise sur Google.



Figure 1.1: Logo de l'entreprise Trèfle.





Figure 1.2: L'usine Trèfle.

## 1.2 Mission et gamme de produit Trèfle :

### 1.2.1 Mission et activités de Trèfle :

La mission de la laiterie **Trèfle** est la production et la commercialisation d'une gamme de produits (**Yaourt crème dessert, jus à base ou non de lait, lait UHT, lait caillé, fromage à base de pâte molle, citronnade,...**). De plus, l'entreprise a l'exclusivité de la commercialisation des produits «**Ferrero Rocher**» que l'entreprise importe. Ses activités principales sont donc :

1. La production de produits laitiers et de produits connexes (**Jus de fruits, citronnade,...**),
2. La commercialisation des produits **Trèfle**.
3. L'importation de la poudre de lait.
4. L'importation et la commercialisation des produits «**Ferrero Rocher**».
5. Le transport et la distribution des produits **Trèfle** ou autres.
6. L'exportation des produits **Trèfle**.

## 1.2.2 La gamme de produits :

Laiterie Trèfle propose une large gamme de produits laitiers qui va des différents yaourts et crèmes dessert, au lait UHT et lait caillé au fromage fondu. L'entreprise Trèfle commercialise aussi des boissons telles que différents jus de fruits et citronnades. Le tableau suivant reprend les principaux produits que commercialise l'entreprise Trèfle (voir le [tableau 1.1](#))

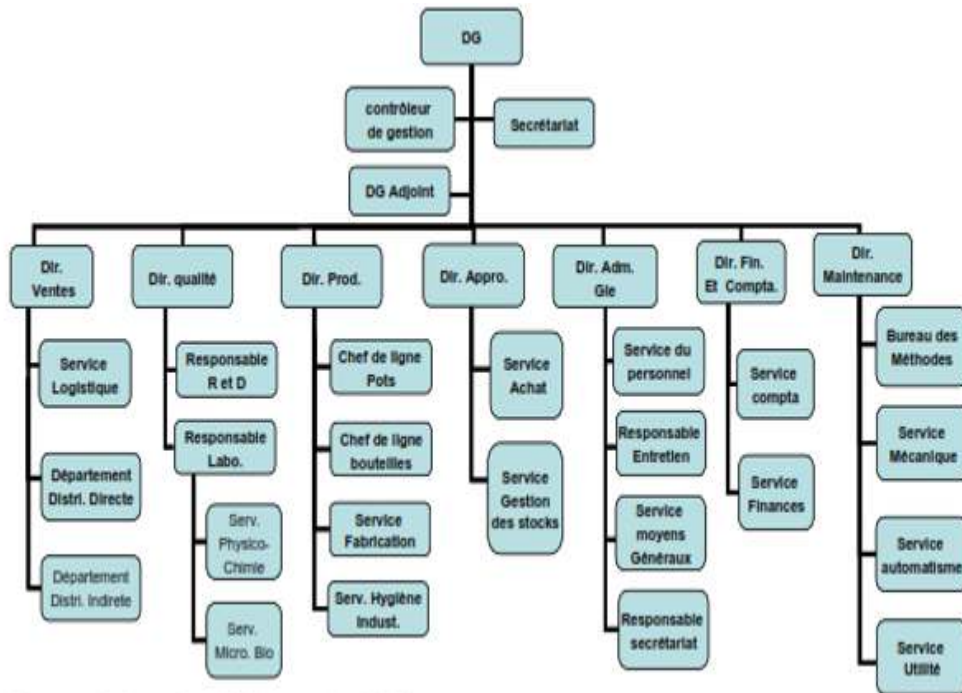
PRODUITS TREFLE	
YAOURT BANDEROLER ETUVER	110 GR
YAOURT BANDEROLER NATURE	110 GR
YAOURT LIGHT	110 GR
YAOURT Brassé aromatisé	100 GR
YAOURT. Brassé Fruité	110 GR
YAOURT. Brassé Fruité TARTISE	100 GR
YAOURT Finesse 0,4%MG	100 GR
YAOURT Brassé Mix	85 GR
YAOURT BOUTEILLE	1 LITRE
CREME DESSERT CHOCO	100 GR
CREME DESSERT CAMEL	100 GR
CREMY PISTACHE	90 GR
CREMY VANILLE	90 GR
CREMY CAMEL	90 GR
CREMY CHOCO	90 GR
LEBEN SACHET	1 LITRE
RAIB BOUTEILLE	1 LITRE
LEBEN BOUTEILLE	1 LITRE
PETIT MALIN NATURE	60 GR
PETIT MALIN AROMATISER	60 GR
LAIT BOUTEILLE	1 LITRE
CITRONADE JUS BOUTEILLE	1 LITRE
FRESH UP JUS BOUTEILLE	1 LITRE
ACTI-FORT	1/2 ET 1LITRE
BELLE DES CHAMPS FROMAGE	KG

*Source : document de présentation de l'entreprise Trèfle*

**Tableau 1.1: Gamme de produits Trèfle.**

### 1.3 Organisation :

L'entreprise est organisée en sept directions représentées par l'organigramme suivant (voir [figure 1.3](#))



Source : Présentation de l'entreprise Trèfle

Figure 1.3: Organigramme de l'entreprise Trèfle.

### 1.4 Laproductionjournalière :

Laproductionjournalière auniveaudel'entrepriseTrèfleestde200000L/J

(Tousproduitsconfondus). Sachantquelesjoursde productionsontaunombrede6joursparsemaineetquelesexpéditions'effectuenten moyenne2foisparsemaine

(Encequiconcernelespartenairesnationaux).Laquantitéàexpédier'élève doncà

$3 \times (200000) = 600000L$ , à chaquetournée.

Lesproduitsà expédiersonrépartiscommesuit(voir [figure 1.4](#))

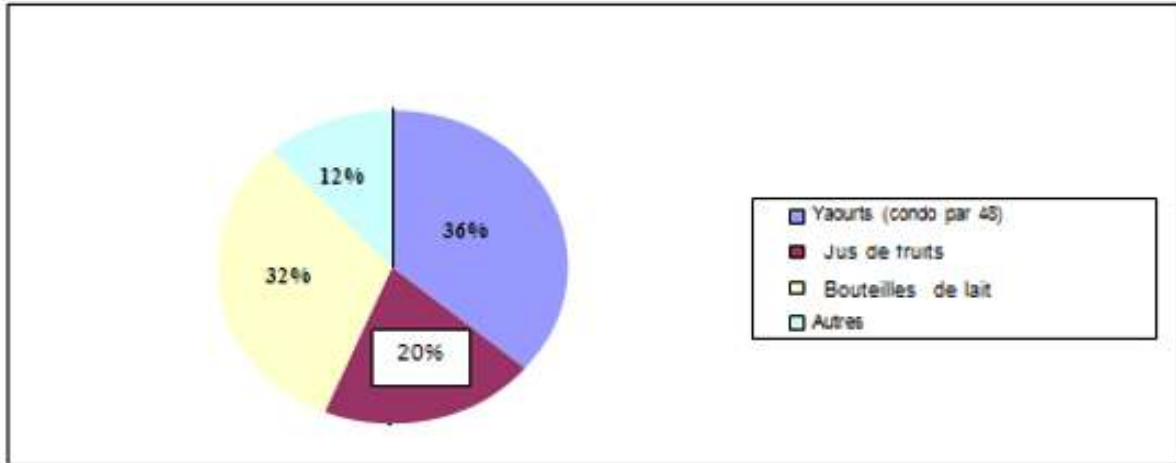


Figure 1.4: Production journalière de Trèfle par type de conditionnement.

## 1.5 Equipement de l'entreprise Trèfle :

### 1.5.1 Machines de conditionnement :

L'entreprise **Trèfle** dispose de **neuf lignes** de conditionnement qui assure les fabrications de différent produit :

1. **Deux machines "A6(24)" (Arcil)** assurant la fabrication de **40000 pots/heure**.
2. **Deux machines "A6(12)" (Arcil)** assurant la fabrication de **20000 pots/heure**.
3. **Deux machines (Sidel)** assurant la fabrication des bouteilles.
4. **Trois machines "IZIA4" (Arcil)** assurant la fabrication de **12500 pots/heure**.  
(Voir [figure 1.5](#))



Figure 1.5: Machine Arcil (IZIA4).

### 1.5.2 Les cuves :

L'entreprise dispose de plusieurs **cuves**, elles sont utilisées pour la **préparation** des produits (lait, yaourt...) et le **stockage** de la matière première et même au **nettoyage** (acide, soude...)

(Voir [figure 1.6](#))



Figure 1.6: Les cuves de la station NEP.

### 1.5.3 STATION DE NETTOYAGE (NEP) :

Le nettoyage **deséquipements** de laiterie était autrefois effectué

(Et continue à l'être en certains endroits) par du personnel armé de brosses et de **solutionsdétergentes**, Qui devait démonter le matériel et pénétrer dans les cuves pour en atteindre lesSurfaces.

Ceci était, non seulement pénible, mais également **inefficace**,

Les produits étaient souvent réinfectés par des équipements imparfaitement nettoyés.

Pour assurer un nettoyage approprié et des résultats **hygiéniques**, L'entreprise**Trèfle** a mis au

Point **des systèmes de nettoyage en place (NEP)** par circulation, adaptés aux

Différentes parties des unités de traitement.

**Les opérations de nettoyage** doivent être effectuées dans le strict respect d'une Méthode soigneusement élaborée, pour atteindre le niveau de propreté désiré. La Suite d'opérations devra donc être rigoureusement la même à chaque fois.

**Remarque :**

Il y'a d'autre matériels utilisés dans l'usine comme (**palettiseur, étiqueteuse.....**)

## **1.6 Conclusion :**

Dans ce chapitre on a présenté les différentes activités de l'entreprise **Trèfle**, ainsi que leur produits laitiers, puis on a cité les équipements industriel, et notamment la station de nettoyage en place **NEP** qu'on va détailler dans le chapitre suivant.

# Chapitre 2 Principe de fonctionnement de nettoyage

## en place

---

### 2.1 Introduction :

Nettoyage en place (**NEP**) est une méthode de nettoyage des surfaces intérieures de tuyaux, des navires, et des équipements de process, sans démontage.

**Auparavant** Les systèmes fermés ont été démontés et nettoyés manuellement. L'avènement de **NEP** était un bon nouvel pour les industries qui avaient besoin de nettoyage interne fréquent de leurs processus. Les industries qui dépendent fortement des **NEP** sont celles qui exigent des niveaux élevés de l'hygiène, et comprennent les produits laitiers, des boissons, de la brasserie, les aliments transformés, les produits pharmaceutiques et cosmétiques.

L'avantage pour les industries utilisatrices **NEP** est que le nettoyage est plus rapide, moins de main-d'œuvre et plus reproductible, il pose moins de risque d'exposition chimique aux personnes et prolonger la durée de vie des équipements.

### 2.2 Méthode de nettoyage de la station NEP:

Nettoyage en place signifie le fait de circuler l'eau de rinçage et les solutions détergentes dans les cuves, tuyauteries et lignes de traitement sans avoir à démonter le matériel.

**Le principe** de Nettoyage en place c'est qu'on va brancher sur la machine

(**Ex** : tank, tuyauteries, échangeur à plaque) un appareil qui va injecter dans les tuyauteries de l'eau sous pression suivant un cycle de nettoyage.

Certaine **NEP** sont intégrés aux machines (pas besoin de brancher).

Le **NEP** aussi dépend du « **TACT** »

(Température, Action mécanique, Concentration en produit, Temps de contact)

- Dans certains systèmes de **NEP** on récupère (recyclage) les « lessives » (eau+détergent) dans un tank (centrale de **NEP**) pour le cycle suivant.
- La puissance des pompes et la vitesse de circulation de l'eau déterminent l'action mécanique.



- La conception de la machine aussi est essentielle : raccords entre les tubes, vidange complète, surfaces inox lisses (ni coin ni recoin) : difficile pour pompes, plaques, tanks...
- Le système doit inclure des contrôles du niveau des liquides dans les tanks, de la concentration en détergent, de la température de l'eau (systèmes d'alarme).
- On doit vérifier manuellement au moins une fois par mois la qualité du nettoyage (démontage).
- Au moment de Nettoyage En Place il faut qu'on respecte les principes du « **TACT** »
- **T** : On fait circuler les solutions détergentes, de désinfection en circuit fermé pour assurer le temps de contact
- **A** : on augmente le débit pour que la vitesse de circulation soit suffisante pour décrocher les souillures !
- **C** : on applique des molécules qui ont un rôle adapté à l'étape : **NaOH** pour éliminer les souillures organiques. Acide phosphorique (présent dans la solution désinfectante appelé ici « CIP Steril Acid ») qui joue 2 rôles :

Rôle de détartrage (élimination des souillures minérales), et rôle antiseptique (destruction des  $\mu.o.$ ).

- **T** : Les T°C des solutions doivent être respectées pour une meilleure efficacité.

Le passage d'un courant de liquides à vitesse élevée sur les surfaces du matériel a un effet décapant mécanique qui déloge les dépôts de souillures.

Ceci vaut uniquement pour l'écoulement dans les tuyauteries, échangeurs de chaleur, pompes, vannes etc.

La technique courante de nettoyage des cuves de grandes dimensions consiste à pulvériser le détergent sur les surfaces supérieures et à le laisser couler jusqu'au bas des parois.

L'effet décapant mécanique est souvent, dans ce cas, insuffisant, mais on peut l'améliorer, dans une certaine mesure, en utilisant des systèmes de pulvérisation spécifiquement conçus, dont un exemple est illustré sur la figure (**Voir [figure2.1...](#)**)

Le nettoyage des cuves exige d'importants volumes de détergent, qui devront circuler rapidement.

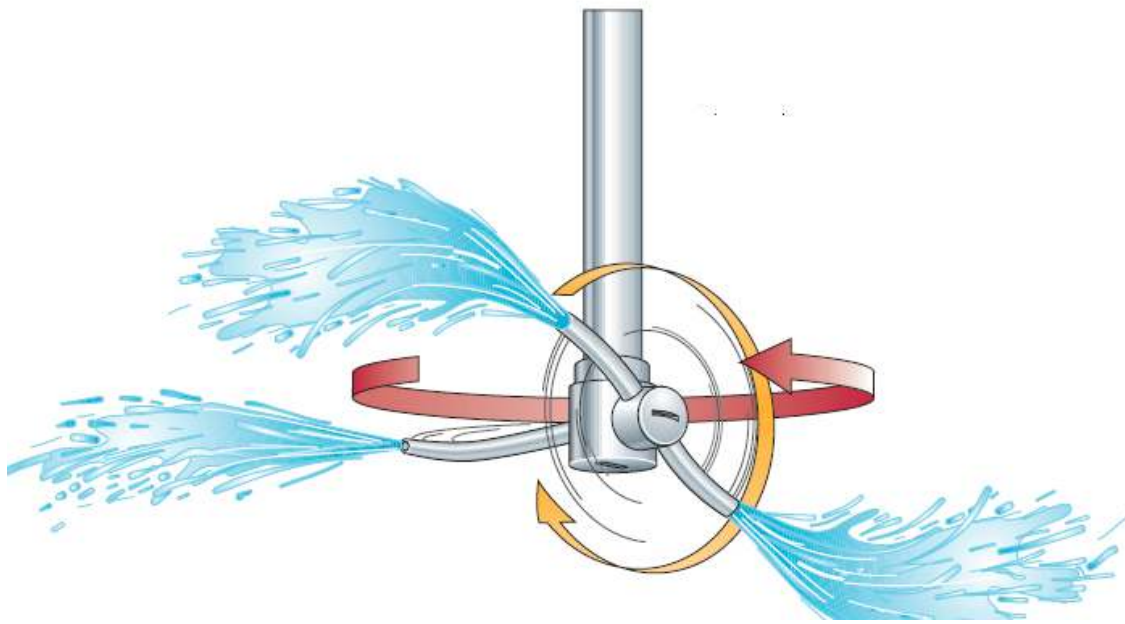


Figure 2.1: Turbine de pulvérisation.

## 2.3 DESCRIPTION DU NETTOYAGE - CYCLES :

La station de nettoyage dispose de **trois** lignes d'envois, chaque ligne est reliée avec deux machines et deux cuves.

Avant chaque démarrage de la station de nettoyage, il faut choisir la ligne sur laquelle on va envoyer les différentes solutions de nettoyage et la machine ou la cuve à nettoyer avec le cycle de nettoyage choisi.

### 2.3.1 Les cycles de nettoyage :

Il existe en tout **six** cycles différents :

1. **COURT (rinçage final à froid)** = envoi d'abord **l'eau récupérée** pendant un temps bien défini, **ensuite** l'envoi de **la soude** pendant une durée précise,  **finalement** l'envoi de **l'eau propre** pendant un temps bien défini.
2. **COURT (rinçage final à chaud)** = envoi d'abord **l'eau récupérée** pendant une durée précise, **ensuite** l'envoi de **la soude** pendant un temps bien défini,  **finalement** **eau chaude** pendant une durée précise.
3. **LONG (rinçage final à froid)** = envoi d'abord **l'eau récupérée** pendant un temps bien défini, **ensuite** l'envoi de **la soude** pendant une durée précise,  **puis** l'envoi de **l'eau propre** pendant un temps bien défini,  **et ensuite** l'envoi de **l'acide** pendant une durée précise,  **finalement** l'envoi de **l'eau propre** pendant un temps bien défini.
4. **LONG (rinçage final à chaud)** = envoi d'abord **l'eau récupérée** pendant un temps bien défini, **ensuite** l'envoi de **la soude** pendant une durée précise,  **puis** l'envoi de **l'eau propre** pendant un temps bien défini,  **et ensuite** l'envoi de **l'acide** pendant une durée précise,  **finalement** l'envoi de **l'eau chaude** pendant un temps bien défini.

5. **RINCAGE A FROID** =envoi seulement de l'**eau propre** pendant un temps bien défini.  
6. **RINCAGE A CHAUD** =envoi seulement de l'**eau chaude** pendant un temps bien défini.

## 2.4 Matériels utilisés :

La station NEP dispose des composants suivants:

### 2.4.1 Automate programmable :

La station **NEP** est programmée par un **PC Intégral** de marque **APV**,  
Dans l'**armoire** de la station **NEP** se trouve des **modules entrées/sorties** et module de **communication** pour envoyer les informations des **capteurs et actionneurs** au **PC Intégral** par un **profibus**.

### 2.4.2 Les électrovannes :

Une électrovanne est un dispositif commandé électriquement, permettant d'autoriser ou d'interrompre par une action mécanique la circulation d'un fluide.

Il existe deux types d'électrovannes : tout ou rien (**TOR**) et proportionnelle (**Vanne modulante**).

#### *a Les tous ou rien (TOR):*

Ce sont des vannes qui peuvent prendre seulement **deux** positions, soit complètement **fermées** ou complètement **ouvertes**.

#### *b Vanne Delta SW4:*

Les vannes d'arrêt et de dérivation **Delta SW4** sont conçues pour être utilisées dans les domaines de l'industrie des brasseries et des boissons, mais aussi pour l'industrie des produits laitiers et autres produits alimentaires ainsi que pour les industries chimiques et pharmaceutiques.

La fonction principale des vannes **Delta SW4** est de fermer (**arrêter**) et de changer la direction du flux par dérivation. Pour une exploitation comprenant un vérin pneumatique avec raccordement d'air est commandé par **ressort**.

Les pièces intérieures du vérin ne demandent aucun entretien.

Le nettoyage interne de la vanne se fait lors du nettoyage **NEP**.

Pour le contrôle d'une vanne de type **pneumatique**, une unité de contrôle est installée sur la partie supérieure du **vérin**.

Les diodes électroluminescentes jaunes sur l'unité de contrôle indiquent la position de la vanne.

En alternative à l'unité de contrôle, **un support** pour les détecteurs de proximité équipé de diodes électroluminescentes pour indiquer la position de la vanne peut être monté sur le vérin.

(Voir [figure 2.2](#) et [figure 2.3](#) et [figure 2.4](#) ...)

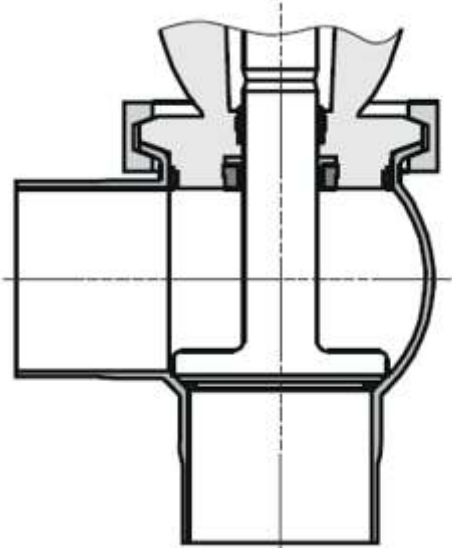


Figure 2.2:vanne d'arrêt.

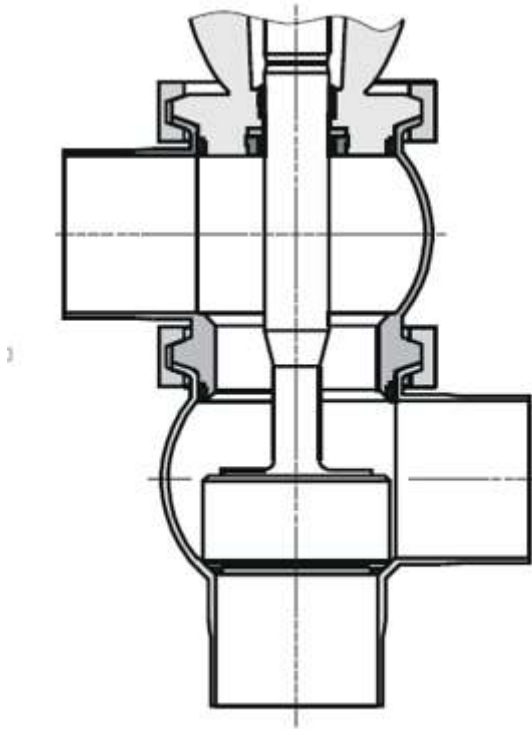


Figure 2.3:vanne de dérivation.



Figure 2.4:Vanne Delta SW4.

**c Vanne Delta DA3+:**

Une vanne à double siège ultra-hygiéniques moderne, conçu pour répondre aux exigences les plus extrêmes de l'industrie de transformation d'hygiène dans le monde entier.

Elle permet le passage de produits différents dans les deux corps de la vanne, une chambre de sécurité entre les deux corps exclu le mélange des différents produits.

(Voir [figure 2.5](#) ...)



Figure 2.5:Vanne Delta DA3+.

### **C.1 VANNE EN POSITION "FERMEE" :**

Les clapets inférieur et supérieur sont fermés par la force du ressort et isolent en toute sécurité les deux circuits **A** et **B**.

La chambre de fuite **L** qui les sépare permet l'évacuation libre et dépressurisée de liquides vers le bas. Les clapets sont équilibrés et ainsi protégés contre des coups de pression.

(Voir [figure 2.6](#) ...)

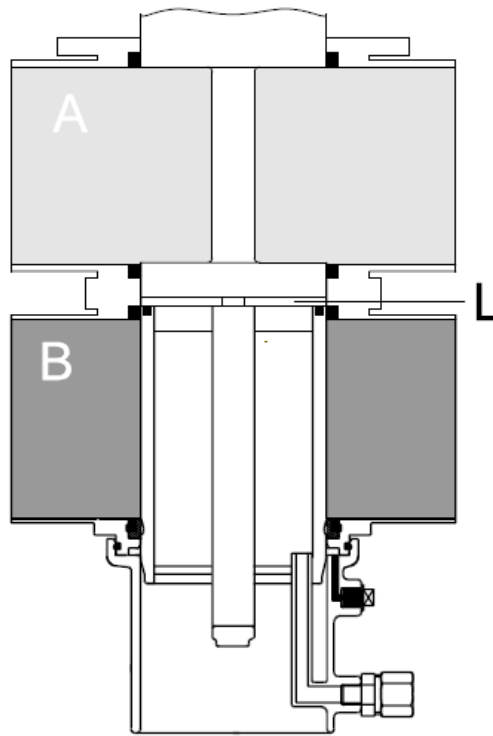


Figure 2.6:Vanne en position « fermée ».

### ***C.2 VANNE EN POSITION "OUVERTE" :***

En commandant l'actionneur, le clapet supérieur vient pousser le joint du siège inférieur. Ainsi la chambre de fuite L est fermée par rapport à la chambre de produit.

Les deux clapets meurent vers le bas à la position ouverte mettant en communication les deux circuits A et B. (Voir [figure 2.7](#) ...)

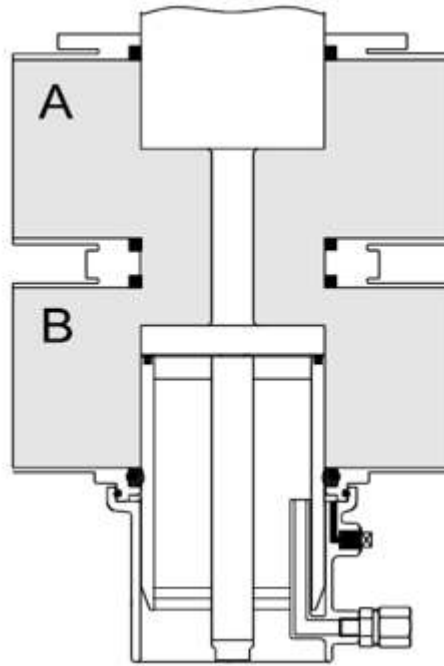


Figure 2.7:Vanne en position « ouverte ».

***d Vannes modulante:***

La vanne de régulation est une soupape utilisée pour contrôler des conditions telles que le débit, la pression, la température et le niveau de liquide en totalité ou en partie d'ouverture ou de fermeture en réponse à des signaux reçus à partir des contrôleurs qui comparent un «**point de consigne**» à une «**variable de processus**» dont la valeur est fournie par des capteurs qui surveillent les changements dans ces conditions. la vanne de régulation est située avant l'échangeur , la boucle de courant utilisée est de **4-20 mA**.  
(Voir [figure 2.8...](#))





Figure 2.8:Vanne modulante.

### 2.4.3 Les capteurs:

#### *a Capteur de température Thermo-résistances Pt 100 (Omnigrad TST) :*

La sonde de température est un dispositif permettant de transformer l'effet du réchauffement en **signal électrique**, elle est équipée d'une **thermorésistante** qui constitue l'élément sensible. **PT** représente le mot platine qui est le matériel principal de **la sonde**, le model utilisé varie sur une plage de **-200c°à +400c°**.

(Voir [figure 2.9...](#))



Figure 2.9:Capteur de température PT100.

***b Capteur de niveau (sonde à tige 11371):***

**La sonde 11371** est utilisée pour la détection de niveau dans les cuves de produits alimentaires liquides, **par ex** : lait, jus de fruit. C'est un détecteur de **niveau conductor**. Grâce à ses matériaux anticorrosion pour **la tige** et l'isolation et à la compatibilité, la sonde est une solution idéale pour **l'industrie agroalimentaire**.  
(Voir [figure 2.10...](#))



Figure 2.10: Capteur de niveau.

#### 2.4.4 Débitmètre (promass 83):

**Promass 83** est un appareil destiné à mesurer le **débit** d'un liquide, sa technique de mesure repose sur le principe de **la force de Coriolis**.

La force de Coriolis dépend de la masse déplacée  $m$ , de sa vitesse  $\Delta V$  dans le système, Lorsque le débit est **nul**, c'est à dire qu'il n'y a pas d'écoulement, les deux tubes oscillent en **Phase (1)**, Lorsqu'il y a un débit massique l'oscillation des tubes est temporisée à **l'entrée (2)** et accélérées à la **sortie (3)**.

(Voir [figure 2.11...](#))

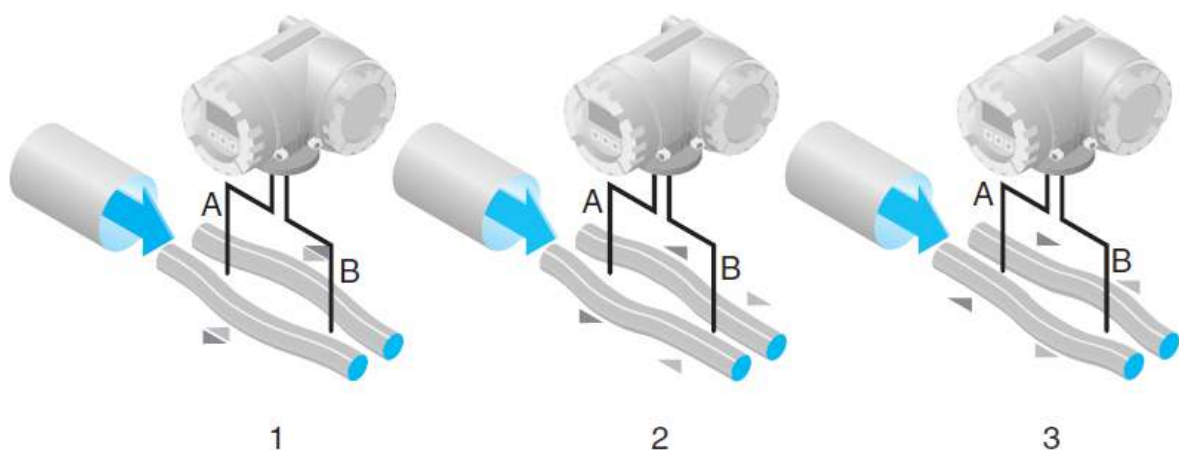


Figure 2.11:Schéma d'un Débitmètre Promass 83.

**Le débitmètre** se retrouve dans chaque ligne de retour, il est utilisé comme un dispositif de **sécurité** pour s'assurer qu'il n'y a pas de **fuites**.(Voir [figure 2.12...](#))



Figure 2.12: Débitmètre Promass E.

#### 2.4.5 Conductivimètre (CLM 253):

Un appareil de mesure de **la conductivité** électrique dans une solution, elle est située dans les cuves de la soude et acide pour vérifier leurs **concentration**, Elle est aussi utilisée dans les lignes de retour, pour contrôler la concentration de l'acide et la soude, si elle est mauvaise la solution s'envoie directement à **l'égout** sinon elle retourne à la **cuve**.(Voir [figure 2.13...](#))



Figure 2.13: Conductivimètre.

#### 2.4.6 Le clapet anti retour:

Un clapet anti-retour est un dispositif qui veiller à ce qu'un liquide se écoule à travers un tuyau dans la bonne direction, et bloque le flux si celui-ci venait à s'inverser.

(Voir [figure 2.14...](#))



Figure 2.14: Clapet anti-retour.

### 2.4.7 L'échangeur tubulaire:

L'échangeur tubulaire est un appareil qui permet de transférer la chaleur entre deux fluides ayant des températures différentes, les deux fluides ne sont pas en contact direct, et le transfert s'effectue à travers une surface d'échange. Au sein de la paroi séparatrice, le mécanisme de transmission de la chaleur est la conduction, et sur chacune des deux surfaces de contact avec les fluides, ce sont presque toujours les phénomènes de convection qui prédominent

(Voir [figure 2.15...](#))



Figure 2.15:L'échangeur tubulaire.

### 2.4.8 Le régulateur de température:

La régulation de température est un système automatique qui permet de maintenir la température de la soude, l'acide et l'eau à une valeur précise.

Régulateur travaille en autonome, c'est lui qui commande la vanne modulante et envoie les alarmes au PC quand la température est très basse ou très haute.

(Voir [figure 2.16...](#))



Figure 2.16:Le régulateur de température (EUROTHERM 3216).

#### 2.4.9 Variateur de vitesse:

Un variateur de vitesse est un équipement permettant de faire varier la vitesse d'un moteur en faisant varier la fréquence et la tension du moteur. (Voir [figure 2.17...](#))

Le variateur de vitesse est utilisé au niveau des pompes d'envoi.



Figure 2.17:Variateur de vitesse.

### 2.4.10 Purgeur de vapeur:

Un purgeur de vapeur est un organe de l'échangeur, c'est un type de vanne automatique qui évacue le condensat (c.-à-d. la vapeur d'eau condensée) sans permettre à la vapeur de s'échapper.

Dans le secteur industriel, la vapeur d'eau est souvent utilisée pour chauffer.

(Voir [figure 2.18](#) et [figure 2.19...](#))

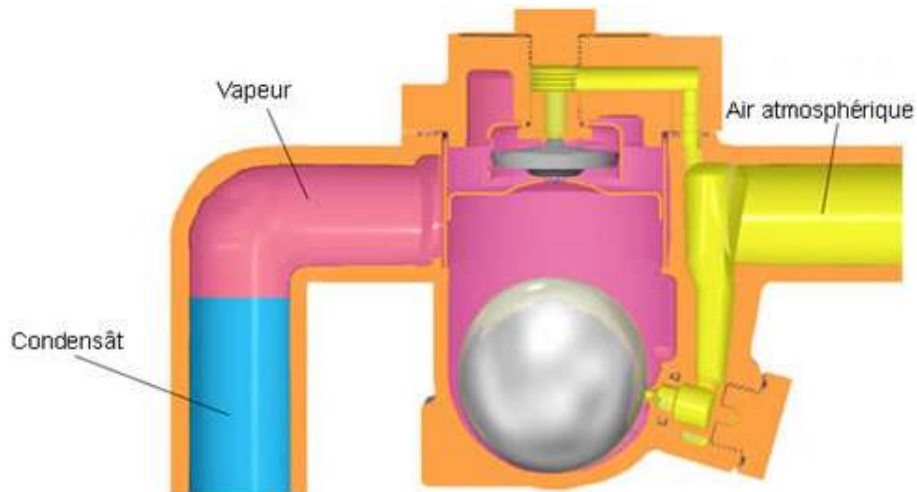


Figure 2.18: Schéma d'un Purgeur de vapeur.



Figure 2.19: Purgeur de vapeur.



#### 2.4.11 Pompe de pression (APV ZMS-4):

La pompe est un moteur triphasé à cage d'écureuil protégé contre les projections d'eau, Toutes les parties de la pompe en contact avec le produit sont en acier Inoxydable antiacide. La pompe est dotée d'une roue montée Directement sur l'arbre du moteur.

Elle est destinée au nettoyage par Circulation de détergents chimiques, le débit peut arriver jusqu'à **70.000 litres/heure**. (Voir [figure 2.20...](#))



Figure 2.20:Pompe de pression.

#### 2.4.12 Systèmes de contrôle industriel(Superviseur):

Systèmes de contrôle industriel est un système informatique qui surveille et contrôle la station de nettoyage.

Le Protocol de communication entre le PC Intégral et l'écran de superviseur se fait par le Profibus. (Voir [figure 2.21](#) et [figure 2.22...](#))

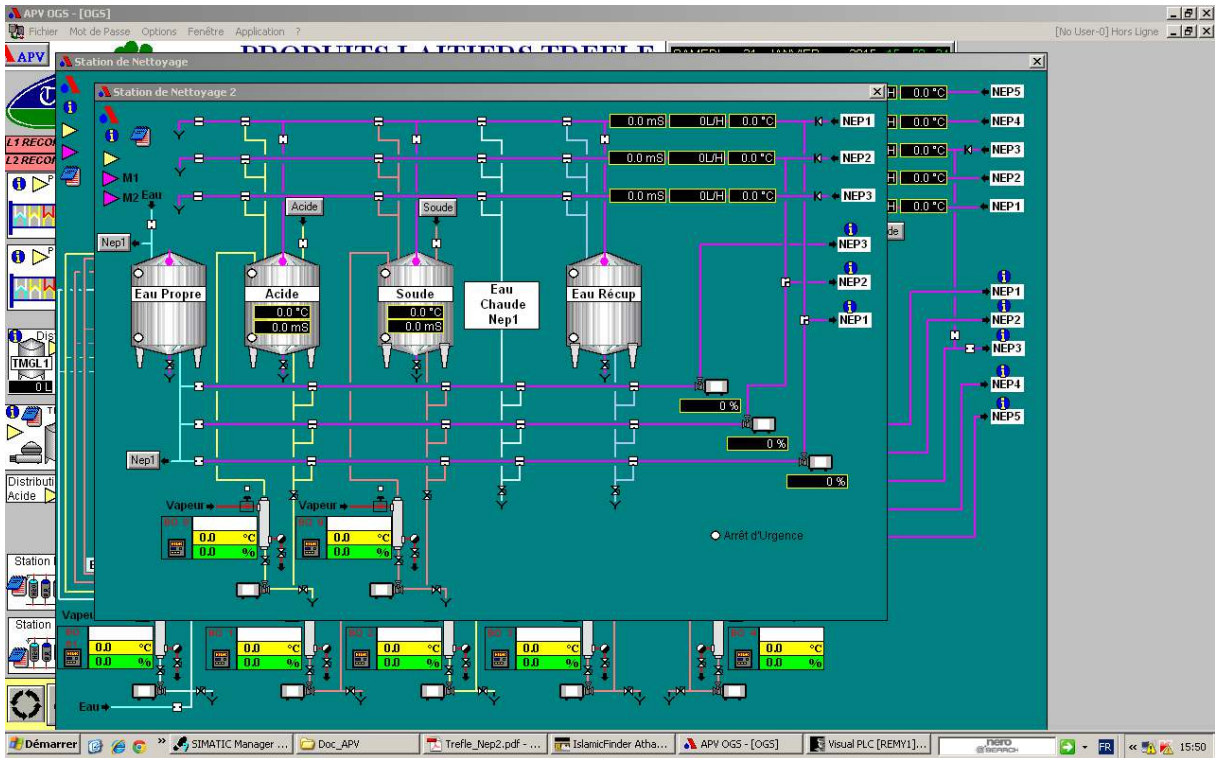


Figure 2.21:représentation graphique de la station de nettoyage.

Cette figure est pour visualiser en temps réel l'état actuel des capteurs et actionneurs.

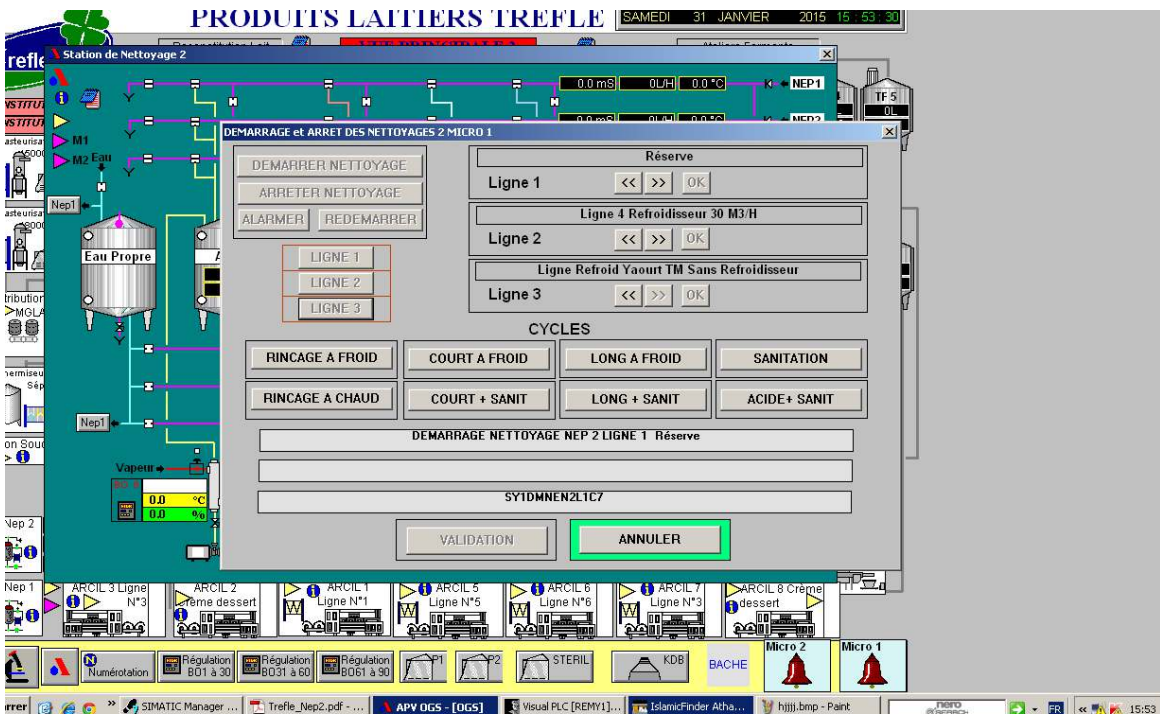


Figure 2.22:fenêtre de commande.

Cette figure est pour choisir la ligne et la machine à nettoyer avec son cycle.

## 2.5 Conclusion :

Dans ce chapitre on a expliqué les différentes Méthode de Nettoyage de la station **NEP**, ainsi que leurs cycles choisis, puis on a cité les équipements qui sont utilisés pour le nettoyage, et notamment les capteurs et les actionneurs.

Dans le chapitre suivant on va expliquer les différents logiciels utilisés et les méthodes de programmation.

# Chapitre 3 Généralités sur STEP7/WinCC flexible et les étapes de programmation

---

## 3.1 Introduction:

Ce chapitre décrit, dans une première partie, une description de l'automate, et nous présenterons les outils qu'on va utiliser pour développer notre application tels que le logiciel de programmation STEP7, le simulateur PLCSIM et le logiciel de supervision WinCC flexible, les étapes que nous avons effectuées pour programmer l'automate à recevoir les données, ainsi que la programmation. Dans la deuxième partie contient notre programmation que nous avons faite sur STEP7 et qui dépend de chaque type de capteurs utilisés dans la station NEP.

## 3.2 Généralités sur les automates programmables (API):

### 3.2.1 Historique:

Les premiers automates programmables ont été introduits aux USA en 1969 pour les besoins de l'industrie automobile américaine.

Le but recherché était de remplacer les armoires à relais, utilisées dans la commande des chaînes de fabrication par des équipements moins onéreux, aussi bien du point de vue du coût d'acquisition que du coût de la maintenance, et flexibles c'est -à-dire faciles à modifier, à utiliser et à entretenir.

Les automates programmables se sont répandus dans tous les secteurs de l'industrie (aéronautique, automobile, chemin de fer, électroménager.....).

Ils représentent aujourd'hui plus de la moitié du marché des équipements utilisés en automatisation, devançant ainsi de loin la technique à relais et l'électronique câblée.

Ce succès des automates programmables est dû en partie à leur facilité de mise en œuvre d'une part mais d'autre part à leur résistance à l'environnement de production.

En 1973, les communications entre les API ont été développées en utilisant ce qu'on appelle le Modbus. Il a permis à l'API de gérer à travers certaine distance des machines qu'ils contrôlaient.

### 3.2.2 Définition:

L'API (Automate Programmable Industriel) ou en anglais PLC (Programmable Logic Controller) c'est un appareil électronique destiné à la commande de processus industriels par un traitement séquentiel (Il contrôle les actionneurs grâce à un programme informatique qui traite les données d'entrée recueillies par des capteurs). Qui comporte une mémoire Programmable par un utilisateur automaticien (et non informaticien) à l'aide d'un langage adapté (Le langage List, Le langage Ladder...etc.) pour le stockage interne des instructions pour satisfaire un objectif, l'Automate Programmable Industriel permet de contrôler, coordonner et d'agir sur l'actionneur comme par exemple un robot, un bras manipulateur etc....

Alors on peut dire que l'API est utilisée Pour automatiser des processus. L'API est structurée autour d'une unité de calcul (processeur), de cartes d'entrées/sorties, de bus de communication et de modules d'interface et de commande.

### 3.2.3 Architecture d'un API:

Un API se compose donc de trois grandes parties :

- **Le processeur :**

Le microprocesseur réalise toutes les fonctions logiques ET, OU, les fonctions de temporisation, de comptage, de calcul... à partir d'un programme contenu dans sa mémoire.

- **La zone mémoire :**

-La mémoire c'est la zone où est stocké le programme. Elle est en général figée, c'est à dire en lecture seulement.

- La mémoire de données utilisable en lecture-écriture pendant le fonctionnement c'est la RAM (mémoire vive). Elle fait partie du système entrées-sorties. Elle fige les valeurs (0 ou 1) présentes sur les lignes d'entrées, à chaque prise en compte cyclique de celle-ci, elle mémorise les valeurs calculées à placer sur les sorties.

- **Les interfaces Entrées/Sorties:**

-Les entrées reçoivent des informations en provenance des éléments de détection (capteurs) et du pupitre opérateur.

-Les sorties transmettent des informations aux pré-actionneurs (relais, électrovannes ...) et aux éléments de signalisation (voyants) du pupitre.

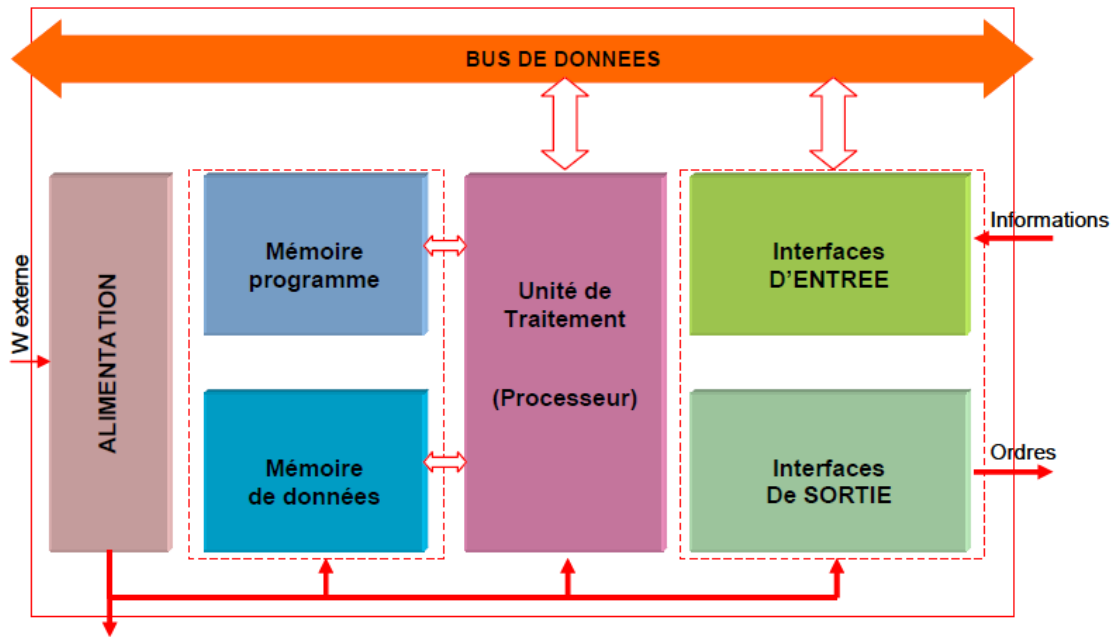


Figure 3.1: Architecture d'un API.

### 3.2.4 TRAITEMENT DU PROGRAMME AUTOMATE:

Tous les automates fonctionnent selon le même mode opératoire :

- Traitement interne : L'automate effectue des opérations de contrôle et met à jour certains paramètres systèmes (détection des passages en RUN / STOP, mises à jour des valeurs de l'horodateur, ...).
- Lecture des entrées : L'automate lit les entrées (de façon synchrone) et les recopie dans la mémoire image des entrées.
- Exécution du programme : L'automate exécute le programme instruction par instruction et écrit les sorties dans la mémoire image des sorties.
- Ecriture des sorties : L'automate bascule les différentes sorties (de façon synchrone) aux positions définies dans la mémoire image des sorties.

Ces quatre opérations sont effectuées continuellement par l'automate (fonctionnement cyclique).

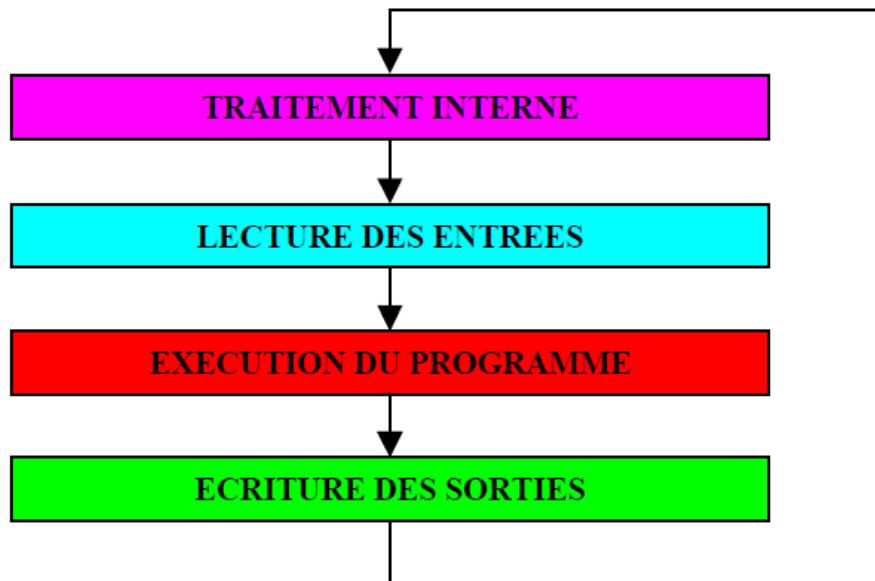


Figure 3.2:TRAITEMENT API.

### 3.2.5 Logiciel de programmation des API SIEMENS "Step7":

#### *a* Qu'est-ce que STEP 7 ? :

Step 7 permet l'accès de base aux automates SIEMENS. Il permet de configurer et de programmer Individuellement un automate (en différents langages). Il prend également en compte le réseau, ce qui permet d'accéder à tout automate du réseau (pour le programmer), et éventuellement aux automates de s'envoyer des messages entre eux. Mais il ne permet pas de faire participer les ordinateurs à l'automatisme, il existe plusieurs versions de Step 7:

-STEP 7-Micro/DOS et STEP 7-Micro/Win pour des applications autonomes simples sur SIMATIC S7 - 200.

STEP 7 pour des applications sur SIMATIC S7-300/400, SIMATIC M7-300/400 et SIMATIC C7.

#### *b* Créer un projet:

Un projet permet de regrouper l'ensemble des données et programmes nécessaires à réaliser une solution d'automatisation. Ces données englobent en particulier :

-Les données de configuration matérielle et les données de paramétrage pour les modules.

-Les données de configuration pour la communication par réseau.

-Les programmes pour modules programmables.

-La tâche principale dans la réalisation d'un projet, consiste à préparer ces données et développer le programme qui assure la tâche d'automatisation de la surveillance du procédé.

Il existe deux méthodes pour créer un projet soit par l'assistant qui se lance automatiquement après chaque ouverture de logiciel Step 7 ou de le créer manuellement (fichier -> nouveau).

### Fenêtre projet :

La fenêtre du projet est la fenêtre principale. Elle est partagée en deux volets :

Le volet gauche représente l'arborescence du projet et le volet droit affiche le contenu de l'objet.

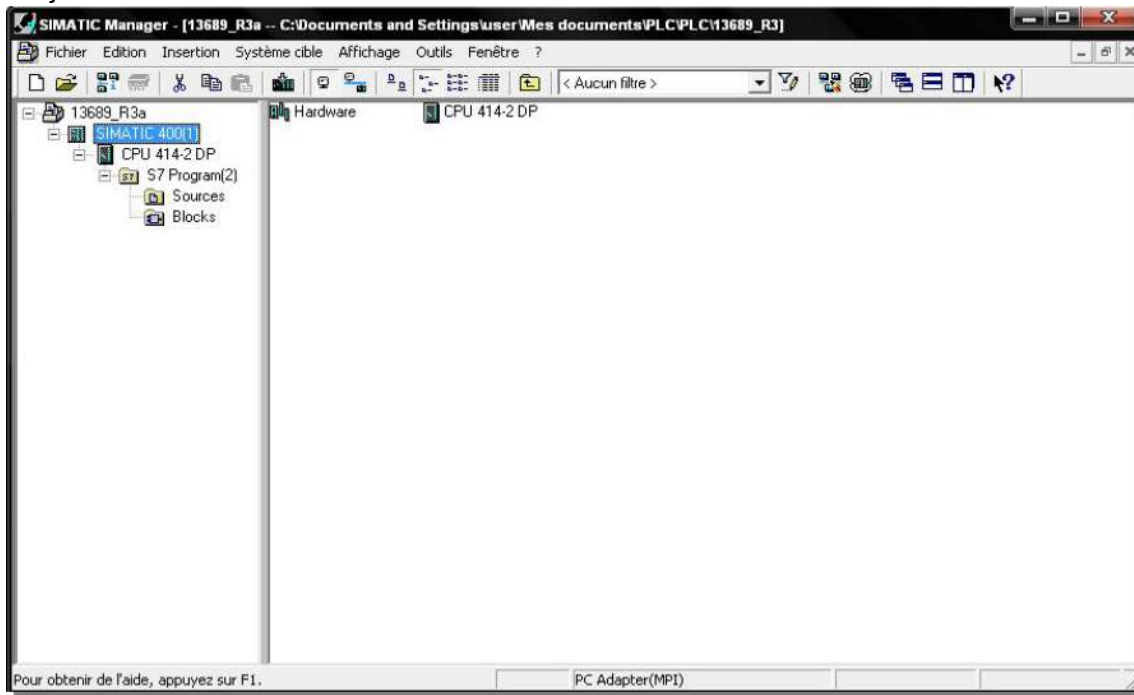


Figure 3.3: Logiciel STEP7.

Les principales actions qu'on peut effectuer à partir de cette fenêtre sont :



Partenaire accessible : Affiche tous les modules accessibles dans le réseau avec leurs adresses.



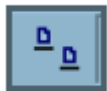
Charger : Permet de charger le programme dans la mémoire.



En ligne : Permet de voir en ligne le déroulement du programme de l'automate.



Passé dans le mode d'affichage hors ligne.



Affiche les objets en utilisant de grandes icônes.



Démarré l'application servant à configurer des réseaux PROFIBUS et ETHERNET.



Permet d'activer/désactiver le simulateur.



### c Configuration matérielle (Partie Hardware):

C'est une étape importante, qui correspond à l'agencement des châssis, des modules et de la périphérie décentralisée. Les modules sont fournis avec des paramètres définis par défaut en usine. Une configuration matérielle est nécessaire pour :

- Modifier les paramètres ou les adresses prérègles d'un module.
- Configurer les liaisons de communication.

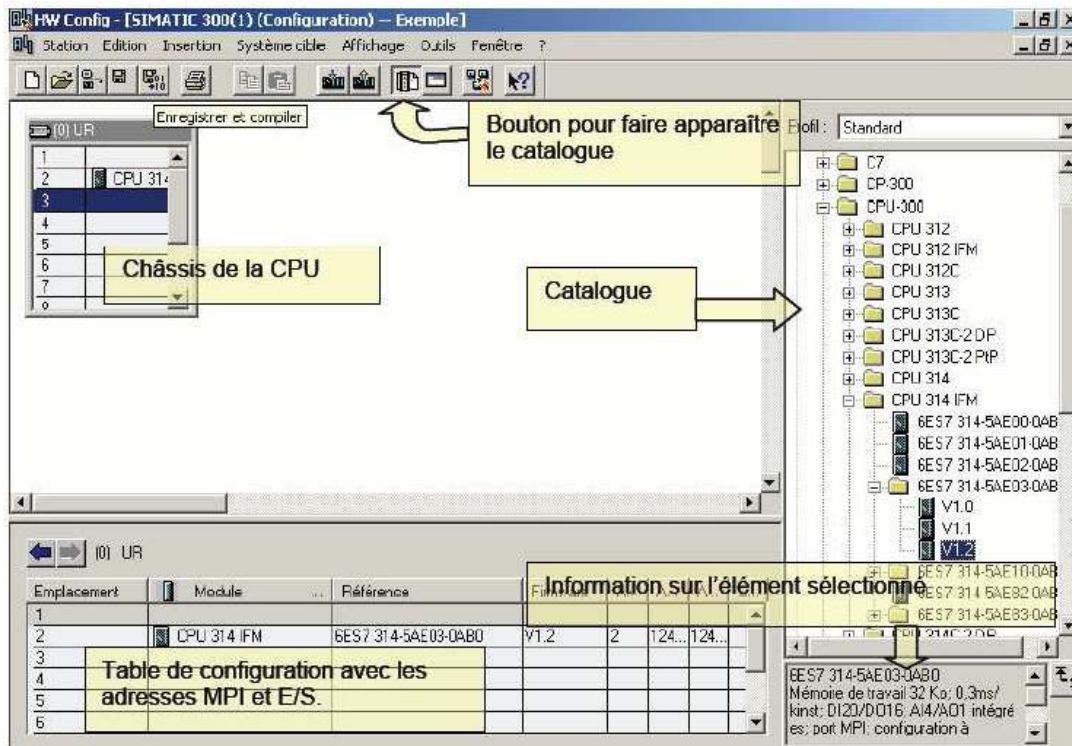


Figure 3.4: Configuration du matériel.

On commence par le choix du châssis selon la station choisie auparavant, Pour la station SIMATIC S300, on aura le châssis " RACK-300 " qui comprend un rail profile.

Il prévoit plusieurs lignes : la première ligne (dans l'ordre de leur implantation physique) est réservée pour l'alimentation, la deuxième ligne est réservée pour l'automate,

La ligne 3 est réservée aux coupleurs (en particulier pour connecter un second rail), puis les modules entrées et sorties TOR et Analogique.

### d Bloc utilisateur :

STEP 7 offre les blocs utilisateur suivants pour la programmation structurée :

**Les blocs d'organisation (OB)** constituent l'interface entre le système d'exploitation et le programme utilisateur. Un bloc d'organisation OB1 vide est créé automatiquement dans le dossier blocs lors de création d'un projet.

Le système d'exploitation exécute OB1 de manière cyclique. Quand son traitement est achevé,

le système écrit dans la mémoire image des sorties, met à jour la mémoire image des entrées et redémarre l'exécution de l'OB1 à nouveau.

**Le bloc fonctionnel FB** est subordonné au bloc d'organisation. Il renferme une partie du programme qui peut être appelée autant de fois que l'on veut dans l'OB1.

Les FB sont très utiles car ils permettent l'abstraction des logiques complexes et récurrentes au sein d'un programme, augmentant l'efficacité et la rapidité de mise en œuvre de fonctionnalités présentes sur plusieurs emplacements dans le programme.

Ils offrent le gain de mémoire obtenu en multipliant des instances de variables au lieu de retrouver la même portion de code plusieurs fois. Ils permettent également de protéger des parties de codes de programme en les rendant inaccessibles à un utilisateur non autorisé à modifier le bloc.

Tous les paramètres formels et toutes les données statiques du bloc fonctionnel sont stockées dans un bloc de données DB séparé qui est associé au bloc fonctionnel. Ce type de bloc est appelé DB d'instance.

Contrairement au DB d'instance, le DB globale est utilisé par tous les autres FB et OB.

Les données stockées dans le DB sont de type booléen, nombre entier, réel. Il y a aussi les dates, heures et chaînes de caractères.

**Fonction FC** ne possède pas une zone de mémoire propre (elle n'a pas besoin de Bloc de données contrairement au bloc fonctionnel), mais elle peut transmettre des paramètres. Les données locales d'une fonction sont perdues après l'exécution de la fonction. La structure d'un programme en STEP7 est définie comme suit :

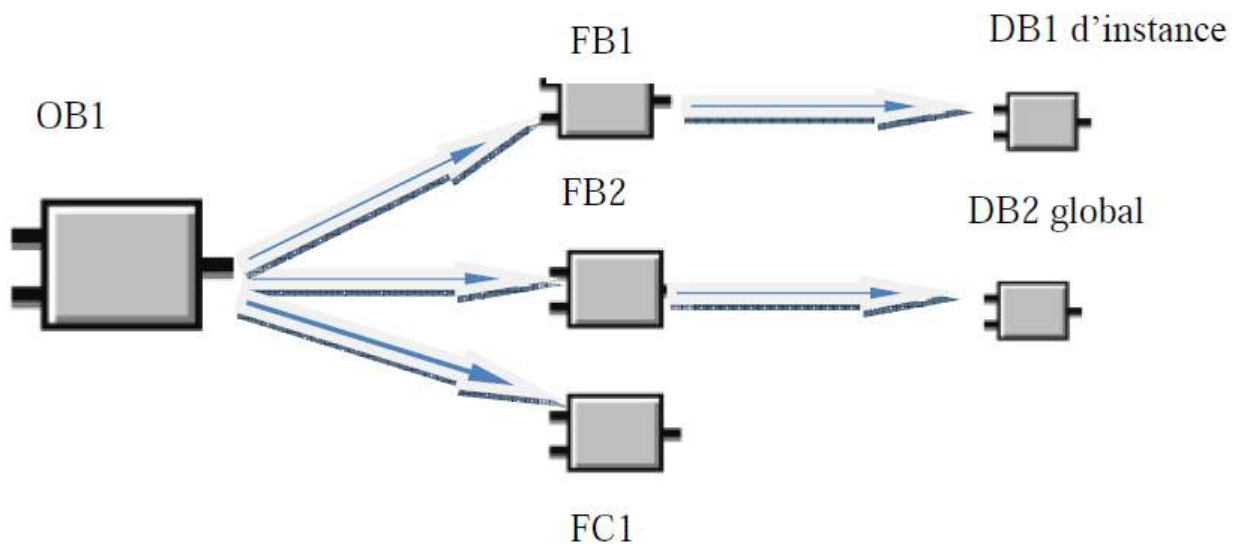


Figure 3.5: Structure du programme STEP7.

Cette structure offre plusieurs avantages:

- Ecrire clairement des programmes importants.
- Standardiser certaines parties du programme.
- Simplifier l'organisation du programme et faciliter sa modification.

-Simplifier le test du programme, car on peut l'exécuter section par section.

Les blocs de code (OB, FB, FC) comportent une section de déclaration des variables, une section

D'instructions et possèdent aussi des priorités.

Les variables représentent les ressources de base pour l'exécution de programme. Les variables d'entrées ou de sorties sont reliées à des entrées et des sorties physiques dans le programme, on leur donne une adresse explicite liée au matériel. Ces variables représentent l'état réel du système au sein du programme.

Certaines variables locales déclarées et utilisées dans un bloc ne seront utilisées que localement c'est à dire qu'elles ne peuvent pas être utilisées par un autre bloc.

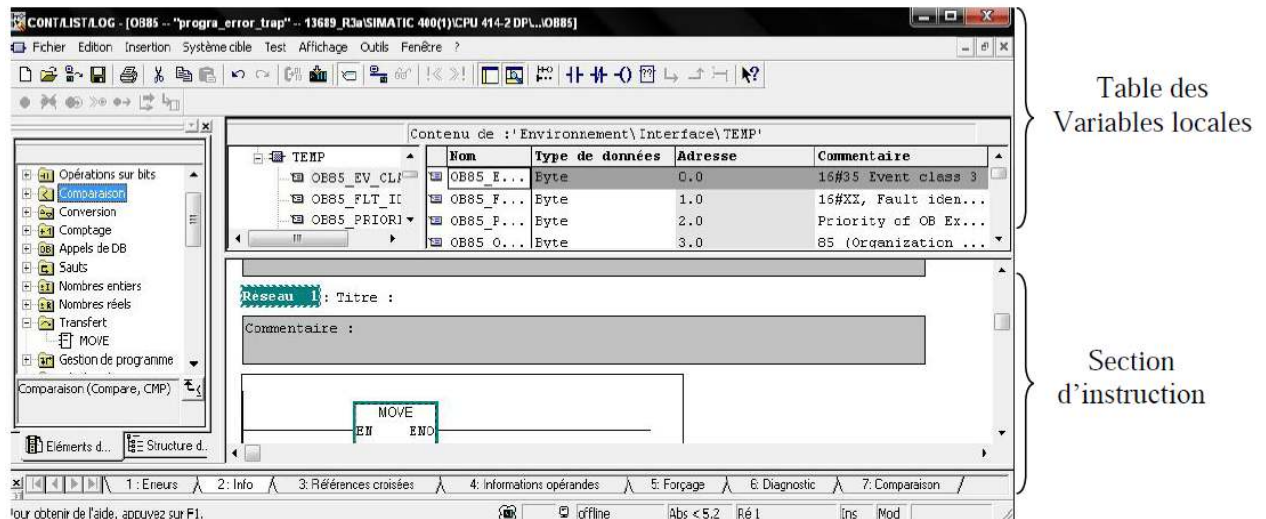


Figure 3.6: Fenêtre de programmation.

Le volet gauche affiche les différents opérandes qu'on peut insérer dans le réseau tels que les opérandes de comparaison, de conversion et les blocs déjà programmés (FB, FC).

Pour créer un bloc dans un projet, on passe par les étapes suivantes.

- Création d'un bloc OB, FB, FC dans le gestionnaire de projet SIMATIC.
- Edition de la table des déclarations des variables
- Edition de la partie d'instruction : le choix du langage (LIST, CONT, LOG).
- Définir la priorité du bloc (quand et où il sera appelé)

### D.1 BLOC SYSTEME POUR FONCTIONS STANDARD ET FONCTIONS SYSTEME:

Ces blocs peuvent être appelés par l'utilisateur et utilisés dans le programme.

On dispose dans STEP 7 des blocs système suivants

#### -SFB (bloc fonctionnel système):

Un bloc fonctionnel système (SFB) est un bloc fonctionnel intégré à la CPU S7. Comme les SFB font partie du système d'exploitation, ils ne sont pas chargés en tant que partie du programme. Comme les FB, les SFB sont des blocs avec mémoire. Vous devez donc également créer pour les SFB des blocs de données d'instance que vous chargez dans la CPU en tant que partie du programme.

Les CPU S7 proposent des SFB :

- pour la communication via des liaisons configurées,
- pour des fonctions spéciales intégrées (par exemple, SFB29 HS\_COUNT dans la CPU 312 IFM et la CPU 314 IFM).

### **-SFC (fonction système):**

Une fonction système (SFC) est une fonction préprogrammée et intégrée dans la CPU S7. Vous pouvez appeler les SFC à partir de votre programme. Comme ces fonctions font partie du système d'exploitation, elles ne sont pas chargées en tant que partie du programme. Comme les FC, les SFC constituent des blocs sans mémoire.

Les CPU S7 proposent des fonctions système pour :

- des fonctions de copie et de blocs,
- le contrôle du programme,
- la gestion de l'horloge et du compteur d'heures de fonctionnement, le transfert d'enregistrements logiques,
- le transfert, en mode de fonctionnement multiprocesseur, d'événements d'une CPU à toutes les CPU enfichées,
- la gestion des alarmes horaires et temporisées,
- la gestion des événements d'erreur synchrone, des événements d'alarme et des événements d'erreur asynchrone,
- l'information sur les données système statiques et dynamiques, p. ex. le diagnostic,
- la mise à jour de la mémoire image du processus et le traitement de champ binaire,
- l'adressage de modules,
- la périphérie décentralisée,
- la communication par données globales,
- la communication via des liaisons non configurées,
- la création de messages relatifs aux blocs.

### **D.2 EDITEURMNEMONIQUES:**

Les mnémoniques sont les noms que l'on attribue aux variables globales de l'API. L'emploi des mnémoniques à la place des adresses absolues (ex E0.0=MOTEUR) améliore considérablement la lisibilité et la clarté d'un programme et aide à isoler des défauts éventuels. Les mnémoniques ainsi définies sont utilisables dans tout le programme utilisateur d'un module programmable.

	Etat	Mnémonique	Opérande	Type de d	Commentaire
1		1B1	E 124.4	BOOL	
2		1B2	E 124.5	BOOL	
3		1Y1	A 124.0	BOOL	
4		1Y2	A 124.1	BOOL	
5		2B1	E 124.6	BOOL	
6		2Y1	A 124.2	BOOL	
7		3Y1	A 124.3	BOOL	
8		A2	E 124.3	BOOL	
9		arret	E 125.1	BOOL	
1		aut	E 125.2	BOOL	
1		B3	E 124.2	BOOL	
1		dcy	E 125.0	BOOL	
1		delay1	M 100.5	BOOL	
1		delay2	M 100.6	BOOL	
1		delay3	M 100.7	BOOL	
1		fin	M 100.0	BOOL	
1		G7_STD_3	FC 72	FC 72	
1		init	M 100.1	BOOL	
1		ip_inf	A 124.7	BOOL	
2		reset	E 125.3	BOOL	
2		start	M 100.2	BOOL	
2		TIME_TCK	SFC 64	SFC 64	Read the System Time
2		vreset	A 125.1	BOOL	
2		vstart	A 125.0	BOOL	

Figure 3.7:table de Mnémoniques.

## D.2 DIFFERENTS TYPES DE VARIABLES CONTENUES DANS STEP7:

Dans l'environnement STEP7 on utilise plusieurs types de variables qu'on doit déclarer au préalable, le tableau résume les types de variables utilisées :

Mot-clé	Taille (en bits)	Exemple d'une constante De ce type
BOOL	1 bit	1 ou 0
Byte	8 bits	B#16#A6
Word	16 bits	W#16#FADA
Dword	32 bits	DW#16#CAC40EDF
Char	8 bits	` W `
S5Time	16 bits	S5T#5s-200ms
Int	16 bits	123
DInt	32 bits	65539 ou L#-5
Real	32 bits	1.0 ou 34.5E-12
Time	32 bits	T#2D-1H-3M-45s-12ms
Date	16 bits	D#2002-06-20
Time of-Day	32 bits	TOD#12 :23 :45.12

Tableau 3.1 : Types de variables utilisés dans Step7.

#### D.4 LANGAGE DE PROGRAMMATION:

##### **Le langage LOG**

C'est un langage graphique, utilisant les symboles de l'électronique numérique (portes logiques). Il n'y a rien de spécial à dire, c'est très intuitif. On peut utiliser plusieurs entrées pour une même porte, placer des inverseurs sur les entrées.... Ici, on découpe son programme en plusieurs réseaux (en général quand un ensemble de blocs n'est pas relié au reste, ou un réseau par sortie...)

##### **Le langage LIST**

C'est un langage textuel, qui est le plus proche du comportement interne de l'automate (correspond à peu près à l'assembleur dans un ordinateur). Le système sait toujours traduire du CONT ou du LOG en LIST, mais pas l'inverse.

Le programme se compose d'une suite de lignes, chacune spécifiant un code opération suivi d'un opérande (et un seul). L'opérande peut être une adresse absolue (E0.0) ou un mnémonique entre guillemets (si les mnémoniques ont été définis, bien sûr). Comme on ne peut pas utiliser

Deux opérandes dans une même ligne,

##### **Le langage CONT**

Le schéma à contact CONT est un langage de programmation graphique. La syntaxe des instructions fait penser aux schémas de circuits. Les éléments d'un schéma de circuit tels que contacts à fermeture, contacts à ouverture sont rassemblés dans un réseau. Un ou plusieurs réseaux forment la section des instructions complètes d'un bloc de code.

Ce langage permet de suivre facilement le trajet du courant entre les barres d'alimentation en passant par les contacts, les éléments complexes et les bobines.

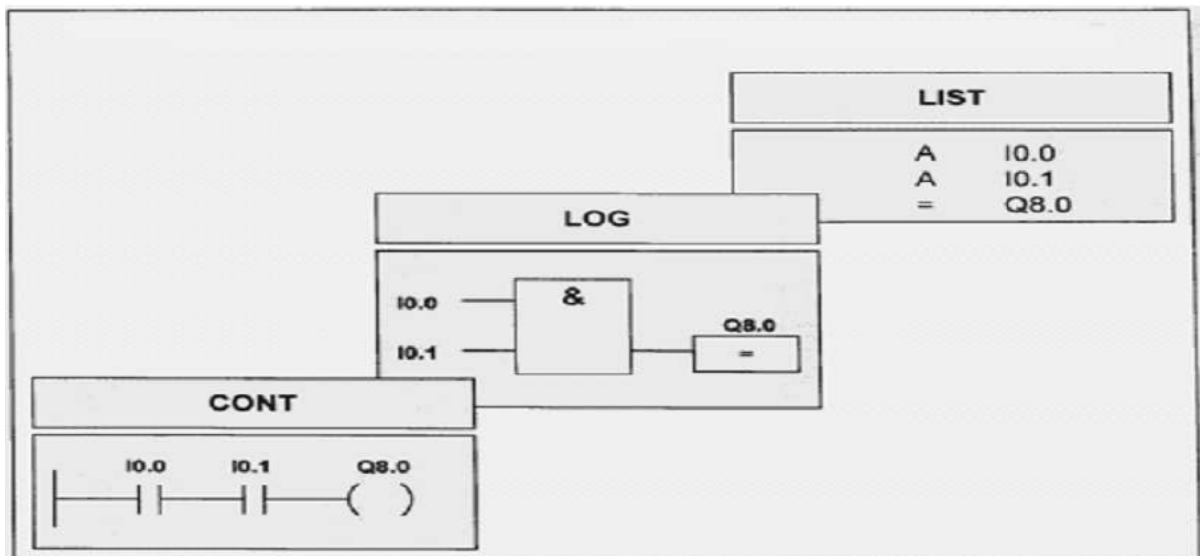


Figure 3.8:langages de programmation STEP 7 .

##### **Le langage GRAPH**

Le Graph (langage proche du Grafcet mais moins puissant) n'est pas un langage de base dans S7. Mais il suffit de créer un bloc fonctionnel (FB) pour avoir le choix de l'utiliser.

Pour les transitions, on peut choisir le langage CONT ou LOG.

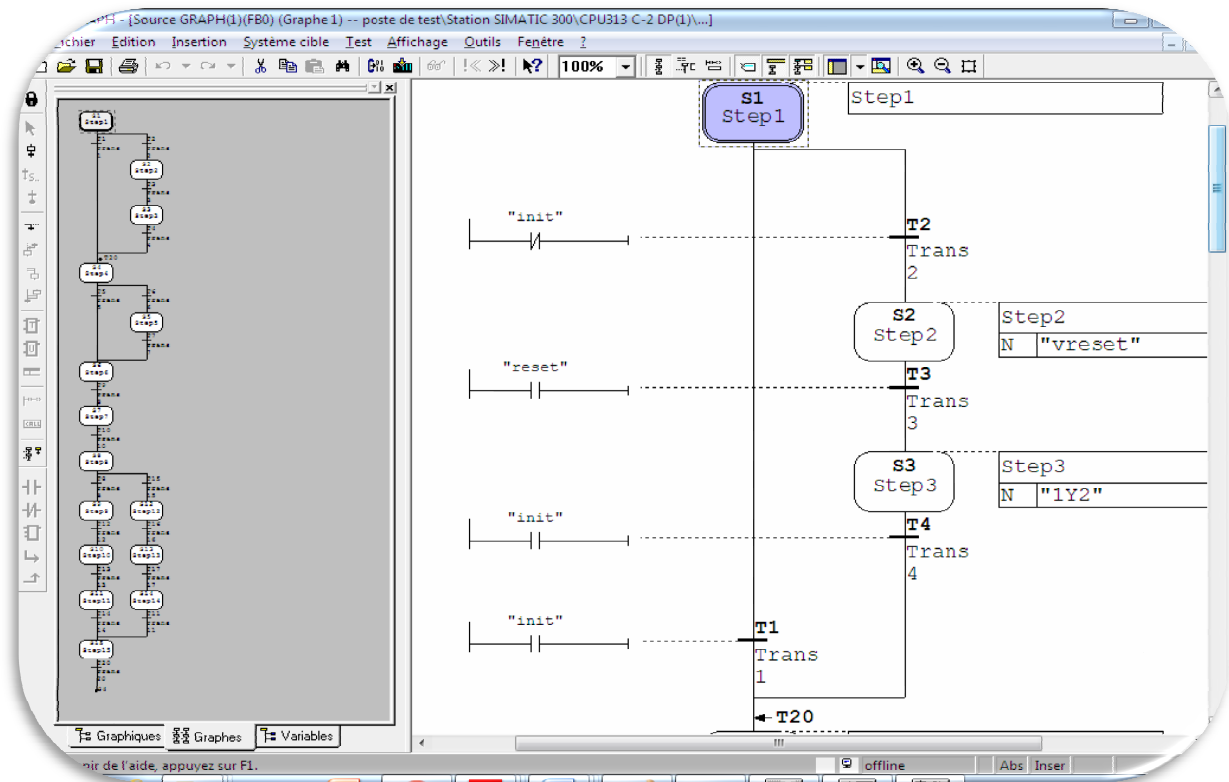


Figure 3.9: langage graph.


### Remarque


Chaque mode de représentation du programme a ses avantages mais aussi ses limitations. Si quelques règles ont été respectées lors de la programmation, la compilation est possible dans les trois modes de représentation. Les programmes d'automatisation programmés en CONT ou LOG sont en principe toujours traduisibles en LIST. Le langage liste à l'avantage d'être traité plus rapidement par l'automate que les langages LOG et CONT.


### D.5 SIMULATEUR S7-PLCSIM:


Le simulateur S7-PLCSIM permet d'exécuter et de tester le programme utilisateur dans un API qu'on veut simuler. Il fournit une interface simple avec le programme utilisateur pour visualiser et forcer différents objets comme des entrées et des sorties.


Ces objets sont:

-Entrées  : Permet d'accéder aux données enregistrées dans la zone mémoire des entrées. L'adresse par défaut est l'octet 0 (EBO).

-Sorties  : Permet d'accéder aux données enregistrées dans la zone mémoire des sorties. L'adresse par défaut est l'octet 0 (AB0).

-Mémento  : Permet d'accéder aux données enregistrées dans la zone des mémentos (M) .L'adresse par défaut est l'octet 0 (MB0).

-Temporisation  : Permet d'accéder aux temporisations utilisées par le programme. La temporisation par défaut est T0.

-Compteur  : Accède aux compteurs utilisés par le programme. Le compteur par défaut est Z0.

La fenêtre principale du simulateur S7-PLCSIM est présentée comme suit :

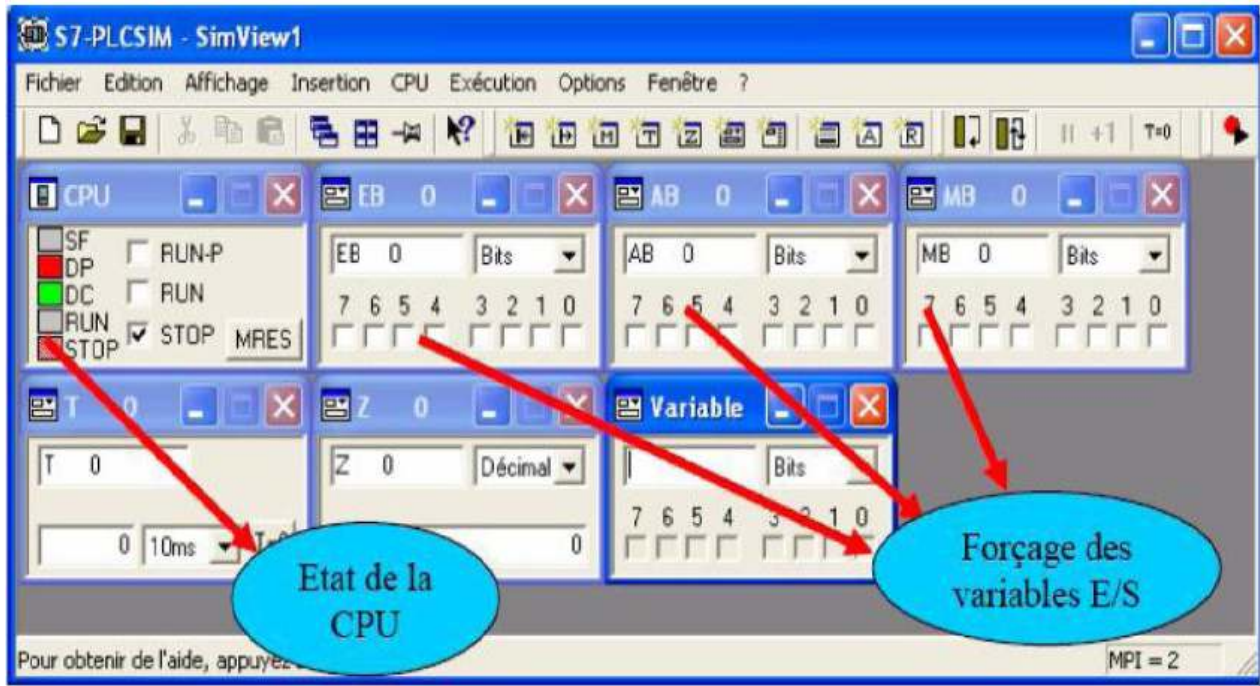


Figure 3.10:Interface de simulation PLCSIM.

Le simulateur permet aussi de visualiser simultanément le déroulement du programme de l'API en montrant les parties du code concernées par les variables d'objets modifiées

Deux modes d'exécution du programme sont définis :

Cycle unique : exécute un cycle de programme, puis attend le démarrage de l'exécution du cycle suivant.

Cycle continu : exécute le programme de la même manière que dans un API réel. Il démarre un nouveau cycle aussitôt que le cycle précédent est terminé.

La simulation peut être mise à l'état d'attente en exécutant la commande pause et permet de reprendre l'exécution du programme avec l'instruction à laquelle elle avait été interrompue.

La fenêtre CPU dispose d'un ensemble d'indicateurs qui correspondent aux voyants de signalisation

Sur une CPU réel :

-SF (erreur système) : Avertit que la CPU a détecté une erreur système.

-DP (Périphérique Décentralisé E/S éloigné) : Indique l'état de communication avec



Les entrées /sorties décentralisées éloignées.

-DC (alimentation) : Indique si la CPU se trouve sous ou hors tension.

-RUN : Indique que la CPU se trouve en état de marche.

-STOP : Indique que la CPU se trouve en état arrêt.

### **3.3 Logiciel de programmation WinCC flexible**

#### **3.3.1 Définition:**

WINCC flexible est un système polyvalent qui permet de réaliser des projets de visualisation et de contrôle des commandes dans le domaine de l'automatisation de la production et des processus. Il offre des modules fonctionnels adaptés au monde industriel pour la représentation graphique, la signalisation des alarmes, l'archivage et la journalisation. Avec couplage au processus performant, le WINCC flexible offre un rafraîchissement rapide des vues et un archivage de données fiable, il assure une haute disponibilité du système.

WinCC flexible peut aussi être défini comme l'interaction entre un système et un automate, l'interaction est présentée par un écran avec dynamique icônes, des chiffres et du texte.

Un opérateur peut surveiller la production et le contrôle à un certain niveau à l'aide d'un écran IHM ou PC, il permet de réduire les risques d'erreur humaine.

#### **3.3.2 WinCC flexible Runtime:**

Au Runtime, l'opérateur peut réaliser le contrôle-commande du processus.

Les tâches suivantes sont alors exécutées:

- Communication avec les automates.
- Affichage des vues à l'écran.
- Commande du processus, par exemple spécification de consignes ou ouverture et fermeture de vannes.
- Archivage des données de Runtime actuelles, des valeurs processus et événements d'alarme.

#### **3.3.3 Création de projet :**

Le projet est à la base de la configuration de l'interface graphique.

Nous créons et configurons dans le projet tous les objets indispensables à la commande et au contrôle de nos stations, par exemple :

- des vues, pour représenter et commander les stations.

- des variables qui transmettent les données entre les stations et l'opérateur.
- des alarmes qui affichent au pupitre les états de fonctionnement des stations.



Figure 3.11: création d'un projet avec l'assistant du WinCC flexible.

L'assistant du WinCC flexible nous donne l'accès de donner des paramètres spécifiques pour notre projet comme nous montre la **Figure 3.12**

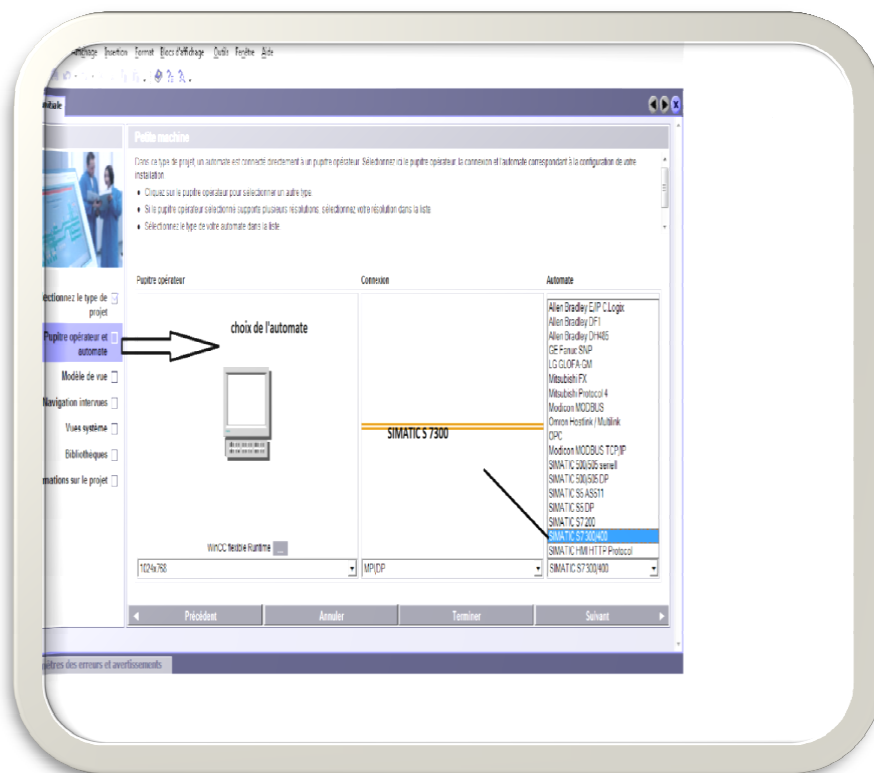


Figure 3.12: la liaison entre WinCC flexible et l'automate.

On suit les étapes demandé par le logiciel on remarque qu'il faut sélectionner les bibliothèques disponibles comme la figure nous le montre **Figure3.13**

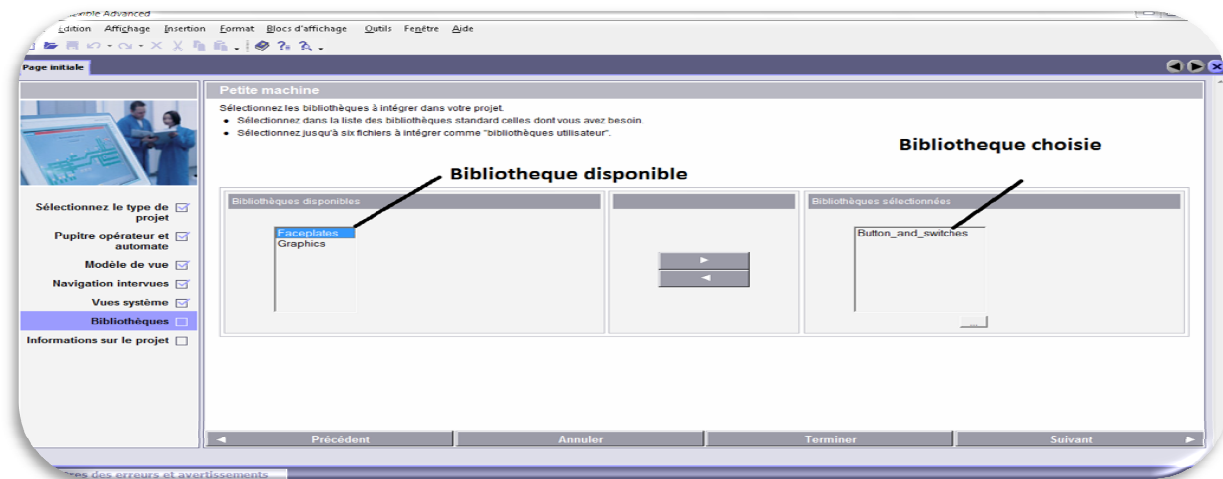


Figure 3.13: bibliothèques de WinCC flexible pour réaliser un projet.

### 3.3.4 Création des vues:

Dans WinCC flexible, chaque projet crée contient principalement des vues que l'on crée pour le contrôle-commande de machines et d'installations. Lors de la création des vues, vous disposez d'objets prédéfinis permettant d'afficher des procédures et de définir des valeurs du processus.

Une vue peut être composée des éléments statique et des éléments dynamiques.

- Les éléments statiques se sont les objets qui ne changent pas au Runtime.
- Les éléments dynamique varient en fonction de la procédure, ils permettent de visualiser les paramètres du processus à partir de la mémoire de l'automate ou à partir de la mémoire du pupitre de l'opérateur sous formes des courbes ou des champs d'entrées / sorties par exemple.

Les différents outils et barres de l'éditeur des vues sont représentés dans la figure qui suit :

- **Barre des menus** : La barre des menus contient toutes les commandes nécessaires à l'utilisation de WinCC flexible. Les raccourcis disponibles sont indiqués en regard de la commande du menu.
- **Barres d'outils**: La barre d'outils permet d'afficher tout dont le programmeur à besoin.
- **Zone de travail**: La zone de travail sert à configurer des vues, de façon qu'il soit le plus compréhensible par l'utilisateur, et très facile à manipuler et consulter les résultats.
- **Boîte à outils** : La fenêtre des outils propose un choix d'objets simples ou complexes qu'on insère dans les vues, par exemple moteur, vanne, capteur, bouton, cuve..... .
- **Fenêtre des propriétés** : Le contenu de la fenêtre des propriétés dépend de la sélection actuelle dans la zone de travail, lorsqu'un objet est sélectionné, on peut éditer les propriétés de l'objet en question dans la fenêtre des propriétés.

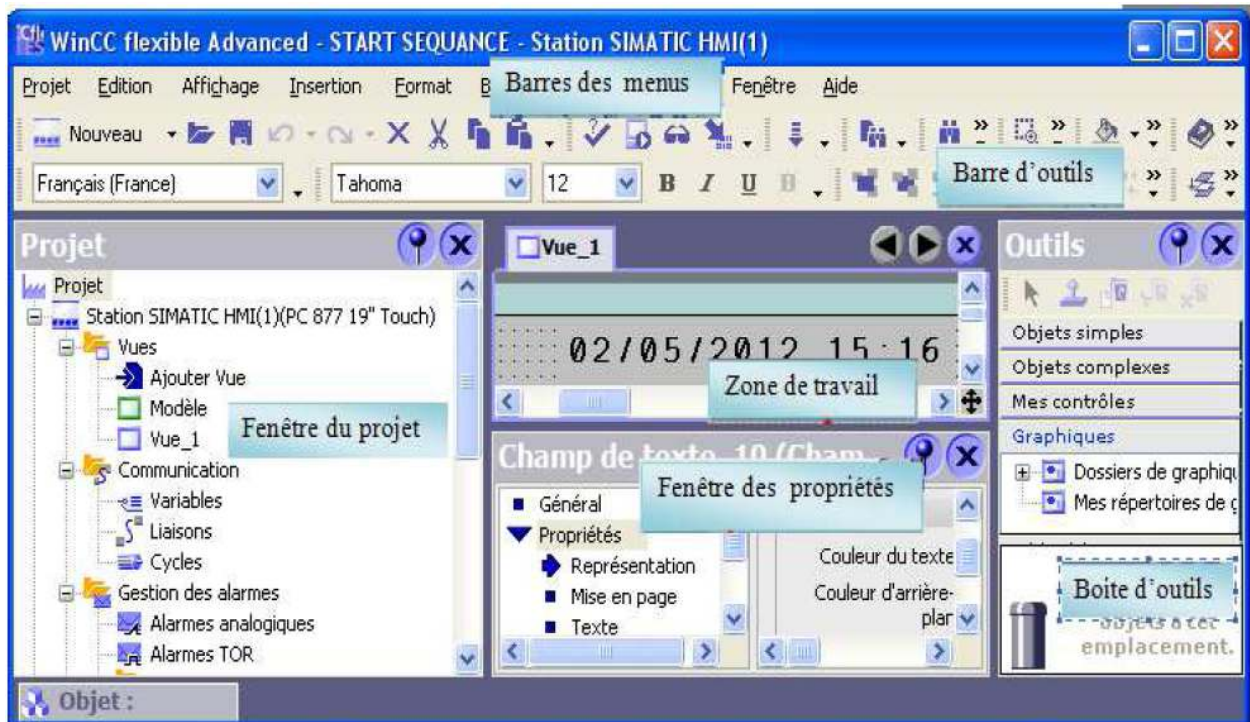


Figure 3.14:Vue d'ensemble du logiciel WinCC flexible.

### 3.3.5 Création de la variable :

Il est indispensable de créer une table de variable sur WinCC flexible qui contient les différents entrées / sorties nécessaire pour la visualisation et le contrôle des paramètres de système.

Il Ya deux type de variable (variable interne, variable externe)

- **variable interne :**

Les variables internes ne possèdent aucun lien avec l'automate.

Les variables internes sont enregistrées dans la mémoire du pupitre.

- **variable externe :**

Les variables externes permettent de d'échanger des données entre un pupitre operateur et un automate

Les variables externes étant une image d'une cellule mémoire de l'automate.

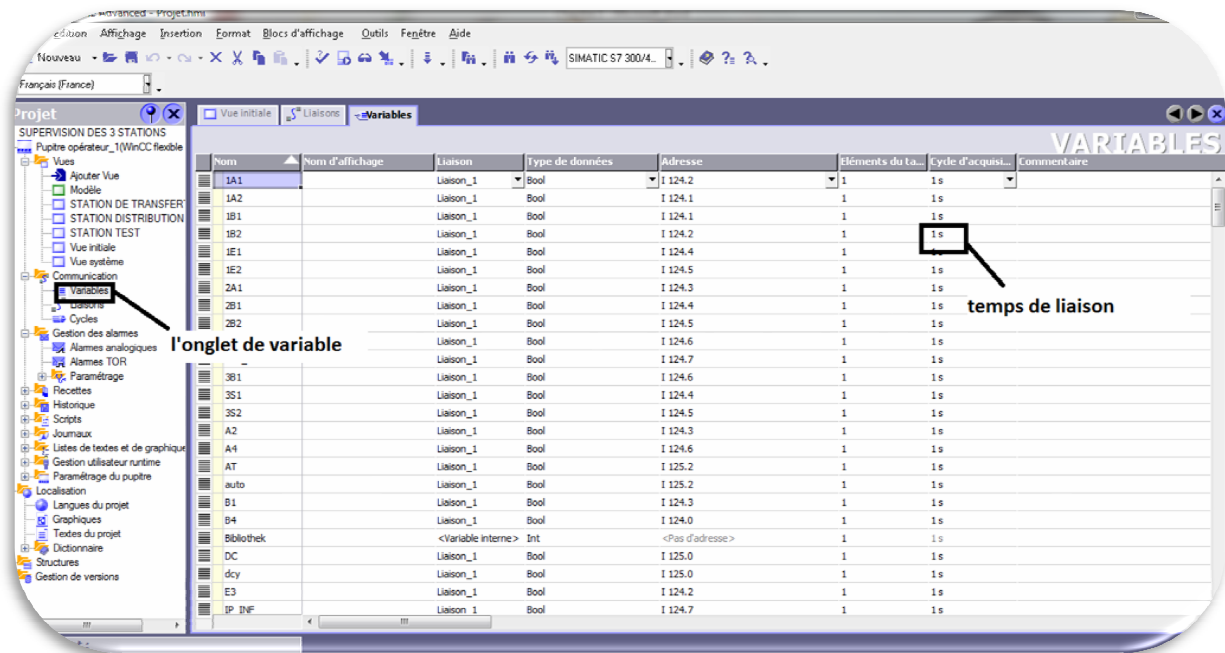


Figure 3.15:table des variables.

**Remarque :**

Le temps de liaison est le temps que doit mettre WinCC flexible entre deux lectures dans la mémoire de l'automate.

**3.3.6 . Intégration du projet STEP7:**

Avant de commencer la réalisation de l'interface de supervision, il est indispensable, de créer une liaison directe entre WINCC flexible et notre automate (le projet du Step7). Ceci dans le but que WINCC flexible puisse aller lire les données qui se trouvent dans la mémoire de l'automate.

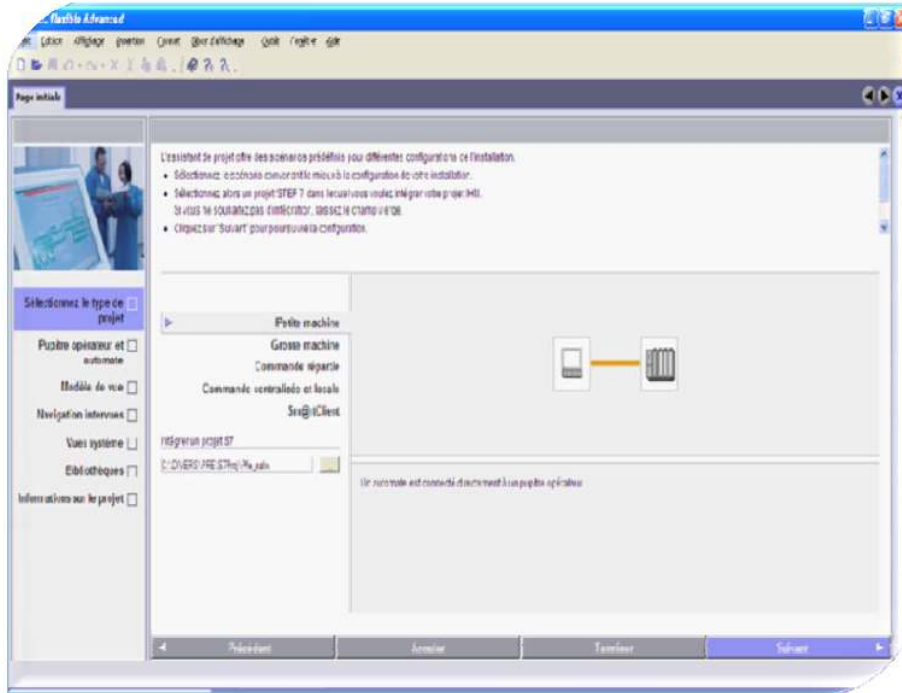


Figure 3.16: Intégration du projet Step7 dans WinCC flexible.

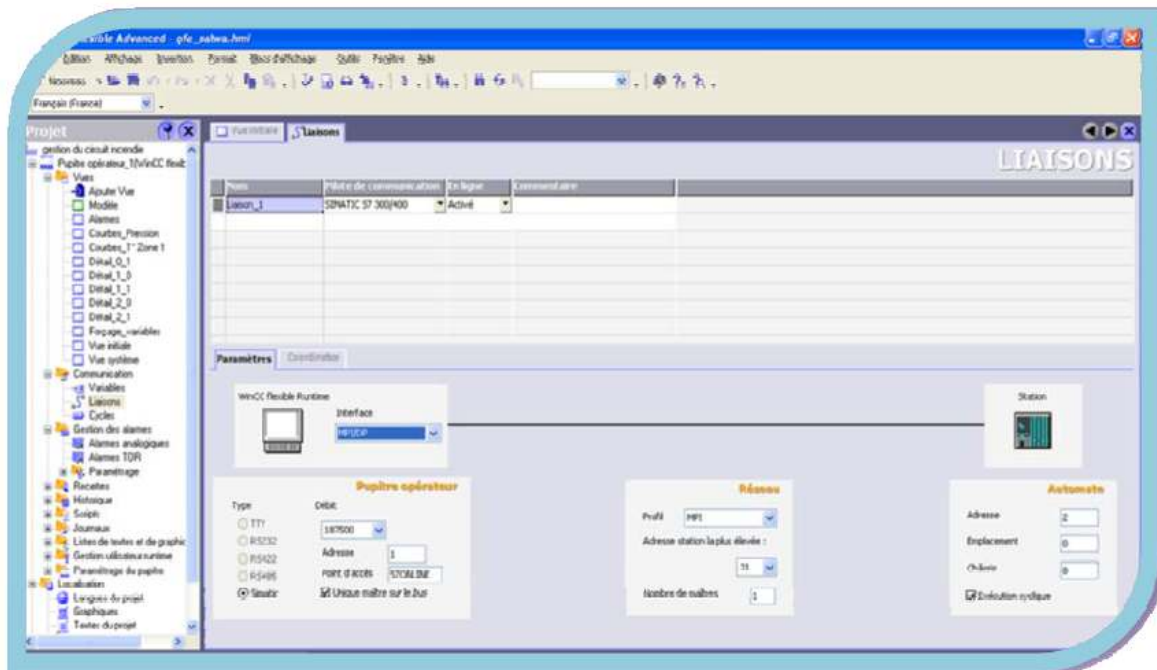


Figure 3.17: Liaison avec l'automate S7300.

### 3.3.7 Créer des alarmes :

Il existe de type d'alarme (Les alarmes définies par l'utilisateur, Les alarmes système)

- **Les alarmes définies par l'utilisateur :**

Vous configurez les alarmes à afficher des états de processus ou de mesurer et de communiquer des données de processus qui vous recevez de l'automate sur le pupitre opérateur.

Il y a deux aussi pour les alarmes définies par l'utilisateur

- **alarme TOR :**

Le pupitre operateur déclenche une alarme lorsqu'un certain bit est mis a 1 ou 0 dans L'automate. .

- **alarme analogique :**

Le pupitre operateur déclenche une alarme lorsqu'une variable déterminée dépasse une valeur limite ou une valeur supérieure.

- **Les alarmes système :**

Les alarmes système sont prédéfinies dans ces dispositifs pour afficher certains états du système dans le Pupitre opérateur ou l'automate.

Les deux alarmes définies par l'utilisateur et les alarmes système sont déclenchées par le Pupitre opérateur ou l'automate et peut être affiché sur le pupitre opérateur.



### **3.4 Cahier de charges :**

Notre projet intitulé «Etude de la station NEP de l'entreprise Trèfle» consiste à élaborer une application qui a pour objectif de faire la supervision et le contrôle en temps réel des paramètres de système de nettoyage en place.

Notre travail est subdivisé en deux parties : partie programmation sur STEP7 et partie supervision sur WINCC flexible.

La première partie consiste à concevoir un programme répondant aux fonctions principales du système de nettoyage des machines et des cuves ainsi la préparation des cuves qui envoient les solutions détergentes en utilisant STEP7. Le programme ainsi développée doit permettre la :

- Suivi en temps réel des paramètres de température dans les cuves.
- Suivi en temps réel des paramètres de concentration dans les différentes cuves et lignes.
- Suivi en temps réel des paramètres de débit dans les lignes de nettoyage.
- Gestion des alarmes du système.

Dans la deuxième partie, on utilisera WINCC flexible qui consiste à superviser et à surveiller notre système.

### **3.5 Les besoins de la station NEP :**

Après une étude très sensible de notre cahier de charges et une très longue analyse on a estimé les exigences de la station NEP en côté matériels :

#### **3.5.1 CPU 313C-2DP :**

Qui a les caractéristiques suivantes :

Mémoire de travail 64 Ko; 0,1ms/kinst; DI16/DO16 intégrées; 3 sorties d'impulsion (2,5kHz); comptage et mesure sur 3 voies avec codeurs incrémentaux 24V (30kHz); ports MPI+ DP (maître DP ou esclave DP); configuration multi-rangées jusqu'à 31 modules; échange direct de données possible (émetteur et récepteur); équidistance; routage; communication S7

(FB/FC chargeables); firmware V2.6

### **3.5.2 . LES MODULES**

Nous avons choisi suivant les exigences de l'installation les modules suivants :

- **trois** Modules d'entrées analogiques (AI8xRTD) spéciaux pour les thermistances et les sondes de température à résistance platine Pt100.

Qui ont les caractéristiques suivantes :

Modules d'entrées analogiques AI8xRTD, 16 bits (interne 24 bits selon le procédé Sigma-Delta); RTD: Pt100, Pt200, Pt500, Pt1000, Ni100, Ni120, Ni1000, Cu10; reparamétrable en ligne

- Module de sortie analogique (AO4x12Bit).

Qui a les caractéristiques suivantes :

Module de 4 sorties analogiques, 12 bits, pas prévu pour une configuration avec modules de bus actifs

- Module d'entrées/sorties TOR (DI16/DO16)

### **3.5.3 Le câble de communication profibus :**

(Process Field Bus) est le nom d'un type de bus de terrain propriétaire et de son protocole, inter-automates et de supervision. Il est devenu peu à peu une norme de communication dans le monde de l'industrie ces dix dernières années

### **3.5.4 Le pupitre digital PC IL 77 15" Touch 1.4.0.0 :**

L'appareil fait partie de la gamme des pupitres opérateur basés sur PC, le Panel PC IL77 satisfait aux exigences de base du domaine industriel et offre une performance élevée Elle est conçue pour fonctionner en continu 24 heures sur 24, L'appareil se compose d'une unité de dialogue à écran tactile.

### **3.6 Comment on a programmé la station N.E.P :**

Pour comprendre plus précisément le fonctionnement du processus C'est-à-dire L'interaction entre la partie commande et la partie opérative et pour développer une solution de conduite programmable, la modélisation de ce cycle en GRAFCETs'avère nécessaire.

A partir du cahier des charges nous avons créé un GRAFCET qui modélise les séquences de fonctionnement de la station NEP.

#### **3.6.1 Legrafcet**

L'interprétation du cahier des charges est donnée par le Grafcet, et l'exemple le plus simple qu'on a utilisé est celle de préparation de cuve EAU PROPRE (voir la figure...**3.29.**)

Et l'organigramme suivant explique les étapes de notre application (voir fig.3.18)

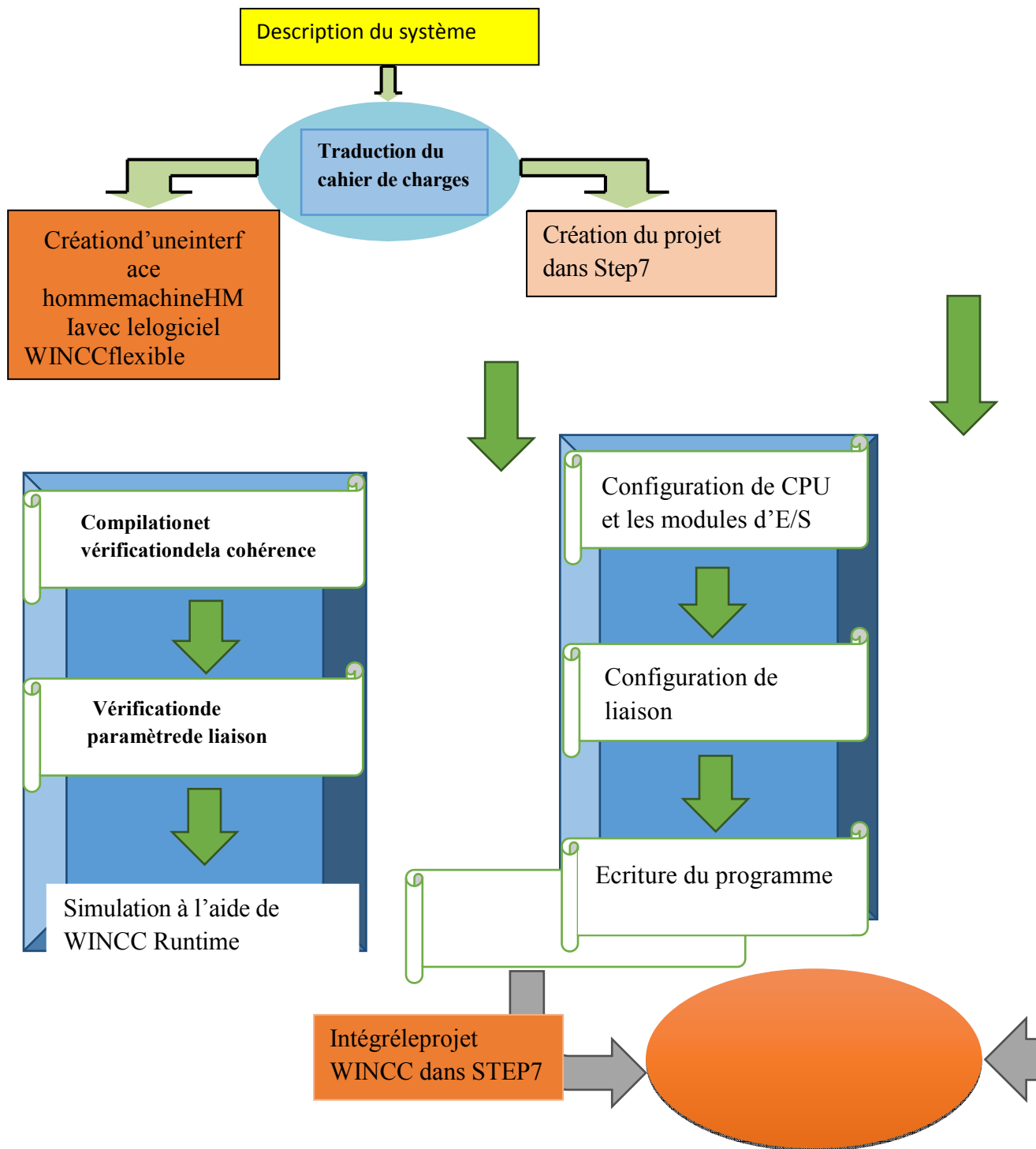


Figure 3.18: Organigramme de l'application.

### 3.6.2 Création de projet S7

#### a .Création du projet dans SIMATIC Manager

Afin de créer un nouveau projet STEP7, il est possible d'utiliser

« L'assistant de création de projet » ou bien créer le projet soi-même et le configurer directement, cette dernière est un peu plus complexe, mais nous permet aisément de gérer notre projet.

En sélectionnant l'icône SIMATIC Manager, on aura la fenêtre principale qui s'affiche, pour sélectionner un nouveau projet et valider. Comme le projet est vide il nous faut insérer une station SIMATIC 300.

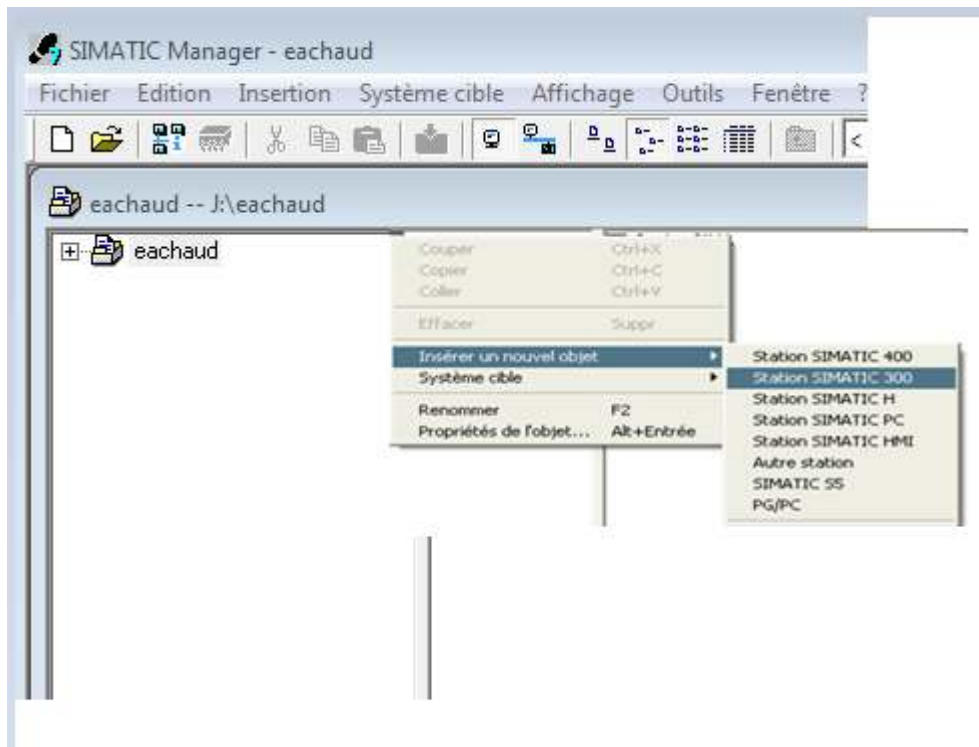


Figure 3.19: Le SIMATIC Manager.

#### b . Configuration du matériel dans STEP7

Le matériel utilisé est imposé par l'installation existante.

Insertion d'une station SIMATIC 300  
RACK 300

On sélectionne une CPU 313C-2DP et les E/S TOR et ANALOGIQUES  
**(VOIR FIGURE 3.20 et fig. 3.21)**

PROFIBUS(1): Réseau maître DP (1)

Emplacement	Module	Référence	Firmware	Adresse MPI	Adresse d'entrée	Adresse de sortie	C...
1							
2	CPU313 C-2 DP(1)	6ES7 313-6CF03-0AB0	V2.6	2			
X2	DP				1023*		
2.2	DI16/DO16				124...125	124...125	
2.4	Comptage				768...783	768...783	
3							
4	AI8xRTD	6ES7 331-7PF00-0AB0			256...271		
5	AI8xRTD	6ES7 331-7PF00-0AB0			272...287		
6	AI8xRTD	6ES7 331-7PF00-0AB0			288...303		
7	AO4x12Bit	6ES7 332-5HD00-0AB0				304...311	
8							
9							
10							
11							

Figure 3.20: configuration de matériels.

Emplacement	Module	Référence	Firmware	Adresse MPI	Adresse d'entrée	Adresse de sortie	Commentaire
1							
2	CPU313 C-2 DP(1)	6ES7 313-6CF03-0AB0	V2.6	2			
X2	DP				1023*		
2.2	DI16/DO16				124...125	124...125	
2.4	Comptage				768...783	768...783	
3							
4	AI8xRTD	6ES7 331-7PF00-0AB0			256...271		
5	AI8xRTD	6ES7 331-7PF00-0AB0			272...287		
6	AI8xRTD	6ES7 331-7PF00-0AB0			288...303		
7	AO4x12Bit	6ES7 332-5HD00-0AB0				304...311	
8							
9							
10							
11							

Figure 3.21: configuration de la sortie analogique.

### c . Paramétragedesmodulesanalogiques:

La connaissance de la méthode d'adressage de chaque module analogique représente une étape primordiale dans l'acquisition des données. Lors du raccordement des capteurs à l'automate, nous sommes sensés de savoir et respecter un certain adressage. L'automate ne peut recevoir ni traiter les données que si l'instrument de mesure est branché à la bonne voie du module d'entrée analogique, avec l'adresse d'entrée correspondante sur STEP7.

Les figures suivantes montrent les adresses de début et fin des modules d'Entrées/Sorties.

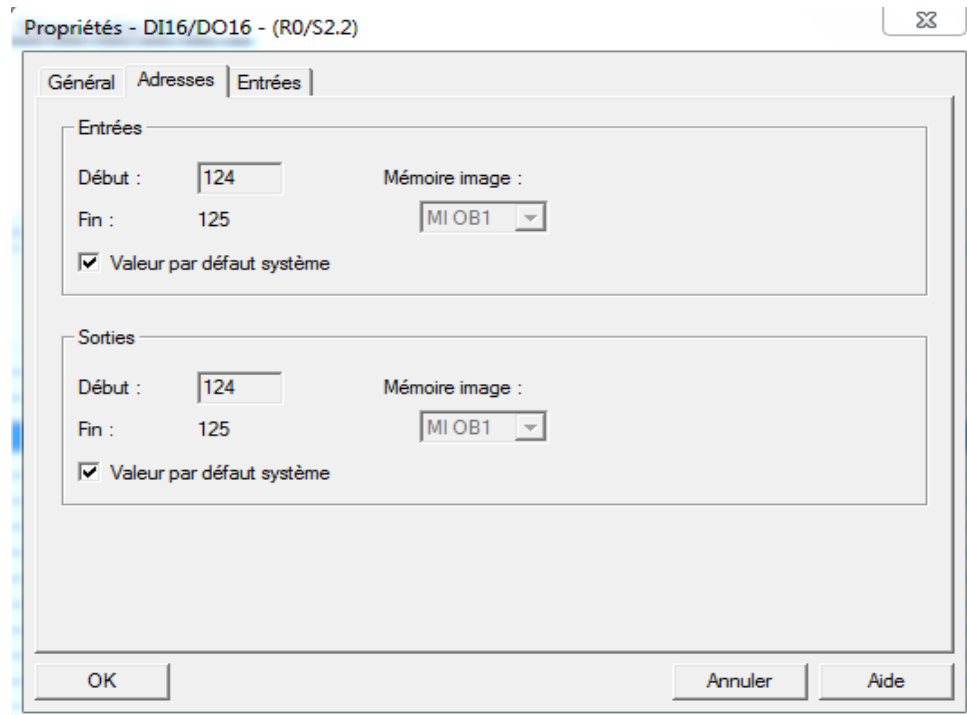


Figure 3.22:adresses du module DI16/DO16.

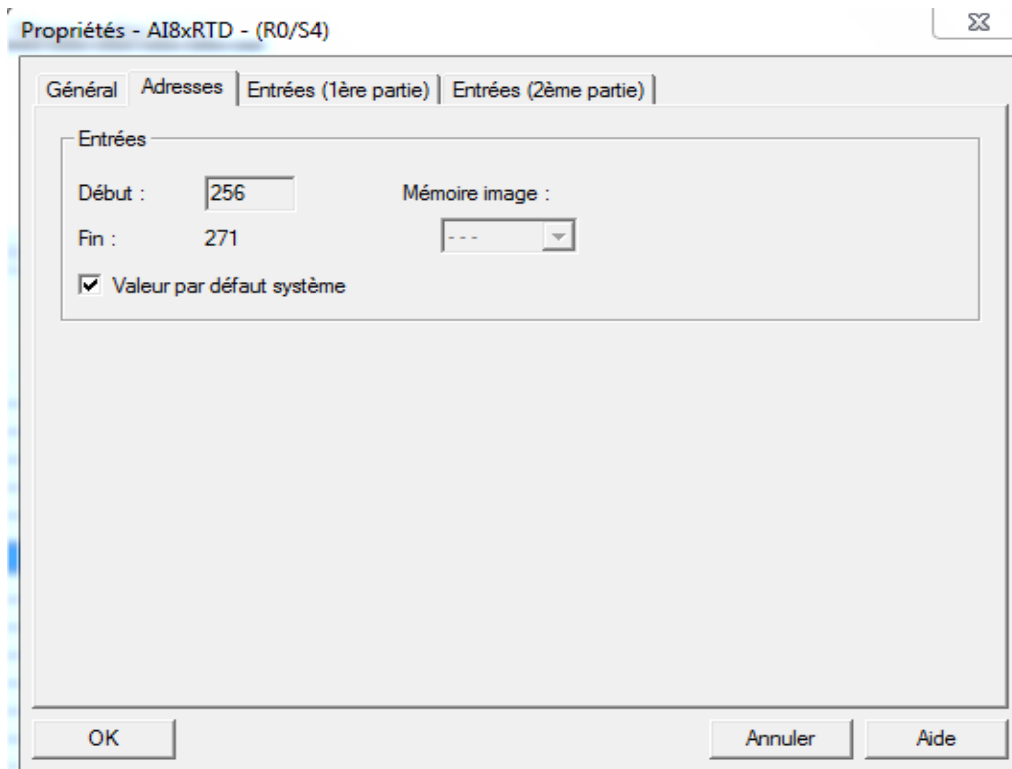


Figure 3.23:adresses du 1er module AI8xRTD.

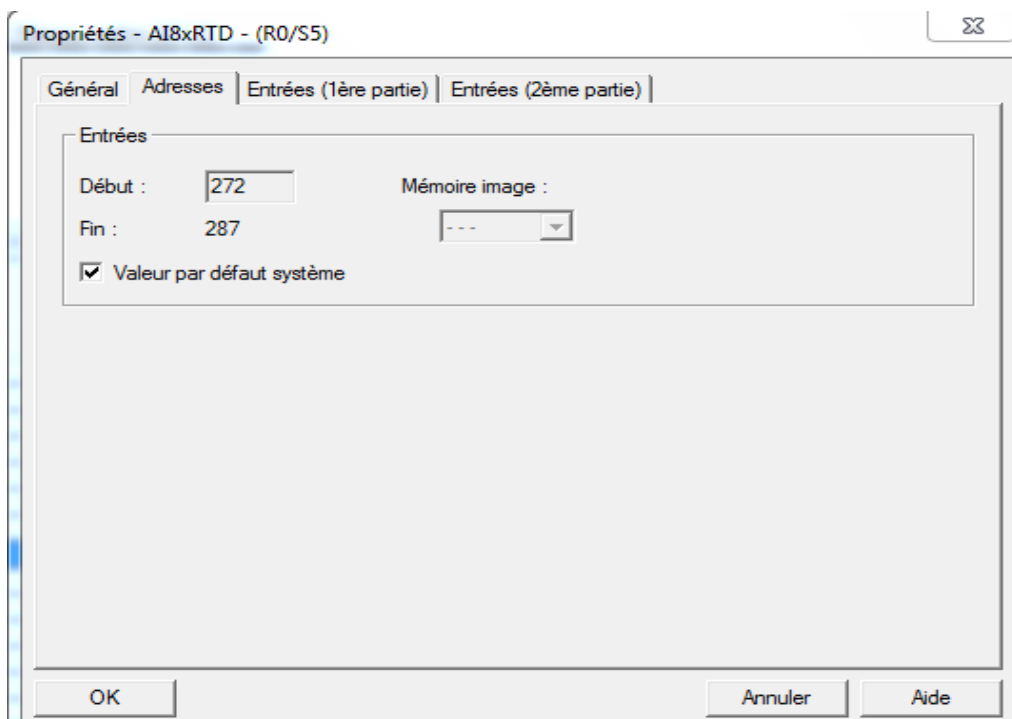


Figure 3.24:adresses du 2eme module AI8xRTD.



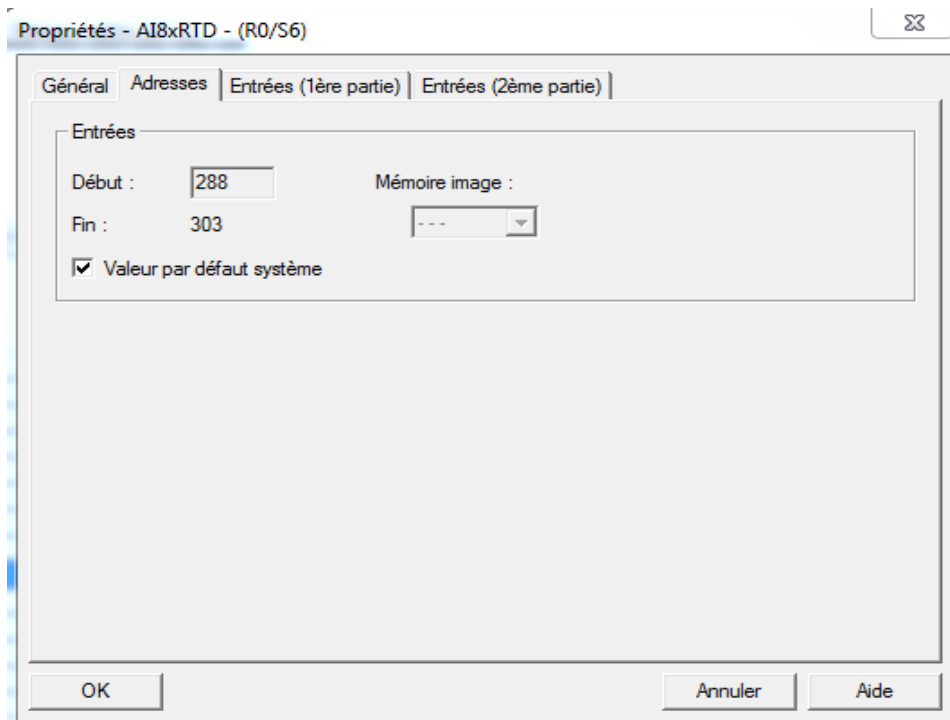








































































Figure 3.25:adresses du 3eme module AI8xRTD.

### 3.6.3 Les blocs d'organisation et fonctionnels

Notre programme STEP7 contient plusieurs blocs (figure 3.26):  
Bloc d'organisation OB1 et OB35  
Blocs fonctions FC et blocs fonctionnels FB et blocs de données DB

Nom de l'objet	Nom symbolique	Langag...	Ta...	Type
 Données système	---	---	---	SDB
 OB1	Cycle Execution	CONT	15...	Bloc d'organisati
 OB35	CYC_INT5	CONT	10...	Bloc d'organisati
 FB1	CYCLE EAU CHAUD LIGNE 1	CONT	994	Bloc fonctionnel
 FB2	cycle eau propre LIGNE 1	CONT	944	Bloc fonctionnel
 FB3	CYCLE EAU CHAUD LIGNE 2	CONT	10...	Bloc fonctionnel
 FB4	CYCLE EAU PROPRE LIGNE 2	CONT	962	Bloc fonctionnel
 FB5	CYCLE EAU CHAUD LIGNE 3	CONT	10...	Bloc fonctionnel
 FB6	CYCLE EAU PROPRE LIGNE 3	CONT	998	Bloc fonctionnel
 FB7	4 CYCLES M1 LIGNE 1	CONT	17...	Bloc fonctionnel
 FB8	4 CYCLES M2 LIGNE 1	CONT	17...	Bloc fonctionnel
 FB9	4 CYCLES CUVE 1 LIGNE 1	CONT	17...	Bloc fonctionnel
 FB10	4 CYCLES C2 LIGNE 1	CONT	17...	Bloc fonctionnel
 FB11	4 CYCLES M3 LIGNE 2	CONT	18...	Bloc fonctionnel
 FB12	4 CYCLES M4 LIGNE 2	CONT	18...	Bloc fonctionnel
 FB13	4 CYCLES C3 LIGNE 2	CONT	18...	Bloc fonctionnel
 FB14	4 CYCLES C4 LIGNE 2	CONT	18...	Bloc fonctionnel
 FB15	4 CYCLE M5 LIGNE 3	CONT	19...	Bloc fonctionnel
 FB16	4 CYCLE M6 LIGNE 3	CONT	19...	Bloc fonctionnel
 FB17	4 CYCLE C5 LIGNE 3	CONT	19...	Bloc fonctionnel
 FB18	4 CYCLE C6 LIGNE 3	CONT	19...	Bloc fonctionnel
 FB19	nettoyage EAU RECUPéré	CONT	580	Bloc fonctionnel
 FB20	NETTOYAGE CUVE SOUDE	CONT	692	Bloc fonctionnel
 FB21	NETTOYAGE CUVE ACIDE	CONT	692	Bloc fonctionnel
 FB22	preparation eau propre	CONT	184	Bloc fonctionnel
 FB23	préparation eau chaud	CONT	340	Bloc fonctionnel
 FB24	préparation cuve soude	CONT	320	Bloc fonctionnel
 FB25	PREPARATION CUVE ACIDE	CONT	320	Bloc fonctionnel
 FB26	reset ligne 2	CONT	116	Bloc fonctionnel
 FB27	reset ligne 3	CONT	114	Bloc fonctionnel
 FB28	pause ligne 3	CONT	38	Bloc fonctionnel
 FB29	reset ligne 1	CONT	104	Bloc fonctionnel
 FB41	CONT_C	SCL	14...	Bloc fonctionnel
 FC1	capteur température L1	CONT	118	Fonction
 FC2	capteur température L2	CONT	118	Fonction

Nom de l'objet	Nom symbolique	Langag...	Ta...	Type
 Données système	---	---	---	SDB
 OB1	Cycle Execution	CONT	15...	Bloc d'organisator
 OB35	CYC_INT5	CONT	10...	Bloc d'organisator
 FB1	CYCLE EAU CHAUD LIGNE 1	CONT	994	Bloc fonctionnel
 FB2	cycle eau propre LIGNE 1	CONT	944	Bloc fonctionnel
 FB3	CYCLE EAU CHAUD LIGNE 2	CONT	10...	Bloc fonctionnel
 FB4	CYCLE EAU PROPRE LIGNE 2	CONT	962	Bloc fonctionnel
 FB5	CYCLE EAU CHAUD LIGNE 3	CONT	10...	Bloc fonctionnel
 FB6	CYCLE EAU PROPRE LIGNE 3	CONT	998	Bloc fonctionnel
 FB7	4 CYCLES M1 LIGNE 1	CONT	17...	Bloc fonctionnel
 FB8	4 CYCLES M2 LIGNE 1	CONT	17...	Bloc fonctionnel
 FB9	4 CYCLES CUVE 1 LIGNE 1	CONT	17...	Bloc fonctionnel
 FB10	4 CYCLES C2 LIGNE 1	CONT	17...	Bloc fonctionnel
 FB11	4 CYCLES M3 LIGNE 2	CONT	18...	Bloc fonctionnel
 FB12	4 CYCLES M4 LIGNE 2	CONT	18...	Bloc fonctionnel
 FB13	4 CYCLES C3 LIGNE 2	CONT	18...	Bloc fonctionnel
 FB14	4 CYCLES C4 LIGNE 2	CONT	18...	Bloc fonctionnel
 FB15	4 CYCLE M5 LIGNE 3	CONT	19...	Bloc fonctionnel
 FB16	4 CYCLE M6 LIGNE 3	CONT	19...	Bloc fonctionnel
 FB17	4 CYCLE C5 LIGNE 3	CONT	19...	Bloc fonctionnel
 FB18	4 CYCLE C6 LIGNE 3	CONT	19...	Bloc fonctionnel
 FB19	nettoyage EAU RECUPéré	CONT	580	Bloc fonctionnel
 FB20	NETTOYAGE CUVE SOUDE	CONT	692	Bloc fonctionnel
 FB21	NETTOYAGE CUVE ACIDE	CONT	692	Bloc fonctionnel
 FB22	preparation eau propre	CONT	184	Bloc fonctionnel
 FB23	préparation eau chaud	CONT	340	Bloc fonctionnel
 FB24	préparation cuve soude	CONT	320	Bloc fonctionnel
 FB25	PREPARATION CUVE ACIDE	CONT	320	Bloc fonctionnel
 FB26	reset ligne 2	CONT	116	Bloc fonctionnel
 FB27	reset ligne 3	CONT	114	Bloc fonctionnel
 FB28	pause ligne 3	CONT	38	Bloc fonctionnel
 FB29	reset ligne 1	CONT	104	Bloc fonctionnel
 FB41	CONT_C	SCL	14...	Bloc fonctionnel
 FC1	capteur température L1	CONT	118	Fonction
 FC2	capteur température L2	CONT	118	Fonction

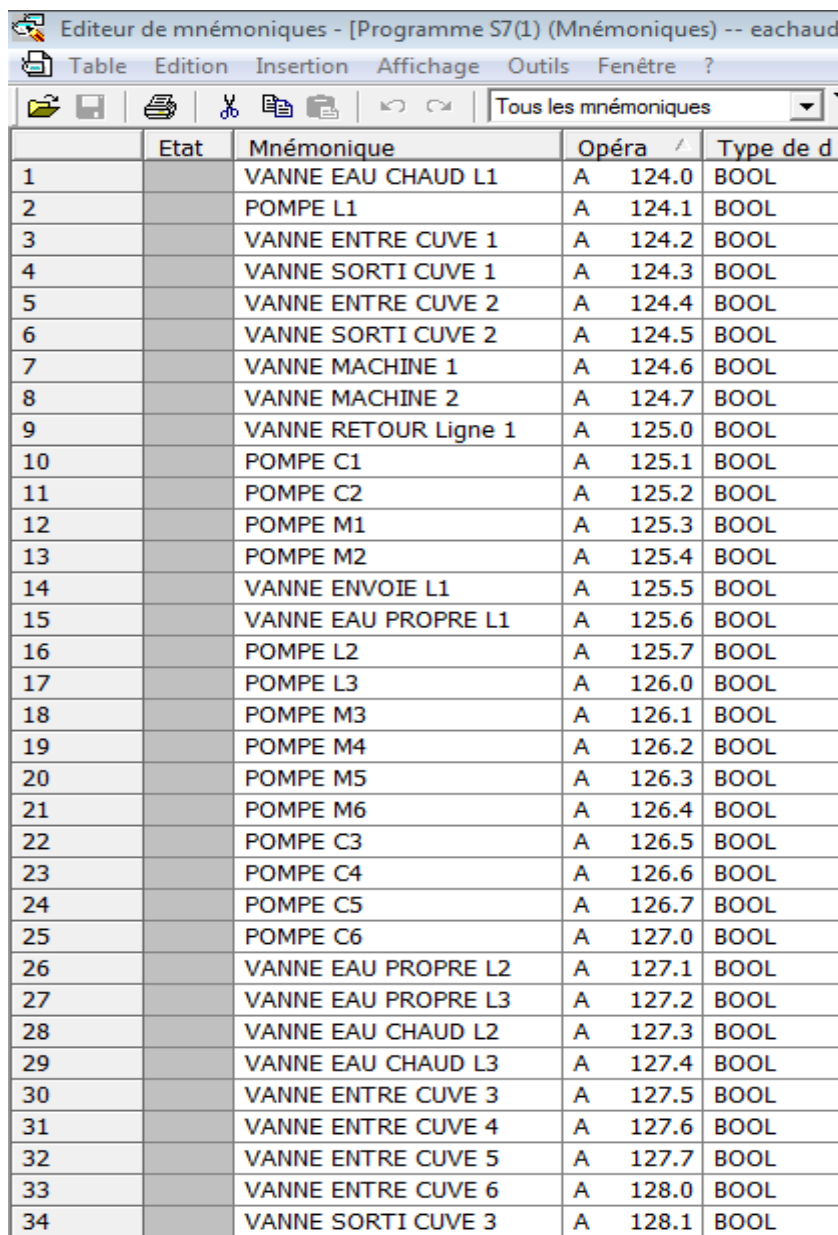
Nom de l'objet	Nom symbolique	Langag...	Ta...	Type
FC3	capteur température L3	CONT	118	Fonction
FC4	capteur concentration L1	CONT	118	Fonction
FC5	capteur concentration L2	CONT	118	Fonction
FC6	capteur concentration L3	CONT	118	Fonction
FC7	CAPTEUR NIVEAU ACIDE	CONT	142	Fonction
FC8	CAPTEUR NIVEAU SOUD E	CONT	142	Fonction
FC9	CAPTEUR NIVEAU EAU RECUP	CONT	130	Fonction
FC10	débit mètre L1	CONT	120	Fonction
FC11	débit mètre L2	CONT	120	Fonction
FC12	débit mètre L3	CONT	120	Fonction
FC13	température cuve eau ch	CONT	218	Fonction
FC14	CAPTEUR T ACIDE	CONT	218	Fonction
FC15	CAPTEUR T SOUDE	CONT	218	Fonction
FC16	CAPTEUR CONC CUVE SOUDE	CONT	152	Fonction
FC17	CAPT CONCENT CUVE ACIDE	CONT	152	Fonction
FC105	SCALE	LIST	244	Fonction
FC106	UNSCALE	LIST	324	Fonction
DB1		DB	36	DB d'instance du FB 1
DB2		DB	36	DB d'instance du FB 2
DB3		DB	36	DB d'instance du FB 3
DB4		DB	36	DB d'instance du FB 4
DB5		DB	36	DB d'instance du FB 5
DB6		DB	36	DB d'instance du FB 6
DB7		DB	36	DB d'instance du FB 7
DB8		DB	36	DB d'instance du FB 8
DB9		DB	36	DB d'instance du FB...
DB10		DB	36	DB d'instance du FB 9
DB11		DB	36	DB d'instance du FB...
DB12		DB	36	DB d'instance du FB...
DB13		DB	36	DB d'instance du FB...
DB14		DB	36	DB d'instance du FB...
DB15		DB	36	DB d'instance du FB...
DB16		DB	36	DB d'instance du FB...
DB17		DB	36	DB d'instance du FB...
DB18		DB	36	DB d'instance du FB...
DB19		DB	36	DB d'instance du FB...
DB20		DB	36	DB d'instance du FB...
DB21		DB	36	DB d'instance du FB...
DB22		DB	36	DB d'instance du FB...
DB23		DB	36	DB d'instance du FB...
DB24		DB	36	DB d'instance du FB...
DB25		DB	36	DB d'instance du FB...
DB26		DB	162	DB d'instance du FB...
DB27		DB	36	DB d'instance du FB...
DB28		DB	36	DB d'instance du FB...
DB29		DB	36	DB d'instance du FB...
DB30		DB	162	DB d'instance du FB...
DB31		DB	162	DB d'instance du FB...

Figure 3.26: Bloc d'organisation et blocs de fonctions et blocs fonctionnels et blocs de données.

### 3.6.4 . Création de programme :

#### a . Travailler avec l'éditeur de Mnémoniques

Le programme est créé pour gérer notre application, il est indispensable de créer la table des Mnémoniques pour utiliser la méthode d'adressage relatif dans le but de bien organiser le programme et qu'il soit compréhensible aux autres.



The screenshot shows a software window titled 'Editeur de mnémoniques - [Programme S7(1) (Mnémoniques) -- eachaud]'. The menu bar includes 'Table', 'Edition', 'Insertion', 'Affichage', 'Outils', 'Fenêtre', and '?'. The toolbar contains icons for file operations and editing. A dropdown menu is open, showing 'Tous les mnémoniques'. Below the toolbar is a table with the following data:

	Etat	Mnémonique	Opéra /	Type de d
1		VANNE EAU CHAUD L1	A 124.0	BOOL
2		POMPE L1	A 124.1	BOOL
3		VANNE ENTRE CUVE 1	A 124.2	BOOL
4		VANNE SORTI CUVE 1	A 124.3	BOOL
5		VANNE ENTRE CUVE 2	A 124.4	BOOL
6		VANNE SORTI CUVE 2	A 124.5	BOOL
7		VANNE MACHINE 1	A 124.6	BOOL
8		VANNE MACHINE 2	A 124.7	BOOL
9		VANNE RETOUR Ligne 1	A 125.0	BOOL
10		POMPE C1	A 125.1	BOOL
11		POMPE C2	A 125.2	BOOL
12		POMPE M1	A 125.3	BOOL
13		POMPE M2	A 125.4	BOOL
14		VANNE ENVOIE L1	A 125.5	BOOL
15		VANNE EAU PROPRE L1	A 125.6	BOOL
16		POMPE L2	A 125.7	BOOL
17		POMPE L3	A 126.0	BOOL
18		POMPE M3	A 126.1	BOOL
19		POMPE M4	A 126.2	BOOL
20		POMPE M5	A 126.3	BOOL
21		POMPE M6	A 126.4	BOOL
22		POMPE C3	A 126.5	BOOL
23		POMPE C4	A 126.6	BOOL
24		POMPE C5	A 126.7	BOOL
25		POMPE C6	A 127.0	BOOL
26		VANNE EAU PROPRE L2	A 127.1	BOOL
27		VANNE EAU PROPRE L3	A 127.2	BOOL
28		VANNE EAU CHAUD L2	A 127.3	BOOL
29		VANNE EAU CHAUD L3	A 127.4	BOOL
30		VANNE ENTRE CUVE 3	A 127.5	BOOL
31		VANNE ENTRE CUVE 4	A 127.6	BOOL
32		VANNE ENTRE CUVE 5	A 127.7	BOOL
33		VANNE ENTRE CUVE 6	A 128.0	BOOL
34		VANNE SORTI CUVE 3	A 128.1	BOOL

35		VANNE SORTI CUVE 4	A	128.2	BOOL
36		VANNE SORTI CUVE 5	A	128.3	BOOL
37		VANNE SORTI CUVE 6	A	128.4	BOOL
38		VANNE MACHINE 3	A	128.5	BOOL
39		VANNE MACHINE 4	A	128.6	BOOL
40		VANNE MACHINE 5	A	128.7	BOOL
41		VANNE MACHINE 6	A	129.0	BOOL
42		VANNE ENVOIE L2	A	129.1	BOOL
43		VANNE ENVOIE L3	A	129.2	BOOL
44		VANNE ACIDE L1	A	129.3	BOOL
45		VANNE ACIDE L2	A	129.4	BOOL
46		VANNE ACIDE L3	A	129.5	BOOL
47		VANNE SOUDE L1	A	129.6	BOOL
48		VANNE SOUDE L2	A	129.7	BOOL
49		VANNE SOUDE L3	A	130.0	BOOL
50		VANNE EAU RECUPERE L1	A	130.1	BOOL
51		VANNE EAU RECUPERE L2	A	130.2	BOOL
52		VANNE EAU RECUPERE L3	A	130.3	BOOL
53		VANNE EGOUT L1	A	130.4	BOOL
54		VANNE EGOUT L2	A	130.5	BOOL
55		VANNE EGOUT L3	A	130.6	BOOL
56		VANNE RETOUR SOUDE L1	A	130.7	BOOL
57		VANNE RETOUR SOUDE L2	A	131.0	BOOL
58		VANNE RETOUR SOUDE L3	A	131.1	BOOL
59		VANNE RETOUR ACIDE L1	A	131.2	BOOL
60		VANNE RETOUR ACIDE L2	A	131.3	BOOL
61		VANNE RETOUR ACIDE L3	A	131.4	BOOL
62		VANNE RETOUR ER L1	A	131.5	BOOL
63		VANNE RETOUR ER L2	A	131.6	BOOL
64		VANNE RETOUR ER L3	A	131.7	BOOL
65		VANNE RETOUR EC L1	A	132.0	BOOL
66		VANNE RETOUR EC L2	A	132.1	BOOL
67		VANNE RETOUR EC L3	A	132.2	BOOL
68		VANNE RETOUR Ligne 2	A	132.3	BOOL

69		POMPE SOURCE EP			
70		VANNE SOURCE EP			
71		VANNE SOURCE ACIDE			
72		VANNE SOURCE SOUDE			
73		POMPE SOURCE ACIDE			
74		POMPE SOURCE SOUDE			
75		pompe chauffage EC			
76		pompe chauffage SO			
77		pompe chauffage AC			
78		VANNE CHAUFFAGE EC			
79		VANNE CHAUFFAGE SO			
80		VANNE CHAUFFAGE AC			

81		NB CUVE 1	E	124.0	BOOL
82		NB CUVE 2	E	124.1	BOOL
83		NB BAC 1	E	124.2	BOOL
84		NB BAC 2	E	124.3	BOOL
85		NB BAC 3	E	124.4	BOOL
86		NB BAC 4	E	124.5	BOOL
87		NB BAC 5	E	124.6	BOOL
88		NB BAC 6	E	124.7	BOOL
89		NB CUVE 3	E	125.0	BOOL
90		NB CUVE 4	E	125.1	BOOL
91		NB CUVE 5	E	125.2	BOOL
92		NB CUVE 6	E	125.3	BOOL

Figure 3.27: quelques mnémoniques d'entrées et de sorties.

### **b . Le programme en langage contacts**

Il existe plusieurs méthodes pour la programmation l'un de cette méthodes est la programmation en langage contacts autrement dit (**Ladder**) ou (**Cont**) nous avons passé par une étape plus importante pour arriver au langage **Cont** c'est la Conversion de GRAFCET vers Ladder cette méthode est la plus facile elle consiste a converti les étapes et les transitions et les actions en contactes (fermés et ouverts) dans notre cas on a converti les actions et les transitions en sorties/entrées ,les étapes en bascules **SR** et aussi en entrées ,l'utilisation de bascule **SR** rend la tâche plus simple à la place de mettre plusieurs réseaux on a fait seulement quelque réseaux grâce à cette bascule.

**SR** (Bascule mise à '0', mise à '1') propre au step 7. Cette opération exécute la mise à '1' si l'état de signale est '1' à l'entrée Set '0' à l'entrée R.

Si l'état de signale est '0' à l'entrée Set '1' à l'entrée R et si l'état de signale est à '1' à l'entrée Set R, la bascule est mise à '0'.

### **c . Simulations des programmes en utilisant S7-PLCSIM**

L'application S7-PLCSIM nous permet de simuler le fonctionnement d'un automate programmable S7-300. Nous pouvons tester nos programmes de commande à partir de S7-PLCSIM sans besoin de faire la liaison au matériel S7-300.

S7-PLCSIM fournit une interface utilisateur graphique permettant de visualiser et de modifier des variables du programme de commande, d'exécuter la CPU de simulation en mode Cycle unique ou Cycle continu, ainsi que de modifier l'état de fonctionnement de l'AP de simulation.

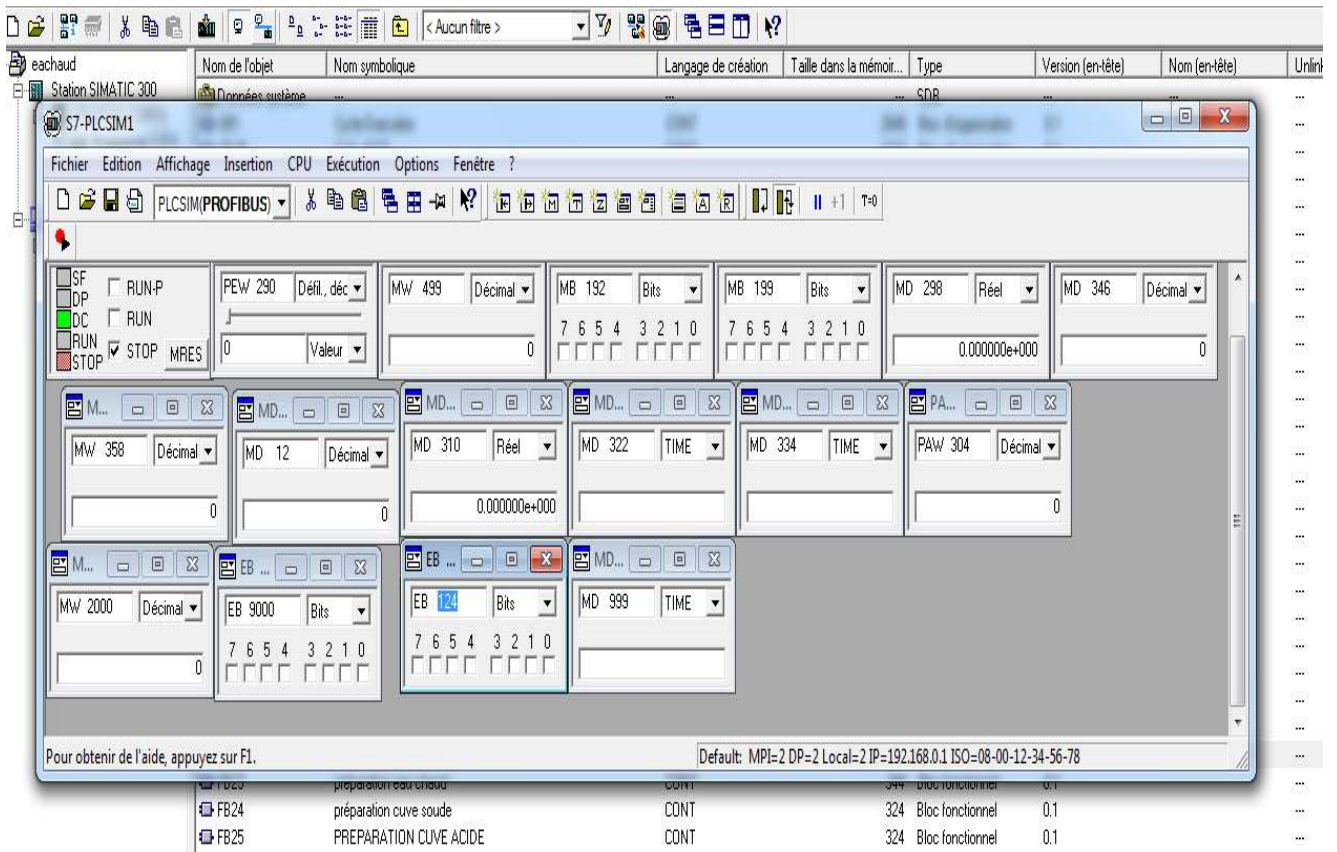


Figure 3.28:S7-PLCSIM1 en Mode Stop.

**d . Un exemple plus simple de programmation :**

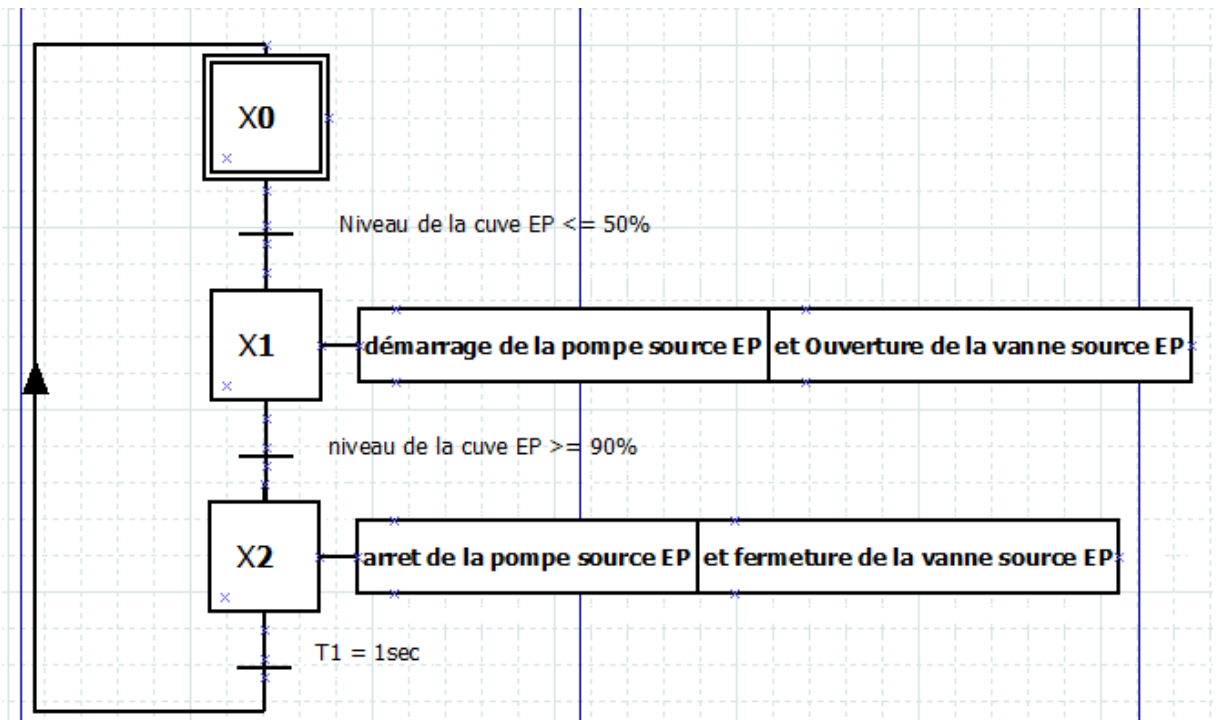


Figure 3.29:simple GRAFCET de Préparation CUVE EAU PROPRE (EP).



Le bloc **FB22** représente la préparation de cuve EP

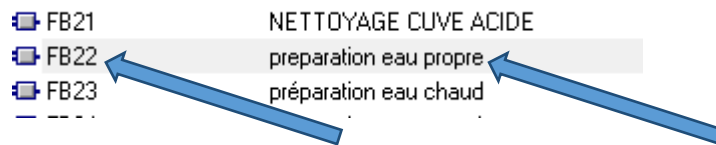


Figure 3.30:le Bloc FB22.

Nous avons développé le GRAFCET de préparation cuve EP au niveau de bloc FB22 dans 6 réseaux.

Le premier réseau représente le démarrage de la pompe source EP et l'ouverture de la vanne source EP si le niveau de la cuve EP inférieur ou égal à 50% (les sorties A132.4 et A132.5 sont à 1 si le bit M193.6 est à 1).

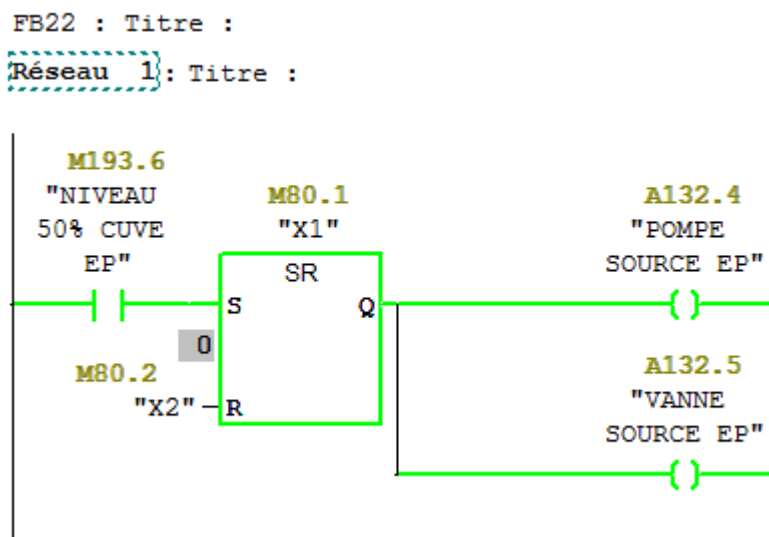


Figure 3.31:Conversion de l'étape X1 de GRAFCET en langage Contact.

Le réseau 2 représente l'arrêt de la pompe source EP et la fermeture de la vanne source EP si le niveau de la cuve EP supérieur ou égal à 90% (les sorties A132.4 et A132.5 sont à 0 si le bit M193.7 est à 1) pour cela on a utilisé les reset ---(R) ---.

**REMARQUE :** on a modifié la valeur de temporisateur T1 de 1sec à 1M40sec seulement pour la simulation le temporisateur T1 est un temporisateur del'opération---**(SE)(Démarrer temporisationsousformederetardàla montée)**

Cetteopérationdémarrrelatemporisationindiquéeavecla<valeur de

temps>donnéesileRLG présenteun

frontmontant.L'interrogationà1del'étatdesignalfournitunrésultatégalà1lorsquela  
valeurdetemps s'est écouléesanserreuret queleRLGesttoujourségal à

1. Sile RLG passede1à0alorsquelatemporisat

s'exécute,latemporisat

ionestremiseàZéro.Danscecas,l'interrogationà1del'étatdesign  
al fournit unrésultat égal à 0.  
Le rôle de T1 est de reseter l'étape X2.

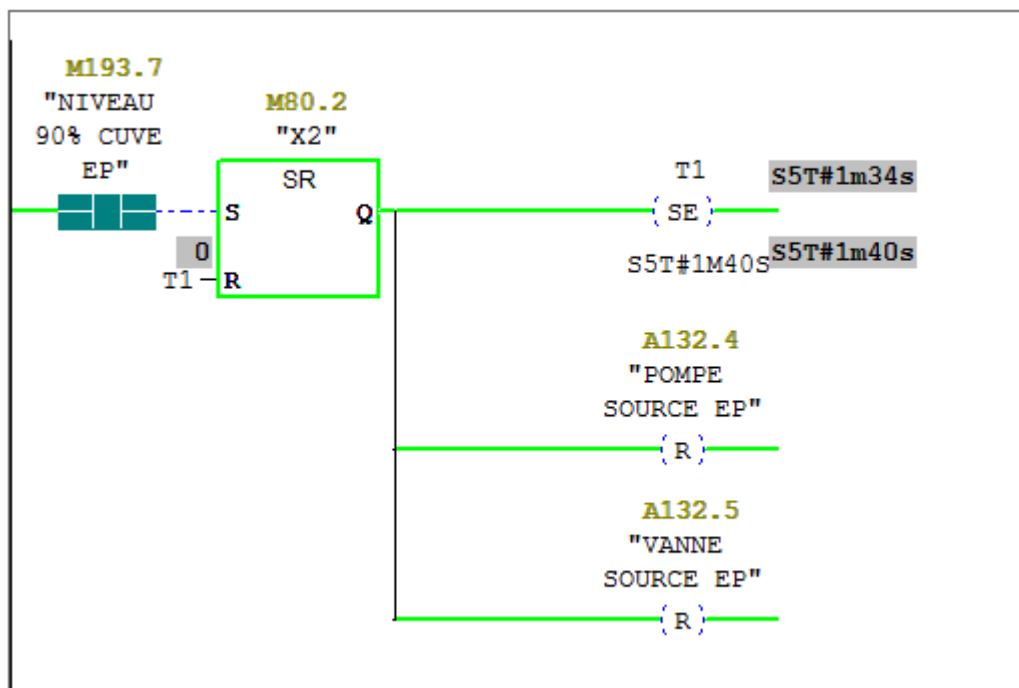


Figure 3.32:Conversion de l'étape X2 de GRAFCET en langage Contact.

Le réseau 3 représente la fonction « SCALE » le bloc FC105.

Dans ce réseau , nous avons traité un exemple de l'acquisition de niveau de la cuve EP.

L'acquisitiondesvaleursde niveau de la cuve EP sefaitàpartirdelafonctionFC105

« SCALE »qui représente une fonction de mise à l'échelleinstallée dans la bibliothèque du STEP7.

•Description du bloc de fonction :

La fonction mise à l'échelle (FC 105 "SCALE") prend une valeur entière (IN) et la convertit selon l'équation ci-après en une valeur réelle exprimée en unités physiques, comprises entre une limite inférieure (LO\_LIM) et une limite supérieure (HI\_LIM). Le résultat est écrit dans le paramètre OUT (voir figure ...). Si la valeur entière d'entrée se situe en dehors de la plage définie pour son type (bipolaire ou unipolaire), la sortie (OUT) est saturée à la valeur la plus proche de la limite inférieure (LO\_LIM) ou supérieure (HI\_LIM) et une erreur est signalée.

**Plages de mesure :**

- **BIPOLAIRE:** La valeur entière d'entrée est supposée être comprise entre -27648 et 27648
- **UNIPOLAIRE:** La valeur entière d'entrée est supposée être comprise entre 0 et 27648.

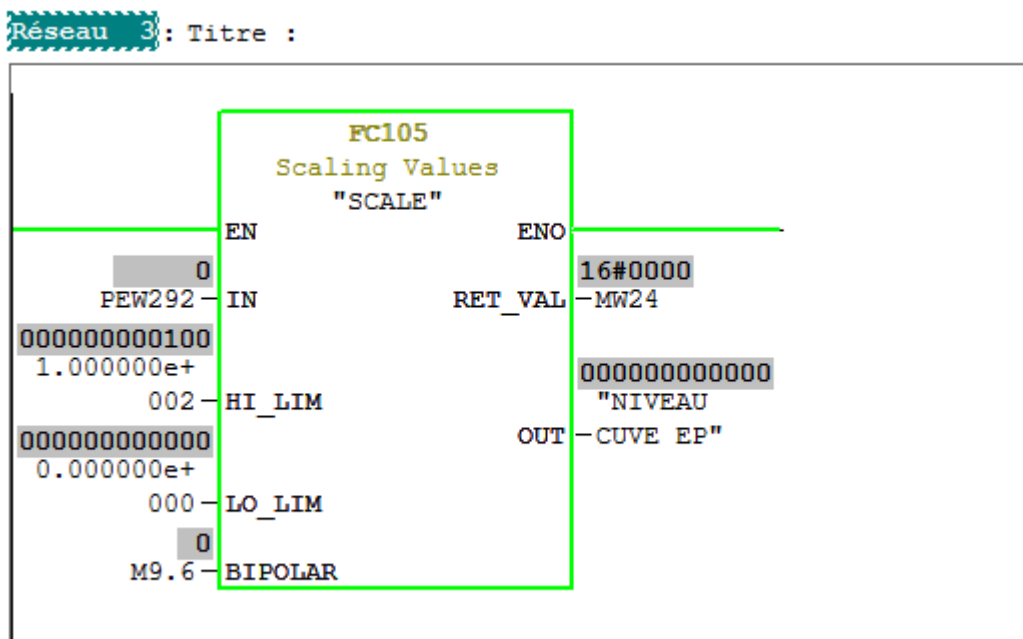


Figure 3.33: le Bloc Fonction "SCALE" FC105.

· Test de la fonction:

Nous avons testé la fonction FC105 et l'acquisition des valeurs de niveau de la cuve EP à l'aide du simulateur intégré dans STEP7 « S7-PLCSIM ». La figure ci-dessous montre un exemple de chargement de la fonction FC105.

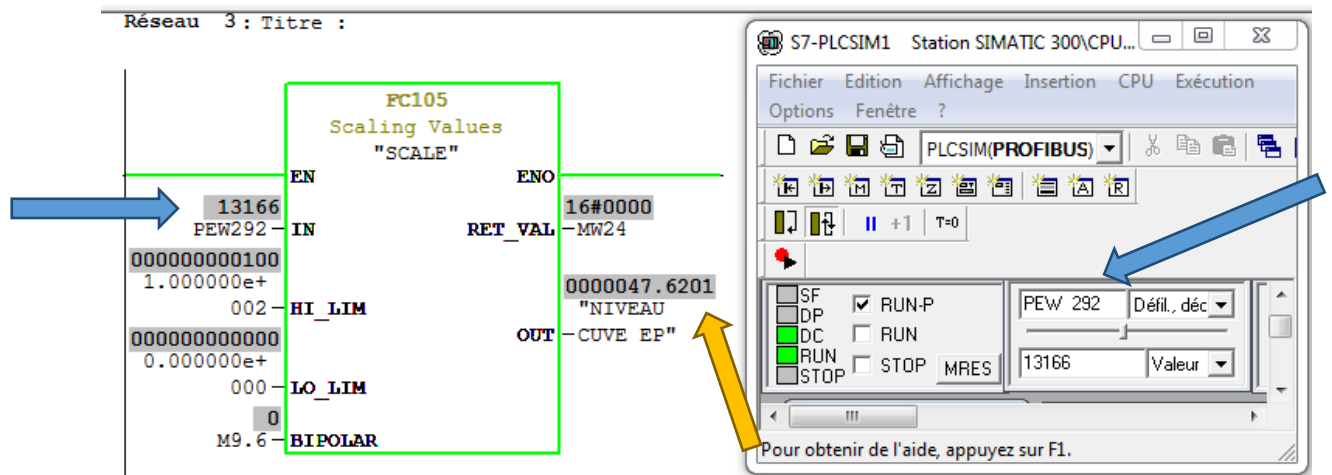


Figure 3.34: Chargement de fonction FC105.

Le réseau 4 représente une comparaison si le bit MD258 est inférieur ou égal à 50% cette condition mettre le bit M193.6 à 1.

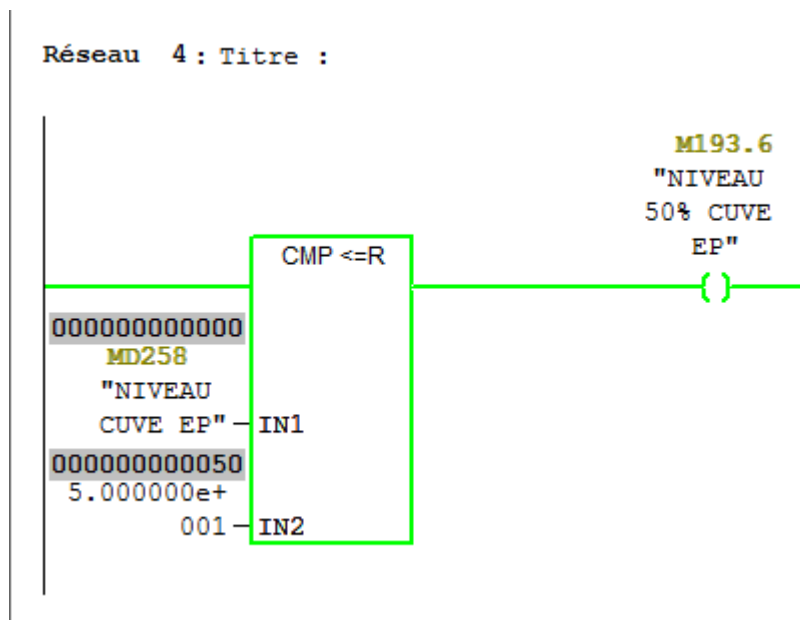


Figure 3.35: Fonction de comparaison.

Et si on fait varier la valeur de PIW292 plus de 50% au niveau de S7-PLCSIM on aura le bit M193.6 à 0 :

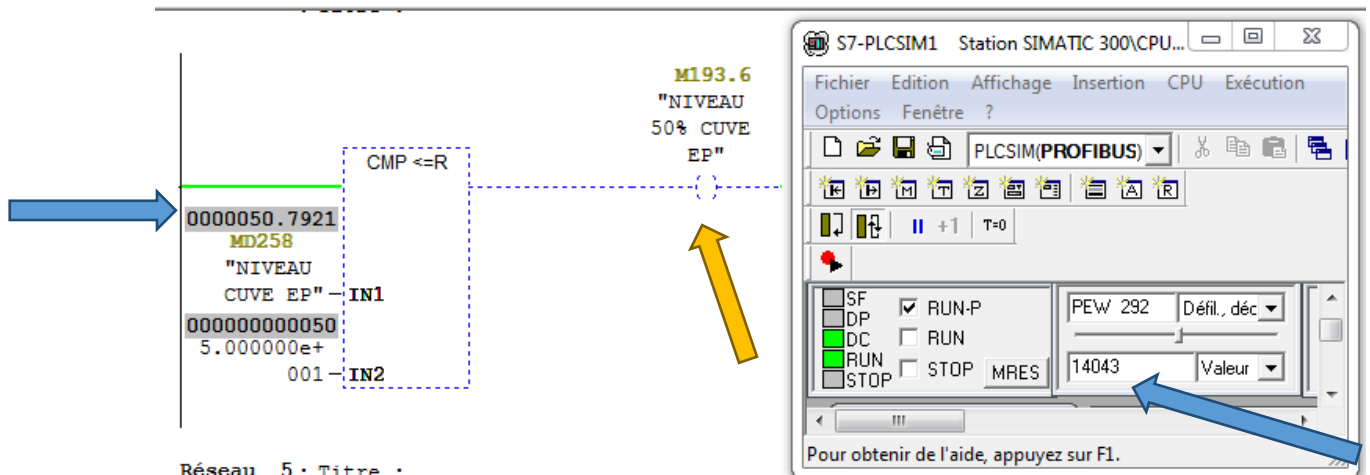


Figure 3.36:fonction de comparaison.

Le réseau 5 représente une comparaison si le bit MD258 est supérieur ou égal à 90% cette condition mettre le bit M193.7 à 1 dans la figure suivante le bit est à 0 parce que le niveau de la cuve EP est inférieur à 90%.

Réseau 5 : Titre :

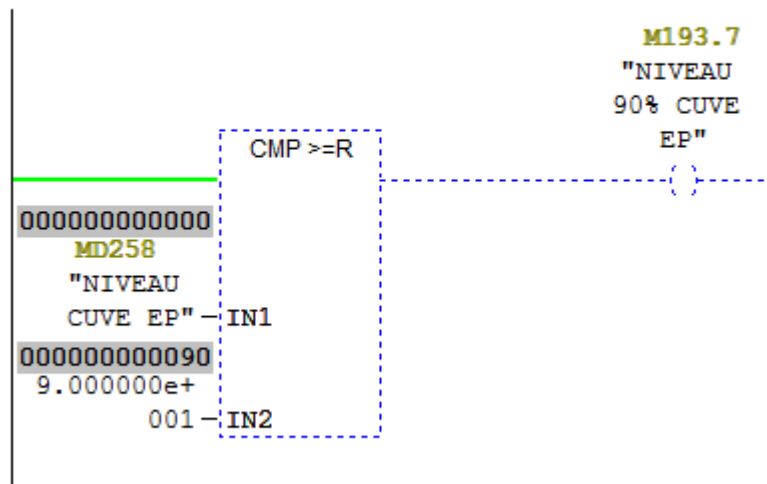


Figure 3.37:Fonction de comparaison.

Et si on fait varier la valeur de PIW292 plus que 90% au niveau de S7-PLCSIM on aura le bit M193.7 à 1 :

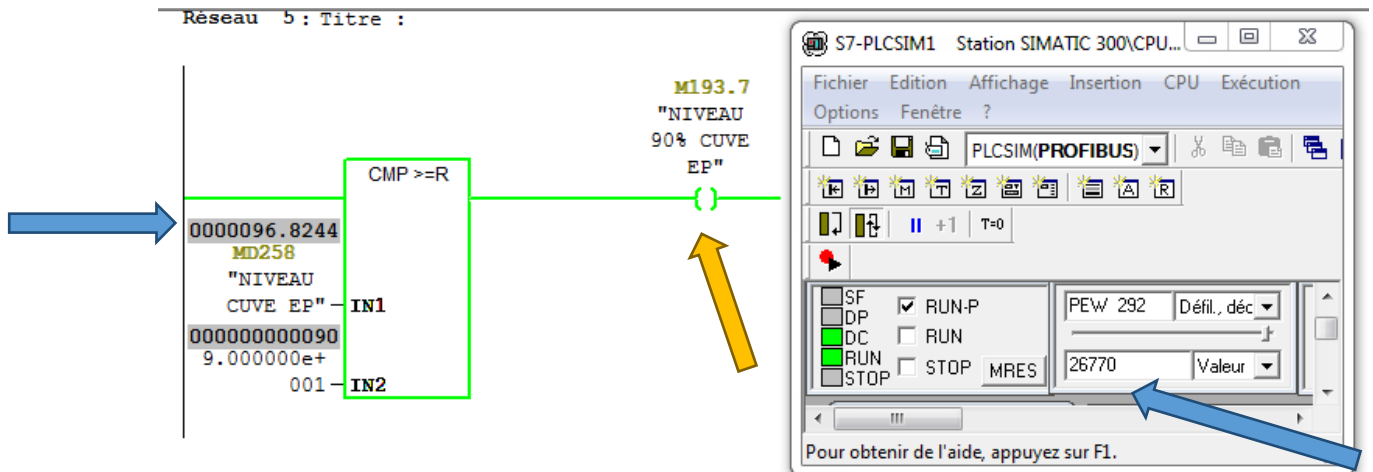


Figure 3.38:fonction de comparaison.

Et pour résoudre le fameux problème des alarmes on a utilisé l'exemple suivant parmi les exemples qu'on met.

Le réseau 6 représente le déclenchement d'une Alarme lorsque le niveau de la cuve EP est inférieur ou égal à 14% (lorsque le bit M188.5 est à 1).

Réseau 6 : Titre :

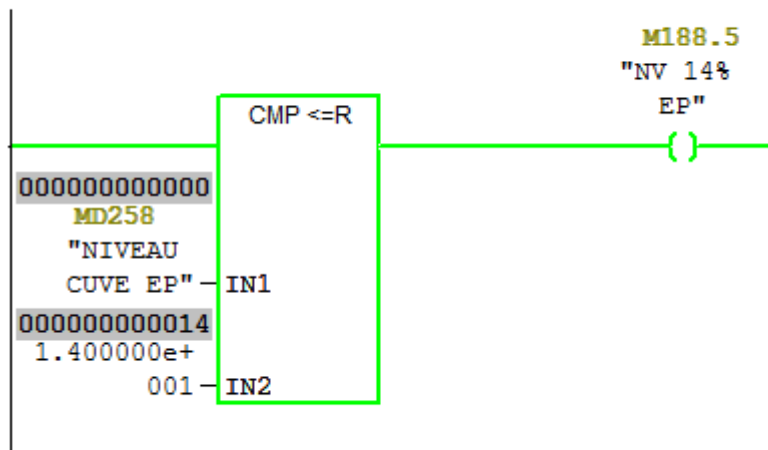


Figure 3.39:Alarme de niveau de la Cuve EP.

Et si on fait varier la valeur de PIW292 plus que 14% au niveau de S7-PLCSIM on aura le bit M188.5 à 0 :

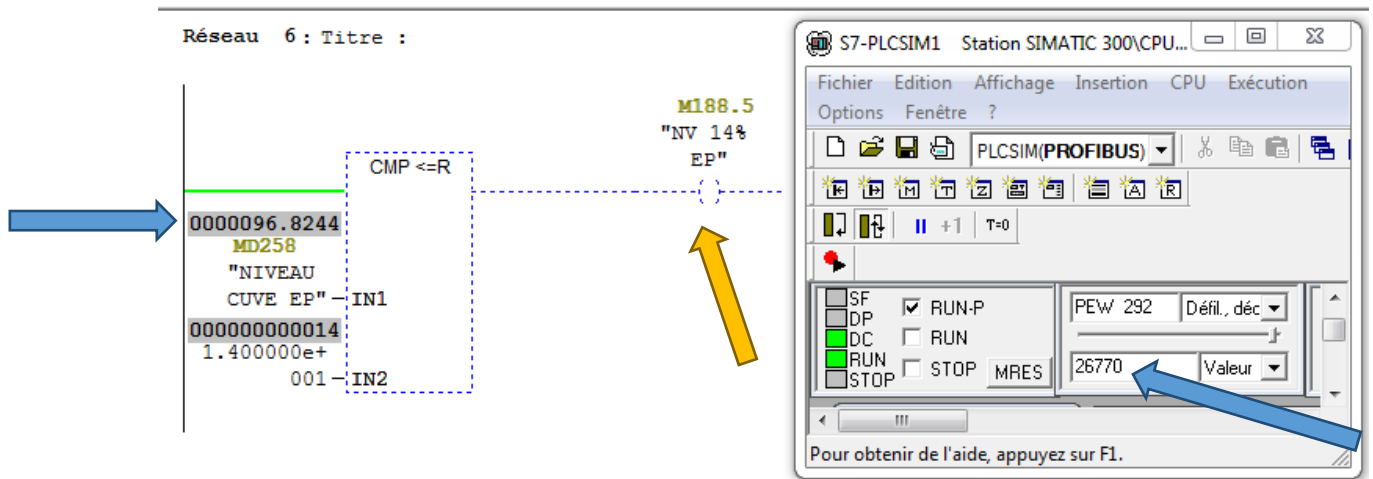


Figure 3.40: pas d'alarme le niveau est supérieur à 14%.

### D.1 L'APPEL DES BLOCS DE SOUS-PROGRAMME:

Avant la simulation de bloc FB22 il existe une tâche obligatoire est :

L'Appel de ce bloc dans le bloc d'organisation OB1 c'est une tâche plus simple il suffit de choisir blocs FB à gauche et faire glisser le bloc FB22 dans le réseau de OB1 après on charge le programme et on simule.

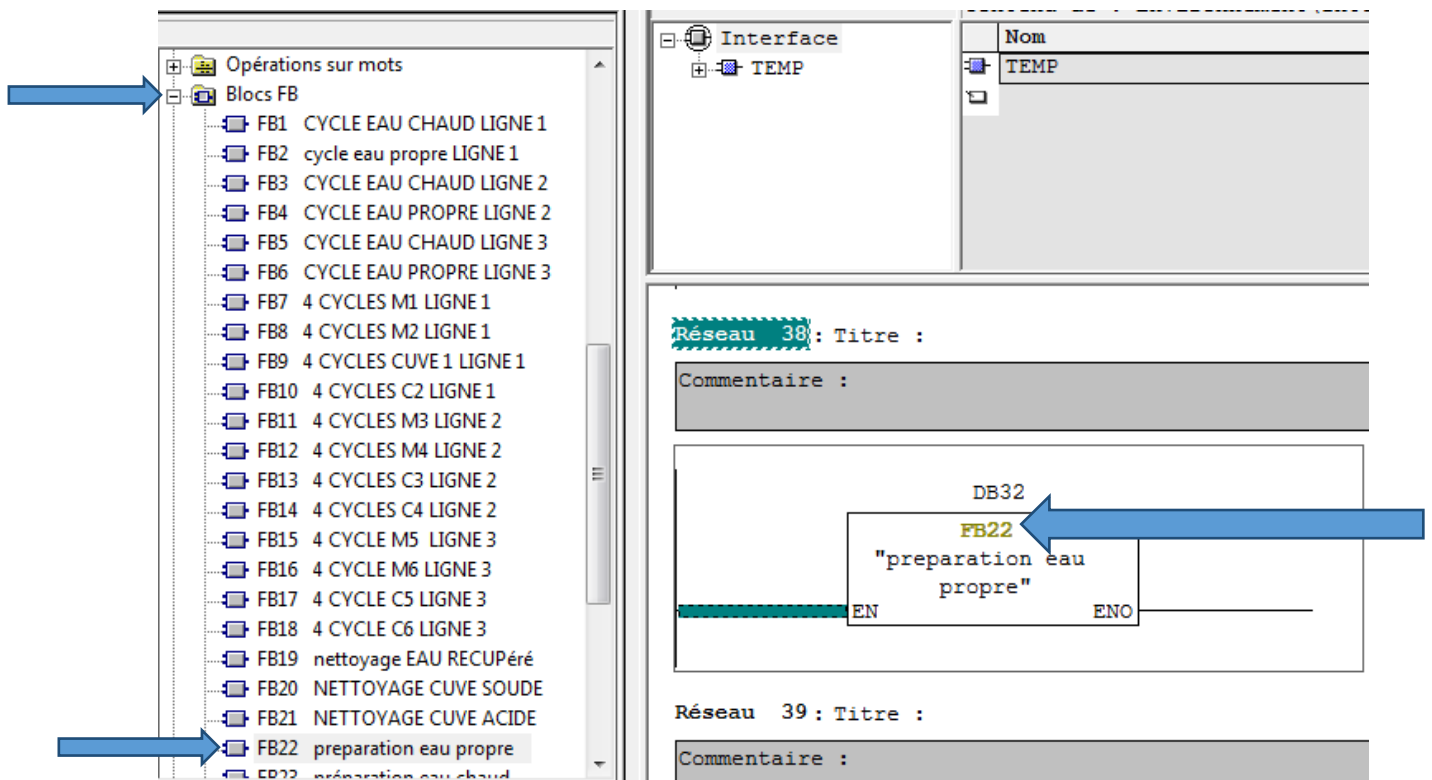


Figure 3.41: appel de FB22 dans OB1.

## D.2 ETAPE DE DEMARRAGE CYCLE EC :

Au niveau de bloc FB1 qu'elle existe cette étape, le réseau 1 de bloc FB1 contient les bits suivants :

- M0.6 qui représente le bit de choix de cycle EAU CHAUDE LIGNE 1.
- M0.0 qui représente le bit de l'étape initiale.
- M0.1 qui représente le bit de choix de la LIGNE 1.
- M194.4 qui représente le bit de la température niveau bas d'eau chaude.
- (M0.2 ou M0.3 ou M0.4 ou M0.5) représentent Respectivement le choix de la Machine 1 ou Machine 2 ou Cuve 1 ou Cuve 2.
- M599.3 qui représente le bit d'annuler le nettoyage de la LIGNE 1.

D'abord l'appel de bloc FB1 au niveau de bloc OB1 comme de suite :

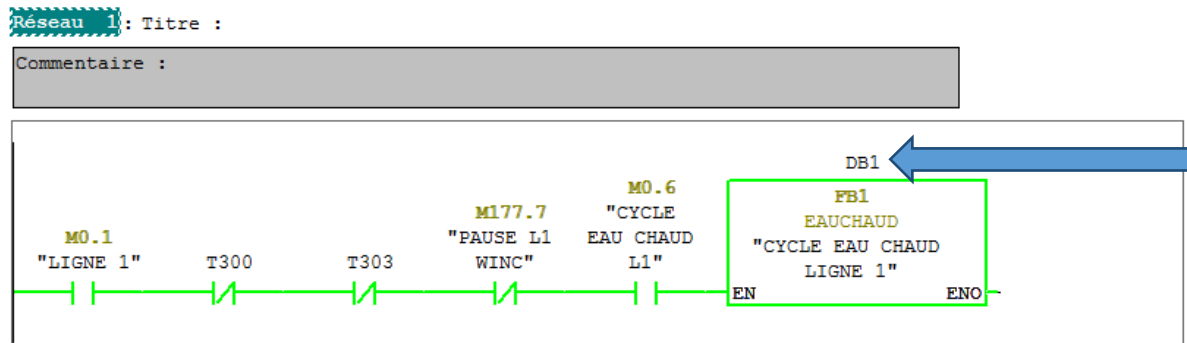


Figure 3.42:appel de bloc FB1.

**REMARQUE :** comme représente la figure précédente pour les blocs FB il est nécessaire de créer un bloc de donnée par l'écriture de DBx en dessus de bloc FBx on a mis DB1 pour le FB1.

Ensuite l'étape de démarrage nettoyage cycle eau chaude ligne 1 :



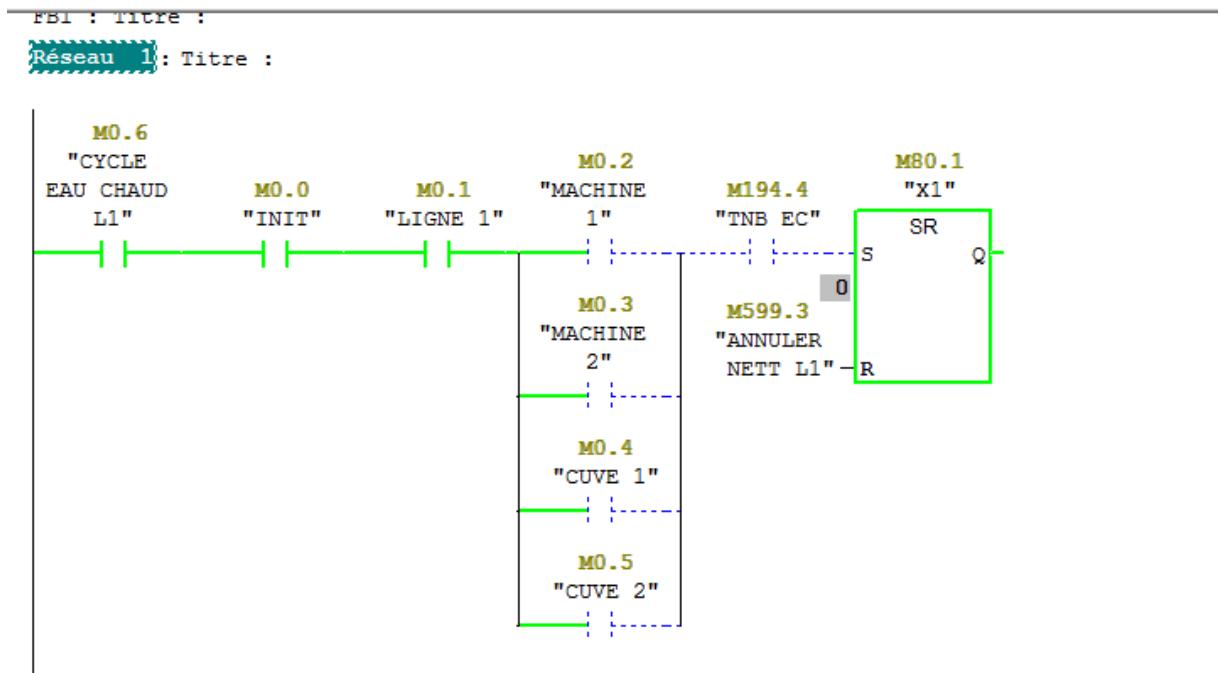


Figure 3.43: réseau 1 de bloc FB1.

### D.3 D'AUTRE OPERATION QU'ON A UTILISEE DANS NOTRE PROGRAMME :

Au niveau de bloc FC13 nous avons utilisé dans le réseau 4 deux opérations plus importante et plus utilisable dans la programmation ce sont :

**L'opération ROUND** (Arrondir par excès ou par défaut) :

#### Description de l'opération :

Cette opération lit le contenu du paramètre d'entrée IN comme nombre à virgule flottante et le convertit en nombre entier de 32 bits, le résultat, qui est le nombre entier le plus proche, est rangé dans le paramètre de sortie OUT, Si le nombre à virgule flottante se situe exactement entre deux nombres entiers, le nombre pair est pris comme résultat, en cas de débordement, ENO est mis a 0.

**L'opération DI\_R** (convertir entier de 32 bits en réel) :

#### Description de l'opération :

Cette opération lit le contenu du paramètre d'entrée IN comme nombre entier de 32 bits et le convertit en nombre à virgule flottante, le résultat est rangé dans le paramètre de sortie OUT, ENO et EN ont toujours un état de signal identique.

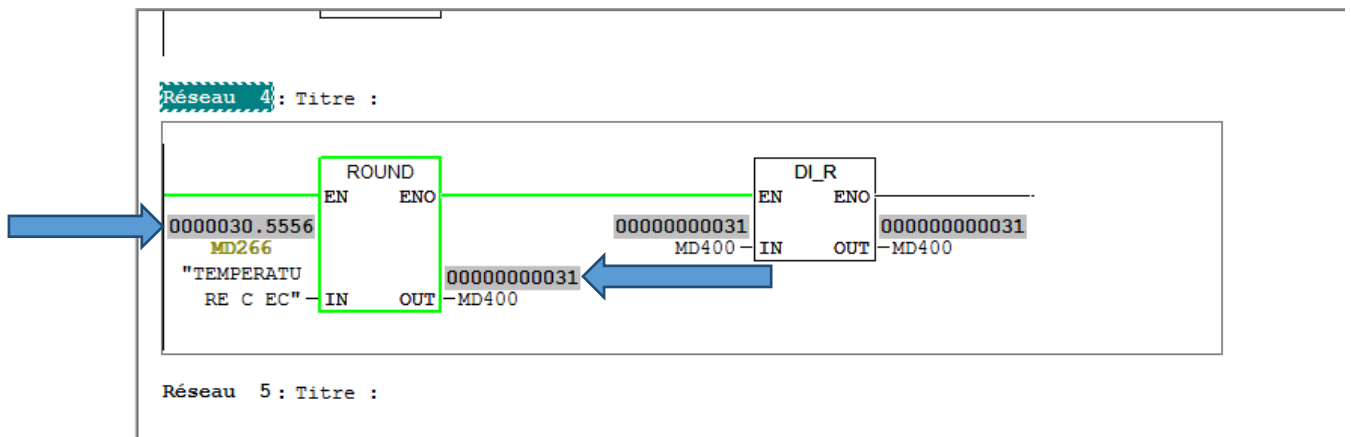


Figure 3.44:l'opération ROUND et l'opération DI\_R.

Parmi les problématiques de la station NEPest de faire un programme qui permet aux ingénieurs et operateurs de mettre la station NEP qui est en court de nettoyage en mode pause et de reprendre le nettoyage (continu depuis la dernière étape) et après annuler le nettoyage ou la préparation, et dans notre programme nous avons introduit cette étape au niveau de bloc d'organisation OB1 comme de suite :

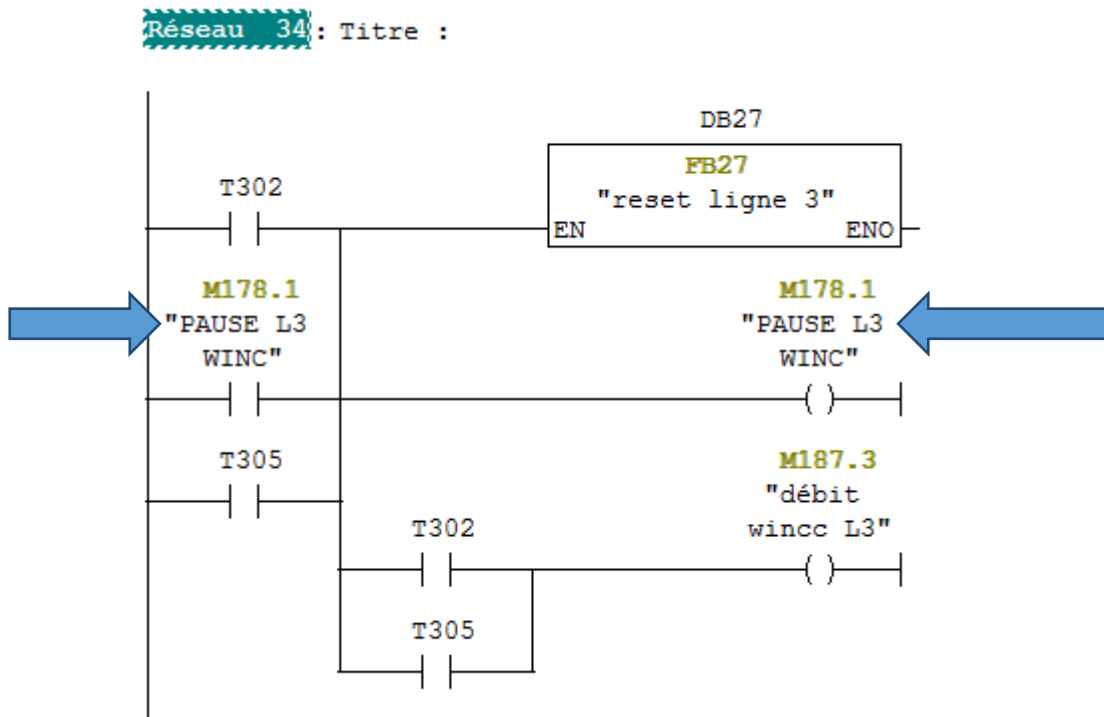


Figure 3.45:Pause de la LIGNE 3.

**REMARQUE :** Notre bloc OB1 est constitué de plus de 51 réseaux la moitié des réseaux sont pour l'appel des blocs FB et FC.

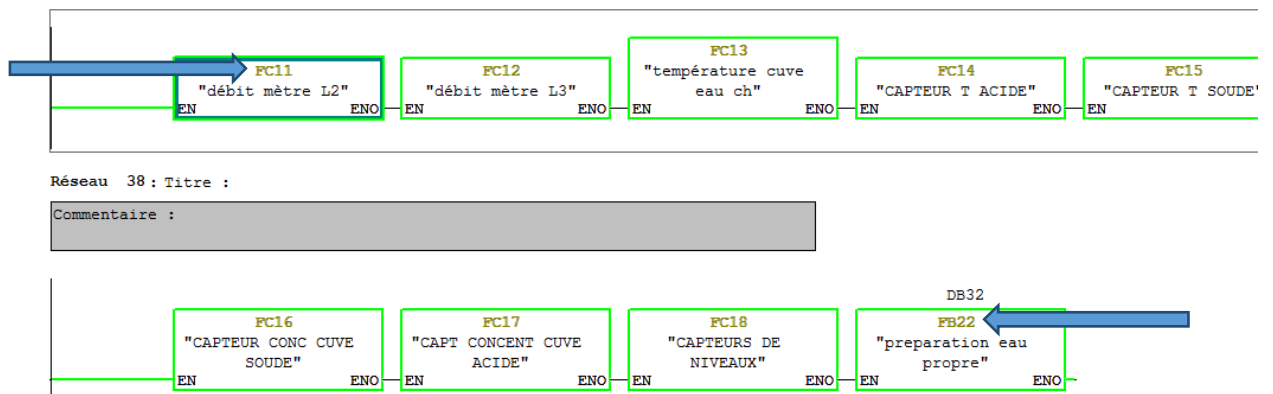


Figure 3.46: l'appel des blocs FB et FC dans OB1.

Au niveau de l'OB1 et dans le réseau 3 nous avons utilisé un temporisateur T27 de base de temps 1sec veut dire à chaque 1sec il marque 1 impulsion, dans le but de résoudre un autre problème de la station NEP ancienne.

### Description de l'opération

**S\_SEVERZ** (Paramétrer et démarrer temporisation sous forme de retard à la montée

Mémorisé)

Cette opération démarre la temporisation précisée en cas de front montant à l'entrée de

Démarrage S. Un changement d'état de signal est toujours nécessaire pour activer une

Temporisation. La valeur de temps indiquée à l'entrée TW continue à s'écouler même si l'état de signal à l'entrée S passe à 0 avant que la temporisation n'ait expiré. L'état de signal à la sortie Q égale 1 lorsque la temporisation a expiré, quel que soit l'état de signal à l'entrée S.

Si l'état de signal à l'entrée S passe de 0 à 1 alors que la temporisation s'exécute, cette

Dernière est redémarrée avec la valeur de temps indiquée.

En cas de passage de 0 à 1 à l'entrée de remise à zéro R, la temporisation est remise à zéro

Quel que soit le RLG à l'entrée S. L'état de signal à la sortie Q est alors 0.

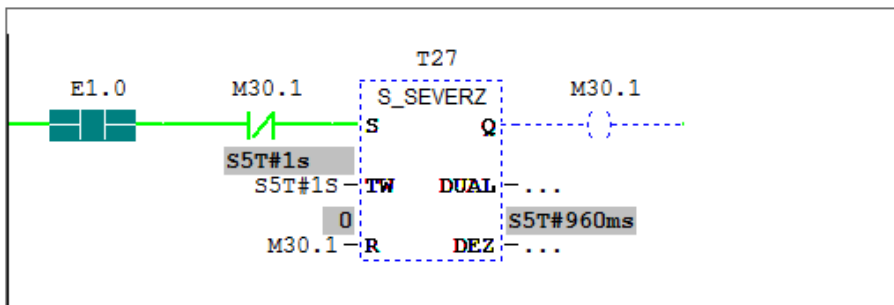
La valeur de temps en cours peut être lue en format binaire à la sortie DUAL et en format

Décimal codé binaire à la sortie DEZ. La valeur de temps en cours correspond à la valeur

Initiale en TW moins la valeur de temps écoulée depuis le démarrage de la temporisation.

Réseau 3: Titre :

Commentaire :



Réseau 4: Titre :

Figure 3.47: temporisation sous forme de retard à la montée.

### 3.6.5 Etude de régulation :

#### a . INTRODUCTION:

La commande PID est dite aussi (correcteur, régulateur, contrôleur), se compose de trois termes P, I et D, d'où le 'P' correspond au terme proportionnel, 'I' pour Terme intégral et 'D' pour le terme dérivé de la commande. Les régulateurs PID sont Probablement les plus largement utilisés dans le contrôle industriel. Même les plus Complexes systèmes de contrôle industriel peut comporter un réseau de contrôle dont Le principal élément de contrôle est un module de contrôle PID.

Le régulateur PID est une simple implémentation de retour d'information (Feedback). Il a la capacité d'éliminer la compensation de l'état d'équilibre grâce à L'action intégrale, et il peut anticiper le futur grâce à une action dérivée.

#### b Principe général d'un correcteur PID :

L'erreur observée est la différence entre la consigne et la mesure. Le PID permet Trois actions en fonction de cette erreur :

- Une action Proportionnelle: l'erreur est multipliée par un gain  $K_p$
- Une action Intégrale: l'erreur est intégrée sur un intervalle de temps  $s$ , puis Multipliée par un gain  $K_i$
- Une action Dérivée: l'erreur est dérivée suivant un temps  $s$ , puis multipliée par un Gain  $K_d$

Les actions dérivées et intégrales ne s'emploient jamais seules mais en combinaison Avec l'action proportionnelle.

### c . Réglage d'un PID :

Le réglage d'un PID consiste à déterminer les coefficients  $K_p$ ,  $K_i$  et  $K_d$  afin d'obtenir une réponse adéquate du procédé et de la régulation. L'objectif est d'être Robuste, rapide et précis. Il faut pour cela limiter le/ou les éventuels dépassements (*Over shoots*).

### d . Les actions PID :

Un régulateur PID est obtenu par l'association de ces trois actions et il remplit essentiellement les trois fonctions suivantes :

- Fonction proportionnelle donne un système plus précis, plus rapide.
- Fonction intégrateur élimine l'erreur statique.
- Fonction dérivée accélère la correction.

### METHODE DE ZIEGLER-NICHOLS :

En 1942, Ziegler et Nichols ont proposé deux approches heuristiques basées sur leur expérience et quelques simulations pour ajuster rapidement les paramètres des Régulateurs P, PI et PID.

La première méthode nécessite l'enregistrement de la réponse Indicielle en boucle ouverte, alors que la deuxième demande d'amener le système Bouclé à sa limite de stabilité.

#### D.1. Méthode de la réponse indicielle :

Pour obtenir les paramètres du régulateur PID, il suffit d'enregistrer la réponse Indicielle du processus seul (c'est-à-dire sans le régulateur), puis de tracer la tangente au point d'inflexion de la courbe. On mesure ensuite sa pente  $p$ , le retard apparent  $L$  correspondant au point d'intersection de la tangente avec l'abscisse et le gain  $K/E$ , On peut alors calculer les coefficients du régulateur choisi à l'aide du tableau 1.

Généralement, les gains  $K_p$  proposés par Ziegler-Nichols sont trop élevés et conduisent à un dépassement supérieur à 20%. Il ne faut donc pas craindre de réduire  $K_p$  d'un Facteur 2 pour obtenir une réponse satisfaisante.

Type	$K_p$	$T_i$	$T_d$
P	$1/4(pLK_0) = 1/4(aK_0)$		
PI	$0.9/4(pLK_0) = 0.9/4(aK_0)$	$3L$	
PID	$1.2/4(pLK_0) = 1.2/4(aK_0)$	$2L$	$0.5L$

Tableau 3.2: Paramètres PID obtenus à partir d'une réponse indicielle (ZNT).

## D.2 Méthode du point critique :

Cette méthode est basée sur la connaissance du point critique du processus. Expérimentalement, on boucle le processus sur un simple régulateur proportionnel dont on augmente le gain jusqu'à amener le système à osciller de manière permanente; On se trouve ainsi à la limite de stabilité. Après avoir relevé le gain critique  $K_{cr}$  du Régulateur et la période d'oscillation  $T_{cr}$  de la réponse, on peut calculer les paramètres Du régulateur choisis à l'aide du tableau 2. Ici également, les valeurs proposées Conduisent à un temps de montée relativement court malheureusement assorti d'un

Dépassement élevé. Cette situation n'étant pas toujours satisfaisante, on peut être Amené à corriger les coefficients proposés et, en particulier, à diminuer le gain  $K_p$ . On Notera que les paramètres  $T_i$  et  $T_d$  proposés par les deux méthodes de Ziegler-Nichols Sont dans un rapport constant égal à 4 Le régulateur possède donc deux zéros Confondus valant  $-1/(2T_d) = -2/T_i$ .

Type	$K_p$	$T_i$	$T_d$
P	$0.5 K_{cr}$		
PI	$0.4 K_{cr}$	$0.8 T_{cr}$	
PID	$0.6 K_{cr}$	$0.5 T_{cr}$	$0.125 T_{cr}$

Tableau 3.3: Paramètres PID obtenus à partir du point critique (ZNf).

Pour notre cas on a utilisé pour déterminer les paramètres de régulation  $K_p, K_i (T_i)$  et  $K_d (T_d)$ , les méthodes empiriques (Méthode de Ziegler et Nichols) et nous avons réglés ces paramètres **manuellement**.

### e . Procédure expérimentale :

Et pour résoudre le problème de régulateur on a passé par la détermination des coefficients de régulation P, I, D.

La figure 3.48 représente le schéma simplifié d'un échangeur thermique de chaleur. Le but de l'installation est la régulation de la température  $T_c$  du fluide à la sortie de l'échangeur.

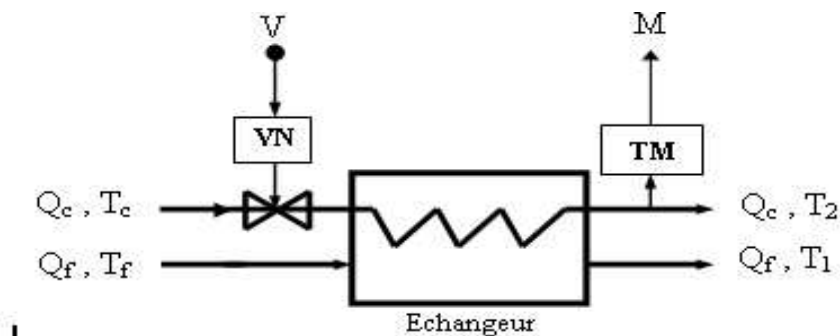


Figure 3.48. Echangeur.

On note:

- $Q_c$ : débit du fluide chaud de température  $T_c$ , c'est la grandeur de réglage. Il est ajusté par un ensemble servomoteur-vanne en agissant sur la tension  $V$ .

- $Q_f$ : débit du fluide froid de température  $T_f$ .

-  $T_2$ : température à la sortie de l'échangeur, grandeur à régler.

-  $M$ : grandeur mesurée par le capteur-transmetteur de mesure  $T_M$ .

-  $VN$ : servomoteur-vanne.

Le schéma fonctionnel de l'installation en boucle fermée est représenté par la

Figure 3.49 où  $H_1, H_2, H_3$  représentent les fonctions de transfert perturbatrices (entre les entrées secondaires et la grandeur réglée) et  $G$  représente la fonction de transfert réglante (entre l'entrée réglante et la grandeur réglée).

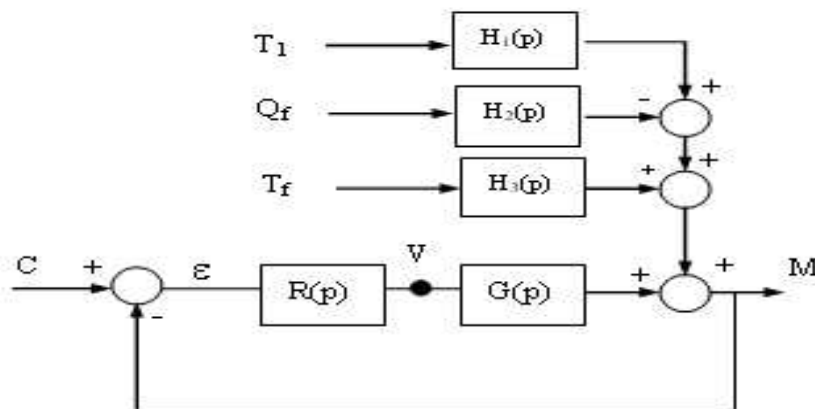


Figure 3.49 : Schéma fonctionnel de l'installation.

$R(p)$  est un régulateur PID:

$$R(p) = K_p \left( 1 + T_d p + \frac{1}{T_i p} \right)$$

Les résultats des essais en boucle ouverte (réponse indicielle) avec la méthode de réglabilité sont présentés ci-dessous:

### E.1. METHODE DE REGLABILITE

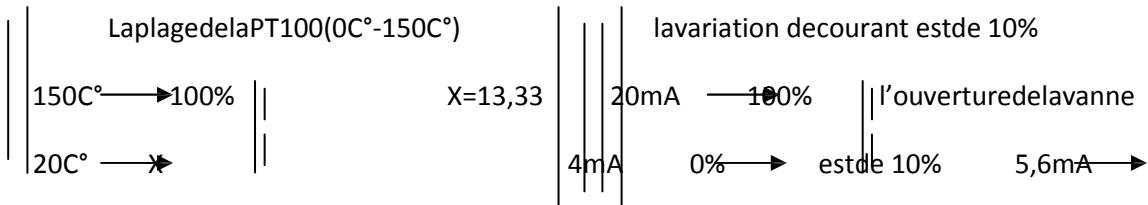
Le régulateur est en mode manuel. On provoque une variation  $\Delta I = 10\%$  (Un échelon  $\Delta U$ )

sur l'entrée de commande (la vanne). La réponse du procédé est dessinée par MATLAB à l'aide des points mesurés expérimentalement

Comme la montre la figure 3.50

Temps(s)	0	40	80	120	160	200	240	280	320	360	400	440
Température(C°)	20	20,5	23,5	30,4	43,6	54	63	67,9	71	71,7	72	72

Tableau 3.4. Les mesures des températures en C° par rapport au temps.



Temps(s)	0	40	80	120	160	200	240	280	320	360	400	440
Température(%)	13,33	13,7	15,3	20,3	29,1	36	42	45,33	47,33	47,8	48	48

Tableau 3.5. Les mesures des températures en % par rapport au temps.

**ProgrammeMATLABpour dessiner laréponseindicielle (figure4.53):**

Close all;

T(s)= 0:40:460;%letemps

P(%) = [13.33 13.7 15.320.329.1364245.3347.3347.84848];% pourcentage dela  
températuremesuréT(c°).

xx= 0:10:460;

yy = **spline**(T(s), P(%),xx);

Plot(T(s), P(%),'o', xx, yy); % la réponseindicielledusystème title ('la réponseindicielledusystème')



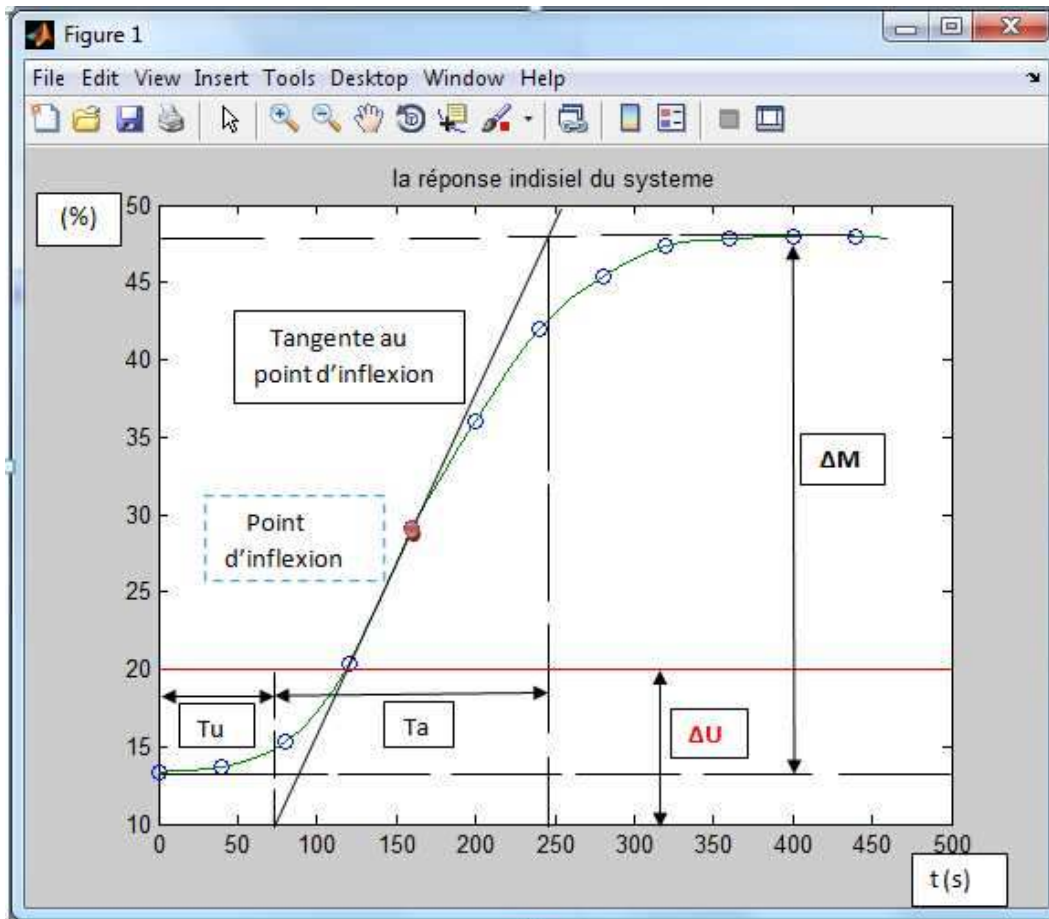


Figure 3.50. Réponse indicielle du système enregistré expérimentalement.

A partir de la réponse indicielle, on relève les paramètres suivants:

$$K = \Delta M / \Delta I = [(48 - 13.33) / 10],$$

$K = 3.47$
$T_u = 72s$
$T_a = 168s$

le gain statique.

le temps de retard.

la constante du temps.

Le coefficient de réglabilité  $r = T_u / T_a = 0.428$

$$0.2 < r < 0.5 \text{ alors: } K_p = [0.5(1 + 0.5*r)] / K*r = 0.40$$

$$K_i = T_a (1 + 0.5*r) = 203.86$$

$$K_d = T_a * (0.5*r / (0.5*r + 1)) = 29.54$$

$K_p = 0.40$
--------------

$K_i = 203.86$
----------------

$K_d = 29.54$
---------------

## ***f . Intégration des paramètres derégulation dans logiciel STEP7***

Pour créer une régulation, de la phase de structuration à son appel par le programme, En passant par son paramétrage, vous aurez à peine besoin de programmation.

Mais il vous faudra bien connaître STEP 7.

Le bloc FB 41 « CONT\_C » sert à régler des processus industriels à grandeurs D'entrée et de sortie continues sur les automates programmables SIMATIC S7. Le Paramétrage vous permet d'activer ou de désactiver des fonctions partielles du régulateur

PID et donc d'adapter ce dernier au système réglé.

Vous pouvez utiliser le régulateur comme régulateur PID de maintien autonome Mais aussi comme régulateur en cascade, de mélange ou de rapport dans des régulations

À plusieurs boucles. Sa méthode de travail se base sur l'algorithme PID du Régulateur à échantillonnage à sortie analogique, complété le cas échéant par un étage

conformateur d'impulsions assurant la formation des signaux de sortie à modulation de largeur d'impulsions pour régulations à deux ou trois échelons avec organes de réglage proportionnels.

En plus des fonctions traitant la consigne et la mesure, le FB réalise un régulateur PID prêt à l'emploi avec sortie continue de la grandeur de réglage et possibilité D'influencer à la main la valeur de réglage.

Il propose les fonctions partielles suivantes.

Branche de consigne

La consigne est entrée en format de virgule flottante à l'entrée SP\_INT.

Branche de mesure

La mesure peut être lue en format de périphérie ou de virgule flottante. La fonction

CRP\_IN convertit la valeur de périphérie PV\_PER en un nombre à virgule flottante

Compris entre -100 et +100 % selon la formule suivante :

Sortie de CPR\_IN = PV\_PER \* 100 / 27648

La fonction PV\_NORM normalise la sortie de CRP\_IN selon la formule suivante :

Sortie de PV\_NORM = (sortie de CPR\_IN) \_ PV\_FAC + PV\_OFF

La valeur par défaut de PV\_FAC est 1 et celle de PV\_OFF est 0.

Traitement de la valeur de réglage manuelle

Vous pouvez passer du mode automatique au mode manuel et inversement.

En mode manuel, la grandeur de réglage est asservie à la valeur de réglage manuelle.

L'intégrateur (INT) est positionné et aligné de façon interne sur LMN - LMN\_P -

DISV et le différentiateur (DIF) est forcé à 0. Ainsi, le passage au mode automatique

S'effectue sans à-coup.

Traitement de la valeur de réglage La fonction LMNLIMIT permet de limiter la grandeur de réglage à des valeurs que vous indiquez. Le dépassement de ces limites par la grandeur d'entrée est signalé par des bits.

La fonction LMN\_NORM normalise la sortie de LMNLIMIT selon la formule suivante:

$LMN = (\text{sortie de LMNLIMIT}) \_ LMN\_FAC + LMN\_OFF$

La valeur par défaut de LMN\_FAC est 1 et celle de LMN\_OFF est 0.

La valeur de réglage est disponible aussi en format de périphérie. La fonction CRP\_OUT convertit la valeur à virgule flottante LMN en une valeur de périphérie

Selon la formule :

$LMN\_PER = LMN \cdot 100 / 27648$

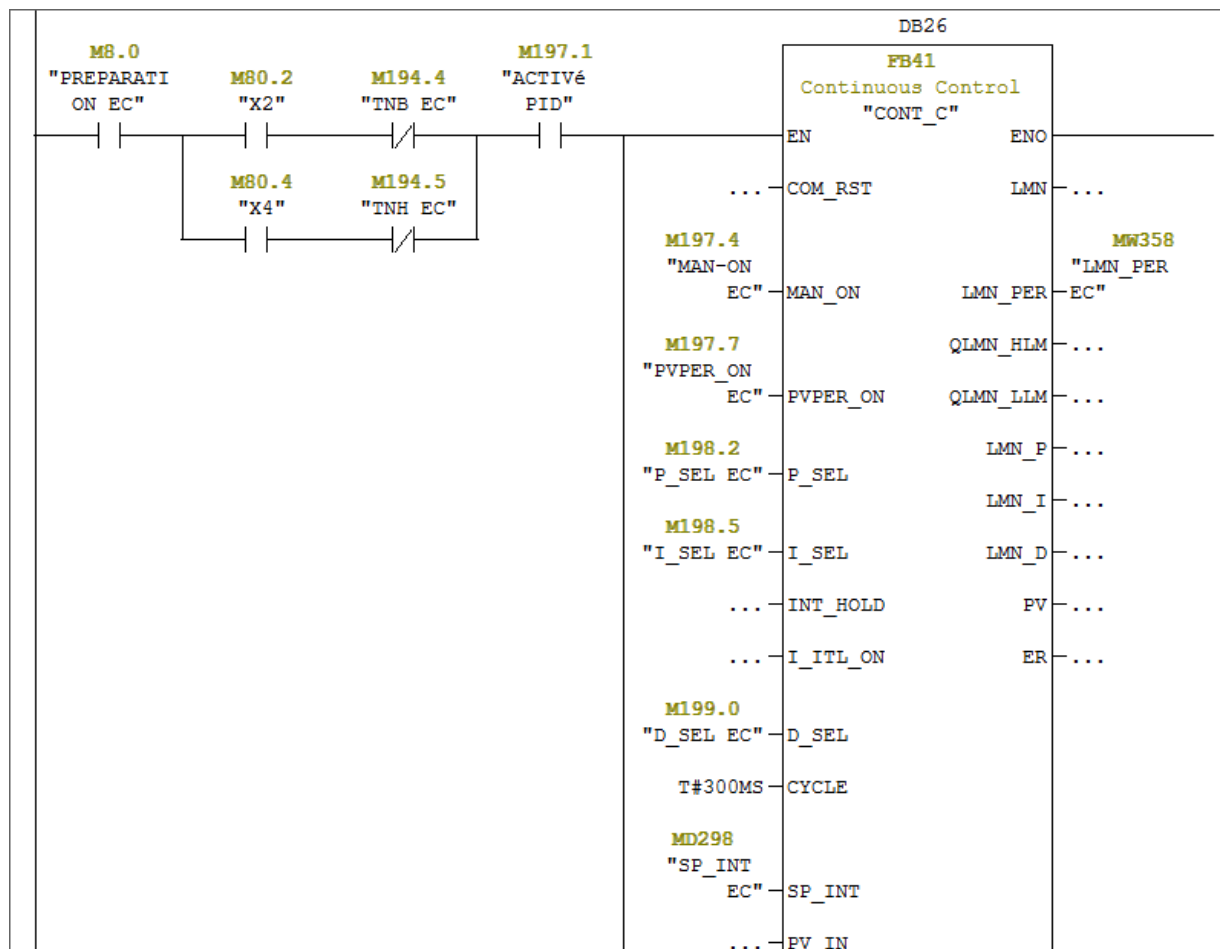
### Démarrage et redémarrage

Le bloc FB 41 « CONT\_C » dispose d'un sous-programme de démarrage qui est Exécuté quand le paramètre d'entrée COM\_RST = TRUE.

A la mise en route, l'intégrateur est positionné de façon interne sur la valeur d'initialisation I\_ITVAL. En cas d'appel dans un niveau d'alarme d'horloge, il continue À travailler à partir de cette valeur.

Toutes les autres sorties sont positionnées sur leur valeur par défaut.

Le bloc ne procède Pas à un contrôle interne d'erreur. Le mot indicateur d'erreur RET\_VAL n'est pas employé.



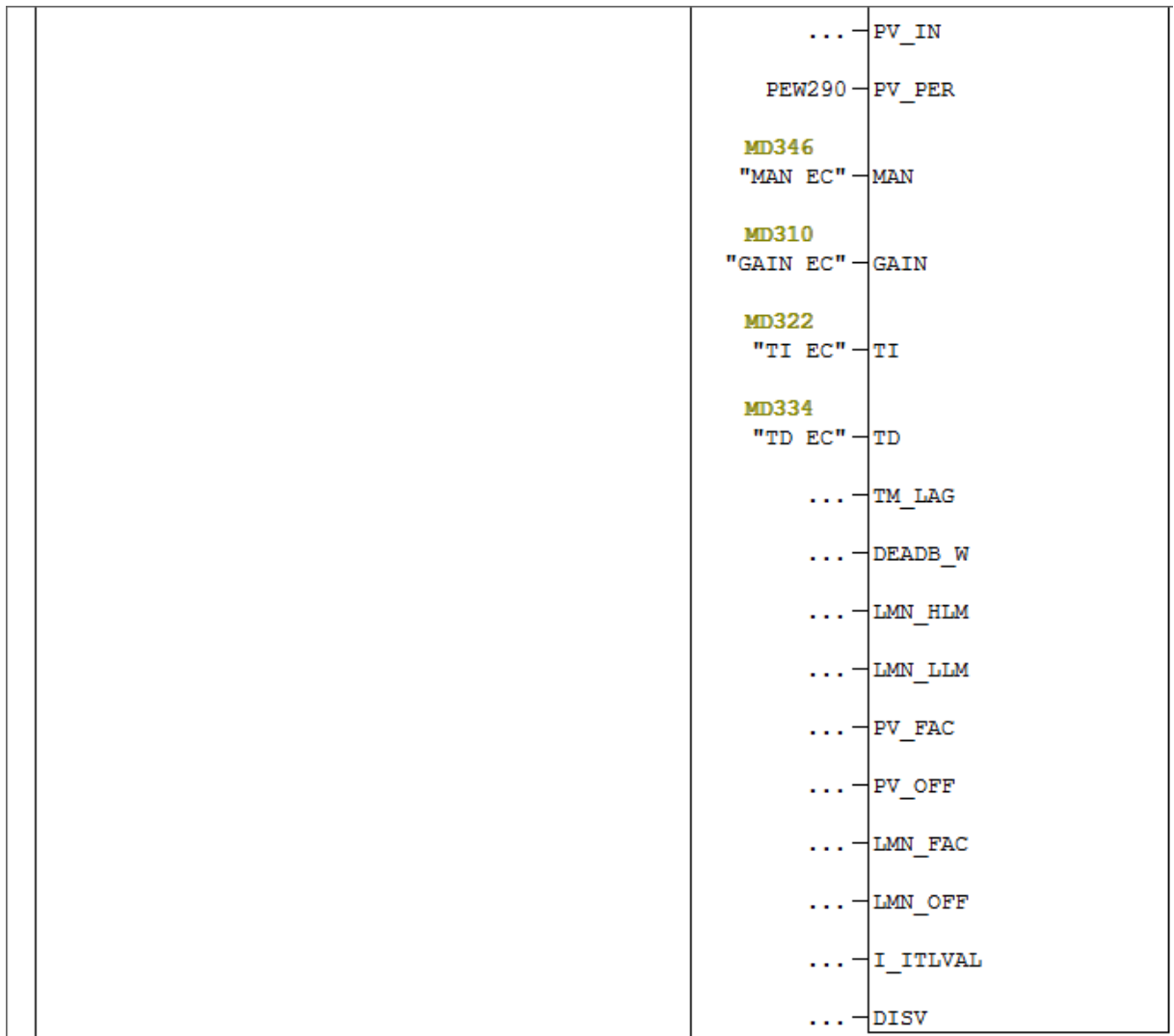


Figure 3.51: utilisation de bloc FB41.

Nous avons utilisé le bloc OB35 pour faire l'appel au bloc FB41 et on a introduit une opération qui nous a aidées à transférer la valeur de LMN\_PER vers la case mémoire MW800 de type Word.

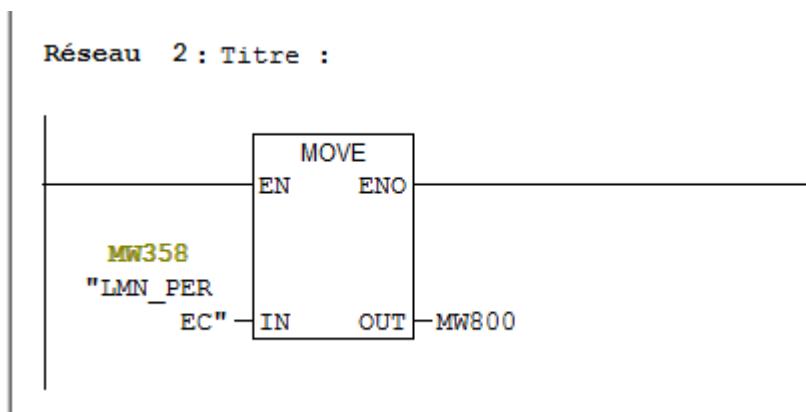


Figure 3.52: opération de transfert.

### **Description de l'opération**

#### **MOVE (Affecter valeur)**

Cette opération est activée par l'entrée de validation EN. La valeur indiquée dans l'entrée IN est copiée à l'adresse précisée dans la sortie OUT. L'état de signal de ENO est identique à celui de EN. L'opération MOVE ne permet de copier que des octets, des mots ou des doubles mots. Pour copier des types de données utilisateur tels que des tableaux ou des structures, vous devez faire appel à la fonction système "BLKMOV" (SFC 20).

## **3.7 Conclusion**

Dans ce chapitre on a vu la description de logiciel de programmation STEP7 et de supervision WinCC flexible par la suite on a défini comment créer un projet, la configuration et la liaison entre le PLCSIM et le WinCC flexible dans le champ SIEMENS, et ensuite on a essayé de présenter l'outil de la station NEP, ces différents composants de programme de l'automate, les variables et les entrées/sorties du système dans un tableau mnémotechnique. Dans le chapitre suivant on va présenter un tableau qui résume les différentes alarmes utilisées. Nous aurons, ensuite, présenter l'interface graphique du système et les résultats obtenus.

# Chapitre 4 Interface de supervision de la station

NEP

---

## 4.1 . INTRODUCTION :

Après avoir exposé les étapes de programmation de la station NEP dans le chapitre précédent, on va présenter dans ce chapitre les différentes vues qu'on a réalisées dans le logiciel WINCC flexible ( pupitre PC IL 77 15" Touch) qui nous permet de contrôler et surveiller la station de nettoyage.

## 4.2 Supervision sous WINCC FLEXIBLE :

### 4.2.1 L'interface HMI configurée par «WinCC flexible»

#### *a Création de la station HMI*

Dans le projet PFE on a introduit un nouvel objet qui est la station HMI en choisissant le type de pupitre sur lequel, les informations seront transmises, pour notre application on utilise un PC IL 77 15" Touch 1.4.0.0 en liaison avec l'automate.

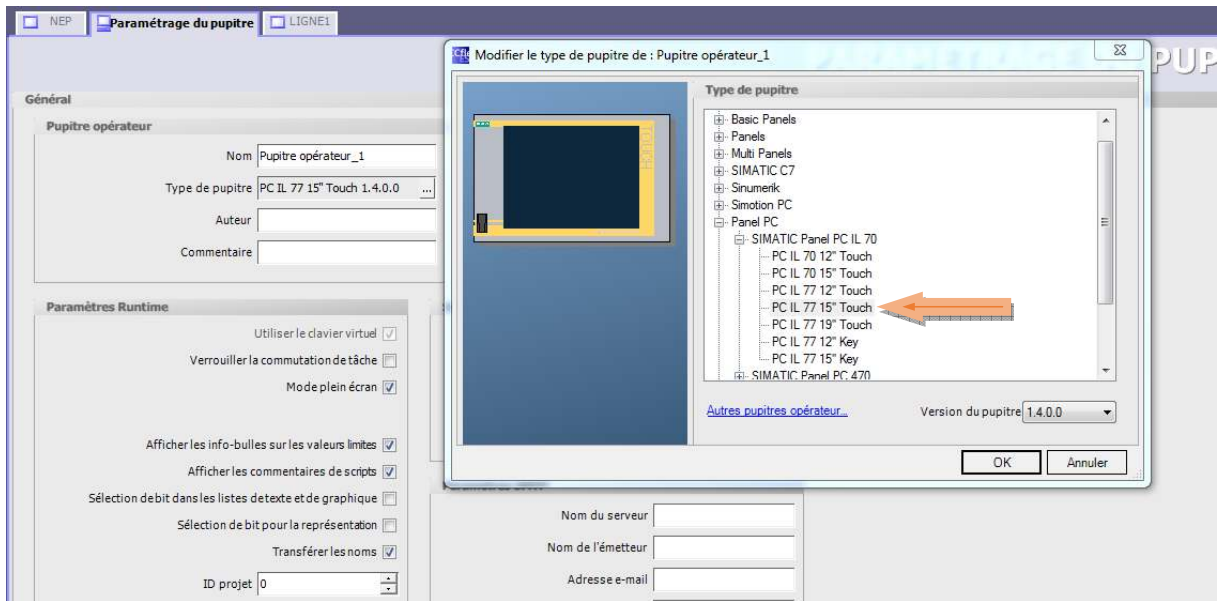


Figure 4.1 : le choix de pupitre.

## 4.2.2 INTEGRATION DU PROJET STEP7 :

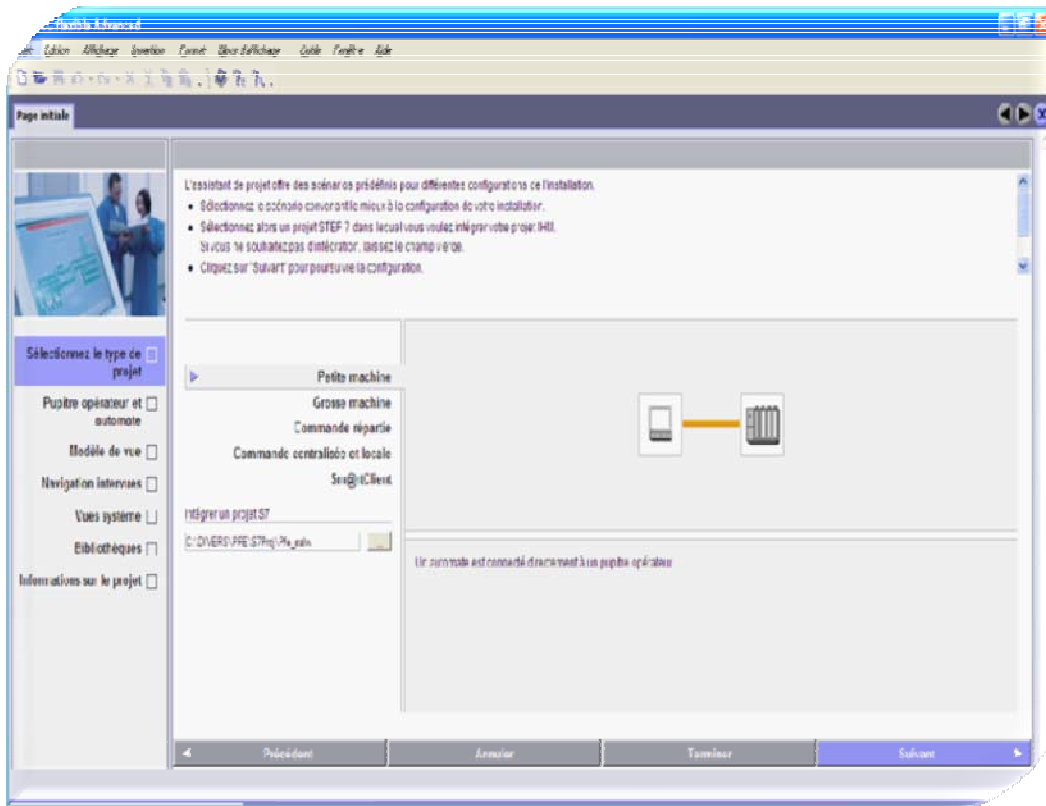


Figure 4.2 : Intégration du projet Step7 dans WinCC flexible.

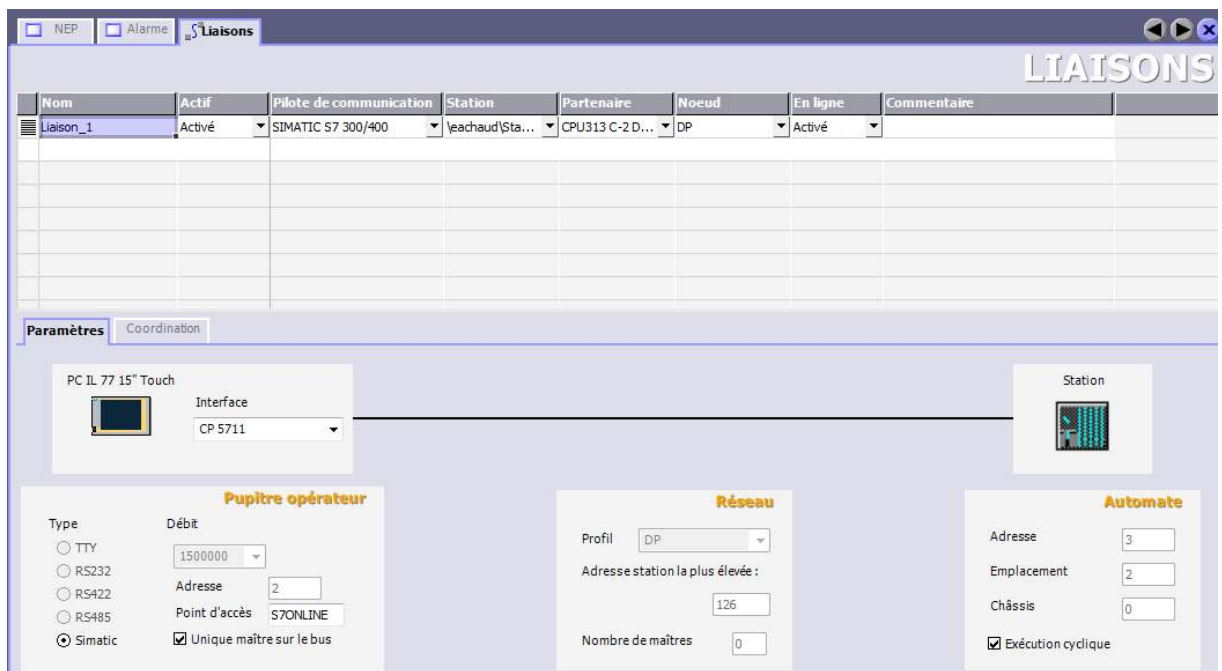


Figure 4.3 : Liaison avec l'automate S7300.

#### **a Création des vues:**

Dans WinCC flexible, les vues sont créées pour la supervision et le contrôle des paramètres du système etc' est le but de notre projet de faire un contrôle en temps réel des paramètres de la station NEP.

#### **b Création de la table de variable:**

Maintenant, la liaison entre le projet du WinCC flexible et l'automate est établie et les vues de la supervision du système sont créées. Donc il est possible d'accéder à toutes les zones mémoire de l'automate (bloc de données, mémoire des entrées/sorties). Ainsi, pour la configuration des alarmes, il est indispensable de créer une table de variables sur WinCC flexible qui contient les différents entrées/sorties nécessaires pour la visualisation et le contrôle des paramètres du système.

La figure ci-dessous représente un exemple des variables que nous avons utilisés dans notre projet.



Nom	Nom d'affichage	Liaison	Type de données	Mnémorique	Adresse	Éléments du tableau	Cycle d'acquisition
LONG F L2		Liaison...	Bool	LONG F L2	M 188.7	1	1 s
LONG F L3		Liaison...	Bool	LONG F L3	M 188.6	1	1 s
M1 WINCC		Liaison...	Bool	M1 WINCC	M 4.2	1	100 ms
M2 WINCC		Liaison...	Bool	M2 WINCC	M 4.3	1	100 ms
M3 WINCC		Liaison...	Bool	M3 WINCC	M 4.4	1	100 ms
M4 WINCC		Liaison...	Bool	M4 WINCC	M 4.5	1	100 ms
M5 WINCC		Liaison...	Bool	M5 WINCC	M 4.6	1	100 ms
M6 WINCC		Liaison...	Bool	M6 WINCC	M 4.7	1	100 ms
MACHINE 1		Liaison...	Bool	MACHINE 1	M 0.2	1	100 ms
MACHINE 2		Liaison...	Bool	MACHINE 2	M 0.3	1	100 ms
MACHINE 3		Liaison...	Bool	MACHINE 3	M 2.4	1	100 ms
MACHINE 4		Liaison...	Bool	MACHINE 4	M 2.5	1	100 ms
MACHINE 5		Liaison...	Bool	MACHINE 5	M 2.6	1	100 ms
MACHINE 6		Liaison...	Bool	MACHINE 6	M 2.7	1	100 ms
MENU		<V...>	Bool	<indéfini>	<Pas d'adresse>	1	1 s
mode pause		Liaison...	Word	mode pause	MW 176	1	1 s
mode pause 2 ...		Liaison...	Word	mode pause 2 et 3	MW 177	1	1 s
NB BAC 1		Liaison...	Bool	NB BAC 1	I 124.2	1	100 ms
NB BAC 2		Liaison...	Bool	NB BAC 2	I 124.3	1	100 ms
NB BAC 3		Liaison...	Bool	NB BAC 3	I 124.4	1	100 ms
NB BAC 4		Liaison...	Bool	NB BAC 4	I 124.5	1	100 ms
NB BAC 5		Liaison...	Bool	NB BAC 5	I 124.6	1	100 ms
NB BAC 6		Liaison...	Bool	NB BAC 6	I 124.7	1	100 ms
NB CUVE 1		Liaison...	Bool	NB CUVE 1	I 124.0	1	100 ms
NB CUVE 2		Liaison...	Bool	NB CUVE 2	I 124.1	1	100 ms

Figure 4.4 : Exemple des variables utilisés dans WINCC flexible.

### c . Navigation entre les vues :

La navigation entre les vues de l'interface a été conçue pour permettre à l'opérateur de naviguer entre les vues de manière simple et rapide.

## 4.3 Réalisation de l'interface de supervision de la station NEP:

Pour les contrôles et la supervision des paramètres de la station NEP, nous avons réalisé les vues représentées ci-dessous :

### 4.3.1 Vue principale :

Cette vue (figure 4.5) permet l'accès aux utilisateurs par la saisie de nom d'utilisateur et le mot de passe pour qu'on puisse connaître les droits d'accès de chacun.

Il existe deux types d'utilisateur,

Le premier c'est l'opérateur (figure 4.6) qu'il peut faire la commande, le contrôle et le deuxième c'est l'ingénieur (figure 4.7) qu'il a l'accès pour modifier les paramètres de régulation et le temps d'envoi des solutions.

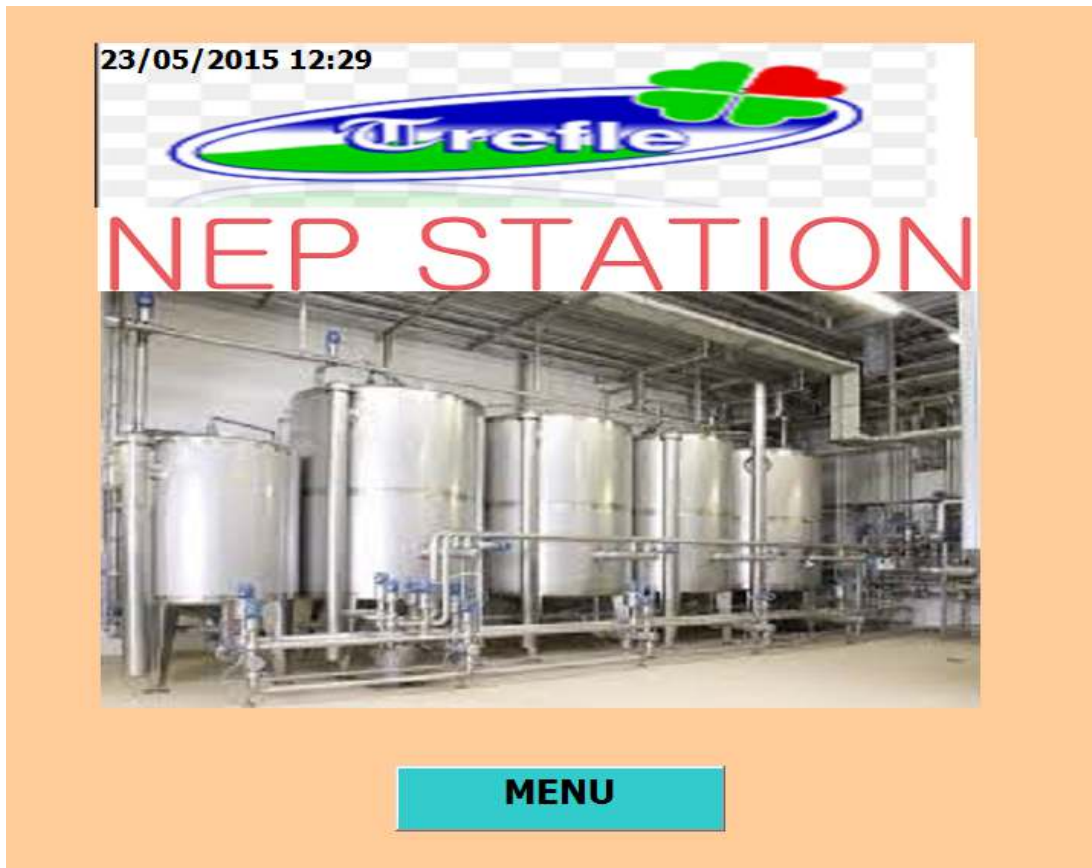


Figure 4.5:vue principale de la station NEP.



Figure 4.6 : vue opérateur.



Figure 4.7 : vue ingénieur.

### 4.3.2 . Vue des alarmes :

Cette vue (figure 4.8) a pour but de nous avertir les niveaux, la température et la concentration des solutions dans les cuves (niveau haut et niveau bas) et si le nettoyage est en mode pause ou il a été annulé.

N°	Heure	Date	Etat	Texte	GR
! 18	14:52:19	23/05/2015	A	la concentration cuve acide a dépassée les limites	0
! 19	14:52:15	23/05/2015	A	la concentration cuve soude n'a pas dépassée le minimum	0
! 15	14:52:01	23/05/2015	A	la temperature cuve acide n'a pas dépassée le minimum	0
! 5	14:51:57	23/05/2015	A	le niveau cuve eau chaud n'a pas dépassé le minimum	0
! 8	14:51:47	23/05/2015	A	le niveau cuve eau propre a dépassé les limites	0

retour

Figure 4.8 vue des alarmes.

### 4.3.3 . Vue des capteurs

La vue des capteurs nous permet de voir la liste des capteurs (figure 4.9) (capteurs TOR et analogique) qui existe dans la station NEP et leurs état actuel, ainsi l'adresse de capteur. Pour faciliter le travail à l'ingénieur.

La même chose pour les pompes (figure 4.10) et les vannes (figure 4.11), ils ont leur propre vue.

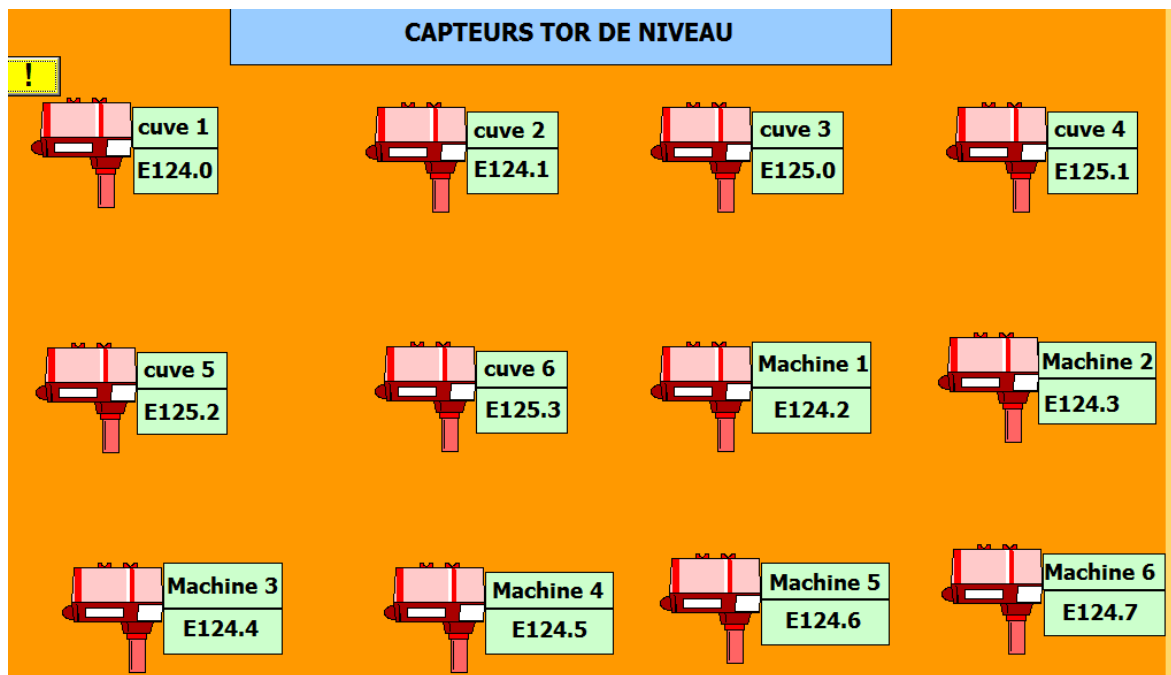


Figure 4.9.vue des capteurs.

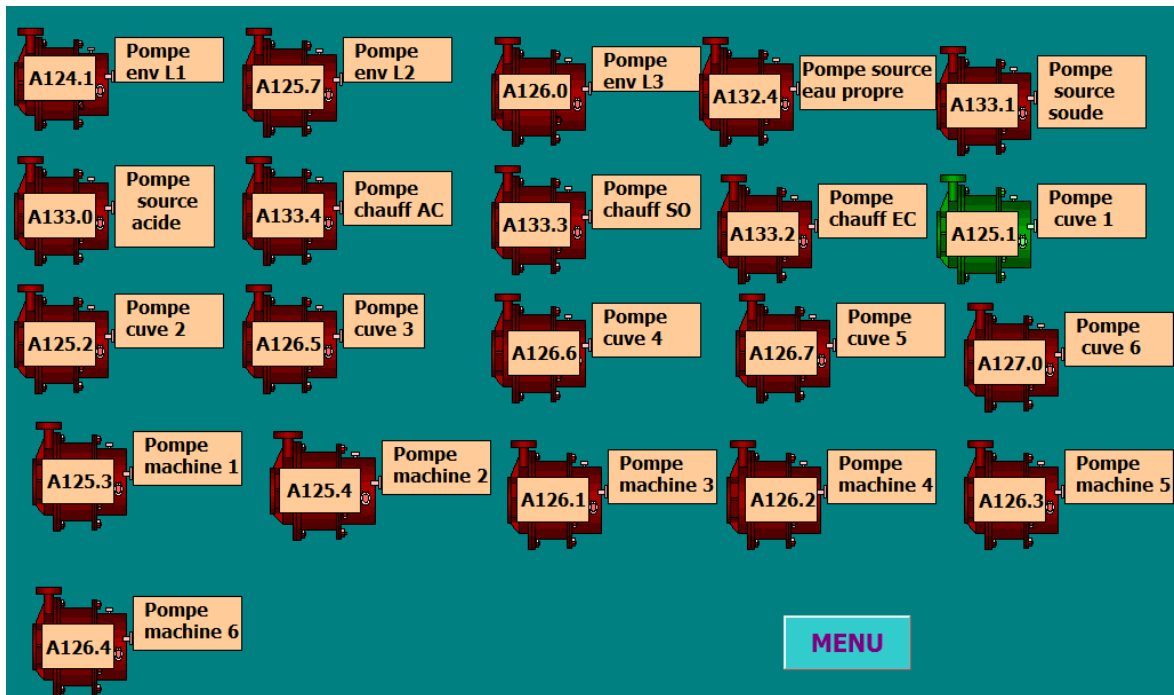


Figure 4.10.vue des pompes.

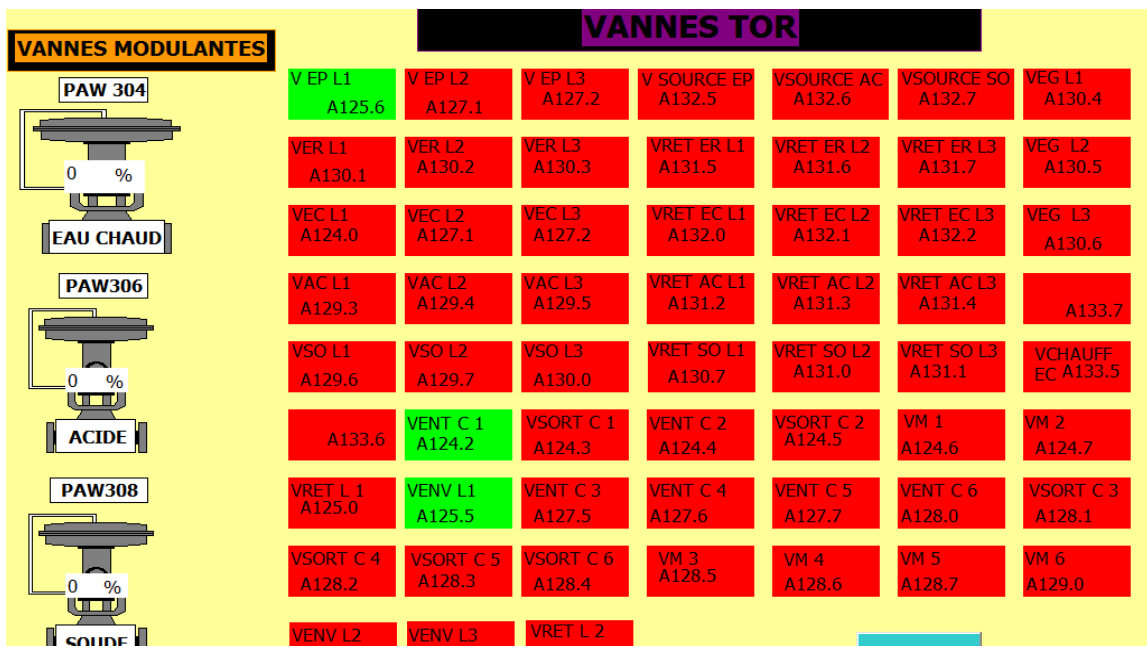


Figure 4.11.vue des vannes.

### 4.3.4 . Vue état des cuves :

La vue état des cuves (figure 4.12) nous permet de savoir si les cuves sont prêtes pour faire le nettoyage des machines ou des cuves.

Les cuves sont prêtes lorsque le niveau de liquide dépasse 20%, la concentration doit être entre 75ms et 80ms et la température soit entre 85 °C et 90 °C

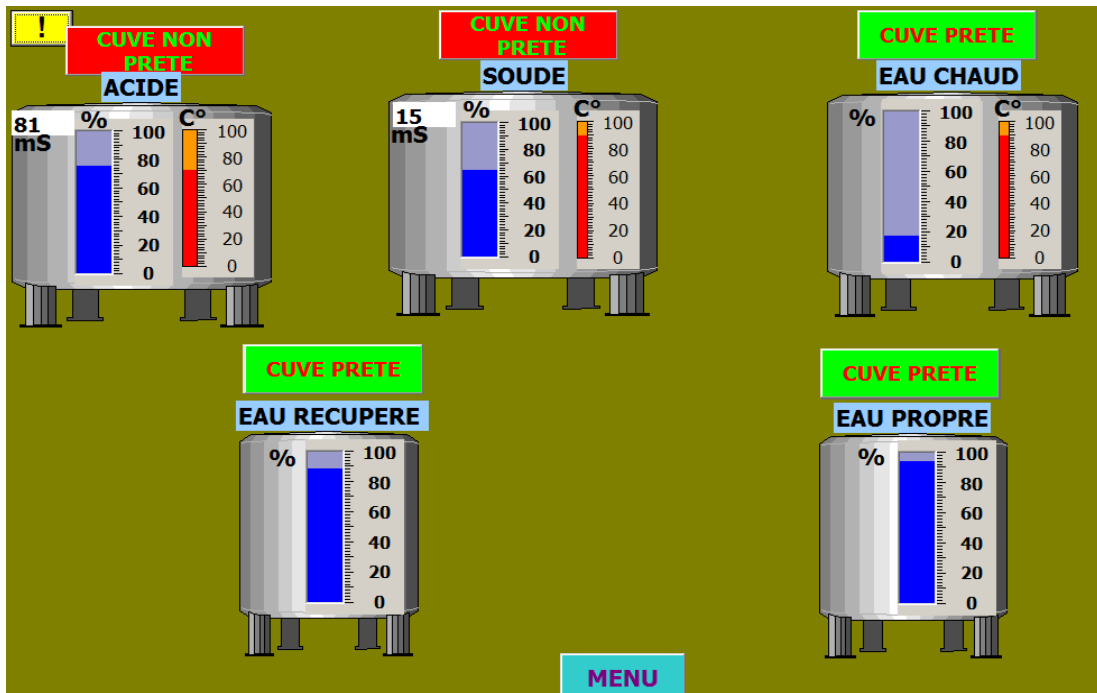


Figure 4.12 : état des cuves.

### 4.3.5 . Vue détail sur le nettoyage

Cette vue nous donne le déroulement de nettoyage actuel, elle nous permet de savoir le cycle qui a été choisi, le matériel à nettoyer et la solution envoyer avec le temps restant.

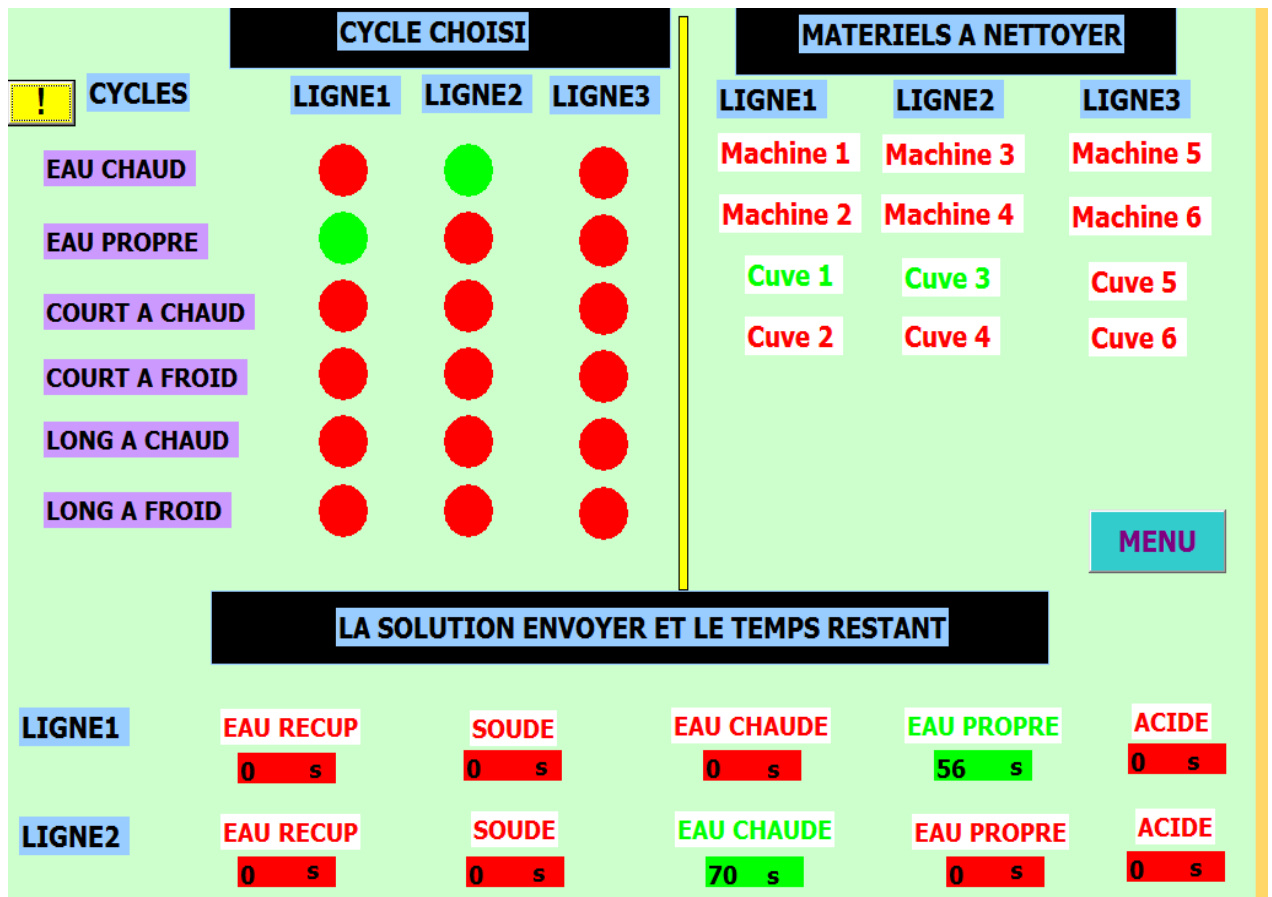
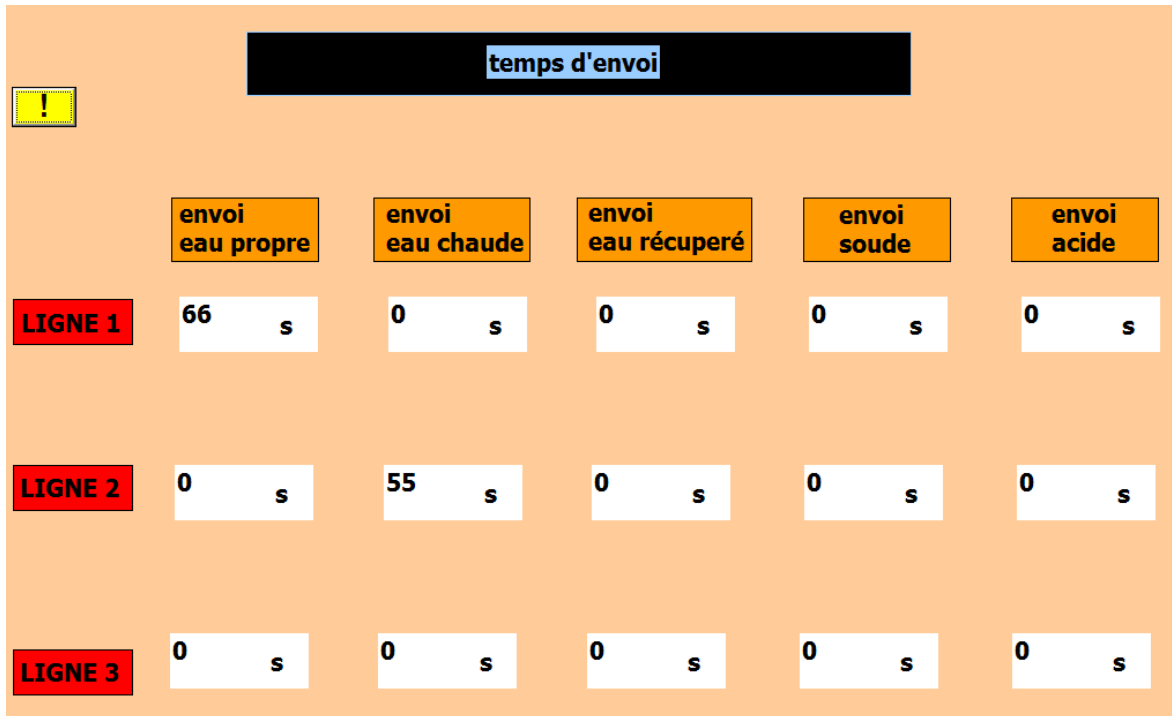


Figure 4.13. l'etat actuel de la station.

### 4.3.6 Vue temps d'envoi

Cette vue permet à l'ingénieur de contrôler et modifier le temps d'envoi des solutions.



	envoi eau propre	envoi eau chaude	envoi eau récupéré	envoi soude	envoi acide
LIGNE 1	66 s	0 s	0 s	0 s	0 s
LIGNE 2	0 s	55 s	0 s	0 s	0 s
LIGNE 3	0 s	0 s	0 s	0 s	0 s

Figure 4.14.vue temps d'envoi.

### 4.3.7 . Vue station NEP

Cette vue (figure4.15) est la représentation graphique de la station NEP, elle contient les différents équipements du système ainsi que leur emplacement exactes comme dans l'installation réelle.

Lorsqu'on appuie sur la ligne 1 (figure 4.16), elle s'affiche la vue des machines et cuves, la même chose pour la ligne 2 et ligne 3.

Ainsi pour la source (figure 4.17) et chauffage (figure 4.18). Elles ont leurs vue.



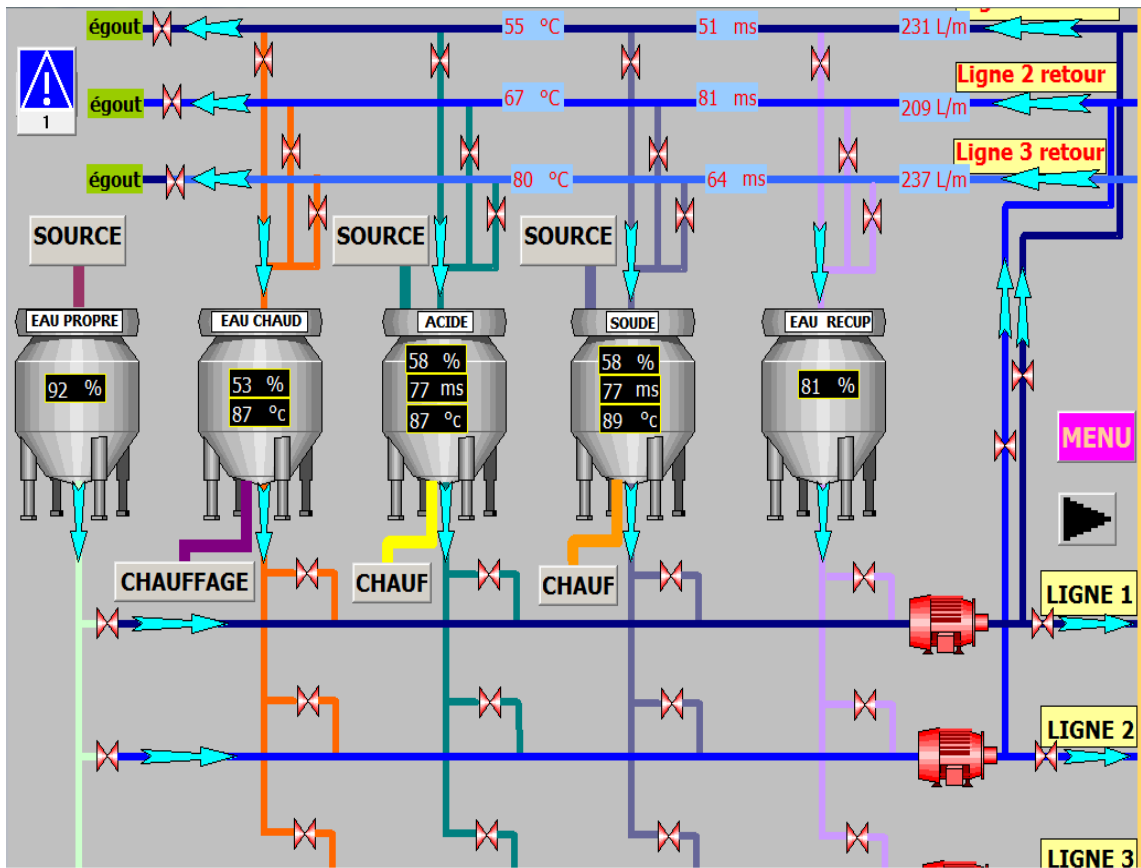


Figure 4.15 : vue station NEP.

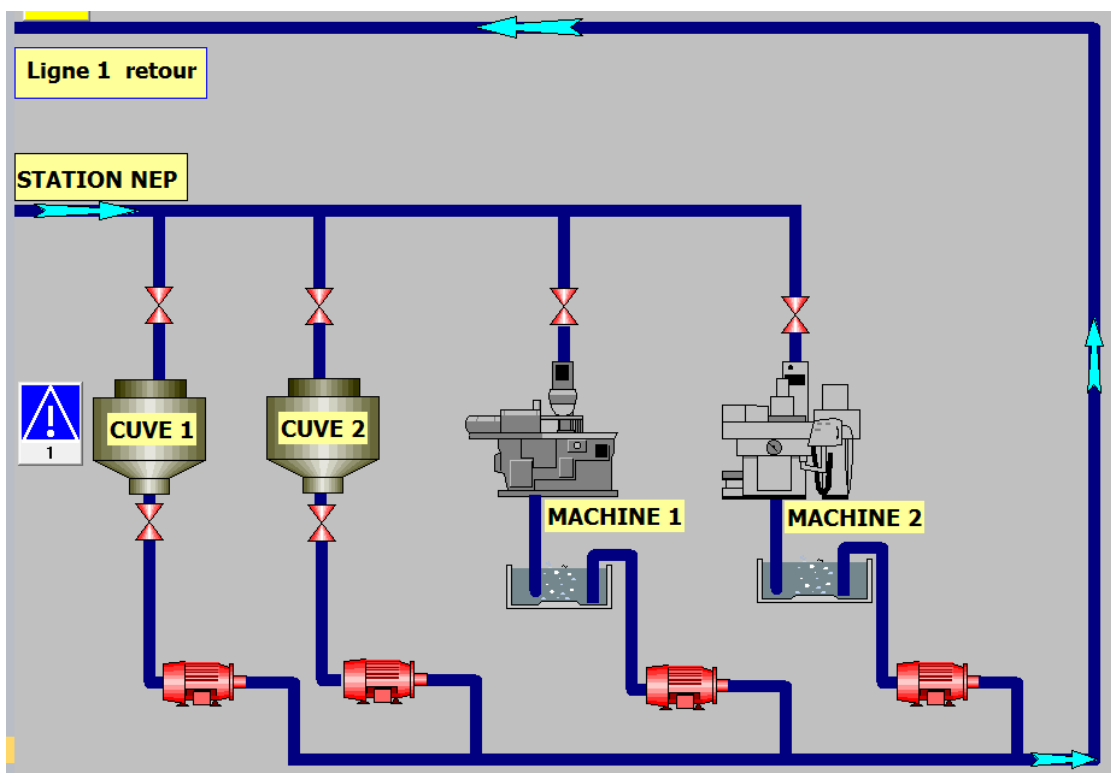


Figure 4.16.vue des machines et cuves pour la ligne 1.

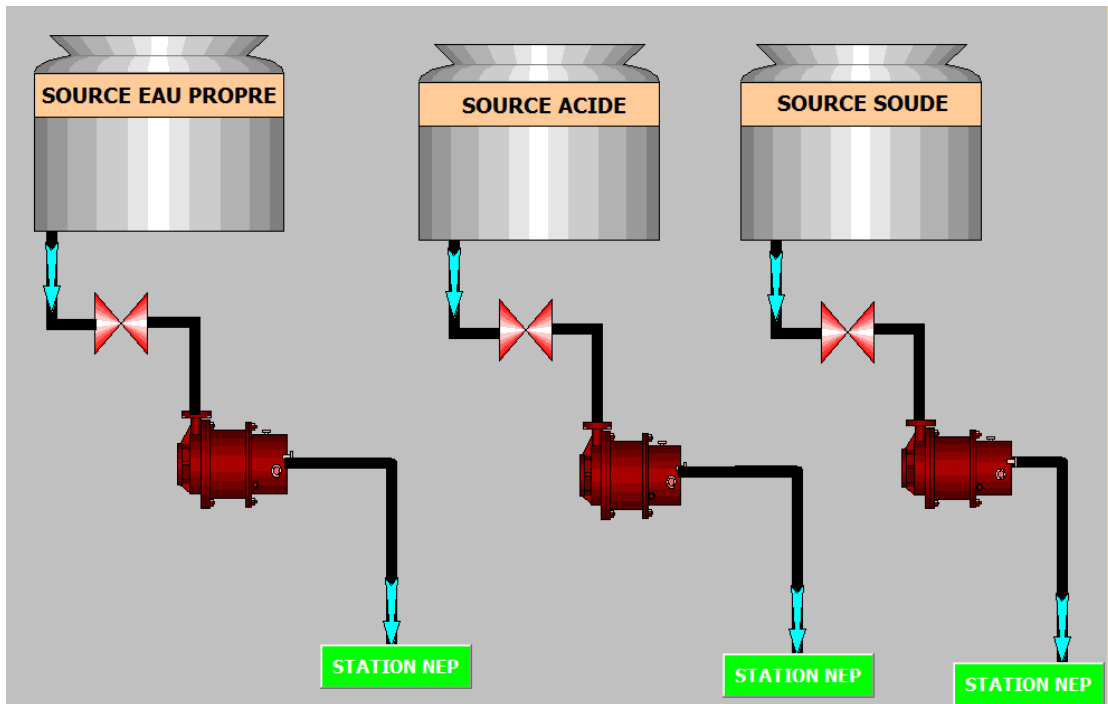


Figure 4.17.vue source.

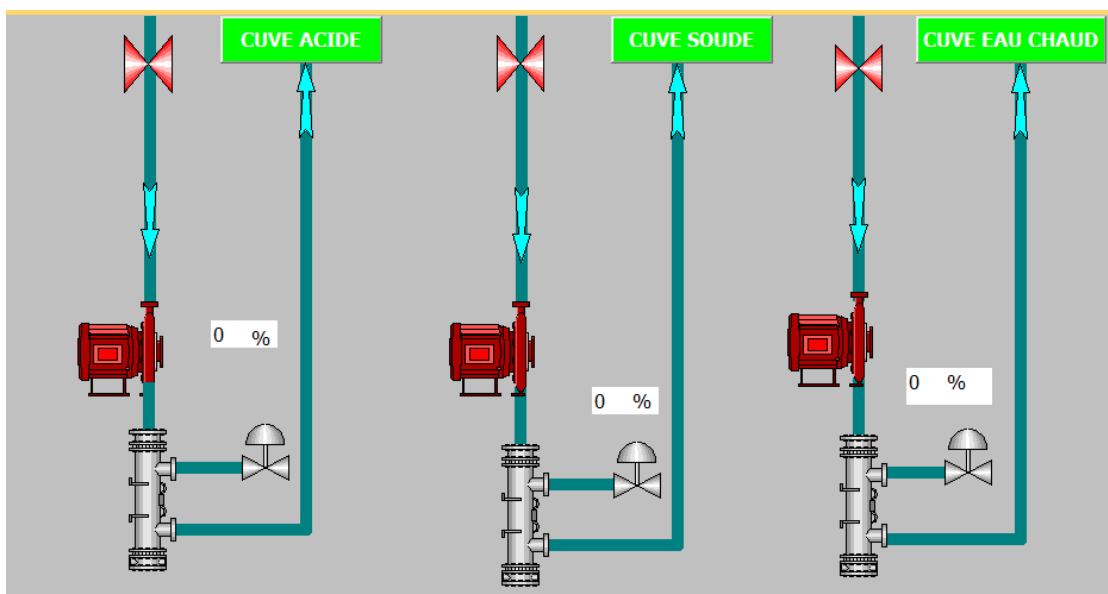


Figure 4.18.vue chauffage.

### 4.3.8 . Vue régulation de température:

Les cuves soude, acide et eau chaude ont un régulateur de température (figure4.19), on a grée pour chaque cuve sa propre fenêtre pour modifier les paramètres de PID.

Après avoir appuyé l'un des boutons, elle apparaitre une notre vue,cette vue (figure 4.20) permet à l'ingénieur de modifier la consigne de température, d'activé les actions de PID et entre leurs valeurs.

Si l'ingénieur veut surveiller l'évolution de la température, il peut visualiser la courbe de température en temps réel.

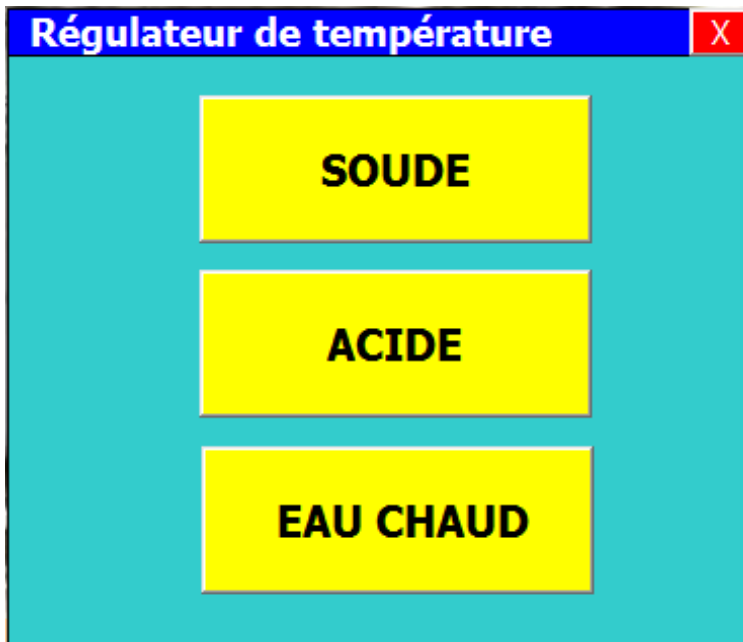


Figure 4.19.choisir le régulateur.

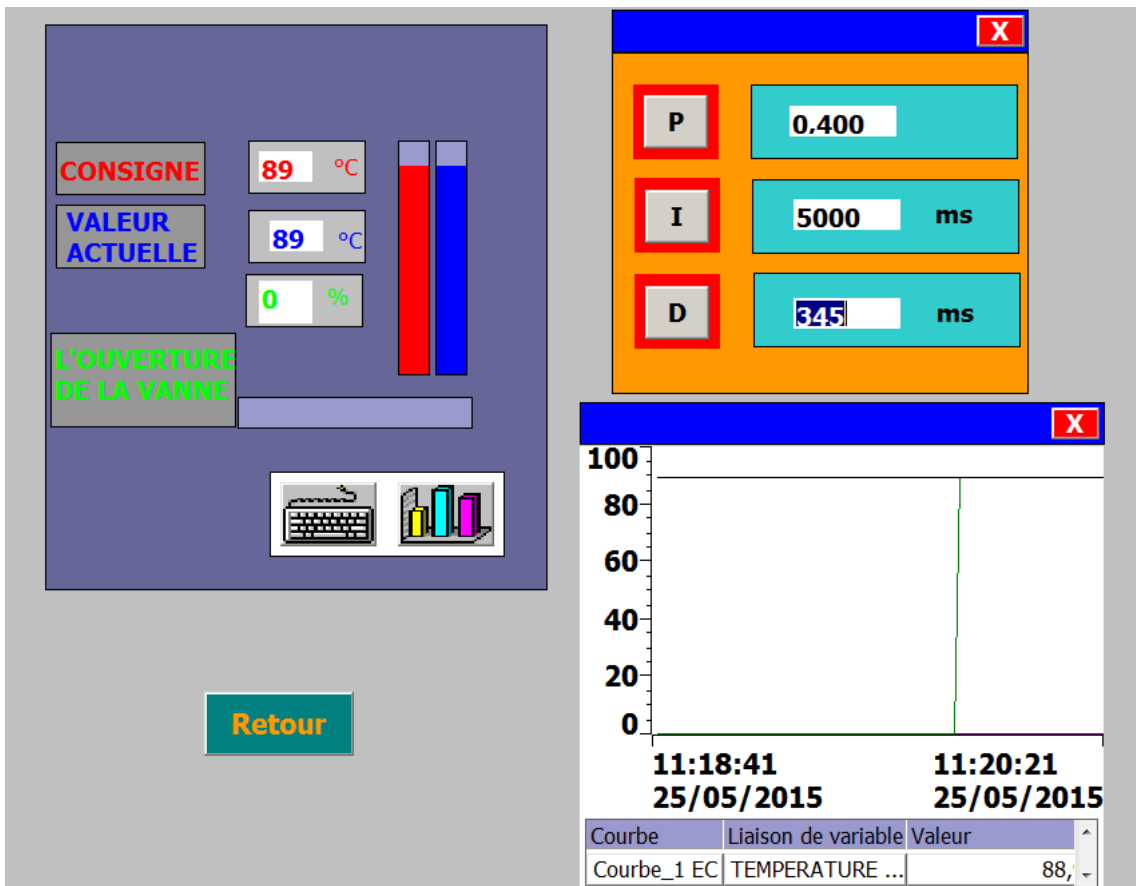


Figure 4.20.Vue régulation de température.

### 4.3.9 . Vue nettoyage ou préparation:

Après avoir appuyé sur le bouton (nettoyage ou préparation) que l'on trouve sur le menu Opérateur et ingénieur, elle apparaitre la vue pour choisir l'un deux (figure 4.21)

- lorsqu'on appuie sur le bouton préparation elle apparaitre une autre vue (figure 4.22), Dans cette vue il existe trois boutons :

- bouton cuve eau chaude :(si on appuie sur ce bouton, la vue change (figure 4.23) et lorsque on appuie sur valider la préparation démarre (figure 4.24))

La même chose pour les deux autres boutons.

- si on choisit le nettoyage, elle surgit la vue pour choisir le matériels à nettoyer ou nettoyage de station NEP (figure 4.25).

Après avoir appuyé sur le bouton choisir le matériel à nettoyer (figure 4.26),

Cette vue nous permet de choisir la machines ou la cuves à nettoyer avec le cycle désigne.

Quand on appuie sur la cuve 3 et le cycle court à chaud (figure 4.27) en suite sur bouton valider pour commencer le nettoyage figure (4.28), l'utilisateur peut annuler le nettoyage ou mettre en mode pause.

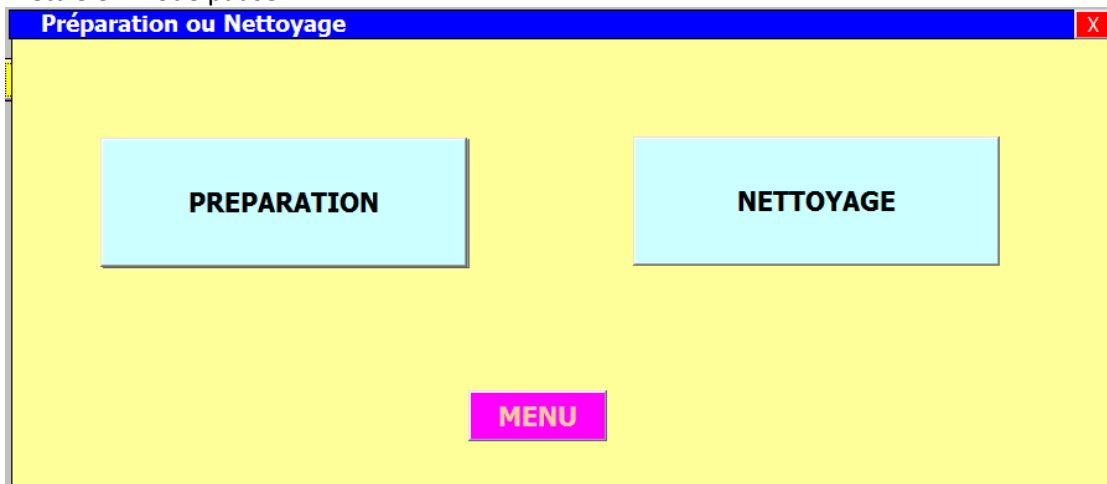


Figure 4.21.choisir nettoyage ou préparation.

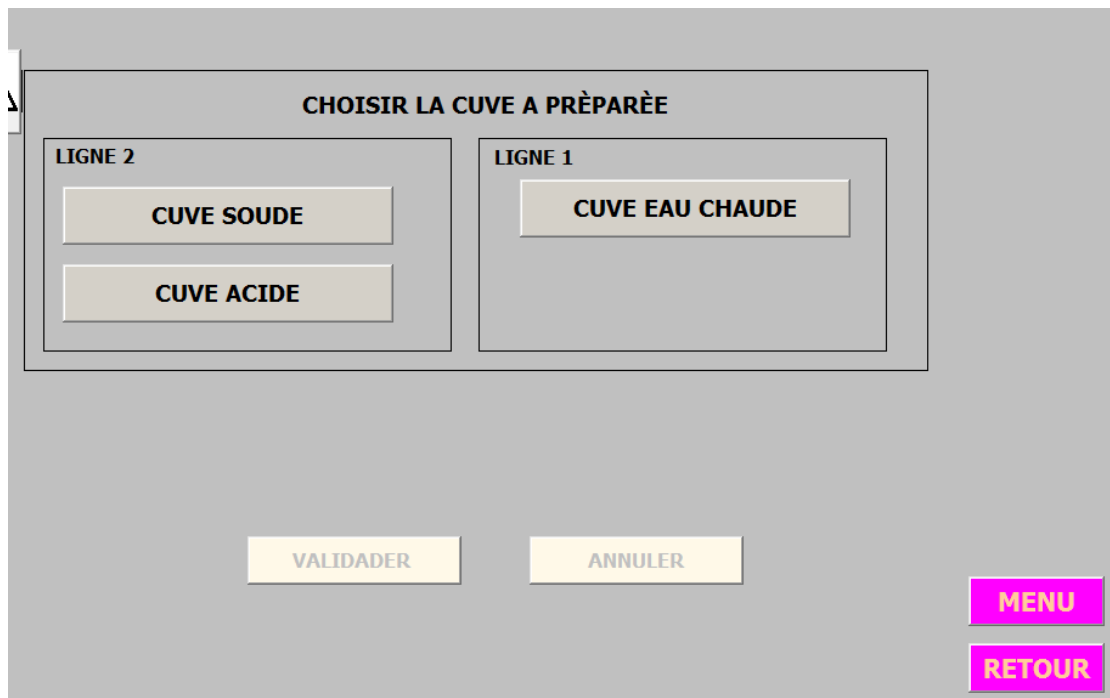


Figure 4.22.choisir la cuve a préparée.

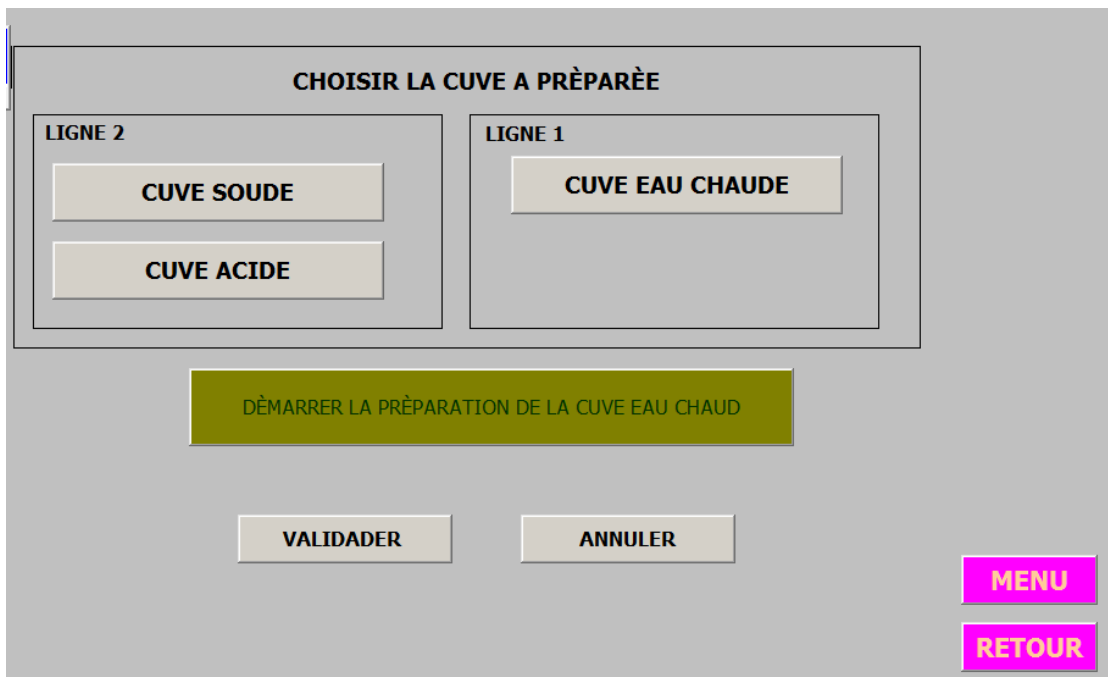


Figure 4.23.préparer la cuve eau chaude.

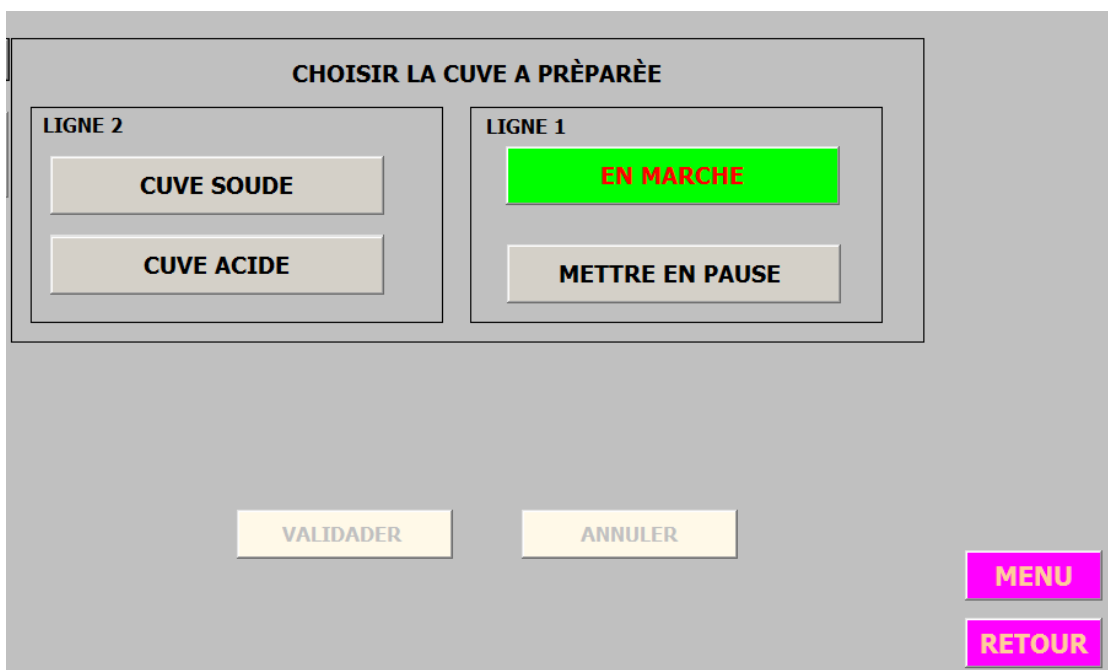


Figure 4.24.préparation d'eau chaude en cours.

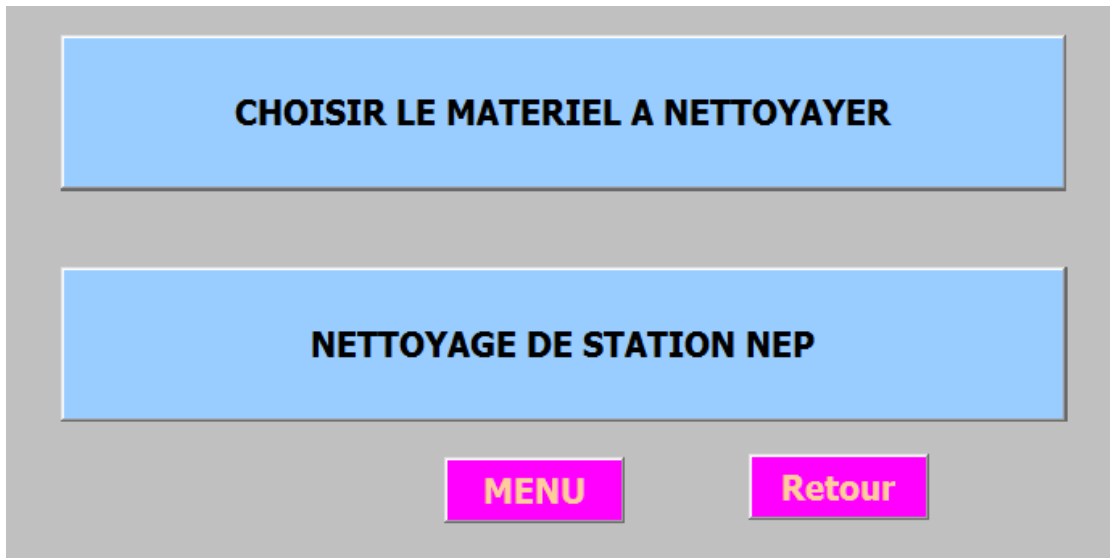


Figure 4.25. le matériels à nettoyer ou nettoyage de station NEP.

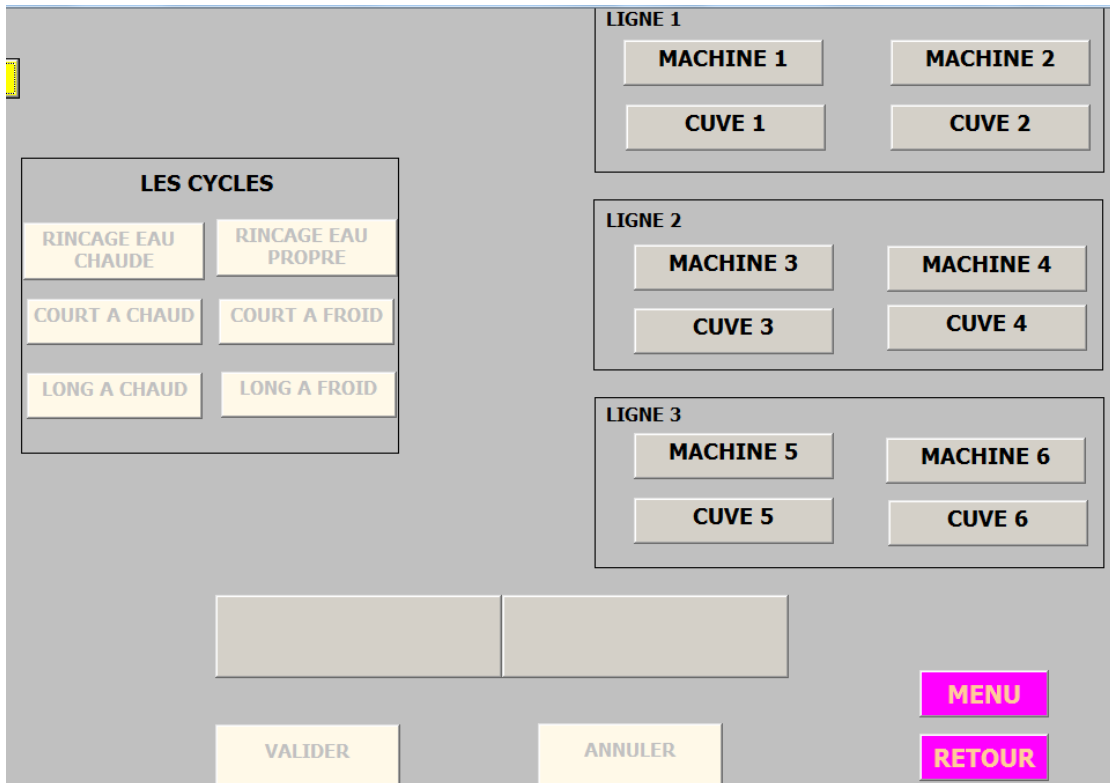


Figure 4.26. le matériels à nettoyer.

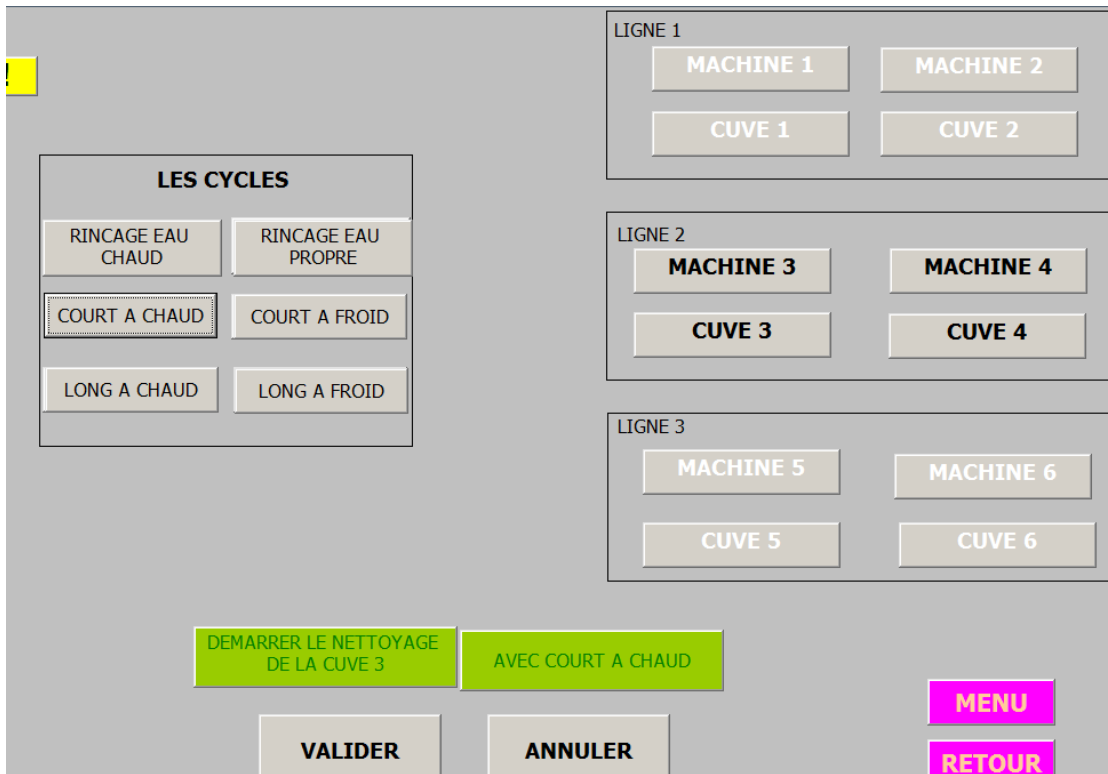


Figure 4.27. nettoyer la cuve 3 avec le cycle court à chaud.

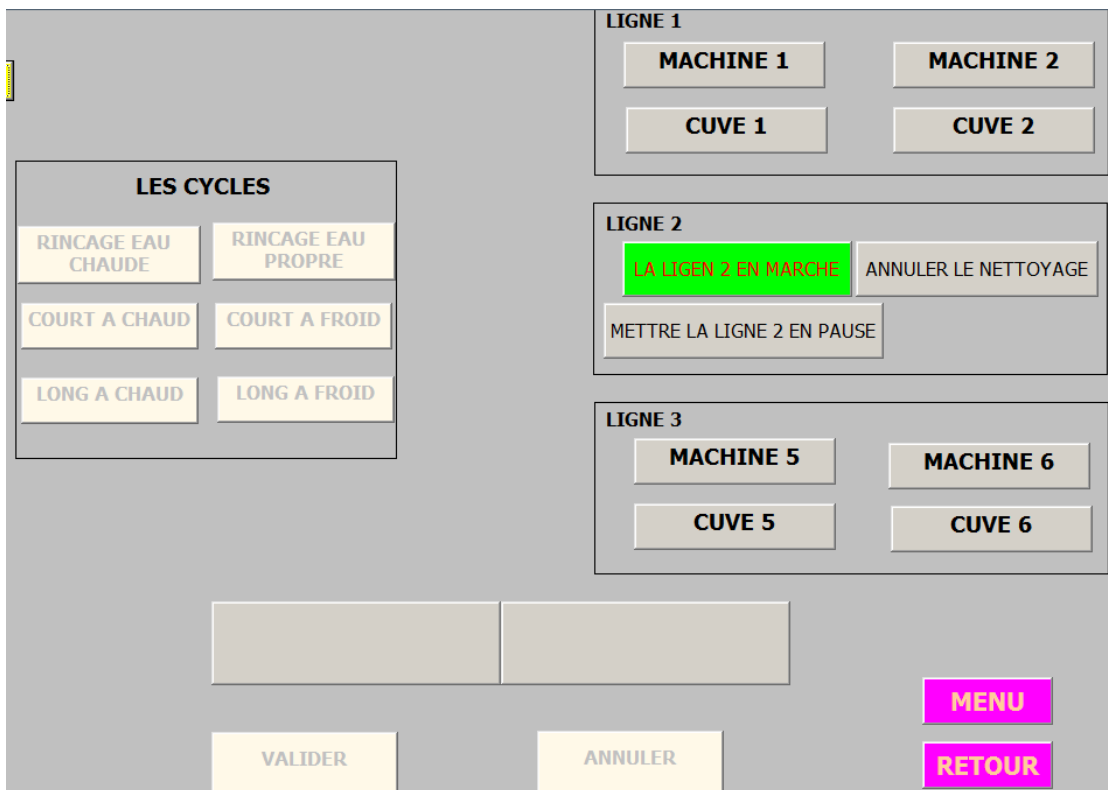


Figure 4.28. La ligne 2 en marche.

#### **4.4 Conclusion :**

L'interface opérateur est nécessaire pour le suivi en temps réel des paramètres d'exploitation de la station NEP. Dès l'apparition d'une anomalie l'opérateur est averti via l'interface de supervision pour qu'il puisse mettre un plan de préventions pour éviter tout risque de fuite ou des dégâts des matériels.



## Conclusion générale

---

Ce projet de fin d'études était globalement très intéressant. Il a permis de découvrir le monde professionnel et à transformer les connaissances de base requises à l'université en compétences professionnelles.

Nous avons fait notre mise en situation professionnelle dans l'entreprise Trèfle qui appartient au champ d'agroalimentaire, et qui est considéré comme le domaine le plus important en Algérie.

Durant toute la période de stage nous avons eu la possibilité de découvrir le milieu industriel, notamment celui d'agroalimentaire, ses infrastructures et ses installations, mais le plus important pour nous c'était de voir de plus près et de manipuler pratiquement toute chose en relation avec notre domaine d'application.

La réalisation nous a poussé à faire appel à toutes nos connaissances et aptitudes d'élèves ingénieur et nous a permis d'appréhender les difficultés que les ingénieurs rencontrent tous les jours dans la vie professionnelle tout en prenant des initiatives personnelles.

Le travail que nous avons réalisé s'inscrit dans le cadre d'une amélioration du procédé d'une station, ainsi que l'exploitation des différents niveaux de la régulation qui intervient dans cette séquence.

Bien que notre stage se soit déroulé dans les bonnes conditions, nous avons rencontré quelques difficultés quand à la disponibilité des ingénieurs de notre spécialité et de certains

outils comme le logiciel « APV », Ce qui a réduit notre champ d'action, pour cela nous avons développé un programme dans le logiciel STEP7 qui sera chargé dans l'automate programmable en vue de commander la station NEP. L'avantage du logiciel STEP7 est de contenir un simulateur d'automate SIEMENS (S7-300 et S7-400), pour simuler des programmes et de faire la communication à l'absence de l'automate réel.

Alors nous avons terminé notre modélisation par l'introduction d'un système de supervision pour garantir et assurer le contrôle et la surveillance du procédé.

La modélisation et la programmation de la séquence de démarrage ainsi que la plateforme de supervision que nous avons créée permettent la compréhension du déroulement de la station NEP

(Les différentes séquences de nettoyage ainsi les étapes de préparation ...)

- **Perspective :**

Ce mémoire étudie les différentes étapes de l'élaboration d'un projet logiciel et l'établissement d'une commande à base d'automate programmable grâce au logiciel de programmation STEP7 et le logiciel de supervision WinCC flexible. À la lumière des résultats obtenus, de nombreuses perspectives s'ouvrent à nous :

- Ajouter les séquences du système de sécurité.
- Ajouter les séquences de variation de la vitesse.
- Ajouter les séquences de régulation.

***Les Grafjets :***

- 1- Grafjet préparation soude.
- 2- Grafjet préparation acide.
- 3- Grafjet préparation eau chaude.

Les Grafjets sont présentés sur des feuilles de format A4.

# Bibliographie

---

- [1] LABADLIA.M et HADJALA.A : 'Etude, Conception et réalisation d'un automate Programmable a 16 entrées et 8 sorties pour commander un Ascenseur', Mémoire D'ingénieur, université de Blida, 1998.
- [2] CHERCHEL.N et HAMMOUM.H : 'Commande d'une machine à injection plastique par un automate S7-300', Mémoire d'ingénieur, université de Blida, 2010.
- [3] HEMICI.B et BELGACEM.M : 'L'automatisation d'une chaîne de production industrielle par l'API S7-300', Mémoire d'ingénieur, École Nationale Polytechnique (ENP), 2005.
- [4] PATRICK.P : 'Automatique, Contrôle et régulation', DUNOD, 2004.
- [5] PATRICK.P : 'Instrumentation et régulation', DUNOD, 2004.
- [6] L'équipe de Tetra Pak : 'Manuel d'opérateur de station NEP Alcip 100', Société Tetra Pak, 2000.
- [7] L'équipe d'Endress Hauser [E+H] : 'Manuel d'opérateur Conductivimètre CLM 223', Société Endress Hauser[E+H], 1999.
- [8] PATRICK.P : 'Manuel de l'API S7-300 et logiciel STEP7', SIMATIC SIEMENS, 2012.
- [9] PATRICK.P : 'Manuels des pupitres digitaux et logiciel WinCC Flexible 2008', SIMATIC SIEMENS, 2012.
- [10] E.BAJIC et B.BOUARD, «Réseau PROFIBUS», Techniques de l'ingénieur, Vol.S8190
- [11] SIEMENS, «PROFIBUS Specification. Normative parts of Profibus-FMS, DP, PA according to the European Standard EN 50170», Vol.2, edition 1.0, PNO, 1998.
- [12] SIEMENS, «PROFIdrive, ProfileDrive Technology », Version 3.0, 2000.
- [13] LOUISJ., MULTONB., BONNASSIEUXY.et LAVABRE M. «Convertisseurs des fréquences pour la variation de vitesse des MAS».
- [14] SIEMENS, «SIMOVERT MASTERDRIVESVC», SIMOVERT, 2005.
- [15] SIEMENS, «STEP 7, Régulation», SIMATIC, 2000.
- [16] [www.siemens.com](http://www.siemens.com)
- [17] Appareils de terrain pour l'automatisation des processus, SIEMENS, 2005.

