

الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne démocratique et populaire

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

جامعة سعد دحلب البليدة
Université SAAD DAHLAB de BLIDA

كلية التكنولوجيا
Faculté de Technologie

قسم الإلكترونيك
Département d'Électronique



Mémoire de Projet de Fin d'Études

présenté par

Aiche Anwar Nadjib

&

Fellag Islam

pour l'obtention du diplôme Master en Électronique option signaux en ingénierie des
systèmes et informatique industrielle

Thème

Commande et contrôle d'un système de chauffage de la matière première de couches bébés « MOLFIX ».

Proposé par : Mr Bennila Nour-Eddine & Boudane mohamed

Année Universitaire 2012-2013



Dédicaces

Je dédie ce modeste travail à :

Mes chères parents, pour leurs soutient moral et financier et d'être la lumière de ma vie, qui m'ont toujours encouragé pour terminer mes études dans de bonnes conditions, en leur espérant une longue vie et que
Dieu les garde.

*A mes frères **Rafik et Mehdi.***

A ma belle-sœur Louiza.

A ma chère grande mere A.Yamna et a tous mes grands parents (qu'ils reposent en paix)

*A ma très chère **Taoug.Amina***

A mes tantes et mes oncles

A tout mes cousins et cousines surtout M.mossab.

A mes chers amis de l'université.

A mon binôme Islam

*A toute la promotion **2012-2013** sans exception.*

Aiche Anwar Nadjib







Dédicaces




Je tiens à dédier ce mémoire :

A ma très chère Mère et à mon cher Père, en témoignage et en gratitude de leurs dévouement, de leurs soutien permanent durant toutes mes années d'études, leurs sacrifices illimités, leurs réconfort moral, pour tout cela et pour ce qui ne peut être dit, mes affections sans limite.

 *A mes frères et sœur : Ahcen, Amine , Nadia.*


 *A mes chers Grands-parents. (Que Dieu les gardes pour moi).*

 *A la mémoire de mon Oncle et ma grand Mère Yousef et Yama que dieu bénit leur âmes.*

 *A mes tantes et mes Oncles et leurs familles.*

 *A toutes les familles : Fellag et Mesmous.*

 *A mon camarade Aiche Anwar Nadjib.*

 *A mes amis fideles et ma meilleure H.Amina.*

Fellag Islam



ملخص:

يتمثل هذا العمل في تطوير وحدة المنظومة المسؤولة عن تلصيق مادتين أوليتين داخلتين في تركيبية حفاظات الأطفال, و هذا عن طريق تسخين مقاومات موضوعة في وسط اسطوانات خاصة بضغط المادتين.

هذا التطوير متمثل في استبدال منظم الحرارة (Thermoregulateur), بالآلي المبرمج (API) الذي يؤمن مهمة تعديل حرارة المقاومات, و شاشة عرض (IHM), التي تعتبر وسيلة لعرض حالة المنظومة و إدخال التعليمات. هذا ما يسمح بالربح في ما يخص دقة تعديل الحرارة و إنقاص كثافة شبكة التوصيلات.

(كلمات المفاتيح): المنظومة, شاشة عرض, الآلي المبرمج, منظم الحرارة.

Résumé : Ce travail consiste à optimiser une unité de processus responsable de soudure de deux matières premières entrant dans la fabrication de couches bébés par le biais de résistances chauffantes placé au centre des cylindres de pressage. Cette optimisation se fait en remplaçant des thermorégulateurs par un API siemens, qui assure la tâche de thermorégulation et une interface homme machine pour l’affichage et l’introduction de consignes. Cela permet de gagné en matière de précisions et de densité de câblages.

Mots clés : API; Thermorégulateurs ; Interface Homme-Machine

Abstract: This work consists in optimizing a unit of process responsible for welding of two raw materials, entering manufacture for babies diapers by heating resistors placed at the center of the cylinders of pressing. This optimization is done by replacing thermoregulators by an PLC siemens , which ensures the spot of thermoregulation and a Human-Machine Interface for the posting and the introduction of instructions. That allows gained accuracy of temperature and decrease density wiring.

Keywords : PLC; thermoregulators ; Human-Machine Interface

Sommaire

Introduction générale	01
------------------------------------	----

Chapitre 1 : présentation de l'unité 'MOLFIX'

1.1 Introduction.....	03
1.2 Présentation du Groupe HAYAT	03
1.3 Présentation de l'unité MOLFIX:	04
1.3.1 Les processus de production	06
1.3.2 La centrale électrique	13
1.4 Conclusion.....	15

Chapitre 2 : Généralités

2.1 Introduction	16
2.2 Automate programmable S7-300	17
2.2.1 Définition	17
2.2.2 Système d'automatisation SimaticS7-300	17
2.3 Logiciel de programmation STEP7	18
2.3.1 Mode d'emploi	19
2.3.2 Principe de conception d'une structure de programme	21
2.4 Interface homme-machine	22
2.4.1 IHM TP 177B	23

2.5 Le logiciel Wincc flexible	24
2.6 Profibus	25
2.6.1 Les différents profils de Profibus	25
2.6.2 Types de transmissions du Profibus DP	26
2.6.3 Structure du télégramme PROFIBUS	26
2.7 Commutateur statique	27
2.7.1 Avantage	28
2.7.2 Domaines d'applications	28
2.8 Thermocouple	28
2.8.1 Thermocouple de type J	29
2.9 Regulation	30
2.9.1 Régulation ou Asservissement.....	30
2.9.2 Chaîne d'une régulation	31
2.9.3 Qualité attendus d'une régulation	31
2.9.4 Régulateur.....	34
2.10 Conclusion	35

Chapitre 3 : Programmation et Réalisation :

3.1 Introduction	36
3.2 Choix des matériels	37
3.2.1 Choix de l'automate programmable	37

3.2.2 Choix de l'interface homme machine	38
3.2.3 Choix du thermocouple et de la résistance	39
3.2.4 Choix du commutateur	40
3.2.5 Choix du module d'entrée analogique	40
3.3 Structure du programme	41
3.3.1 Configuration matériel	42
3.3.2 Fonction de mise en échelle (scale).....	43
3.3.3 Fonction du régulateur PID	46
3.3.4 Communication Profibus API-IHM.....	48
3.3.5 Programmation de l'interface Homme-Machine	51
3.4 Réalisation	53
3.4.1 Schéma de câblage	54
3.4.2 Détermination des paramètres du correcteur PID	55
3.4.3 Méthode d'identification <i>Ziegler-Nichols</i>	55
3.5 Conclusion	58

Chapitre 4 : Développement des résultats

4.1 Introduction	59
4.2 Simulation du bloc FC 105 'scall'	60
4.3 Simulation du bloc FB 58 'TCONT'	61
4.4 Runtime du projet Wincc flexible	61

4.5	Calcule des paramètres PID avec	
	la méthode de ziegler-Nichols	63
4.5.1	Réponse indicielle du système en boucle ouverte	63
4.5.2	Détermination du retard apparent τ et de la tangente α et R.....	63
4.5.3	Détermination des paramètres	64
4.5.4	Résultats du câblage	64
4.5.5	Ajustement des paramètres	66
4.6	Conclusion	67
5	Conclusion generale.....	68

Introduction générale

La majorité des procédés industriels nécessitent le contrôle d'un certain nombre de paramètres physiques mesurables: température, pression, niveau, débit, ph, concentration, vitesse...etc de les préserver à des niveaux prédéterminés, afin qu'ils régissent le comportement du processus, ce contrôle est nommé régulation automatique.

La régulation a pour but de maintenir automatiquement une grandeur physique à une valeur désirée, appelée point de consigne ; Cela se fait en se basant sur la mesure de l'écart entre la grandeur physique à régler et la valeur de consigne, pour délivrer un signal de demande d'énergie. Ce signal vient agir sur l'organe de réglage.

On appelle une régulation agissant sur une température, la thermorégulation. Cette dernière joue un rôle primordial dans une grande part des chaînes de production, puisque le contrôle de la température permet d'éviter des dégâts matériels et humains, et de garantir dans certains cas une meilleure qualité de production et de bonnes conditions de travail.

Ce qui conduit au fait que la performance de système de régulation de température en vue de les rendre les plus autonomes possible et de faciliter leur maintenance, est une tâche très importante. Elle assure un meilleur rendement de la machine et du personnel, d'où l'intérêt de notre travail, qui consiste à remplacer des thermorégulateurs par un automate programmable et une interface homme machine dans un système, qui permet de souder deux matières premières entrant dans la fabrication de couches bébé sous l'effet de la chaleur.

Afin de mieux appréhender l'importance de l'unité de processus en question au sein des lignes de production de la société, nous avons réservé le premier chapitre pour la présentation du groupe HAYAT et de l'unité MOLFIX et définir les différents processus de la

Introduction Générale

machine CB3 (couche bébé 3).Le deuxième chapitre consiste à présenter des généralités sur le matériels et les outils utilisés dans ce travail.

Le troisième chapitre illustre les différentes étapes suivies pour la réalisation de ce projet, enfin le dernier chapitre a été consacré pour la simulation et le développement des résultats obtenus.

On a clôturé ce projet par une conclusion générale qui résume les objectifs atteints, et évalue le travail effectué.

Liste des tableaux

- Tableau 01 : Tableau pour le calcul des paramètres PID,Réglage Ziegler-Nichols en boucle ouverte.....	56
- Tableau 02 : Valeurs d'évolution de la température du système en boucle ouverte.....	57
- Tableau 03 : Essais d'ajustement des paramètres du régulateur.....	66

Conclusion Générale

Le travail présenté se situe dans le domaine de l'automatisation et l'optimisation des processus industriels, afin de faciliter leur maintenance et d'augmenter la qualité de leur production. Cette attente légitime du consommateur engage les industriels dans de nouveaux efforts de recherche en vue d'augmenter les performances de leurs installations.

L'objectif est, d'optimiser et de fiabiliser un des procédés d'une usine de production de couche bébés afin de garantir à tout moment une qualité optimale des produits, et ce, si possible à un coût moindre pour l'exploitant.

Notre but a donc été de remplacer des thermorégulateurs par un automate programmable industriel, et une interface homme-machine pour remédier aux defaults que présentent ces instruments de régulation, et de faciliter la surveillance du processus.

Dans ce travail on a utilisé le logiciel STEP 7 dont on n'avait aucune connaissance, qui a servi de logiciel de programmation de l'automate programmable qui pilote le processus. Cependant la visualisation a été programmé par le logiciel Wincc flexible qu'on a appris à utiliser durant ce projet.

Notre tâche été de programmer l'automate et l'interface de visualisation et de les relier en software et hardware, tout en identifiant le régulateur qui remplacera au sein du programme de manière idéal les thermorégulateurs, ainsi que de prévoir les défaillances que peut présenter le système et de les affiché en tant qu'alarmes dans l'ecran de visualisation.

L'étape la plus délicate été de déterminer le bon compromis entre les paramètres du régulateur, et nous estimons que seuls des essais expérimentaux donnerons l'occasion d'identifier les paramètres applicable en pratique.

Conclusion Générale

Notons que la disponibilité de moyen matériel n'été pas en notre faveur, ce qui a retardé l'avancement du travail et limité le degré performance qu'on pouvait apporter au système.

Compte tenu des résultats obtenus, nous estimons qu'ils sont satisfaisants, et qu'ils pourront être utilisés pour moderniser des procédés de ce genre.

En conclusion on voudrait mentionner que ce travail été bénéfique pour nous, vue qu'il nous a permis d'exploiter nos connaissances théoriques sur le terrain, et nous souhaitons qu'il soit ainsi pour les générations à venir.

2.1 Introduction :

Les généralités présentées dans ce chapitre, démontrent l'importance et l'intérêt des outils et des méthodes utilisées pour l'aboutissement à l'optimisation du procédé de thermorégulation au sein de l'unité de traitement « thermosealing ».

2.2 Automate programmable S7-300 :

2.2.1 Définition :

L'automate programmable industriel (API), ou en anglais Programmable Logic Controller (PLC), est une machine électronique programmable destinée à piloter dans une ambiance industrielle et en temps réel, des procédés logiques séquentiels. Autrement dit, un utilisateur l'utilise pour le contrôle et essentiellement la commande d'un procédé industriel en assurant l'adaptation nécessaire entre tout ce qui est de grande puissance, par rapport à ce qui est de faible puissance côté commande. Son objectif principal est de rendre tout le mécanisme de type "laisser-faire-seul" : le système contrôle ses sorties, décide et agit sur ses entrées afin de maintenir le fonctionnement comme prévu par l'utilisateur. C'est le principe de l'automatisme.

2.2.2 Système d'automatisation SimaticS7-300 :

L'automate utilisé dans notre projet appartient à la gamme *SIMATIC S7* de *SIEMENS*, un mini-automate modulaire pour les applications d'entrée et de milieu de gamme, avec possibilité d'extensions jusqu'à 32 modules, et une mise en réseau par l'interface Multipoint (MPI), PROFIBUS et Industrial Ethernet.

L'automate S7 est constitué d'une alimentation, d'une CPU et de modules d'entrées ou de sorties (Modules E/S). A ceux-ci peuvent s'ajouter des processeurs de communication et des modules de fonction qui se chargeront de fonctions spéciales, telles que la commande d'un moteur pas à pas par exemple.

- **Alimentation :**

Elle convertit la tension réseau (AC 120/230 V) en tension de service DC 24 V et assure l'alimentation du S7-300, ainsi que l'alimentation externe pour les circuits de charge DC 24 V.

- **La CPU S7 300 :**

Elle exécute le programme utilisateur, alimente le bus de fond de panier du S7-300 en 5 V, et communique avec les autres partenaires d'un réseau MPI via l'interface MPI. En outre, une CPU peut être maître ou esclave DP sur un sous-réseau PROFIBUS, elle contient aussi :

- Des LEDs de visualisations comme par exemple « SF » pour signaler un défaut logiciel ou matériel.

-Un commutateur de mode de fonctionnement pour basculer entre les quatre modes de fonctionnement : RUN, RUN-P, STOP, MRES.

-Une pile de sauvegarde pour l'horloge temps réel, ou la sauvegarde du programme utilisateur.

-Interface multipoints (MPI), utilisée pour la console de programmation(PG), le pupitre operateur (OP) ou par la communication au sein d'un réseau MPI. La vitesse de transmission typique est de 187,5 k Bauds

-Carte mémoire, dont le rôle est de sauvegarder le programme utilisateur, le système d'exploitation et les paramètres qui déterminent le comportement de la CPU.

- **modules d'entrées ou de sorties (Modules E/S) :**

Ils assurent le rôle d'interface entre la CPU et le processus, en adaptant les différents niveaux des signaux de processus au S7-300. Ces modules peuvent être de type analogique ou TOR (tout ou rien).

L'automate programmable S7-300 contrôle et commande une machine ou un processus à l'aide du programme STEP7. C'est pour quoi une présentation de ce logiciel est faite dans ce qui suit. [4].

2.3 Logiciel de programmation STEP7 :

STEP 7 est le progiciel de base pour la configuration et la programmation des systèmes d'automatisation SIMATIC. Il s'exécute sous un environnement Windows, à partir d'une console de programmation ou d'un PC.

STEP 7 assiste l'utilisateur dans toutes les phases du processus de création des solutions d'automatisation, comme par exemple :

- ✓ la création et la gestion de projets,
- ✓ la configuration et le paramétrage du matériel et de la communication,
- ✓ la gestion des mnémoniques,
- ✓ la création de programmes, par exemple pour les systèmes cible S7,
- ✓ le chargement de programmes dans des systèmes cible,

- ✓ le test de l'installation d'automatisation,
- ✓ le diagnostic lors de perturbations de l'installation.

2.3.1 Mode d'emploi :

Avant de créer un projet, on peut envisager différentes approches (Figure 2.1). En effet, le logiciel STEP 7, offre une liberté de choix de la procédure à adapter.

Cependant, il est préférable de commencer par configurer le matériel avant la création du programme Correspondants. Il est très facile de remplacer des modules, car les paramètres définis avec STEP 7 sont automatiquement chargés dans le nouveau module à la mise en route.

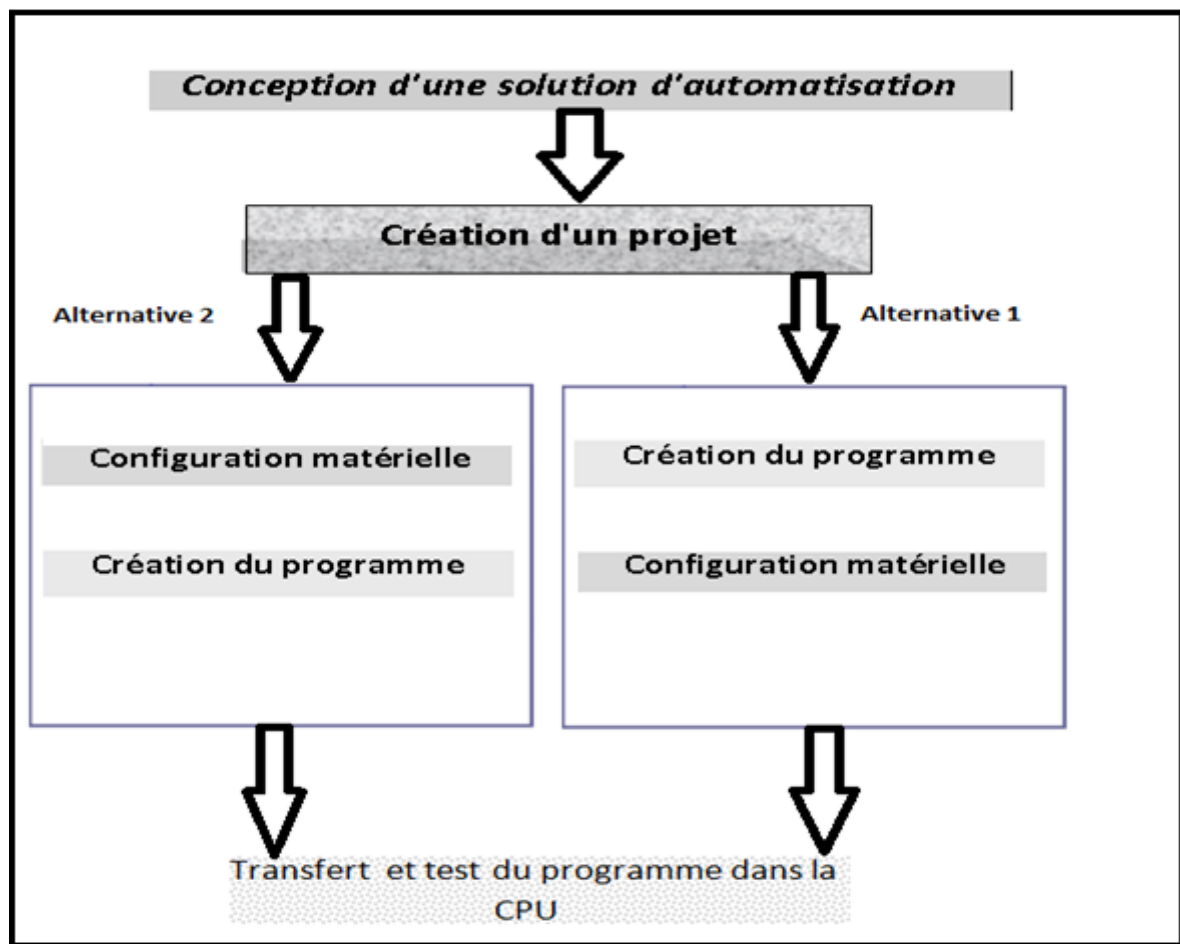


Figure 2.1 : Les approches offertes par le logiciel Step7

- **Configuration matériel et paramétrages :**

L'application de la configuration matérielle de STEP 7 présente l'avantage que les adresses sont prés-sélectionnées, comme l'illustre la figure 2.2. La configuration matérielle nous permet, non seulement de sélectionner les adresses, mais également de modifier les paramètres et les propriétés des modules.

Par "configuration", on entend la disposition de profilés support ou châssis, de modules, d'appareils de la périphérie décentralisée et de cartouches interface dans une fenêtre de station. Les profilés support ou châssis, sont représentés par une table de configuration, dans laquelle l'on peut enficher un nombre défini de modules, tout comme dans les profilés support ou châssis "réels".

STEP 7 affecte automatiquement une adresse à chaque module dans la table de configuration.

Tandis que le "paramétrage", est le réglage des paramètres des modules paramétrables pour la configuration centralisée et pour un réseau. Exemple : une CPU est un module paramétrable. La surveillance du temps de cycle est un paramètre qu'on peut définir, et la définition des paramètres de bus, des maîtres et d'esclaves pour un réseau maître (PROFIBUS) ou d'autres définitions pour l'échange de données entre des composants. Ces paramètres sont chargés dans la CPU qui, lors de son démarrage, les transmet aux modules.[3].

The screenshot shows the HW Config interface. The top part displays a rack configuration with slots 1 through 8. Slot 1 is empty, slot 2 contains a CPU 313C-2 DP, slot 3 contains a DP module, slot 4 contains a DI16/DO16 module, slot 5 contains a Comptage module, and slot 6 contains an AI2x12Bit module. The bottom part shows a detailed table of the configured modules.

slot	Module	Référence	Firmware	Adresse MPI	Adresse d'entrée	Adresse de...	Commentaire
1	CPU 313C-2 DP	6ES7 313-6CG03-0AB0 V2.6		2			
2	DP				1023*		
3	DI16/DO16				124..125	124..125	
4	Comptage				788..789	788..789	
5	AI2x12Bit	6ES7 331-7K802-0AB0			256..258		
6							
7							
8							

Figure2.2 : Configuration matériels

- **Création du programme :**

Le logiciel STEP7, offre trois langages de programmation :

-**CONTACT** : dont la synthèse d'instructions s'inspire des schémas à relais.

-**LIST** : dont la syntaxe des instructions ressemble à celle du langage assembleur.

-**LOGIGRAMME** : langage de programmation graphique qui utilise les boites logiques de l'algèbre de Boole.

2.3.2 Principe de conception d'une structure de programme :

Dans le logiciel STEP7, le programme peut être organisé en plusieurs structures, et subdivisé en parties autonomes, afin de simplifier l'organisation et la modification du programme et ainsi d'en standardiser certaines parties.

Ces structures sont appelées BLOCS. Ces blocs contiennent des parties de programme et sont de ce fait également désignés comme blocs de code comportant une section de déclaration des variables et une section d'instructions. Le nombre de blocs autorisés par type de bloc, ainsi que la longueur maximale de chaque bloc, dépendent de la CPU.

a- Les blocs d'organisation (OB) :

Les blocs d'organisation définissent l'ordre (événements de déclenchement) dans lequel les différentes parties du programme sont traitées. Ils sont appelés par le système d'exploitation selon leur priorité.

- **Bloc d'organisation de traitement cyclique (OB1)** : à l'exécution cyclique du programme utilisateur et comprend les appels correspondant aux blocs fonctionnels FB ou aux fonctions FC.
- **Blocs d'organisation pour l'alarme horaire (OB10 à OB17)** : qui permettent aux alarmes horaires d'être traitées à une date donnée ou à des intervalles de temps définis.
- **Blocs d'organisation pour l'alarme temporisée (OB20 à OB23)** : L'utilité des OB d'alarme temporisée est de programmer l'exécution retardée de certaines parties du programme utilisateur.
- **Blocs d'organisation pour l'alarme cyclique (OB30 à OB38)** : Leur fonction est d'interrompre le traitement de programme cyclique à des intervalles de temps précis.

b- Les blocs fonctionnels (FB) :

Les blocs fonctionnels sont subordonnés aux blocs d'organisation. Ils renferment une partie du programme qui peut être appelée dans l'OB1 ou dans un autre bloc fonctionnel FB.

c- Blocs fonctionnels système (SFB) et fonctions système (SFC) :

Les SFB et SFC sont intégrés à la CPU S7 et nous permettent de réaliser quelques fonctions systèmes importantes.

d- Fonctions (FC) :

Les FC contiennent des routines de programmes pour les fonctions fréquemment utilisées. Une fonction contient un programme qui est exécuté quand cette fonction est appelée par un autre bloc de code.

e-blocs de données (DB) :

Ces blocs ne contiennent pas d'instructions, leur fonction est de mémoriser les données nécessaires au traitement du programme et les données affectées à chaque bloc fonctionnel. On distingue deux types de blocs de données :

- **Blocs de données d'instance :** Un bloc de données d'instance est associé à chaque appel de bloc fonctionnel transmettant des paramètres.
- **Blocs de données globaux :** Ils servent à l'enregistrement des données du programme utilisateur pouvant être utilisées par tous les autres blocs.[4].

2.4 Interface homme-machine :

Les concepts d'automatisation modernes ont, sans cesse, des exigences croissantes en matière de visualisation des processus. Plus particulièrement, il est impératif que la conduite des processus, au niveau machine, fournisse une réponse adaptée aux besoins de simplicité et de performance. L'objectif est de présenter, rapidement et de manière fiable, des données de processus immédiatement compréhensibles par l'opérateur, par exemple, sous la forme d'une courbe graphique. Il est donc indispensable d'archiver les données de processus dès le niveau machine, d'où l'indispensabilité des interfaces homme-machine.

Une interface homme-machine (IHM) est une interface qui permet une interaction entre un être humain et une machine. Les IHM varient considérablement, de panneaux de

contrôle pour centrales nucléaires à l'écran et des touches d'entrée sur un téléphone cellulaire.

Deux composants sont nécessaires dans une interface homme-machine. La première est une entrée, un utilisateur humain a besoin d'une certaine façon d'indiquer à la machine à faire, ou faire des demandes de la machine, ou la régler. Des exemples de dispositifs d'entrée incluent des claviers, des clés, des commutateurs, des écrans tactiles, manettes de jeux et des souris. Tous ces dispositifs peuvent être utilisés pour envoyer des commandes à un système, ou même un ensemble interdépendant de systèmes.

L'interface nécessite également une sortie, ce qui permet à l'appareil de maintenir l'utilisateur humain mis à jour, sur l'état d'avancement des commandes, ou à exécuter des commandes dans l'espace physique. Sur un ordinateur, par exemple, les utilisateurs disposent d'un écran qui peut afficher des informations. Les sorties peuvent également comprendre des choses aussi simples que des voyants d'état qui alertent les gens quand bascule ou des commutateurs ont été activés.

Les SIMATIC Panels de siemens font leurs preuves depuis des années dans les applications et les secteurs les plus divers.

Ils possèdent non seulement un design innovant, mais offrent également des performances élevées. Avec une efficacité de l'ingénierie inégalée jusqu'ici, les pupitres SIMATIC sont clairement structurés :

- **Les SIMATIC HMI Basic Panels** offrent des fonctions de base pour les applications IHM simples.
- **Les SIMATIC HMI Confort Panels** conviennent aux applications exigeantes.

2.4.1 IHM TP 177B :

La série 177 des panneaux représente une nouvelle avancée dans le développement de la bien connue 170 pupitres.

Les TP177B sont conçus pour une meilleure utilisation de texte ou de projets graphiques a base de simple moyens et taches de surveillance dans les machines et installations, Elles offrent la possibilité de raccordement profibus et profinet ,ainsi qu'une mise en service rapide, une mémoire de grande capacité et une haute performance.

2.5 Le logiciel Wincc flexible :

WinCC est un système IHM performant qui est utilisé sous Microsoft Windows 2000 et Windows XP, utilisé pour la programmation graphique des pupitres.

Le contrôle, proprement dit, du processus est assuré par les automates programmables (API). WinCC nous permet juste de visualiser le processus et de concevoir l'interface utilisateur graphique destiné à l'opérateur. Pour ce faire, le processus est visualisé par un graphisme à l'écran. Dès qu'un état du processus évolue, l'affichage est mis à jour.

WinCC facilite la commande du processus. A partir de l'interface utilisateur graphique, l'opérateur peut, par exemple, entrer une valeur de consigne ou ouvrir une vanne.

Lorsqu'un état de processus devient critique, une alarme est déclenchée automatiquement. L'écran affiche une alarme en cas par exemple. de franchissement d'un seuil défini.

Les alarmes et valeurs de processus peuvent être imprimées et archivées sur support électronique par WinCC. Ceci nous permet de documenter la marche du processus et d'avoir accès ultérieurement aux données de production du passé.

Le système de base WinCC se compose des sous-systèmes suivants :

- Système graphique.
- Système de signalisation.
- Système d'archivage.
- Système de journalisation.
- Communication.
- Gestion des utilisateurs.

Ce logiciel est aussi subdivisé en logiciel de configuration (CS) et en logiciel runtime (RT) :

- **Le logiciel de configuration** permet de créer un projet.
- **Le logiciel runtime** permet de mettre le projet en œuvre dans le cadre du processus. Le projet est alors « en Runtime ».

Puisque le contrôle proprement dit du processus est assuré par les API, une communication doit s'établir donc entre WinCC et l'opérateur d'une part et entre WinCC et les automates programmables d'autre part. Cette dernière peut être une communication via profibus. [8].

2.6 Profibus :

PROFIBUS (Process Field Bus) est un réseau de terrain ouvert inventé par Siemens, non propriétaire, répondant aux besoins d'un large éventail d'applications dans les domaines du manufacturier et du process. Son universalité et son indépendance vis-à-vis du constructeur, ainsi que son ouverture, sont garanties par les normes européennes EN 50170, EN 50254, et international IEC61158.

PROFIBUS autorise le dialogue de matériels multi-constructeurs, sans passer par des interfaces spécialisés. Il se prête aussi bien à la transmission de données exigeant des actions réflexes, en des temps de réaction très courts qui peuvent aller jusqu'à 12Mb/s, qu'aux échanges de grandes quantités d'informations complexes. Il est réputé pour sa simplicité d'installation et d'exploitation, des gains en câblage, son immunité face aux parasites

Son architecture repose sur 3 couches inspirées du modèle en 7 couches de l'OSI, suivant la norme ISO 7498.

La couche 1, **physique**, décrit les caractéristiques physiques de la transmission.

La couche 2, **liaison de données**, spécifie les règles d'accès au bus.

La couche 7, **application**, définit les mécanismes communs utiles aux applications réparties, et la signification des informations échangées.

2.6.1 Les différents profils de Profibus :

Grâce à ses différents profils, PROFIBUS permet de répondre de façon efficace à l'ensemble des besoins des installations automatisées

- **Le Profibus-FMS** est le premier à avoir été utilisé pour communiquer plusieurs automates. Puis il fut remplacé par Ethernet, donc peu utilisé de nos jours.
- **Le Profibus-PA** (Process Automation) contrôle et raccorde des équipements de mesure comme des capteurs et actionneurs dans les processus continus, par l'intermédiaire d'un système de contrôle de procédé.

- **Le Profibus-DP** (Decentralised Peripheral ou périphérique décentralisée) est le profil de communication le plus répandu dans l'industrie et le plus prisé pour sa rapidité, ses performances et sa connectique à faible coût. Il est utilisé pour la commande d'actionneurs, ou la vérification de l'état des capteurs, ou bien commander un autre automate programmable.

Sa vitesse peut aller Jusqu'à 12 Mbits / s, et il accepte jusqu'à 32 équipements sans répéteur, et 126 équipements avec répéteur. Ceux-ci régénèrent un signal qui permet de cascader les segments, a condition que leur nombre ne dépasser pas 9 répéteurs entre un équipement et le maître.

Ces profils fonctionnent tous suivant le même protocole, deux réseaux DP et PA peuvent être interconnectés à l'aide d'un adaptateur.[7].

2.6.2 Types de transmissions du Profibus DP :

Profibus DP utilise deux types de services de transmission en envoyant les télégrammes de messages qui sont définis dans la couche 2 et résumées ci-dessous :

- **-SRD (Send and Request Data with acknowledge)** : Avec SRD, les données sont envoyées et reçues dans un seul cycle de télégramme unique. Autrement dit, le maître envoie des données de sortie de l'esclave et reçoit des données d'entrée à partir de l'esclave dans sa réponse, dans un délai de temps spécifié, C'est le service de transmission le plus souvent utilisé dans les PROFIBUS DP qui rend l'échange très efficace.
- **SDN (Send Data with No acknowledge)** : Ce service est utilisé lorsqu'un message doit être transmis simultanément à un groupe d'esclaves, où tous les esclaves ne répondent pas.

2.6.3 STRUCTURE Du TELEGRAMME PROFIBUS (Message) :

-Un télégramme Profibus (figure 2.3) peut contenir jusqu'à 256 octets par message.

SD	LE	LEr	SD	DA	SA	FC	DSAP	SSAP	DU	FCS	ED
1b	1b	1b	1b	1b	1b	1b	1b	1b	1 à 244b	1b	1b

Figure 2.3 Structure du télégramme profibus

- Délimiteur de début (SD)** : Identifie le début d'un télégramme et son format général.
- Longueur du télégramme (LE &Ler)** : Cet octet indique la longueur d'un télégramme.
- Adresse de destination et Adresse source (DA et DS)** : Le dispositif de maître s'adresse à un dispositif esclave spécifique en plaçant l'adresse de l'esclave de 8 bits dans le champ d'adresse du télégramme de DA (DestinationAdresse). Il dispose de sa propre adresse dans le champ d'adresse SA (Sourceadresse).
- Code de fonction (FC)** : spécifie le type du télégramme (requête, réponse, reconnaissance), type de station (passive ou actif / esclave ou maître), la priorité, et la reconnaissance du télégramme (réussie ou de échec).
- **Points Accès au service (SSAP et DSAP)** : Gèrent les échanges de données dans l'en-tête du télégramme.
- Unité de données (DU)** : Cette zone contient les données de la station de DA (données de la requête), ou des données pour la station de SA (données de réponse).
- Cadre de vérification de séquences** : Ce champ contient le contrôle du télégramme.
- Délimiteur de fin (ED)** : Cet octet identifie la fin d'un télégramme Profibus.
- En perpétuelle évolution, PROFIBUS reste le réseau de communication industriel du futur, et c'est pourquoi on a opté pour son utilisation dans notre travail. [6].

2.7 Commutateur statique :

Comme son nom l'indique, un relais sert en tout premier lieu à « relayer », c'est à dire à faire une transition entre un courant faible et un courant fort. Mais il sert également à commander plusieurs organes simultanément grâce à ses multiples contacts synchronisés.

Il permet également la transition entre deux sources différentes, en isolant ces dernières. Il autorise des temporisations, des verrouillages, des impulsions ...les fonctions d'un relais sont aussi nombreuses que différentes.

Un relais statique est par définition un organe ayant la fonction d'un relais, mais réalisé avec des composants électroniques, sans aucune pièce mécanique en mouvement.

Son fonctionnement reste assez simple puisqu'il s'apparente au fonctionnement d'un optocoupleur. En fait, on pilote une led, qui vient à son tour éclairer l'électrode de commande d'un Opto-MOS, ou bien d'un Opto-triac, ou encore d'un Opto-thyristor.

2.7.1 Avantage :

Les relais statiques sont sûrs, faciles à connecter, et ont une longue durée de vie. Ils ne nécessitent pas de service d'entretien, permettent l'exactitude de la température, et réalisent les opérations silencieuses, commutation rapide, nombre de pièces réduit, endurance, résistance aux chocs et vibrations, gradation, pas d'arc et d'étincelle, facile à connecter.

2.7.2 Domaines d'applications :

Bien qu'il y ait littéralement des milliers de différents usages de relais et contacteurs statiques, la majorité est classée dans les applications suivantes :

- **Contrôle de Température** : Il s'agit du plus grand segment d'utilisation des relais statiques, et leur applications s'étend des équipements professionnels alimentaires, machine de moulage/extrusion en plasturgie, équipement de soudure jusqu'à la climatisation et chauffage.
- **Contrôle de Moteurs** : Incluent les ascenseurs, les grues, les équipements de sport, les systèmes de transmission, traqueurs solaires, ventilateurs et contrôle d'électrovannes. [9].

2.8 Thermocouple :

Lorsque deux fils, composés de métaux différents, sont joints à leurs deux extrémités et l'une des extrémités est chauffée, un courant continu circule dans le circuit thermoélectrique. Thomas Seebeck a fait cette découverte en 1821.

Si ce circuit est rompu au centre, il génère une tension de Seebeck, qui est une fonction de la température de jonction et de la composition des deux métaux, c'est le principe d'un thermocouple.

Un thermocouple a au moins deux jonctions (Figure 2.4) : une jonction de mesure appelée soudure chaude, et une jonction de référence appelée soudure froide. En règle générale, la jonction de référence est créée quand les deux fils se connectent à l'appareil de mesure. Cette jonction, est composé en réalité de deux jonctions ; une pour chacun des deux fils, mais parce qu'elles sont supposées être isotherme, alors elles sont considérées comme une seule jonction thermique.

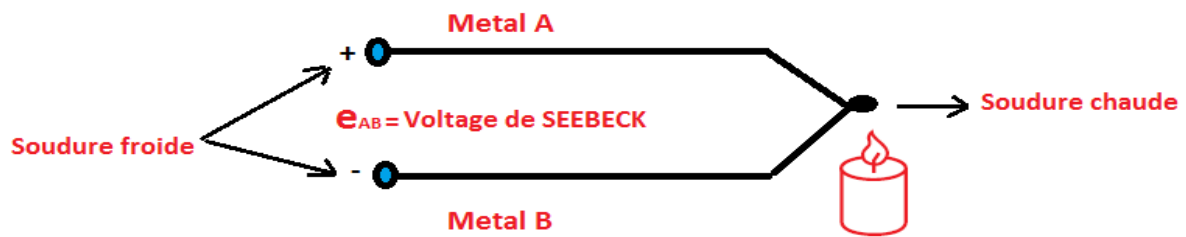


Figure 2.4 Structure interne d'un thermocouple

Autrement dit un thermocouple produit une f.e.m en relation avec la différence de température au niveau de ses deux jonctions, lorsque la température de soudure froide est connue et fixe , celle de la soudure chaude pourra etre connue par la mesure de tension obtenue.

Si la température de la soudure froide ne peut etre connue et constante, elle devra etre compensée à l'aide d'une thermistance, qui corrige la non linearité des courbes de la f.e.m, selon des tables de références.

Un certain nombre de conducteurs peuvent etre associés pour former des thermocouples. Une selection de combinaisons formant les gammes de thermocouple est utilisé actuellement et identifié par un symbole. Les types de thermocouple peuvent différer en matière de stabilité et de plage de temperature a surveiller, ainsi qu'a la f.e.m qui génèrent.

Dans le present travail on a utilisé un thermocouple de type J.

2.8.1 Thermocouple de type J :

Un thermocouple de type j est un thermocouple dont la borne dont la polarité est positif est constitué de « Fer », et la borne negative est une combinaison d'un alliage

de « Cuivre et Nickel » nommé « Constantan ». On reconnaît ce thermocouple par sa gaine isolante qui est de couleur noire et ses fils qui sont noirs et blancs.

Ce type de thermocouple peut mesurer des températures entre -210 et $+1200$ °C. Il est fragile à basse température et sensible à l'humidité, mais possède une très bonne tenue en milieu réducteur. Sa tolérance est donnée par l'écart maximum en degrés Celsius des plages de température :

-**Classe 1** : ± 1.5 °C de -40 °C à 375 °C et $\pm 0.004 \times$ température de 375 °C à 750 °C.

-**Classe 2** : ± 2.5 °C de -40 °C à 333 °C et $\pm 0.0075 \times$ température de 333 °C à 750 °C.

Afin de prolonger un thermocouple jusqu'au dispositif de mesure, il est important d'utiliser un câble et des connecteurs ayant des propriétés identiques au câble du thermocouple, ces câbles peuvent être soit des câbles d'extension ou de compensation.[11].

2.9 Régulation :

Une régulation regroupe l'ensemble des techniques utilisées visant à contrôler une grandeur physique. Par exemple : Pression, température, débit, niveau etc...

La grandeur physique que l'on désire contrôler est la grandeur réglée. Elle donne son nom à la régulation. Exemple : une régulation de température.. Cette grandeur réglée doit prendre une valeur qui est appelée consigne.

Les grandeurs perturbatrices sont les grandeurs physiques qui influencent la grandeur réglée. Elles ne sont généralement pas de même nature que celle-ci.

L'organe de réglage est l'élément qui agit sur la grandeur réglante.

2.9.1 Régulation ou Asservissement

Dans une régulation, on s'attachera à maintenir constante, la grandeur réglée d'un système soumis à des perturbations tandis que dans un asservissement, la grandeur réglée devra suivre les variations de la consigne.

2.9.2 Chaîne d'une régulation :

Afin d'obtenir le fonctionnement désiré, une régulation doit agir en continu sur le procédé (Figure 2.5). Pour cela il faut observer la grandeur à maîtriser par des capteurs, faire une réaction en comparant cette grandeur à celle désirée et, en déterminant l'action à entreprendre par un régulateur, puis agir sur le procédé par cette action. On obtient alors une chaîne de régulation.

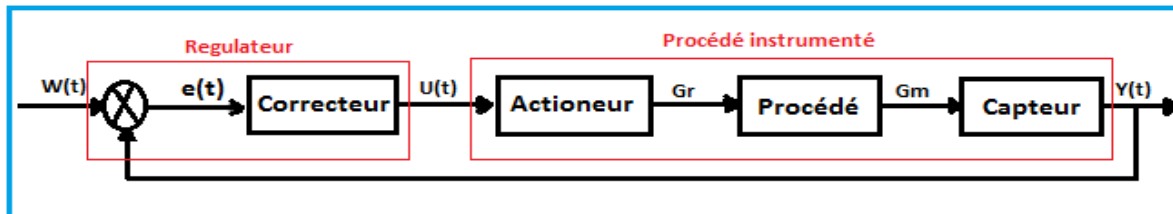


Figure 2.5 : Chaîne d'une régulation

On parle de fonctionnement en boucle ouverte, quand il n'y a pas de retour de l'état du processus, tandis qu'en boucle fermée l'organe de réglage est autonome, c'est le fonctionnement normal d'une régulation par laquelle, le régulateur compare la mesure de la grandeur réglée et la consigne, et agit en conséquence pour s'en rapprocher.

2.9.3 Qualités attendues d'une régulation :

Les performances d'une régulation peuvent se définir à partir de l'allure du signal de mesure, suite à un échelon de consigne. Notons toutefois que les critères de performances classiques peuvent se résumer comme suit :

- **Stabilité** : Cette condition est impérative mais avec un certain degré de stabilité (marge de sécurité). En général, on impose une marge de gain de 2 à 2.5.
- **Précision** : L'exploitant demande à ce que le système possède une bonne précision en régime permanent.
- **Rapidité** : On demande en pratique que le système soit capable rapidement de compenser les perturbations et de bien suivre la consigne.
- **Dépassement** : En général, on recommande un système de régulation dont le régime transitoire soit bien amorti, et dont le dépassement ne dépasse pas 5 à 10% de la valeur nominale.

a-Stabilité :

Le système constitué du procédé et de la boucle de régulation est dit stable (figure 2.6), s'il est soumis à une variation de consigne, la mesure retrouve un état stable, dans le cas contraire le système est dit instable (figure 2.7).

Pour un système stable, le temps écoulé pour retrouver la stabilité constitue le régime transitoire.

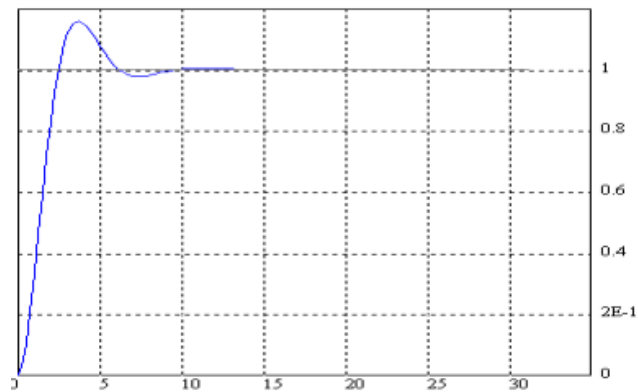


Figure 2.6 Réponse indicielle d'un système stable

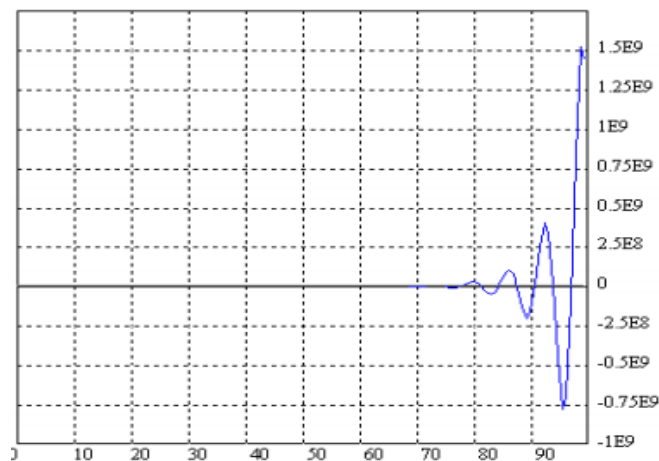


Figure 2.7 Réponse indicielle d'un système instable

b-Précision :

Elle est définie à partir de l'erreur statique ξ en régime stable comme le montre la Figure 2.8 :

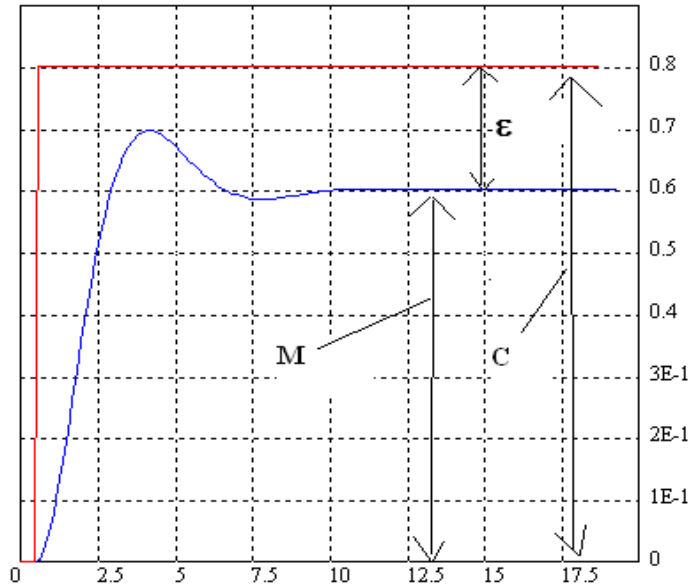


Figure 2.8 : illustration de l'erreur statique sur la réponse indicielle d'un système

Erreur de précision (%) = $(\epsilon/C) \cdot 100$

c-Rapidité :

Elle traduit pratiquement la durée transitoire. Plus précisément, elle s'exprime par le temps de réponse T_e ou temps d'établissement, qui est le temps mis par la mesure pour atteindre sa valeur définitive à $\pm 5\%$ de sa variation, tout en se maintenant dans cette zone des $\pm 5\%$.

Rapidité = temps de réponse T_e

La figure 2.9 représente des réponses oscillatoires amorties.

T_e : correspond au temps mis pour atteindre 95 % ou 105% de la valeur finale de ΔM .

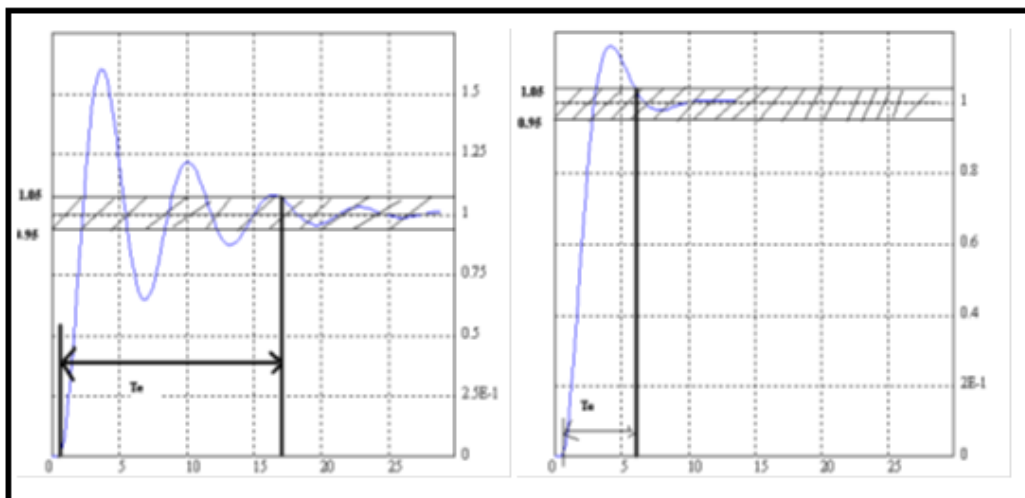


Figure 2.9 : réponses oscillatoires amorties avec T_e différents

d-Compromis précision-rapidité :

Le cahier de charge de régulation comporte plusieurs objectifs, qui sont parfois contradictoires comme, la précision et la rapidité. En effet il est souvent difficile, voir impossible, d'obtenir une très bonne précision avec une très grande rapidité. C'est pour quoi, un réglage optimal d'une régulation sera toujours le fruit d'une recherche du meilleur compromis entre la précision et la rapidité. [10].

2.9.4 Régulateur:

Le régulateur élabore une grandeur de commande en fonction de l'écart de réglage du système à commander. Selon la complexité du régulateur, la grandeur de commande à l'instant précis est fonction de la valeur de l'écart à cet instant, mais aussi aux instants précédents.

- **Régulateur PID :**

Les régulateurs PID répondent à plus du 90% des besoins industriels. Le nombre de régulateurs installés dans une usine pétrolière, par exemple, se compte par milliers. Malheureusement, malgré l'expérience acquise au fil des ans, les valeurs choisies pour les paramètres P, I et D ne sont pas toujours satisfaisantes, ni adaptées au processus à régler.

Un régulateur PID remplit essentiellement trois fonctions :

1. **Action proportionnelle P:** Fournit un signal de commande $u(t)$ en tenant compte de l'évolution du signal de Sortie $y(t)$ par rapport à la consigne $w(t)$ Lorsque P augmente, le temps de montée est plus court mais il y a un dépassement plus important. Le temps d'établissement varie peu et l'erreur statique se trouve améliorée.

2. **Action Intégral I:** Elimine l'erreur statique grâce, Lorsque $I = \frac{1}{Ti}$ augmente, le temps de montée est plus court mais il y a un dépassement plus important. Le temps d'établissement au régime stationnaire s'allonge, mais dans ce cas, on assure une erreur statique nulle. Donc, plus ce paramètre est élevé, plus la réponse du système est ralentie.

3. **Action dérivé D:** Anticipe les variations de la sortie grâce au terme dérivateur. Lorsque D augmente, le temps de montée change peu, mais le dépassement diminue. Le temps d'établissement au régime stationnaire est meilleur. Pas d'influences sur l'erreur

statique. Si ce paramètre est trop élevé dans un premier temps il stabilise le système en le ralentissant trop, mais dans un deuxième temps le régulateur anticipe trop et un système à temps mort élevé devient rapidement instable.

Le régulateur PID classique relie directement le signal de commande $u(t)$ au signal d'écart $e(t)$. Sa description temporelle est la suivante :

$$u(t) = K_p \left(e(t) + \frac{1}{T_i} \int_0^t e(t) dt + T_d \frac{d e(t)}{dt} \right)$$

$$K_i = \frac{K_p}{T_i}; \quad K_d = K_p \times T_d.$$

2.10 Conclusion:

Nous estimons que les généralités, présentées dans ce chapitre, enlève toute ambiguïté concernant les matériels et les méthodes utilisées dans ce travail.

Dans le chapitre suivant, on va présenter les étapes de la réalisation de notre projet.

3.1 Introduction :

Le présent chapitre illustre les étapes suivies et les méthodes pratiquées pour réaliser ce projet. Il consiste à remplacer les six thermorégulateurs qui régulent les températures des résistances dans les cylindres de l'unité qui s'occupent de la soudure d'élastique rassembleur de jambe, avec la feuille de dessus de la couche bébé. Cette opération est supervisée dans une IHM, qui est l'outil qui sert à introduire les consignes voulues, et afficher les températures réelles des résistances, qui sont mesurées par des thermocouples, ainsi que des alarmes, lorsqu'il y a une quelconque défaillance déjà prévue dans le programme.

Tout d'abord, il a fallu faire un diagnostic de l'état présent du processus (figure 3.1) afin de faire le choix du matériel conforme à la modification qu'on doit lui apporter.

Après avoir fait le choix du matériel, l'étape suivante était de structurer notre programme, afin de le traduire en langage adéquat au matériel choisi.

Finalement, un schéma électrique a été fait, afin de simplifier le câblage et faciliter le montage.

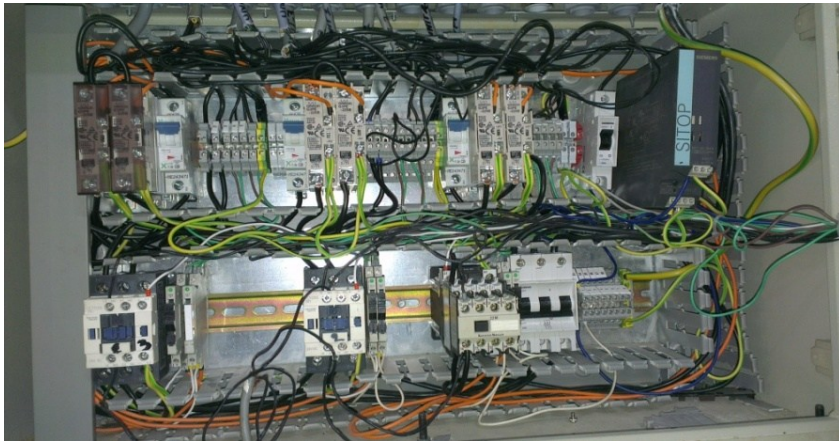


Figure 3.1 : Image de l'état actuel de l'armoire de commande

3.2 Choix des matériels :

3.2.1 Choix de l'automate programmable :

Nous avons opté pour l'utilisation d'une CPU Siemens S7-300 C 2-DP (figure 3.2). Ce choix a été fait, par rapport à sa disponibilité au sein du magasin de l'entreprise, et l'analyse de ses caractéristiques techniques, nous a affirmé que c'est la CPU convenable aux tâches qu'on doit programmer.

-Caractéristiques techniques de la CPU S7-300 C2-DP :

Elle Possède une mémoire intégrée de 64 Ko et une micro carte mémoire de 8Mo maximum, dont la durée de conservation des données est au minimum 10 ans.

-Temps de traitement :

- Opération sur bits : minimum 0.1 μ s.
- Opération sur mots : minimum 0.2 μ s.
- Opérations arithmétiques sur nombre entiers : minimum 2 μ s.
- opération arithmétique sur nombres à virgule flottante : minimum 3 μ s.

-Le nombre maximum de blocs chargeables peut arriver jusqu'à 1024 (FB, FC, DB), avec une taille maximale de 16ko pour chaque bloc.

-Possède 16 entrées numériques (DI), et 16 sorties numériques (DO).

- Peut supporter jusqu'à 4 châssis maximum avec 8 modules par châssis.
- Contient deux interfaces Profibus DP configurables en maître ou esclave, permettant la transmission avec une vitesse pouvant aller jusqu'à 12Mbauds.
- Une interface MPI, l'interface de la CPU avec un PG/OP/PC/TP dont la vitesse de transmission par défaut pour toutes les CPU est de 187,5 kbauds.
- Permet la programmation en langage à contact, liste et logigramme. [8].



Figure 3.2 : Automate S7-300 2 C 2DP

3.2.2 Choix de l'interface homme machine :

Vu que les interfaces qu'on doit programmer ne nécessitent pas une visualisation en couleur, nous avons donc choisi un pupitre Siemens 177B 6'', qui est muni d'un écran d'affichage tactile noir et blanc, et d'un emplacement pour une carte mémoire et d'un commutateur DIP. Cet IHM, illustré dans la figure 3.3 est généralement optimisé pour les projets basé sur WINCC flexible.

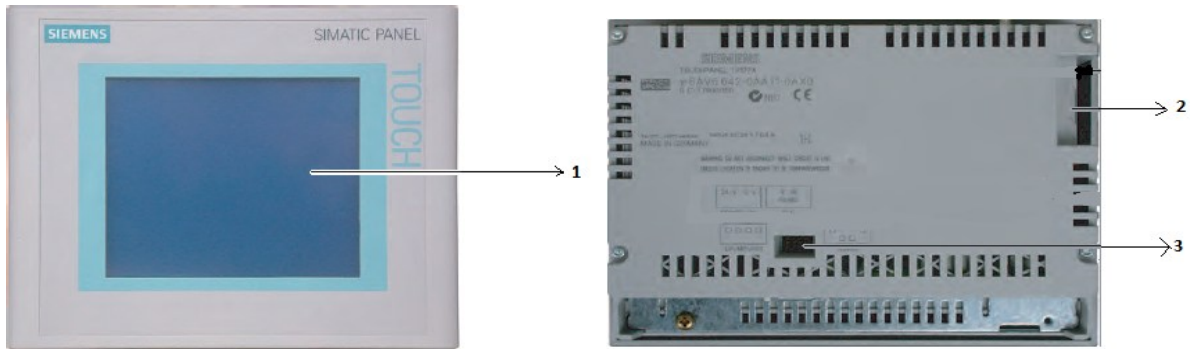


Figure 3.3 Design de l'IHM TP177 B 6"

1 : Ecran d'affichage tactile.

2 : Emplacement carte mémoire.

3 : Commutateur DIP.

3.2.3 Choix du thermocouple et de la résistance :

Notre choix s'est posé sur un thermocouple « RETA » type J par (fils noir et blanc selon CEI 584-3). Ce capteur est intégré dans un cylindre, avec la résistance de chauffage dont la puissance est de 1800 Watt (Figure 3.4), pour que la mesure de la température soit instantanée et plus précise.



Figure 3.4 : Image réelle d'une résistance

3.2.4 Choix du commutateur :

Pour la commutation des courants, on a estimé que les relais statiques (Figure 3.5) sont les plus adaptés à notre travail, vu leur grande vitesse de commutation qui s'adapte à la rapidité que demande le système.



Figure 3.5 : Relais statique 24V DC

3.2.5 Choix du module d'entrée analogique :

Au départ, nous avons choisi un module d'entrées analogique dédié à la lecture des données à partir d'un thermocouple, mais en raison de son indisponibilité, on a dû le remplacer par un transmetteur de température (Figure 3.6), qui a pour mission de convertir la tension en (mV) délivré par le thermocouple en un courant de boucle 4-20mA. Il est ensuite lu par un module analogique dont la référence est SM331-6ES7-7KB01-0AB0 (Figure 3.7) qui s'occupe de la conversion analogique-numérique de l'information d'entrée.



Figure 3.6 : transmetteur de température

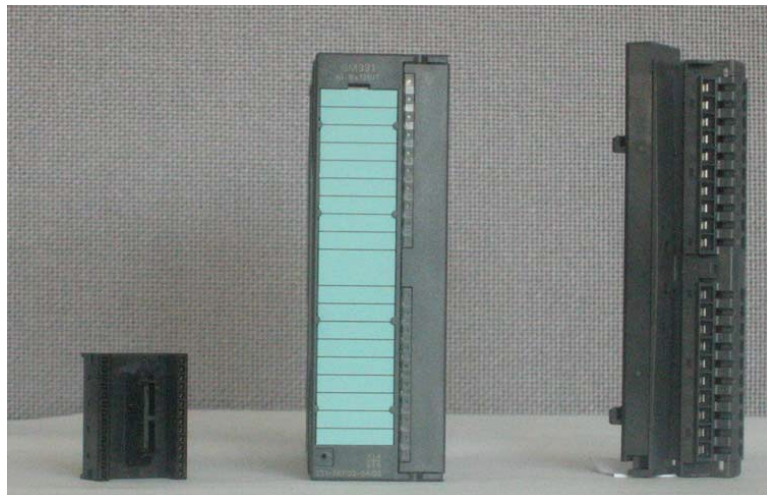


Figure 3.7 : module d'entrées analogique

3.3 Structure du programme :

Après avoir fait le choix du matériel nécessaire pour l'accomplissement de ce travail, l'étape qui suit, était d'établir le programme qui va piloter ce matériel. Ce programme est organisé et structuré de manière à faire une lecture de la donnée à partir des adresses convenables, puis mettre en échelle cette donnée, la comparer avec les consignes, puis la corriger pour enfin donner un ordre adéquat avec cette correction, et afficher l'évolution de cette dernière. Ces étapes sont illustrées dans la figure 3.8 :

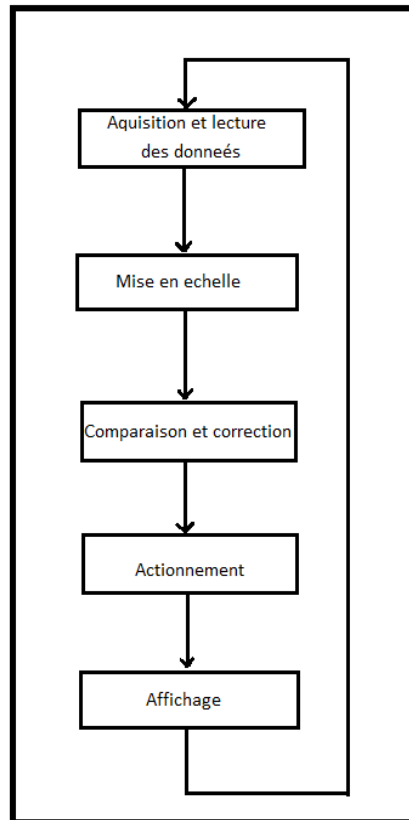


Figure 3.8 : Structure du programme

3.3.1 Configuration matériel :

Après avoir ouvert le logiciel STEP 7, et inséré une nouvelle station simatic S7-300, on a commencé à faire la configuration matériel de notre programme en insérant un profil support qui remplace le châssis. En pratique, un tableau apparaît où on enfiche à partir du catalogue de Step7 notre matériel déjà sélectionné :

- La CPU 313C-2 DP V 2.6, puisque cette API dispose d'une alimentation intégrée, il est enfiché dans le deuxième emplacement du profil support sachant que le premier emplacement est réservé à l'alimentation (PS).
- Le module d'entrées analogiques SM 331 AI 2x12 bits, qui est enfiché dans le châssis au quatrième emplacement.
- Un réseau maître profibus, dont on a besoin pour faire la communication entre la CPU et l'IHM. [8].

Un châssis qui réunit plus de détails sur l'adressage qu'offre la configuration matériels apparaît en dessous comme l'illustre la figure 3.9 :

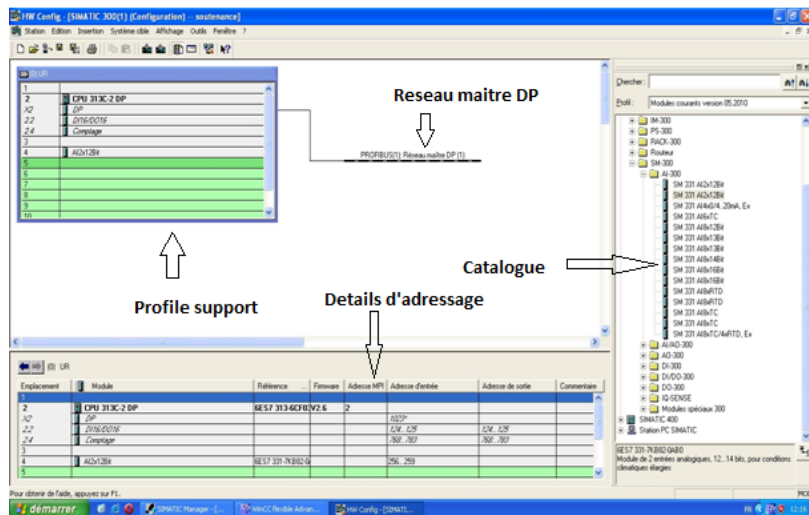


Figure 3.9 : Fenêtre de la configuration matérielle

Enfin on doit enregistrer ce travail et le compiler, et dans la vue d'ensemble de notre projet apparait une icône « Programme S7 ».

3.3.2 Fonction de mise en échelle (scale) :

La grandeur électrique acquise par le module analogique est convertie en une valeur entière. Cette dernière doit être convertie en une valeur réelle exprimée en unité physique qui est dans notre cas la température, ce qui est le rôle de la fonction de mise en échelle. Cette conversion se fait selon l'équation tirée de la courbe de la figure 3.10 :

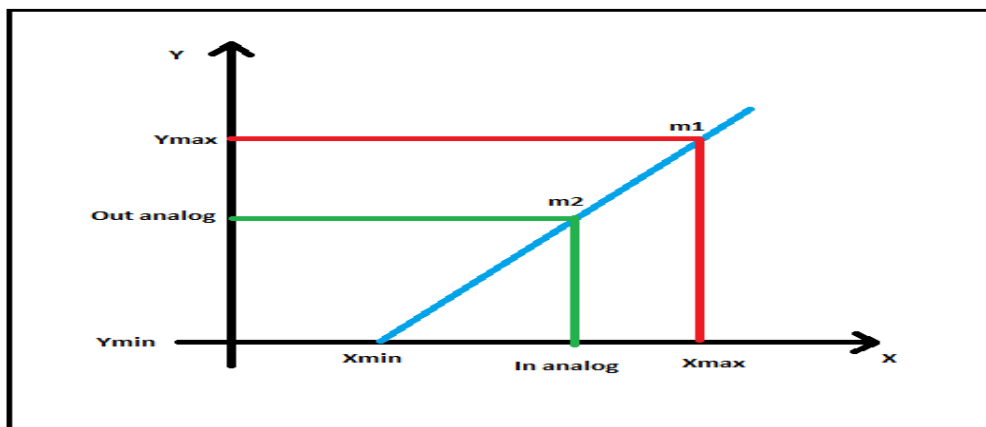


Figure 3.10 : Courbe de mise en échelle

In analog : entrée analogique(grandeur électrique)

4 à 20 mA → (eq : 6400 à 32000)

Xmax : 32000

Xmin : 6400

Out analog= valeur mise en échelle (grandeur réel en unité physique)

Ymax=maximum (ej : 100%)

Ymin=minimum (ej : 0%)

Si on met m1=m2 alors :

$$\frac{Y_{max}-Y_{min}}{X_{max}-X_{min}} = \frac{\text{Out analog}-Y_{min}}{\text{In analog}-X_{min}} \Rightarrow \text{Out analog} = \frac{Y_{max}-Y_{min}}{X_{max}-X_{min}} \times (\text{In analog} - X_{min}) + Y_{min}$$

Le logiciel step 7 est doté d'une fonction nommée « FC 105 scale » prête à l'utilisation. On doit insérer dans cette fonction des entrées pour avoir en sortie la grandeur réelle en unité physique selon l'équation précédente.

Pour programmer cette fonction, on a créé une fonction (FC1) dans l'espace « programme S7 », dont on a affecté le nom symbolique « mise en échelle » et un bloc de données DB1 ou on va mettre les variables à utiliser, comme le montre la figure 3.11 :

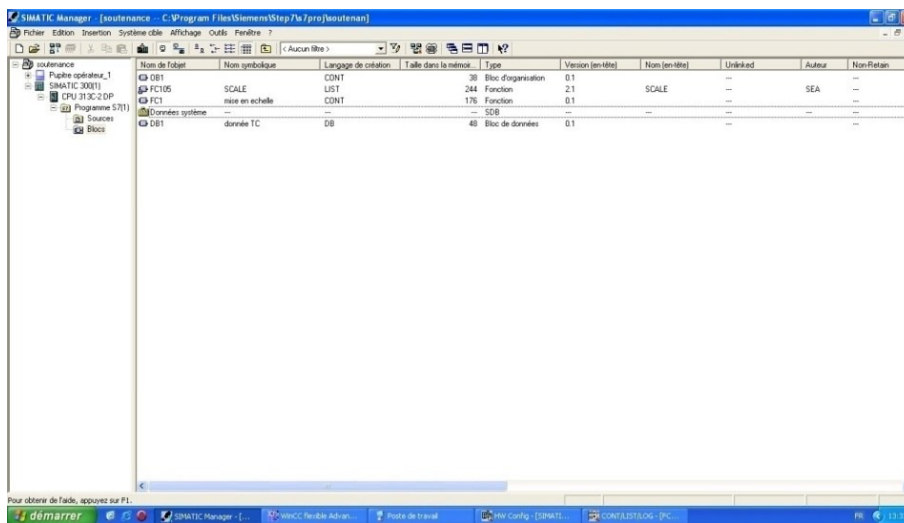


Figure 3.11 : vue des blocs et fonction créée pour la mise en échelle

Dans cette fonction on a créé un nouveau réseau pour faire l'appel de la fonction « FC 105 scale » qui se trouve dans la liste « éléments du programme » sous l'onglet « bibliothèque » comme le montre la figure 3.6, dont les entrées sorties sont :

- **EN** : L'entrée de validation qui active le cadre de fonction si l'état de signal est a "1" .
- **ENO** :La sortie de validation a l'état de signal "1" lorsque la fonction a été exécutée sans erreur.
- **IN** : Valeur d'entrée à convertir selon l'échelle en valeur réelle exprimée en unités physiques.
- **HI_LIM** : Limite supérieure en unités physiques.
- **LO_LIM** : Limite inférieure en unités physiques.
- **BIPOLAR** : L'état de signal "1" signifie que la valeur d'entrée est bipolaire et l'état de signal "0" qu'elle est unipolaire.
- **OUT** : Résultat de la conversion d'échelle.
- **RET_VAL** : Donne la valeur W#16#0000 en retour, lorsque l'opération a été effectuée sans erreur. Pour toute valeur en retour autre que W#16#0000, il faut afficher une alarme système informations d'erreurs. [5].

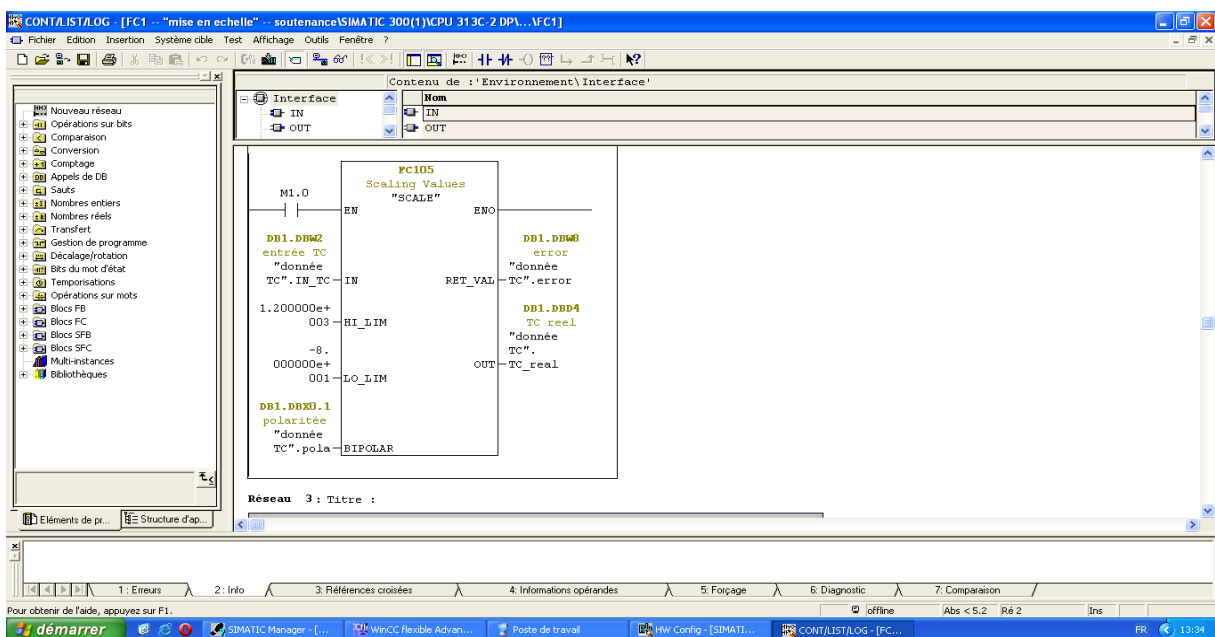


Figure 3.12 : Fonction FC105 Scale pour la mise en échelle

Une fois cette FC1 enregistrée, elle apparaîtra avec la « FC 105 scale » dans la fenêtre BLOCS de l'espace « programme S7 ».

3.3.3 Fonction du régulateur PID :

Afin d'obtenir le degré de chaleur souhaité par le système, une régulation doit se faire, dont le rôle est de comparer la consigne donnée, avec la valeur réelle mesurée par le thermocouple, et corriger l'erreur par un correcteur. Step7 offre cette possibilité puisqu'il propose des fonctions toutes faites, qui réalisent ce genre de tâches. Pour notre travail on a choisi FB 58 'TCONT_CP'.

Le FB 58 "TCONT_CP" sert à la régulation de processus thermiques au moyen d'une commande continue ou à impulsion. En paramétrant ce bloc, on peut activer ou désactiver des fonctions partielles du régulateur PID afin de l'adapter au système à régler. Il suffit, pour cela, d'utiliser l'outil de paramétrage. On peut appeler ce dernier, dans le projet, en effectuant un double-clic sur le DB d'instance qui est dans notre cas DB10.

Une plage de régulation, et une fonction de réduction de l'action P, en cas d'échelon de consigne, permettent d'optimiser la régulation. Alors, grâce à son module d'optimisation, le bloc est capable de régler lui-même les paramètres PI/PID.

Les valeurs, dans les blocs de régulateur, ne sont correctement calculées que si le bloc est appelé régulièrement. Les blocs de régulateur doivent donc, être appelés, dans un OB d'alarme cyclique (OB30-38).

Donc, pour introduire ce bloc fonctionnel au sein de notre programme, on a du créer un OB35, ou nous avons pu faire l'appel du FB58, comme le montre la figure 3.13 :

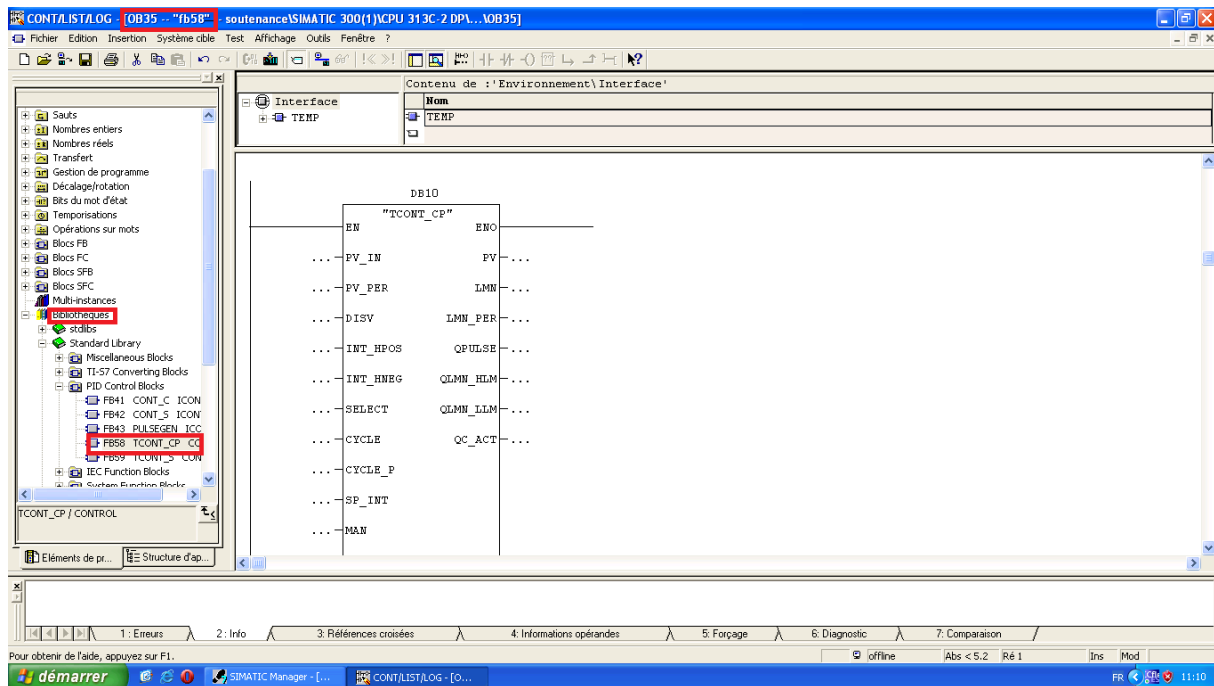


Figure 3.13 : Le bloc fonctionnel de régulation FB58 «TCONT-CP »

Tandis que le paramétrage de ce bloc fonctionnel, s'est fait dans une fonction FC3 qu'on a nommé « Fonction d'activation du PID » ou on a mis au point les paramètres suivants selon les besoin de notre projet :

- **PVPER_ON :**

TRUE La lecture de la mesure s'effectue via la périphérie analogique (PEW xxx) à l'entrée PV_PER.

FALSE La lecture de la mesure s'effectue en format à virgule flottante à l'entrée PV_IN.

- **PV_IN :** L'entrée "entrée mesure" permet de paramétrer une valeur de mise en service ou de commutation une mesure externe en format à virgule flottante.
- **MAN_ON :** La mise à 1 de l'entrée "activation du mode manuel" interrompt la boucle de régulation. La valeur manuelle MAN est spécifiée comme valeur régnante.
- **PULSE_ON :** La mise à 1 du formateur d'impulsions, PULSE_ON = TRUE permet d'activer le formateur d'impulsions.
- **COM_RST :** Le bloc dispose d'une routine d'initialisation, qui est exécutée, lorsque l'entrée COM_RST est activée.
- **SP_INT :** L'entrée "consigne interne" permet de spécifier une consigne.

- **QPULSE** : La Valeur du signal de commande est fournie à la sortie QPULSE en modulation de largeur d'impulsions. Cette valeur sert à alimenter notre résistance de chauffage. [5].

Ces paramètres, et beaucoup d'autres, sont définis dans un bloc de données d'instances qui est dans ce projet DB10. On peut modifier leur valeur initiale et introduire le facteur de gain et les paramètres P, I, D du régulateur de façon manuelle, comme le montre la figure 3.14 :

Adresse	Décl.	Nom	Type	Valeur initiale	Valeur en cours	Commentaire
1	0.0	in	PV_IN	REAL	0.000000e+000	process variable in
2	4.0	in	PV_PER	INT	0	process variable peripherie
3	6.0	in	DISV	REAL	0.000000e+000	disturbance variable
4	10.0	in	INT_HPOS	BOOL	FALSE	integral action hold in positive direction
5	10.1	in	INT_HNEG	BOOL	FALSE	integral action hold in negative direction
6	12.0	in	SELECT	INT	0	selection of call PID and pulse generator
7	14.0	out	PV	REAL	0.000000e+000	process variable
8	18.0	out	LMN	REAL	0.000000e+000	manipulated variable
9	22.0	out	LMN_PER	INT	0	manipulated variable peripherie
10	24.0	out	QPULSE	BOOL	FALSE	output pulse signal
11	24.1	out	QLMN_HLM	BOOL	FALSE	high limit of manipulated variable reached
12	24.2	out	QLMN_LLM	BOOL	FALSE	low limit of manipulated variable reached
13	24.3	out	QC_ACT	BOOL	TRUE	next cycle, the continuous controller is working
14	26.0	in_out	CYCLE	REAL	1.000000e-001	sample time of continuous controller [s]
15	30.0	in_out	CYCLE_P	REAL	2.000000e-002	sample time of pulse generator [s]
16	34.0	in_out	SP_INT	REAL	0.000000e+000	internal setpoint
17	38.0	in_out	MAN	REAL	0.000000e+000	manual value
18	42.0	in_out	COM_RST	BOOL	FALSE	complete restart
19	42.1	in_out	MAN_ON	BOOL	TRUE	manual operation on
20	44.0	stat	DEADB_W	REAL	0.000000e+000	dead band width
21	48.0	stat	I_TLVAL	REAL	0.000000e+000	initialization value of the integral action
22	52.0	stat	LMN_HLM	REAL	1.000000e+000	manipulated variable high limit
23	56.0	stat	LMN_LLM	REAL	0.000000e+000	manipulated variable low limit
24	60.0	stat	PV_FAC	REAL	1.000000e+000	process variable factor
25	64.0	stat	PV_OFFS	REAL	0.000000e+000	process variable offset
26	68.0	stat	LMN_FAC	REAL	1.000000e+000	manipulated variable factor
27	72.0	stat	LMN_OFFS	REAL	0.000000e+000	manipulated variable offset
28	76.0	stat	PER_TM	REAL	1.000000e+000	period time [s]
29	80.0	stat	P_B_TM	REAL	0.000000e+000	minimum pulse/break time [s]
30	84.0	stat	TUN_DLMN	REAL	2.000000e+001	delta manipulated variable for process excitation

Figure 3.14 : Bloc de données DB10

3.3.4 Communication Profibus API-IHM :

Afin d'introduire les consignes et surveiller le bon déroulement de la tâche de chauffage, nous avons utilisé une interface homme machine, qui doit communiquer avec l'API, à travers le protocole de communication Profibus DP, afin de lui transmettre ces consignes et d'afficher les alarmes qu'il détecte après le traitement du programme.

Puisque l'API est programmé avec Step7 et l'IHM avec Wincc flexible , on a du établir la communication entre les deux projets de la manière suivante :

Tout d'abord, on a créé un projet dans Wincc flexible pour y intégrer notre projet établi dans Step7, puis on a choisi le protocole à utiliser, qui est dans notre cas, MPI/DP comme illustré dans les figures 3.15 et 3.16 ci-dessous :

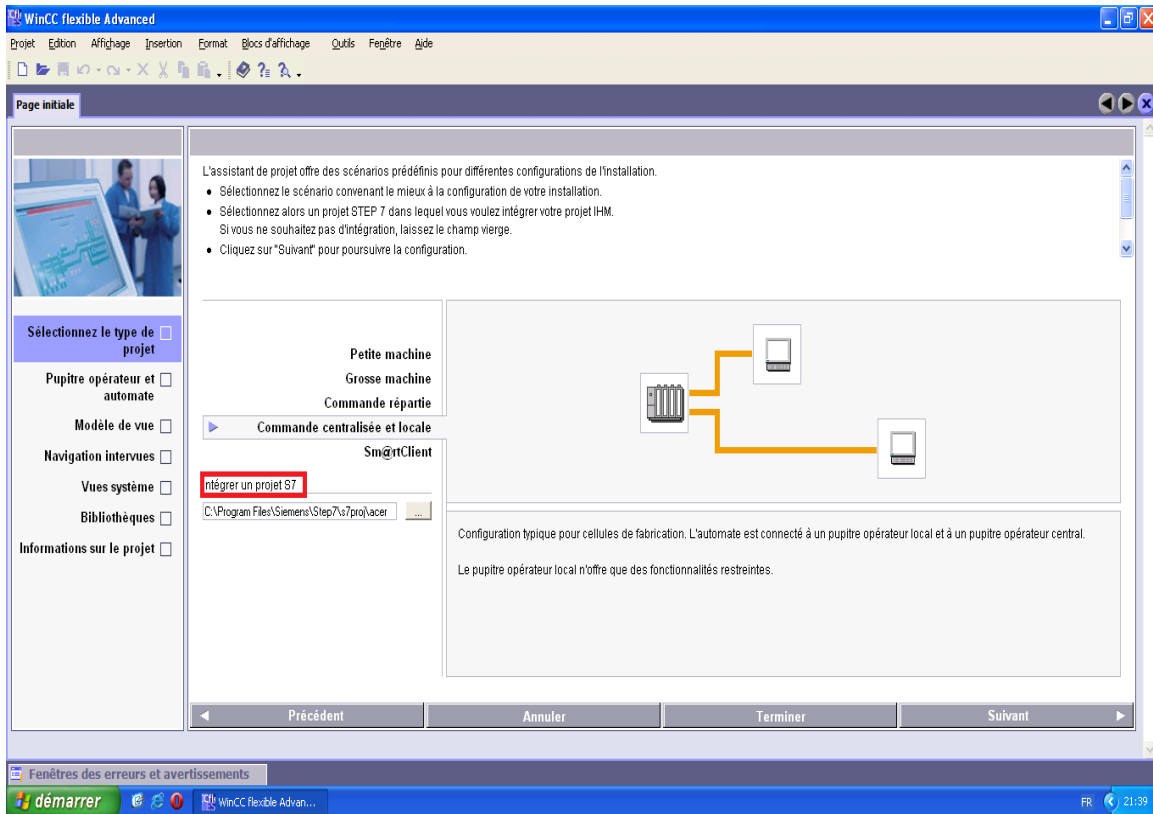


Figure 3.15 : Intégration du projet Step7 dans le projet wincc

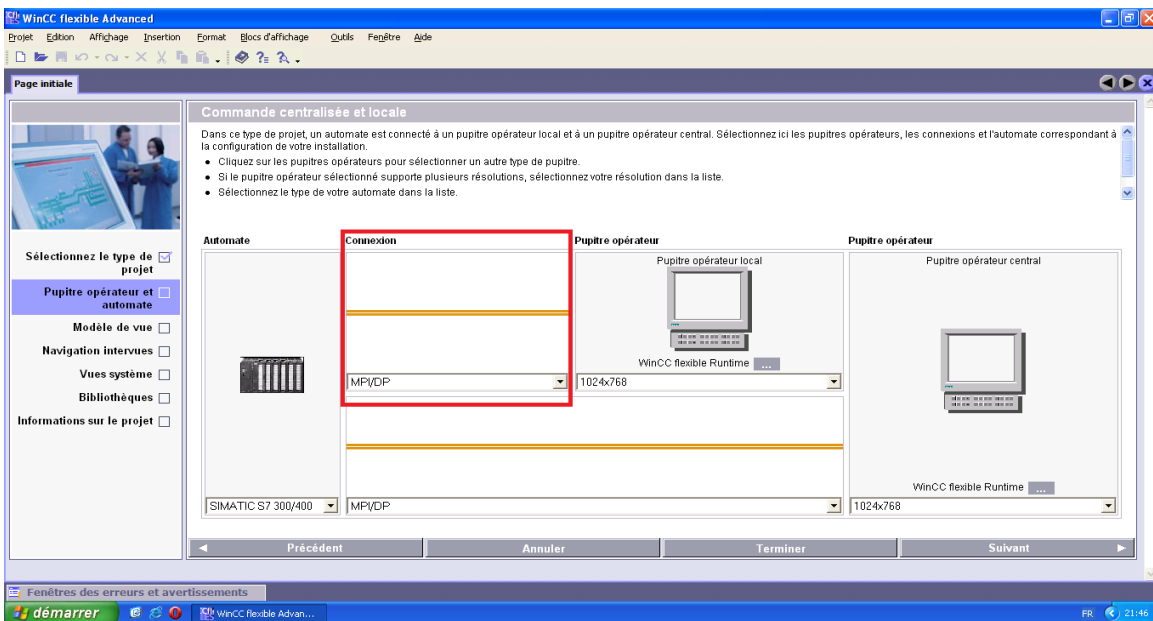


Figure 3.16 : Choix du protocole de communication dans le projet wincc flexible

Cependant, on a du aussi connecter notre API et IHM aux réseaux déjà crée au sein du projet Step 7 dans la configuration matérielle, comme l'illustre la figure 3.17 :

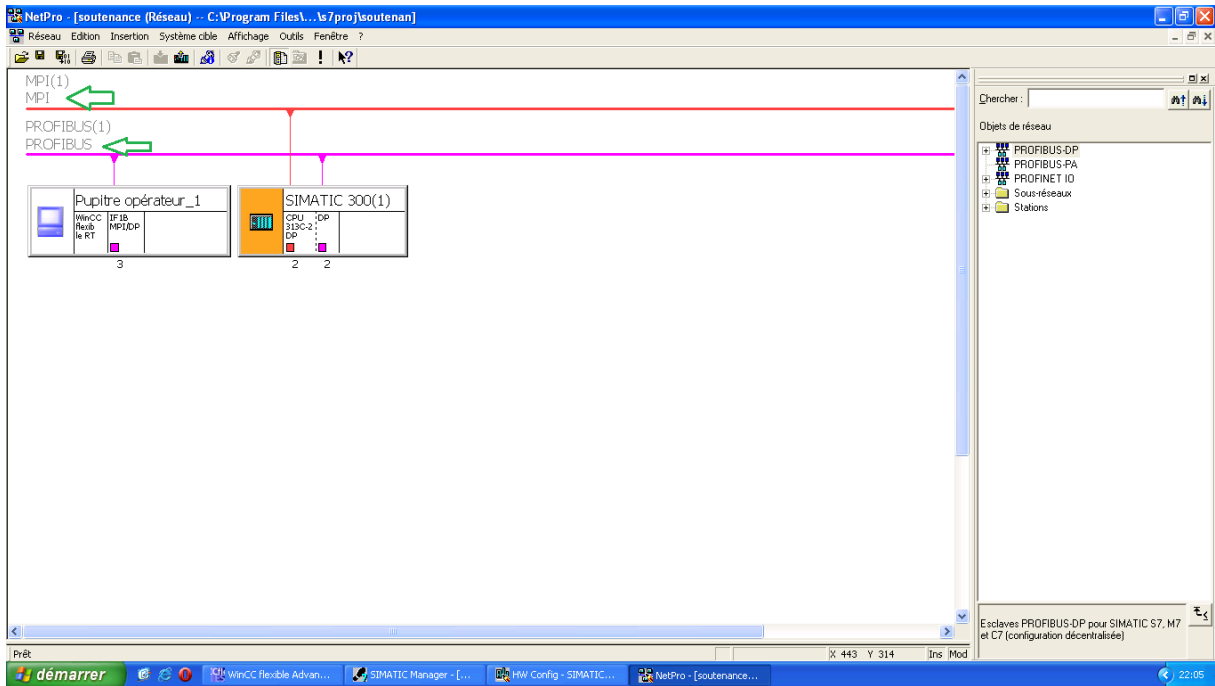


Figure 3.17 : Connexion de l’IHM et de l’API aux réseaux maître DP et MPI

On peut vérifier que la communication est bien faite, en cliquant sur l’onglet « Liaison », lorsqu’on établit notre projet wincc flexible, comme le montre la figure 3.18 :

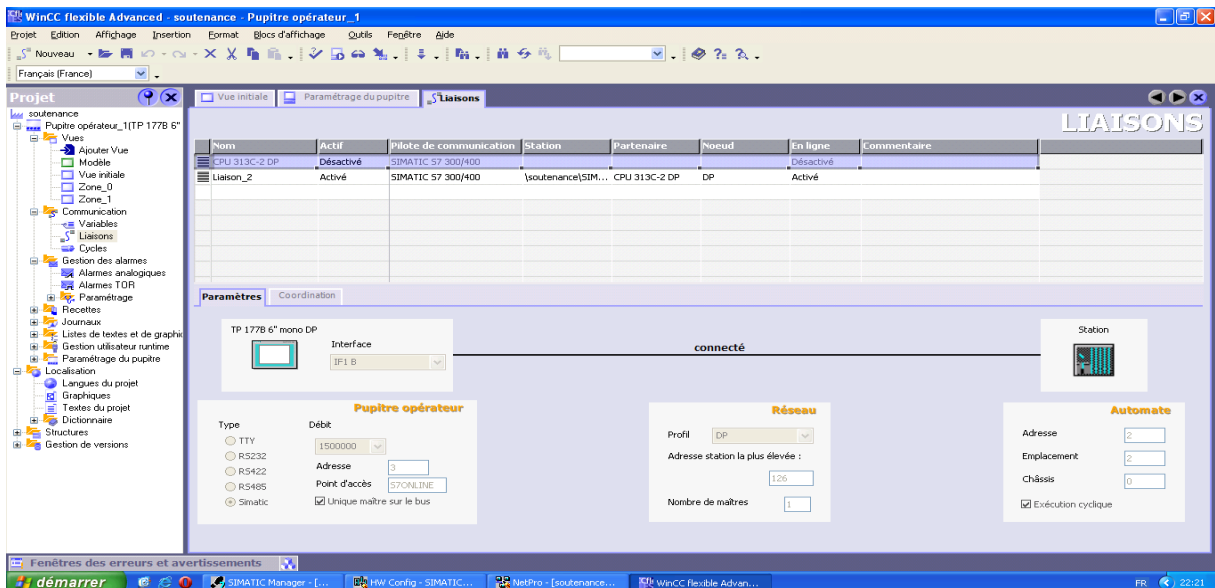


Figure 3.18 : Vérification de la connexion dans Wincc flexible

3.3.5 Programmation de l'interface Homme-Machine :

Dans le but d'introduire les consignes, et de surveiller le comportement du processus et les défauts qu'il peut présenter, on a du programmer l'IHM de manière à faciliter la tâche de surveillance et de maintenance pour le personnel.

Cette programmation a été faite avec le logiciel Wicc Flexible, ou on a structuré nos vues de la manière suivante :

- **Une vue initiale :** Elle sert de page d'accueil ou l'opérateur peut mettre en marche ou en arrêt toutes les résistances chauffantes en même temps, avec le bouton -ON/OFF- qui change de couleur lors d'une mise en marche ou une mise en arrêt. Cette vue offre aussi, la possibilité du contrôle de l'allumage et de l'extinction de chaque résistance indépendamment des autres par des boutons -R1 ,..., R6- .Des boutons sous forme de flèche permettent de naviguer entre les vues, ainsi qu'un bouton spécifique pour la vue, qui contient les alarmes et un champ de date/heure y sont programmés dans cette vue, comme l'illustre la figure 3.19 :

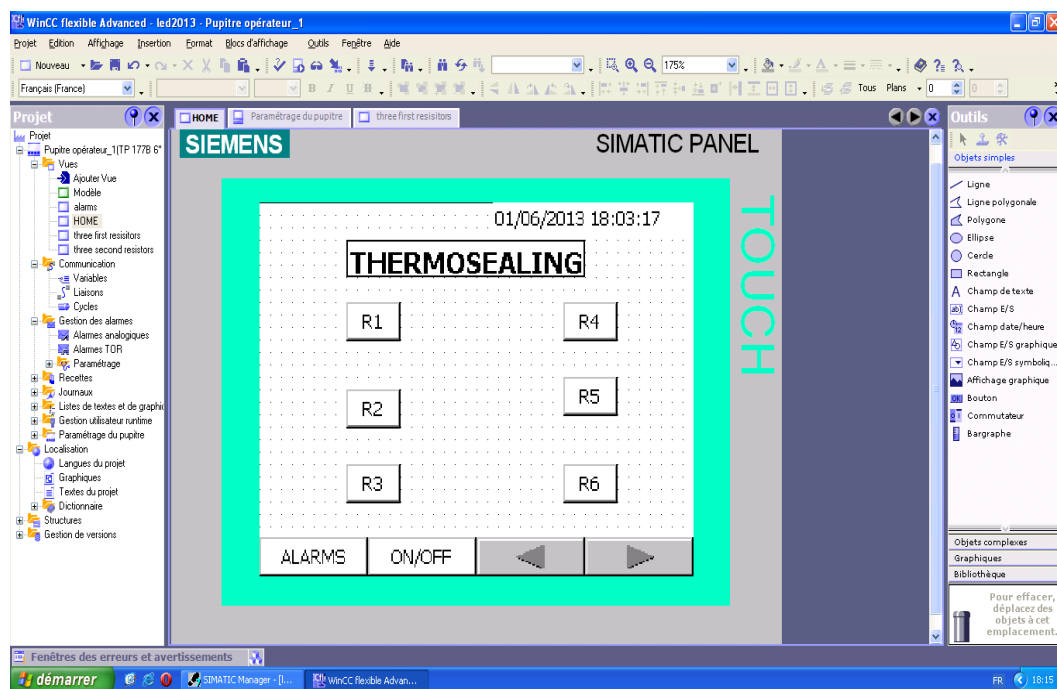


Figure 3.19 : vue initial du programme Wincc flexible

- **Deux vues d'informations :** qui contiennent des champs d'entrées/sorties qui permettent d'introduire la consigne pour chaque résistance, ainsi que des champs

qui affichent la valeur de la température actuelle de chaque résistance, et des champs pour l’affichage de l’écart entre la valeur de consigne et la valeur actuelle de la température de chaque résistance.

Des boutons de navigation et un champ date/heure y sont aussi programmés dans ces deux vues.

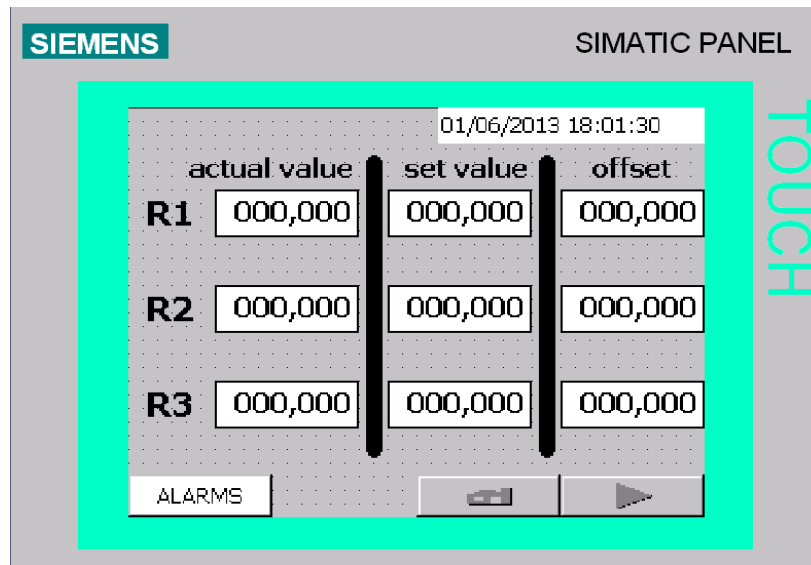


Figure 3.20 : Vue d’information pour les trois premières résistances

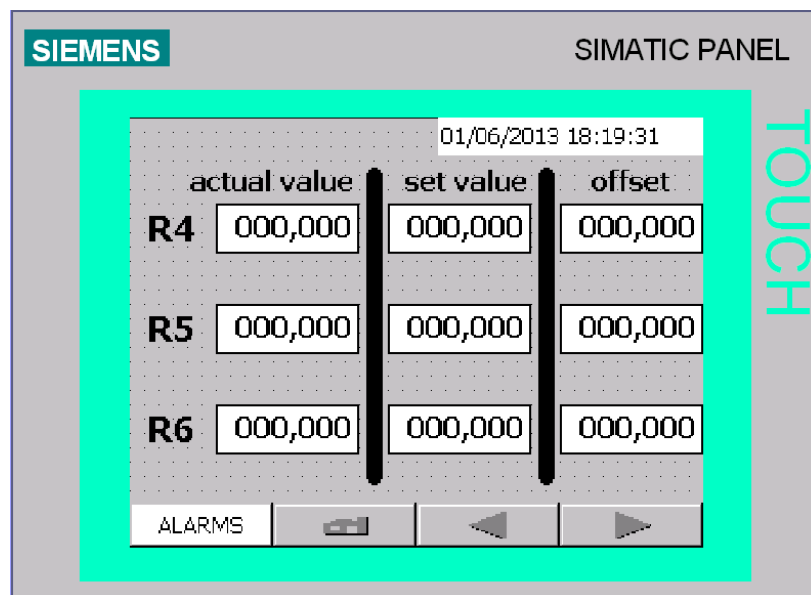


Figure 3.21 : Vue d’informations pour les trois secondes résistances

- **Une vue d'alarmes** : qui affiche toutes les alarmes que nous avons prévues dans une fonction FC3 établie dans le programme step7, un bouton « acknowledge » permet à l'opérateur de mentionner qu'il a reconnu l'alarme.

L'affichage de ces alarmes, est organisé dans un tableau, ou il ya le numéro et le moment précis (date/heure) de l'alarme, comme le montre la figure 3.22 ci-dessous :

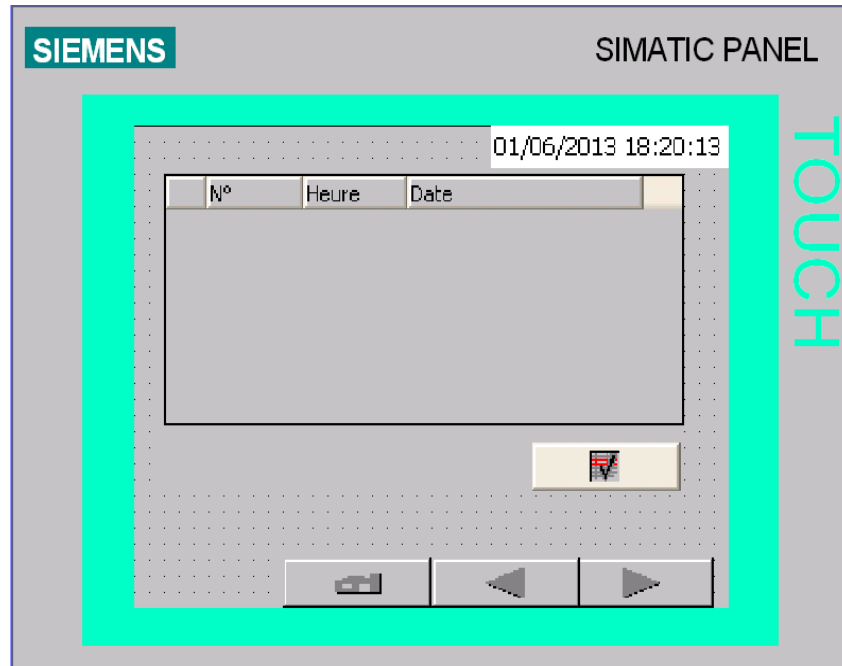


Figure 3.22 : Vue d'alarmes

3.4 Réalisation :

La réalisation et le câblage de ce projet, ont été effectués au sein de l'atelier de l'entreprise M S electronics , où il a fallu dessiner un schéma de câblage, et tracer la réponse du système en boucle ouverte, sans régulateur, qui permet de déterminer les paramètres du régulateur, qu'on a ensuite ajusté en faisant des essais expérimentaux en se basant sur l'approche successive de chaque valeur du régulateur.

3.4.1 Schéma de câblage :

Pour faire le câblage de notre projet on a suivi le schéma de la figure 3.23 :

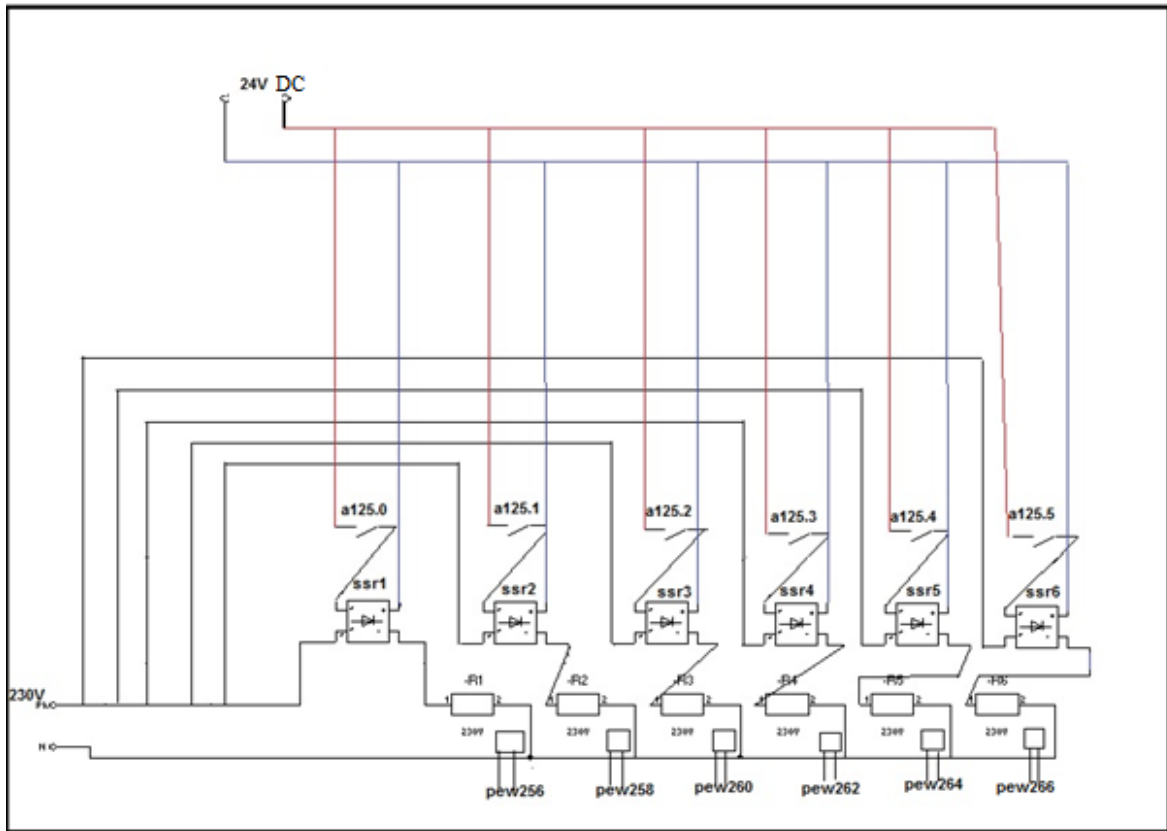


Figure 3.23 : Schéma de câblage

A partir de la prise du secteur 220v, on a branché notre alimentation stabilisée, qui sert d'alimentation pour l'API et du module digital aux bornes 21 et 30, ainsi que le module analogique, dont le schéma de câblage apparait dans la figure 3.24.[12]. :

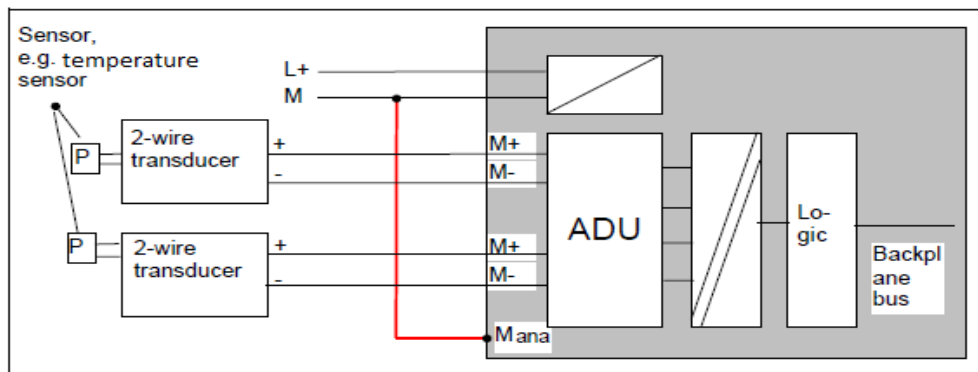


Figure 3.24 : Schéma de câblage d'un convertisseur a 2 fils au module analogique

Pour la lecture de la donnée du thermocouple, on a branché les deux fils de ce dernier dans l'entrée thermocouple du transmetteur, dont on a branché les sorties aux bornes 2 et 3 du module analogique qu'on a configuré à la lecture en courant 4-20mA.[12]

Le relais statique est excité avec 24V, à partir de la borne 22 du module numérique pour qu'il commute en 220 V qui sert d'alimentation pour la résistance.

Pour la visualisation des températures, une IHM est câblée avec l'API par le biais d'un câble profibus, tandis qu'on peut remplacer ce pupitre par une station PG/PC qui peuvent être reliés à l'API par un câble MPI.

3.4.2 Détermination des paramètres du correcteur PID :

Pour arriver aux objectifs décrits dans le cahier des charges de la régulation d'un procédé, il faut analyser le comportement du procédé, c'est à dire connaître sa fonction de transfert régnante .En effet le réglage du correcteur à mettre en œuvre, dépend essentiellement de la nature de cette fonction de transfert.

L'identification consiste à déterminer la fonction de transfert d'un système, peut être effectuée par une mise en équation du système ce qui peut être délicat voire impossible lorsque le procédé est complexe, ce qui mène à une identification expérimentale comme le cas de ce présent projet. [10].

L'identification des paramètres de notre correcteur repose sur l'analyse de réponse temporelle, observée directement sur le système soumis à un signal d'entrée déterminé.

3.4.3 Méthode de détermination des paramètres PID «Ziegler-Nichols»:

Cette méthode permet d'ajuster rapidement les paramètres des régulateurs P, PI et PID en enregistrant la réponse indicielle du système en boucle ouverte, seul (c'est-à-dire sans le régulateur) puis calculer la pente α , le retard apparent τ , et ΔT (°C) variation de la sortie et on calcule les coefficients du régulateur choisi à l'aide du tableau 1.[10] :

$$\text{Prenant : } R = \frac{\alpha}{\Delta T}$$

Type de régulateur	Gain Kp	Ti	Td
Proportionnel P	$\frac{0.9}{R * \tau}$	-	-
PI	$\frac{1.27}{R * \tau}$	3.3* τ	-
PID	$\frac{1.27}{R * \tau}$	2* τ	0.5* τ

Tableau 01 : Tableau pour le calcul des paramètres PID. Réglage Ziegler-Nichols en boucle ouverte.

Pour appliquer cette méthode sur notre système, on a dû désactiver la fonction régulateur PID FB58 'TCONT-CP', puis on a excité le système avec une entrée échelon unité pendant une durée, de 1min30sec.

D'habitude, l'enregistrement de la réponse indicielle se fait par un enregistreur, mais vu l'indisponibilité de cet instrument, on a tracé la courbe à partir de deux mesures du thermocouple, dont la moyenne à chaque seconde a donné les valeurs du tableau 2.

Après avoir calculé les paramètres PID avec la méthode *Ziegler-Nichols*, on a dû boucler notre système et intégrer la fonction régulateur dans le programme, puis on a fait plusieurs essais par la méthode d'approche successive, dans le but d'ajuster les paramètres, pour que notre système soit conforme à la demande du cahier de charge.

Temps (s)	Température (C°)	Temps (s)	Température (C°)
0	28.7083	45	121.155
1	28.8498	46	122.335
2	29.0712	47	124.77
3	29.3663	48	125.00
4	30.1042	49	127.426
5	31.06434	50	131.195
6	31.9488	51	133.403
7	32.908	52	135.985
8	34.0148	53	140.787
9	35.1215	54	142.309
10	36.8186	55	143.806
11	38.4416	56	147.065
12	39.9175	57	150.299
13	41.3932	58	153.251
14	43.3116	59	153.700
15	46.3368	60	156.75
16	47.8125	61	160.70
17	49.8785	62	165.72
18	51.8781	63	170.59
19	54.3056	64	172.14
20	56.5191	65	174.208
21	59.2491	66	177.674
22	61.0938	67	180.33
23	63.6024	68	183.945
24	66.48	69	187.635
25	68.546	70	191.619
26	71.646	71	197.817
27	73.7109	72	200.178
28	76.441	73	204.826
29	79.4661	74	206.155
30	81.6059	75	209.696
31	83.957	76	211.91
32	86.4757	77	214.271
33	88.6892	78	217.444
34	91.0503	79	220.064
35	94.2231	80	222.387
36	97.5434	81	225.704
37	99.9045	82	228.438
38	102.339	83	233.381
39	104.898	84	236.775
40	107.431	85	240.76
41	110.234	86	242.899
42	113.186	87	247.769
43	115.621	88	253.008
44	116.801	89	255.000

Tableau 02 : Valeurs d'évolution de la température du système en boucle ouverte

3.5 Conclusion :

Dans cette partie, on a expliqué les étapes de notre travail, qui a été subdivisé en deux parties, programmation et réalisation.

La bonne compréhension des exigences du cahier de charge mène à une meilleure structuration de la partie programmation, qui à son rôle facilite la partie réalisation qui consiste à relier les entrées/sorties attaqués par les instructions du programme, aux capteurs et actionneurs par un câblage.

Le réglage des paramètres d'un quelconque régulateur nécessite un calcul analytique, mais quand ce dernier est trop complexe, il est remplacé par des méthodes expérimentales, qui peuvent tout de même donner des résultats non satisfaisant, mais servant de base pour un ajustement des paramètres.

Dans le chapitre suivant on va développer et interpréter les résultats obtenus dans les essais d'ajustement des paramètres ainsi que dans la réalisation finale.

Chapitre 4 Résultats obtenus et interprétation

4.1 Introduction :

Dans ce chapitre, on va développer et interpréter les résultats obtenus après la programmation et la réalisation. Il comportera les résultats de simulation des blocs du programme, le calcul analytique des paramètres PID ainsi que leur ajustement successifs, et les résultats d'application de ces paramètres sur le système.

4.2 Simulation du bloc FC 105 'scall' :

Ce bloc dispose d'une sortie de validation dont le signal prend l'état "1" lorsque la fonction a été exécutée sans erreurs (figure 4.1) :

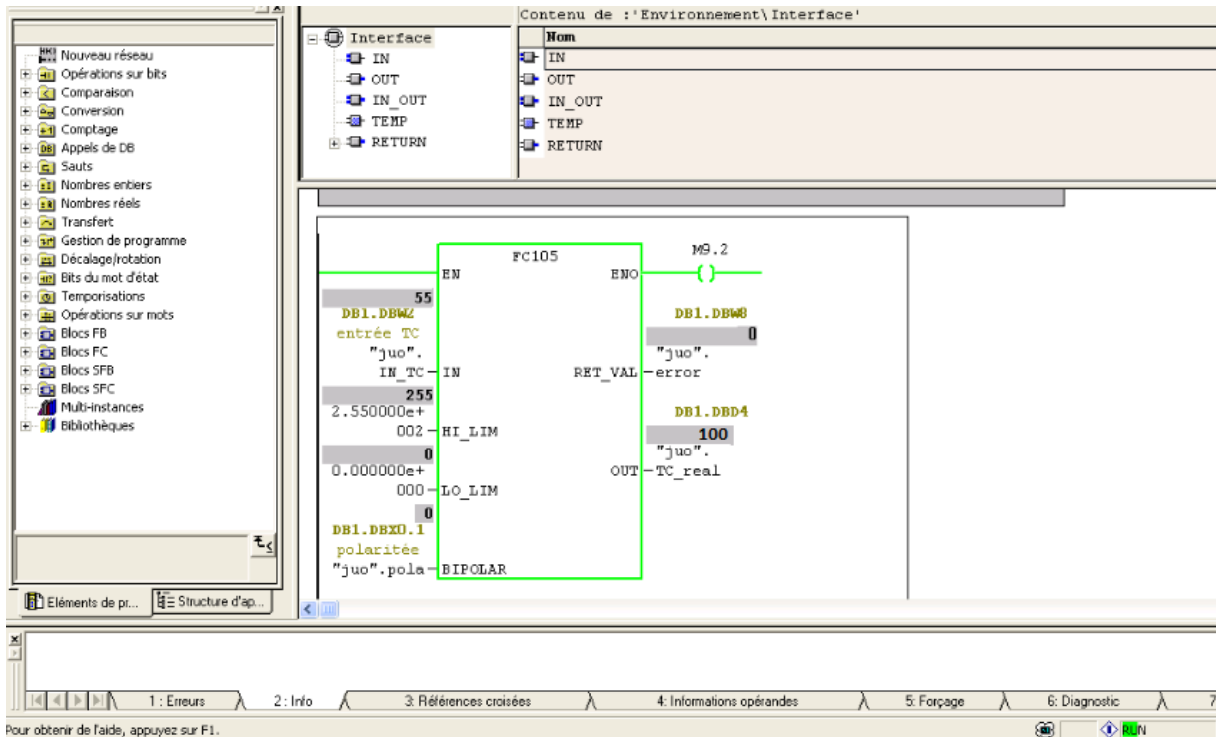


Figure 4.1 : vérification de la fonction FC 105 'Scall'

Comme le montre la figure 4.1 la fonction mise en échelle qu'on a utilisé dans notre programme s'est déroulé avec succès vu que la sortie ENO est mis a « 1 », et ainsi on a vérifié avec un pistolet infrarouge de détection de température (figure 4.2) pour confirmer que la température réelle de la résistance est la même affichée dans notre fonction.



Figure 4.2 : Vérification de la température de la résistance

4.3 Simulation du bloc FB 58 'TCONT' :

Comme la fonction précédente, celle de régulation contient aussi une sortie qui permet de vérifier qu'elle a été exécutée sans erreurs, (figure 4.3) montre la simulation de cette fonction :

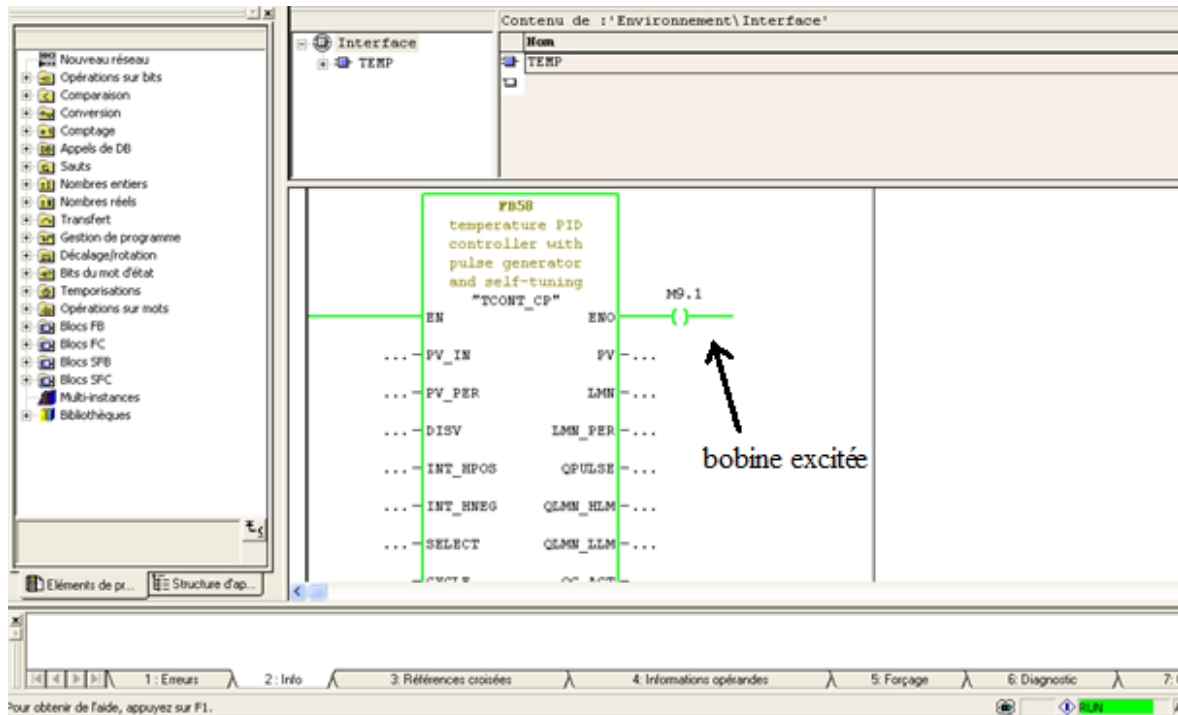


Figure 4.3 : Vérification de la fonction FB58 'TCONT'

Comme on peut le constater, la fonction de régulation se déroule sans erreurs, ce qui veut dire qu'elle accomplit avec succès la comparaison et la correction de la donnée acquise.

4.4 Runtime du projet Wincc flexible :

Afin de vérifier que les consignes introduites dans les vues du projet wincc flexible qui sert à la programmation de l'IHM sont prises en compte par le programme Step 7, on met ce projet en mode Runtime et on introduit différentes valeurs de consignes, puis on vérifie si ces valeurs sont mises à jour dans le DB adéquat (figure 4.4 et figure 4.5) :

SP_INT	REAL	0.000000e+000	140.0	0.000000e+000	internal setpoint
MAN	REAL	0.000000e+000	100.0	Le paramètre 'SP_INT' est en cours de visualisation !	
COM_RST	BOOL	FALSE	FALSE	FALSE	complete restart
MAN_ON	BOOL	TRUE	FALSE		
DEADB_W	REAL	0.000000e+000	0.0		
I_ITLVAL	REAL	0.000000e+000	0.0		
LMN_HLM	REAL	1.000000e+002	100.0		
LMN_LLM	REAL	0.000000e+000	0.0		
PV_FAC	REAL	1.000000e+000	1.0		
PV_OFFS	REAL	0.000000e+000	0.0		
LMN_FAC	REAL	1.000000e+000	1.0		
LMN_OFFS	REAL	0.000000e+000	0.0		
PER_TM	REAL	1.000000e+000	1.0		
P_B_TM	REAL	0.000000e+000	0.0		
TUN_DLMN	REAL	2.000000e+001	20.0		
PER_MODE	INT	0	0		
VPVPER_ON	BOOL	FALSE	FALSE		

Figure 4.4 : Vérification que la consigne 140 °C est lise dans le DB 10

34.0	in_out	SP_INT	REAL	0.000000e+000	95.0	0.000000e+000	internal setpoint
38.0	in_out	MAN	REAL	0.000000e+000	100.0	Le paramètre 'SP_INT' est en cours de visualisation !	
42.0	in_out	COM_RST	BOOL	FALSE	FALSE	FALSE	complete restart
42.1	in_out	MAN_ON	BOOL	TRUE	FALSE		
44.0	stat	DEADB_W	REAL	0.000000e+000	0.0		
48.0	stat	I_ITLVAL	REAL	0.000000e+000	0.0		
52.0	stat	LMN_HLM	REAL	1.000000e+002	100.0		
56.0	stat	LMN_LLM	REAL	0.000000e+000	0.0		
60.0	stat	PV_FAC	REAL	1.000000e+000	1.0		
64.0	stat	PV_OFFS	REAL	0.000000e+000	0.0		
68.0	stat	LMN_FAC	REAL	1.000000e+000	1.0		
72.0	stat	LMN_OFFS	REAL	0.000000e+000	0.0		
76.0	stat	PER_TM	REAL	1.000000e+000	1.0		
80.0	stat	P_B_TM	REAL	0.000000e+000	0.0		
84.0	stat	TUN_DLMN	REAL	2.000000e+001	20.0		
88.0	stat	PER_MODE	INT	0	0		

Figure 4.5 : Vérification que la consigne 95 °C est lise dans le DB 10

Sous mode Runtime, on peut aussi provoquer des défaillances et vérifier si les alarmes appropriées s'affichent dans la vue d'alarmes du projet Wincc (figure 4.6) :

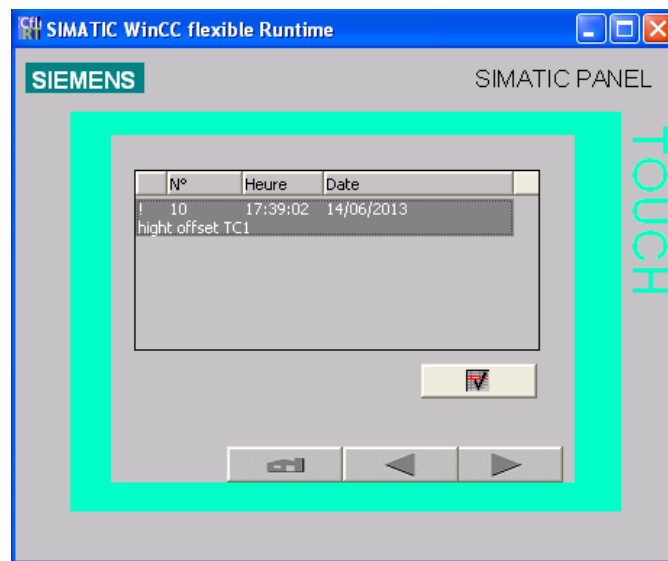


Figure 4.6 : Figure d'alarme affichant une alarme de défaillance

On remarque que les alarmes s'affichent avec la date et l'heure, de la mise en défaut du matériel.

4.5 Calcul des paramètres PID avec la méthode de ziegler-Nichols :

4.5.1 Réponse indicielle du système en boucle ouverte :

En se basant sur les mesures présentées dans le tableau 01 du chapitre précédent, on a établi un programme sous matlab qui permet de tracer l'évolution de la température du système en boucle ouverte par rapport au temps, on a obtenu la courbe de la figure 4.7 :

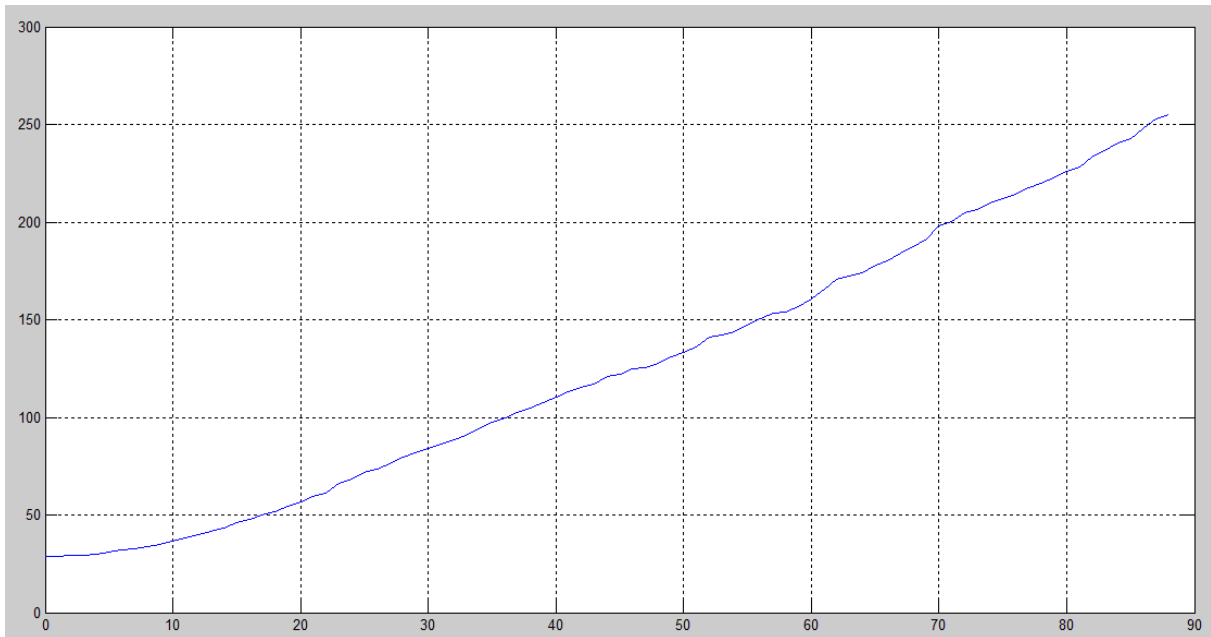


Figure 4.7 : réponse indicielle du system en boucle ouverte

On remarque que cette réponse est une courbe intégratrice donc c'est un système naturellement instable.

4.5.2 Détermination du retard apparent τ et de la tangente α et R:

Pour cela, on trace une asymptote et son intersection avec l'axe des temps, qui donnera la valeur du retard apparent, comme le montre la figure 4.8, on calcule sa tangente de manière analytique.

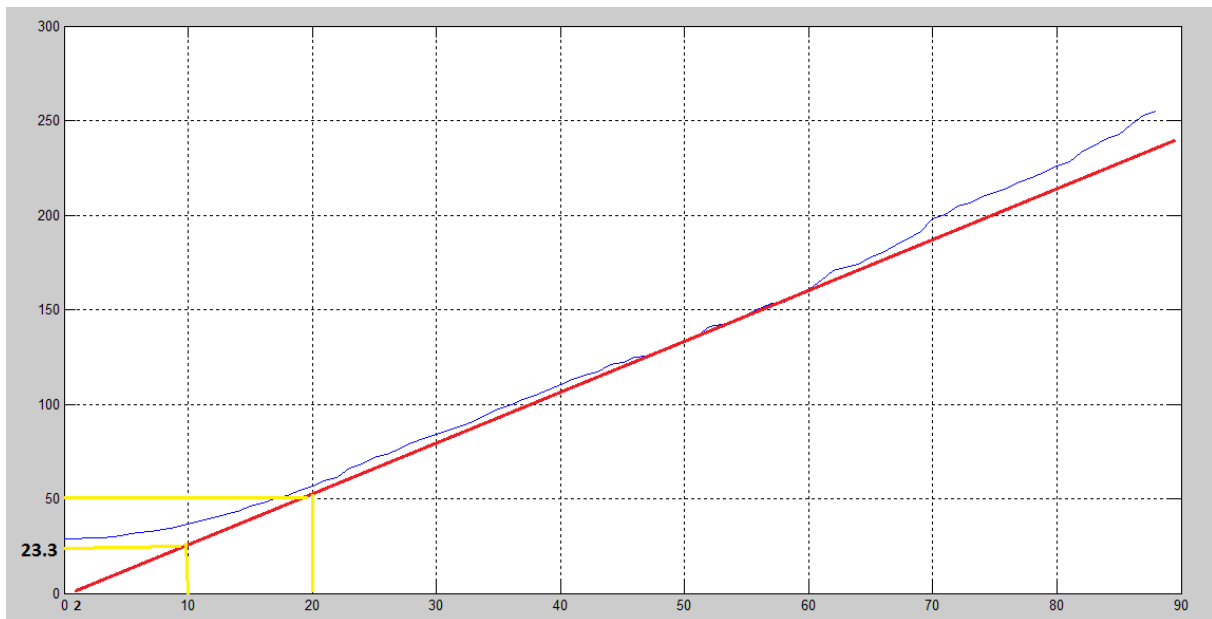


Figure 4.8 : détermination du temps de retard

Comme le montre la figure $\tau = 2$ s et $\alpha = \frac{50-23.3}{10} = 2.67$ et $R = \frac{2.67}{26.7} = 0.1$.

4.5.3 Détermination des paramètres :

En se basant sur la méthode de ziegler-Nichols , et des calculs du tableau 02 on a eu les résultats suivants :

$K_p = 6.35, \quad T_i = 4, \quad T_d = 1$
--

4.5.4 Résultats du câblage :

En se basant sur le schéma présenté dans la figure 3.23 on a câblé notre matériel et on a chargé le programme dans la CPU, qu'on a mis en mode RUN, donc on a eu les résultats présentés dans les figures suivantes :



Figure 4.9 : Image de l'API câblé

On remarque que la LED 'RUN' est allumée en vert, ainsi que la LED 'SF' est éteinte ce qui signifie qu'il n'y a aucune erreur système ou matériel.

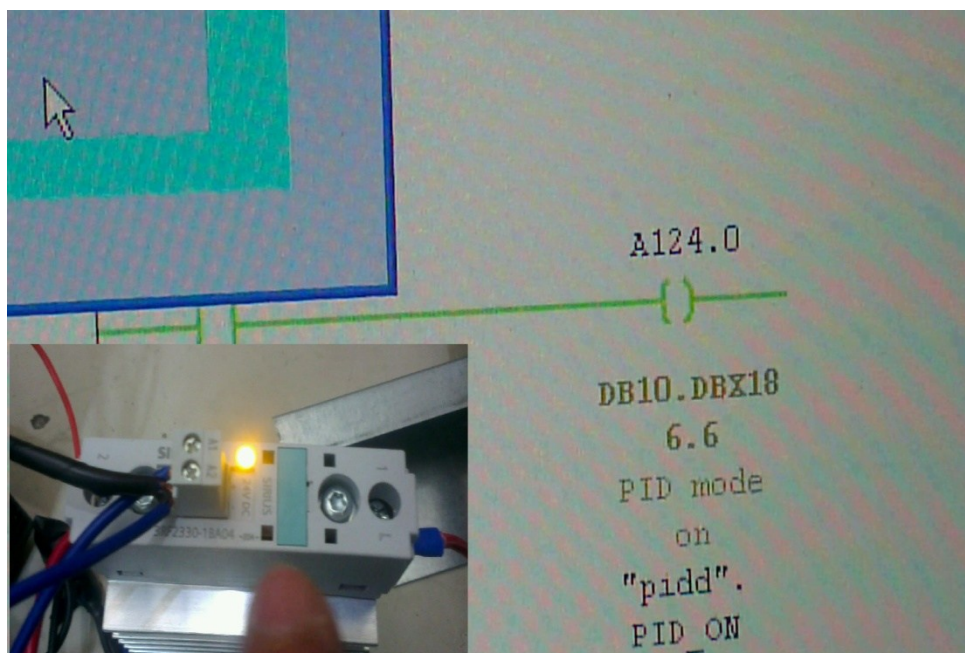


Figure 4.10 : Vérification de la cohérence entre le programme et le relais statique

La figure 4.10 démontre que le relais statique répond parfaitement à l'excitation que lui impose le programme.

4.5.5 Ajustement des paramètres :

Le tableau 03 ci-dessous illustre les principaux essais qu'on a effectués pour ajuster les paramètres du régulateur :

paramètres			Consignes (C°)	Réponse du SYS (C°)	dépassements
Kp	Ti	Td			
6.35	4	1	De 100→150	190	26.66%
6.35	4	1	De 80→100	135	35%
3.1	4	1	De 100→150	172	14%
3.1	2	1	De 70→100	142	42%
2.5	4	1	De 86→100	107	7%
2.5	4	1	De 72→100	115	15%
1.9	5	1	De 72→100	109	9%
1.5	5.5	1	De 86→100	102.5	2.5%
1.5	5.5	0.5	De 80→110	119	9%
1.5	6	1	De 100→130	144.5	11.5%
1	10	0.5	De 80→100	119	8%
1	10	0.5	De 90→110	112	1.8%
1	10	0.5	De 110→125	127.5	1.5%
1	10	0.5	De 125→130	130	0%
1	10	0.5	De 100→110	112	1%
1	10	0.5	De 110→150	158	5%
1	10	0.5	De 65→110	117	5.5%
1	10	1	De 26→120	139	15%
1	10	1	De 76→82	82	0%
1	10	1	De 82→100	103	3%
1	10	1	De 100→150	160	8%
1	10	1	De 55→65	64.8	0%
1	10	1	De 85→115	119.5	3.9%
1	10	1	De 115→155	163	5.16%
1	10	1	De 50→55	55.9	0.09%
1	10	1	De 65→85	87.5	2.09%

Tableau 03 : Essais d'ajustement des paramètres du régulateur

Comme le montre le tableau 03, on a débuté l'ajustement à partir des paramètres calculés par la méthode de Ziegler-Nichols, mais on a remarqué que le système démontrait des dépassements importants, donc on a commencé à réduire graduellement l'action P, et augmenté le paramètre T_i , vu que cela diminuait l'action intégrale $K_i = \frac{K_p}{T_i}$, puisque ces deux actions qui sont responsables du pourcentage du premier dépassement qui ne doit pas être au delà de 10% de la consigne, vers la fin on a essayé de jouer sur la rapidité du système afin de le rendre le plus rapide possible sans influencer sa précision.

Donc en dernier lieu on a opté pour les paramètres $K_p=1, K_i=0.1, K_d=1$

Le système ainsi réglé, a présenté une meilleure réponse avec des dépassements acceptables et une très bonne précision.

4.6 Conclusion :

La présentation des résultats illustre, au mieux, les objectifs atteints et permet de les comparer avec ceux tracés par le cahier de charge.

Pour s'assurer qu'un projet de ce type se déroule comme prévu, il faut vérifier que les fonctions de sa partie programmation s'exécutent sans erreurs, de même que son câblage ne doit présenter aucune défaillance.

Afin d'avoir une bonne régulation de la température, il faut trouver le bon compromis entre les paramètres qui performant au mieux le système en matière de stabilité, précision, rapidité et dépassement.

1.1 Introduction :

Dans ce chapitre, on va faire la présentation de l'unité 'molfix' pour la production des produits de soins pour bébés, afin de faire connaissance et de donner une idée générale sur l'installation, avec tous ses différents processus et ainsi que les utilités qui assurent le fonctionnement de l'unité.

1.2 Présentation du Groupe HAYAT :

Créé en 1937, HAYAT HOLDING est un groupe industriel d'origine Turque, présent aujourd'hui dans 70 pays, représentant les 5 continents avec plus de mille produits. Ce groupe est composé de 15 sociétés dont HAYAT PRODUITS D'ENTRETIEN ET DE SANTE qui représente la plus grande organisation dans son secteur. Avec une capacité de production annuelle de 1,5 milliards d'unités de couches bébés, de 1,5 milliards de serviettes hygiéniques et de 950 millions de feuilles de lingettes, cette dernière dont fait partie le groupe SARL HAYAT DHC Algérie.

SARL HAYAT DHC Algérie, fondée en 2005 dans la wilaya de Blida est présent dans le secteur des produits d'entretien ménagers. Elle s'est attelée à la production de tout ce qui touche à l'hygiène corporelle en mettant au marché, des lingettes, des serviettes périodiques, des mouchoirs humides et des couches bébés de première qualité.

Cette société est composée d'unités de production de Silicate, Labsa, détergent liquide et en poudre, glycérine, savon liquide, cosmétique, serviette hygiénique, papier

hygiénique et couche bébé. Afin de garantir la satisfaction absolue du consommateur, Hayat DHC développe ses produits en utilisant les dernières technologies.

1.3 Présentation de l'unité MOLFIX:

Les produits de soins pour bébés et d'hygiène sont fabriqués au sein des unités de production d'hygiène du Groupe Hayat, nommé 'molfix' illustré dans la figure 1.1, situé sur la route nationale 29 zone d'activité Bouinan-Blida. Cette unité a démontré son professionnalisme en obtenant le certificat de l'ISO 9001 en 2008, ce qui lui a permis de s'imposer sur plusieurs marchés dans les pays africains et du Moyen Orient après avoir satisfait les besoins du marché local.

La production de couches se fait par quatre lignes de production avec des machines à grande vitesse de production, compétitives au niveau international en utilisant une haute technologie et un haut degré d'automatisation. Une centrale électrique indépendante du réseau sonelgaz est installée afin de subvenir aux besoins de l'unité en énergie électrique.

Pour garantir une meilleure qualité et hygiène du produit, un laboratoire de contrôle de qualité, géré par du personnel qualifié, est intégré au sein de l'unité.

Afin d'assurer une production continue, des groupes de mécaniciens et électriciens veillent sur la maintenance et l'optimisation des machines de l'unité. Dans ce qui suit un aperçu des processus dont ce personnel côtoient. [13]

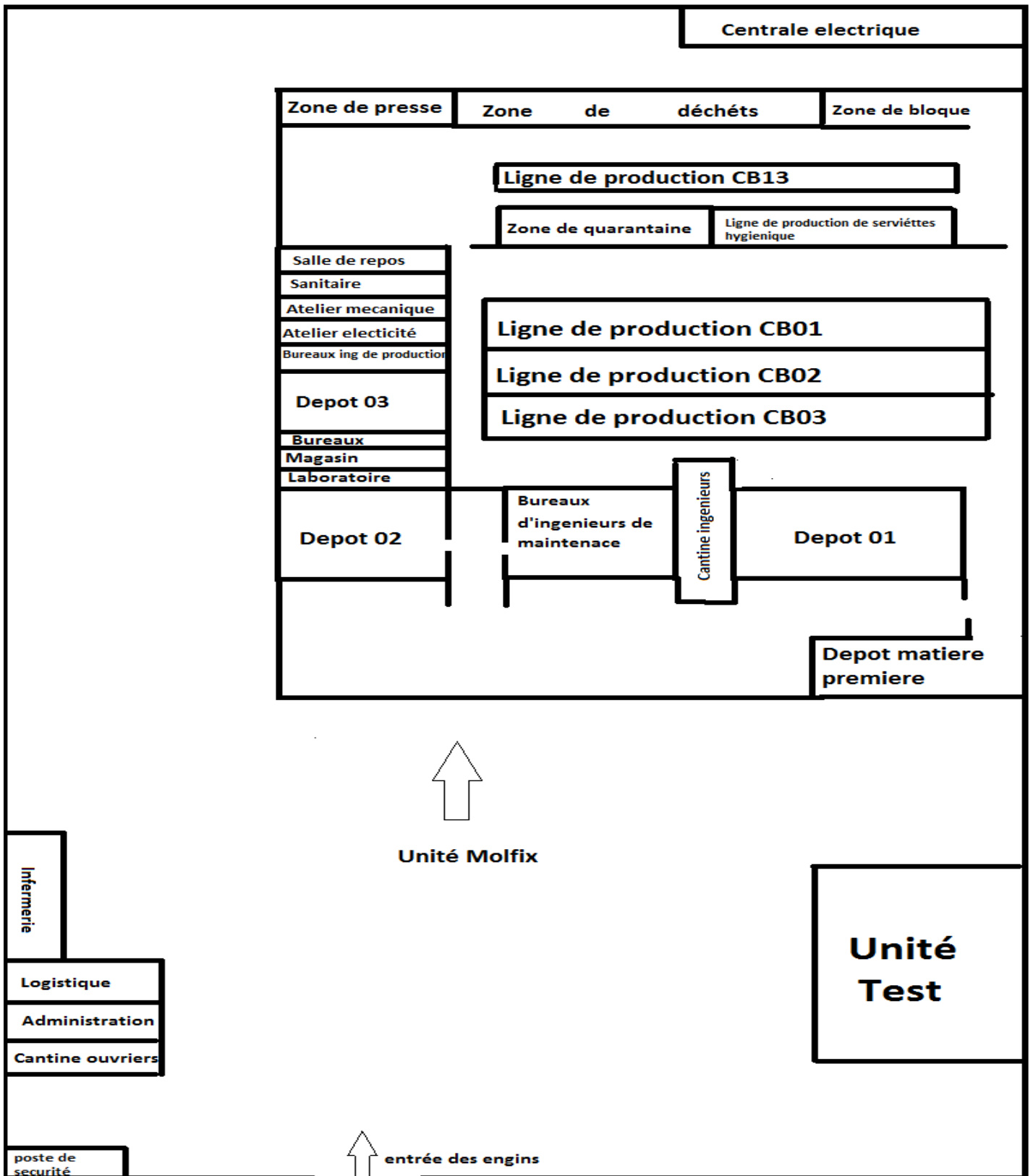


Figure 1.1 : Schéma synoptique de l'unité molfix

1.3.1 Les processus de production :

Pour concevoir des couches bébés molfix, quatre lignes de production identiques sont mises en place pour assurer chacune, la production d'une taille de ce produit.

Chaque machine doit traiter et assembler différentes matières premières. Ce traitement doit passer par plusieurs processus composés d'unités de traitements, afin d'obtenir le produit fini, qui est composé de plusieurs matériaux, comme présenté dans la figure 1.2:

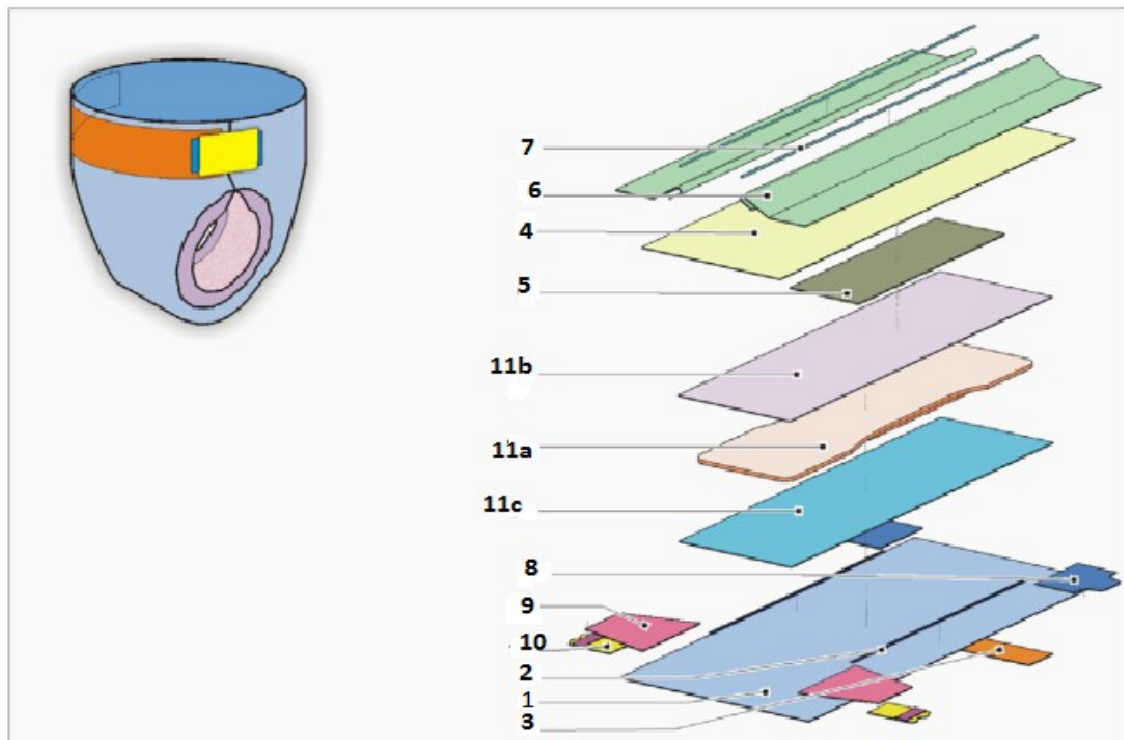


Figure 1.2: Composition d'une couche bébé molfix

1. **Backsheet (polyéthylène)** : feuille arrière de la couche.
2. **leg elastics** : élastique de jambes.
3. **frontal tape** : ruban frontal
4. **non-woven topsheet** : feuille de dessus non-tissé.
5. **acquisition layer** : couche d'acquisition

6. **leg gathers non woven** : rassembleur de jambe, non-tissé
7. **leg gathers elastics** : élastique rassembleur de jambe
8. **front wings** : ailes de devant
9. **rear wings** : ailes arrière
10. **rear wings tape** : ruban adhésif d'aile arriver.

11. **Absorbing core** : qui est le noyau absorbant composé des matières suivantes :
 - 11a Fluff + SAP
 - 11b Upper tissue
 - 11c Lower tissue

Un processus est défini comme étant une chaîne d'unité de traitement exécutant un ordre de montage partiel d'un produit.

Les processus et les unités de traitement formant les lignes de production au sein de l'unité et assurant le bon déroulement de la production sont les suivants :

1-Les Processus responsable de la fabrication de la partie inférieure de la couche :

La fabrication de la partie inférieure de la couche illustrée dans la figure 1.3, commence de la droite de la ligne de production vers la gauche de manière horizontale dans l'ordre suivant :

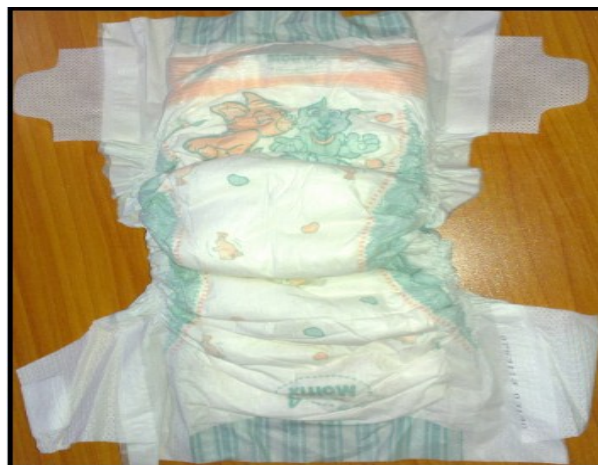


Figure 1.3 : Image réelle de la partie inférieure de la couche

- **Backsheet (Poly application) :** Dans ce processus la matière de fabrication de la feuille arrière de la couche « polyéthylène » est déroulée dans un dérouleur, puis trainée à une tension pré réglable par le rouleau d'entrée, qui est équipé d'un conditionneur permettant d'accélérer ou décélérer les servomoteurs pour commander la tension de tirage du polyéthylène, dont le mouvement latérale, est contrôlé par le guide de bordures. Cette matière est stratifiée dans le groupe de stratification « lamination group », et une colle lui est appliquée par l'applicateur de colle, car toutes les parties de la couche doivent coller en elle, et enfin cette dernière est refroidie par une unité de refroidissement « cooled unit », pour ne pas perdre sa forme, cela se fait hors tapis, une fois la feuille arrière prête, elle est envoyée sur le tapis horizontal pour se coller avec ce qui suit [1].
- **Noyau absorbant :** Cette partie de la machine a pour mission de produire les éléments absorbants, qui se trouvent au noyau de la couche et qui ont pour rôle d'absorber les liquides et de les geler. Ce processus est, en fait composé de trois unités indépendantes qui sont :
 - a-SAP+ Fluff (Cellulose) :** Ce sont deux matières responsables du gel des liquides dans la couche, Ces deux matières sont préparées hors tapis, dans des drum (barils) qui leurs donnent leur forme, puis ils sont envoyés dans le tapis.
 - b-Upper tissu et lower tissu :** Ces deux unités sont identiques, sauf que l'une d'elle contient un applicateur de colle pour pouvoir coller le tissu supérieur fabriqué par « upper tissu » et le tissu inférieur fabriqué par « lower tissu ».

Ces deux tissus couvrent le SAP et le FLUFF afin de garder la forme donnée par les drum.

- **Leg elastic application (Application d'élastique de jambes) :** L'unité de nettoyage de bande d'élastique plat « flat elastic band cleaning unit » guide le matériau bande élastique plat, à diviser pour fournir les élastiques à mettre autour de l'ouverture de jambes, à l'intérieur d'une caisse d'aspiration afin d'éliminer le talc appliqué sur sa surface. Un rouleau

d'entrée tire le matériau qui est orienté par l'élastique splitter, afin de fournir le bon nombre de fil nécessaire d'élastique, qui seront trainés et enfilés puis fixés sur le backsheet, tout en étant guidé par une unité appelée, applicateur d'élastique courbé « curved elastic applicator ».

- **Frontal tape :** La matière de fabrication du ruban frontal est déroulée dans un dérouleur « unwinder », de manière à approvisionner constamment la ligne de production avec la matière première. Ce ruban frontal venant du dérouleur est tiré par une unité de processus appelée rouleau d'entrée « Infeed », puis ses mouvement latéraux sont guidés et contrôlés par le guide de bordure « edge guider », qui est muni de photocellules communiquant avec une unité de traitement afin de détecter et d'éviter l'en sabotage du matériau. L'applicateur de colle étale, par le biais de ses pistolets, une colle sur la surface du ruban frontal, puis ce dernier est coupé en portions et collé sur une autre matière appelée polyéthylène (backsheet).

2-les processus responsable de la fabrication de la partie supérieure de la couche :

Cette partie de la couche, illustrée dans la figure 1.4 , se fabrique de la gauche vers la droite de la ligne de production se déplaçant sur un convoyeur en biais, de manière à se croiser avec la partie inférieure sur le tapis horizontal pour que l'une soit collée sur l'autre.



Figure 1.4 : Image réelle de la partie supérieure de la couche bébé

- **Topsheet non woven (feuille de dessus non tissé) :**

a-Topsheet center : Dans cette partie de la chaîne de production, un dérouleur contient la matière non tissée « non-woven », destinée à la fabrication de la feuille de dessus de la couche. Le matériau déroulé est tiré par un rouleau d'entrée avec une tension conditionnée, puis il est laminé par le groupe de starisation, car cette feuille doit coller avec une matière appelée cuff dans ces deux bordures.

b- Rassembleur de jambe non tissé (Leg gathers non woven) : cette partie est faite à base du cuff, qui est déroulé en feuille de 190mm de largeur. Cette feuille est coupée en deux, pour coller chaque partie dans l'une des bordures du topsheet center.

Les deux matières précédentes, qu'illustre la figure 1.5, doivent être en contact direct avec les liquides, c'est pourquoi on n'utilise pas de colles pour les rassembler mais les cylindres formant l'unité thermosealing (figure 1.6), les presse. Les deux premiers cylindres font une presse primaire, puis les deux seconds les pressent en gravant des ouvertures sur leurs surfaces et puis les deux derniers font une presse finale.

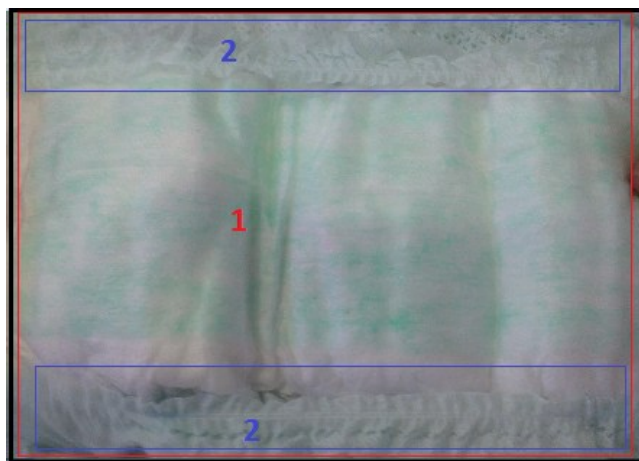


Figure 1.5 : Image réelle des deux matériaux (Topsheet – Leg gather)

1 : Topsheet center (Feuille de dessus non tissé)

2 : Leg gather (Rassembleur de jambe -Cuff-)

La pression ne suffit pas toute seule pour coller ces matériaux, c'est pourquoi des résistances chauffantes sont placées au centre des cylindres, pour qu'ils soient pressés tout en étant chauffés. La température de chauffage peut froisser le « cuff » si elle n'est pas précise, c'est pour quoi cette température est surveillée par des thermocouples et régulée par des thermorégulateurs. Cette régulation présente l'inconvénient de densité du câblage, ainsi que son imprécision, qui entraînent souvent des pertes matérielles. C'est pour quoi notre travail consiste à intervenir, en modifiant ce processus, de manière à le rendre plus précis en matière de régulation, et d'éviter les défaillances qu'il présente actuellement.

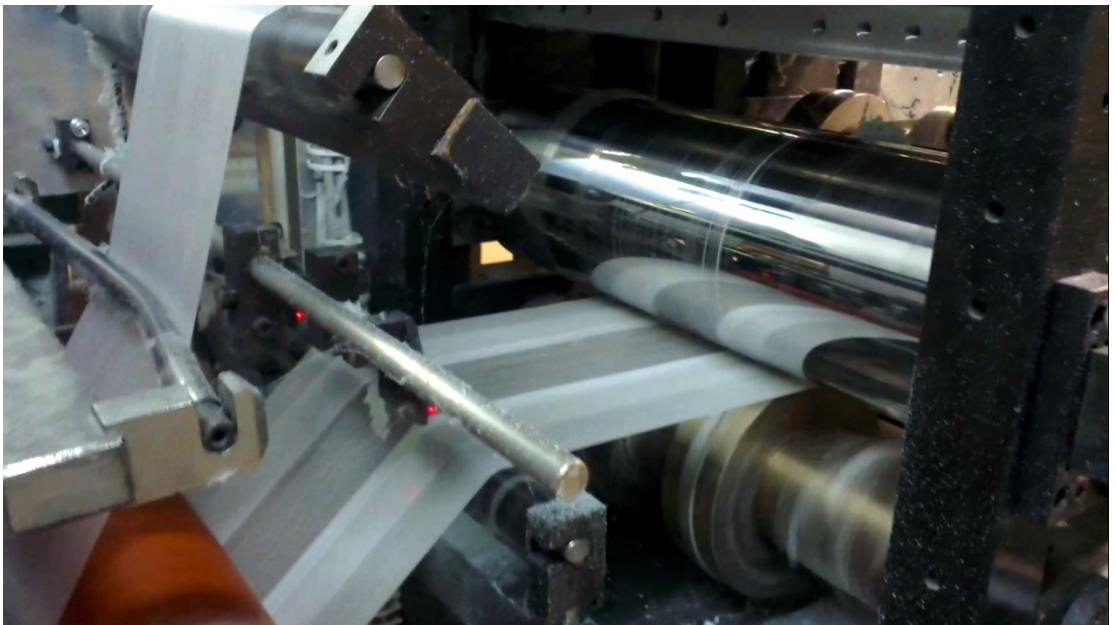


Figure 1.6 : Image réelle de deux cylindres de l'unité thermosealing

C- Couche d'acquisition : Cette partie de la machine façonne une matière qui a pour rôle, de diriger les liquides dans la couche, afin qu'ils soient absorbés. Cette matière doit contenir des trous. La matière première est déroulée dans un dérouleur à grande vitesse, puis elle est tirée par un rouleau d'entrée, tout en étant guidée par un système de guidage, pour être trouée dans des endroits précis, et une colle lui est appliquée pour coller avec le topsheet center.

- **Elastique rassembleur de jambes :** Appelé aussi élastique de la feuille de dessus, ce matériau est déroulé puis acquis par l'élastique inlet et divisé en deux fils, contenant chacun deux élastiques, pour qu'il passe dans le guide des quatre élastiques, afin d'appliquer une colle pour chacun par l'applicateur de colle, pour être collé au topsheet.
- **Ailes des couches:** Ces deux processus indépendants ont pour rôle, de donner la forme adéquate aux matériaux de fabrication des ailes d'avant de la couche « front wings », et des ailes arrière « rear wings ». Après que cette matière soit dirigée et façonnée par les unités de traitement de ces deux processus, ces deux parties ont une texture douce afin d'éviter l'infection de la peau de bébé par le frottement, puisqu' il sert de séparation entre cette dernière et le ruban adhésif.
- **Ruban adhésive d'ailes arrière :** Ce processus met en œuvre la matière permettant de fermer la couche. Cette matière doit être adhésive pour pouvoir coller les deux ailes de la couche, donc un applicateur de colle étale une colle sur le matériau, qui est mis sous pression par le biais du rouleau de pression.

Les deux parties de la couche (supérieure et inférieure) sont rassemblées par la colle et une presse et des lames s'occupent de couper chaque pièce de couche bébé, qui est convoyé vers une unité appelée « optima », qui a pour rôle de faire leur emballage.

Tout ces processus sont visualisés dans des Interfaces homme-machine (figure 1.7), ou l'opérateur peut introduire des consignes qui assurent le bon déroulement de la production. Les IHM de visualisation contiennent des raccourcis vers chaque processus, où l'opérateur peut visualiser l'état du processus et les alarmes qu'il présente, et introduire des consignes qui agissent sur le processus lui-même.

Ces IHM, contiennent des images illustratives de chaque partie de la couche, afin de faciliter la surveillance de la ligne de production.

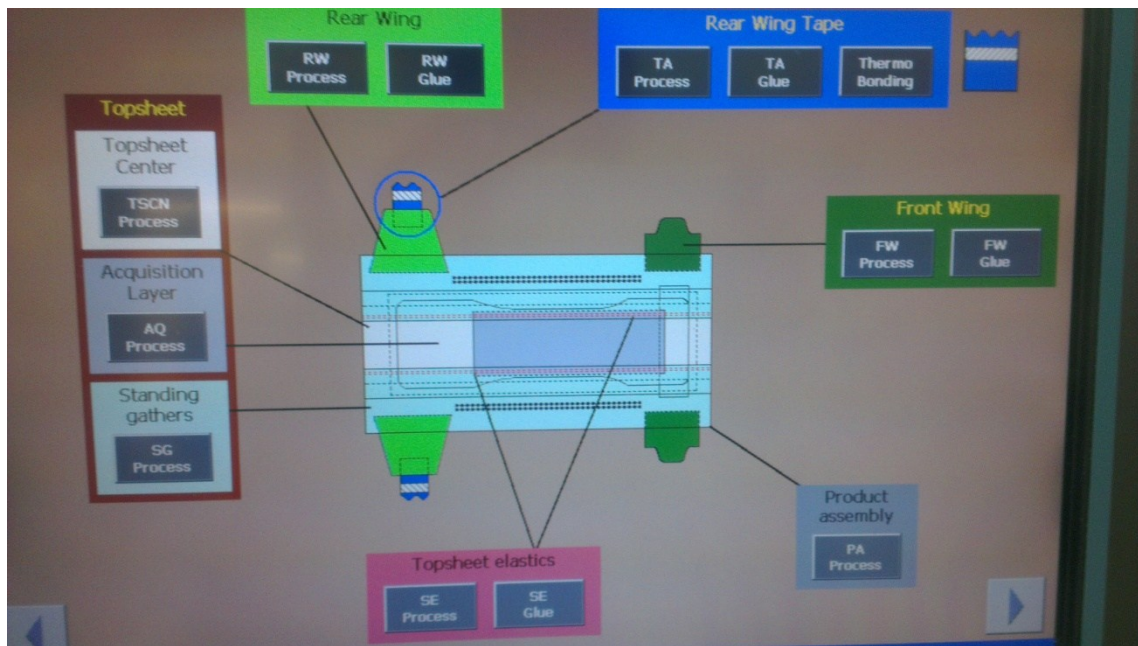


Figure 1.7 : exemple d'IHM de ligne de production

1.3.2 La centrale électrique :

L'alimentation en électricité de tous les processus de l'unité est assuré par une centrale électrique. Elle accomplit cette tâche indépendamment du réseau Sonelgaz, quand elle est en mesure de subvenir à la demande en énergie électrique des équipements. Dans la négative, l'alimentation se fait en parallèle avec le réseau Sonelgaz.

Cette centrale électrique est constituée de cinq moteurs à combustion interne (MCI), dont quatre à gaz et un de secours à gazoile. Chaque moteur entraîne via un arbre à came un alternateur. Le démarrage des moteurs à gaz est rapide, vu qu'ils servent comme groupe d'alimentation pour la production de base.

La figure 1.8 présente un schéma fonctionnel illustrant les éléments essentiels au fonctionnement de la centrale électrique.

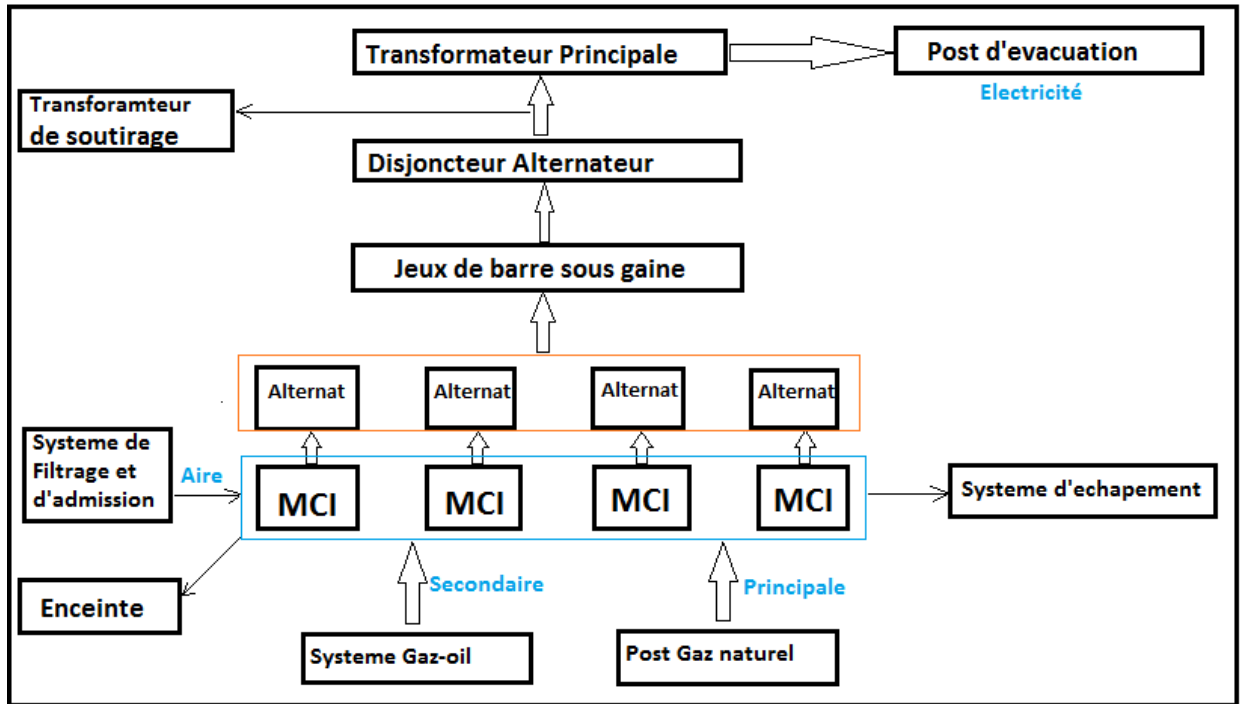


Figure 1.8 : Schéma synoptique de la centrale électrique

Les moteurs sont enfermés dans une enceinte acoustique afin de réduire les bruit sonores qu'ils émettent.

Ces MCI sont associés à d'autres systèmes qui assurent la production incessante de l'énergie électrique, qui sont les suivants :

- **Circuit d'air** : C'est le système permettant la filtration et l'admission de l'air vers les compresseurs des groupes a gaz.
- **Circuit d'échappement de gaz** : Système qui permet l'évacuation des gaz d'échappement des groupes a gaz.
- **Circuit combustible** : C'est les systèmes qui permettent le traitement et l'alimentation en gaz naturel et en gasoil de secours vers les MCI.
- **Électricité** : Sous l'effet de l'énergie mécanique qu'appliquent les MCI sur les alternateurs, ces derniers produisent de l'électricité. Celle ci est en suite transporté a travers un jeux de barres sous gaines, vers le transformateur principal , qui a pour rôle d'élever la tension électrique au niveau requis par le réseau électrique. Cette électricité est enfin évacuée vers le réseau via un post d'évacuation. Quant à l'alimentation électrique des systèmes auxiliaires de la centrale, un transformateur de soutirage assure cette tache. [2]

Le contrôle et la supervision de la centrale sont assurés par un système de contrôle-commande.

1.4 Conclusion :

Ce chapitre, a donné une idée sur le fonctionnement de l'unité avec ses différents processus, ainsi que la disposition et le rôle du processus sur lequel notre travail porte au sein de la ligne de production de couche bébé.

Dans ce qui suit, la présentation des généralités sur le matériel et les méthodes utilisées dans ce projet.

Liste des figures :

Chapitre 1 : Présentation de l'unité molfix

-Figure 1.1 : Schéma synoptique de l'unité molfix.....	05
- Figure 1.2: Composition d'une couche bébé molfix	06
- Figure 1.3 : Image réelle de la partie inférieure de la couche.....	07
- Figure 1.4 : Image réelle de la partie supérieure de la couche bébé.....	09
- Figure 1.5 : Image réelle des deux matériaux (Topsheet – Leg gather).....	10
- Figure 1.6 : Image réelle de deux cylindres de l'unité thermosealing.....	11
- Figure 1.7 : exemple d' IHM de ligne de production	13
- Figure 1.8 : Schéma synoptique de la centrale électrique.....	14

Chapitre 2 : Généralités sur les outils et méthodes utilisés

-Figure 2.1 : Les approches offertes par le logiciel Step7.....	19
- Figure2.2 : Configuration matériels.....	20
- Figure 2.3 Structure du télégramme profibus.....	26
- Figure 2.4 Structure interne d'un thermocouple.....	29
- Figure 2.5 : Chaîne d'une régulation.....	31
- Figure2.6 Réponse indicielle d'un système stable.....	32
- Figure2.7 Réponse indicielle d'un système instable.....	32
- Figure 2.8 : illustration de l'erreur statique sur la réponse indicielle d'un système.....	33
- Figure 2.9 : réponses oscillatoires amorties avec T_e différents.....	33

Chapitre 3 : Programmation et Réalisation

- Figure 3.1 : Image de l'état actuel de l'armoire de commande.....	37
- Figure 3.2 : Automate S7-300 2 C 2DP.....	38
- Figure 3.3 Design de l'IHM TP177 B 6".....	39
-Figure 3.4 : Image réelle d'une résistance.....	39
- Figure 3.5 : Relais statique 24V DC.....	40
- Figure 3.6 : transmetteur de température.....	41
- Figure 3.7 : module d'entrées analogique.....	41
- Figure 3.8 : Structure du programme.....	42
- Figure 3.9 : Fenêtre de la configuration matérielle.....	43
- Figure 3.10 : Courbe de mise en échelle.....	43
- Figure 3.11 : vue des blocs et fonction créé pour la mise en échelle.....	44
- Figure 3.12 : Fonction FC105 Scale pour la mise en échelle.....	45
- Figure 3.13 : Le bloc fonctionnel de régulation FB58 «TCONT-CP ».....	47
- Figure 3.14 : Bloc de données DB10.....	48
- Figure 3.15 : Intégration du projet Step7 dans le projet wincc.....	49
- Figure 3.16 : Choix du protocole de communication dans le projet wincc flexible.....	49
- Figure 3.17 : Connexion de l'IHM et de l'API aux réseaux maitre DP et MPI.....	50
-Figure 3.18 : Vérification de la connexion dans Wincc flexible.....	50
- Figure 3.19 : vue initial du programme Wincc flexible.....	51
- Figure 3.20 : Vue d'information pour les trois premières résistances.....	52

- Figure 3.21 : Vue d'informations pour les trois secondes résistances.....	52
-Figure 3.22 : Vue d'alarmes.....	53
- Figure 3.23 : Schéma de câblage.....	54
- Figure 3.24 : Schéma de câblage d'un convertisseur a 2 fils au module analogique.....	54

Chapitre 4 : Développement des résultats

-Figure 4.1 : vérification de la fonction FC 105 'Scall'	60
- Figure 4.2 : Vérification de la température de la résistance.....	60
- Figure 4.3 : Vérification de la fonction FB58 'TCONT'	61
- Figure 4.4 : Vérification que la consigne 140 °C est lise dans le DB 10.....	62
- Figure 4.5 : Vérification que la consigne 95 °C est lise dans le DB 10.....	62
- Figure 4.6 : Figure d'alarme affichant une alarme de défaillance	62
- Figure 4.7 : réponse indicielle du system en boucle ouverte.....	63
- Figure 4.8 : détermination du temps de retard.....	64
- Figure 4.9 : Image de l'API câblé.....	65
- Figure 4.10 : Vérification de la cohérence entre le programme et le relais statique.....	65

Listes des acronymes et abréviations :

A :

AC : Alternatif Curent.

AI : Analog Input.

API: Automate Programmable Industriel.

C :

CPU : Computer Process Unit.

CS : Configuration Software.

D :

DA : Destination Address.

DB : Data Bloc.

DC : Direct Curent.

DHC : Détergeant Hygiène Corporelle.

DI : Digital Input.

DIP : Dual In-line Packet.

DO : Digital Output.

DP : Decentralized Peripherals.

DS : Source Address.

DSPA : Destination Service Access Point.

DU : Data Units.

E :

ED : End Delimiter.

EN : European Normalization.

F :

FB : Fonctional Bloc.

FC : Function.

FC : Function Code.

F.E.M : Force Electro-Motrice

H:

HI-LIM : Hight Limit.

HMI : Human-Machine Interface.

I:

IAP : l'Institut Algérien du Pétrole

IEC : International Electrothechnical Commission.

IHM : Interface Homme-Machine.

ISO : International Standard Organization.

L :

LEn : Net Data Length.

LED : Light Emitting Diode.

LEr : Length repeated.

LO-LIM : Low Limit.

M:

Matlab : Matrice Laboratory.

MCI : Moteur a Combustion Interne.

MPI : Multi-Points Interface.

MRES : Memory Reset.

O :

OB : Organisation Bloc.

OP : Operateur Panel.

OSI : Open System Interconnections.

P:

PA : Process Automation.

PC : Panel Computer.

PG : Programming Gadget .

PID : Promotional Integral Derive.

PLC : Programmable logical Control.

Profibus : Programable Field Bus.

PS : Power Supply.

R:

RET-VAL : Return Value.

RT : Runtime.

S:

S7 : Step 7.

SAP : Super Absorbent.

SARL : Société A Responsabilités Limités.

SD : Start Delimiter.

SDN : Send Data with No acknowledge.

SF : System Fault.

SFC : Special Function.

SM : Signal Module.

SRD : Send and Request Data with acknowledge.

SSPA : Source Service Access Point.

T :

TOR : Tout Ou Rien.

TP : Touch Panel.

Bibliographie

- [1].Documentation HAYAT DHC Algérie : setting and adjustments ,operating and maintenance manual.
- [2].Documentation SONEGAZ : central a cycle simple, guide technique générale 02, année 2010
- [3].N-KANDI : automate programmable industrielle, configuration matériel, document de formation IAP boumerdes
- [4].N-KANDI : automates programmable industrielle, langage de programmation, document de formation IAP, boumerdes
- [5].L'aide du logiciel Siemens Step7 version 5.5
- [6].ACROMAG: introduction to profil DP , technical references ,2002
- [7].[www.agilicom.fr/ tutoriel-profile bus-dp.html](http://www.agilicom.fr/tutoriel-profile-bus-dp.html)
- [8].Site Internet : www.automation.siemens.com
- [9]. celduc relais : relais statique principe, manuel technique.
- [10]. PatrickProuvost : automatique contrôle et régulation, Dunod, 2007
- [11].Thermatec: process control ,thermocouple ,article

[12].Günter wellenreuther, dieterzastrow : automatisieren mit SPS, theorieund praxis, 2008.

[13].Site Internet: www.Hayat.com.dz