

Université de Toulon

IUT de Toulon

Département Génie Electrique et Informatique Industrielle (GEII)

# Automatisme

## Mesure de la température

écrit le 25 mars 2024

par

Bruno HANNA

*Encadrant universitaire :* Olivier REYMOND

---



# Table des matières

<b>Introduction</b>	<b>1</b>
<b>1 Lecture des Entrées TOR, Mesure de la Température, Codage d'un Grafcet</b>	<b>2</b>
1.1 Configuration du Schéma Système . . . . .	2
1.2 Interface IHM . . . . .	3
1.2.1 Interface Visuelle . . . . .	3
1.2.2 Interface de Pilotage . . . . .	3
1.3 Programme . . . . .	4

# Introduction

Cette section débute avec une exploration de la programmation via le logiciel Unilogic, un outil conçu pour le développement d'applications destinées à une variété d'automates produits par PL Systems. Unilogic se distingue par sa capacité à gérer des tâches complexes, notamment la lecture d'entrées Tout Ou Rien (TOR), la mesure précise de températures et le codage de grafsets, éléments essentiels dans l'automatisation industrielle.

La configuration matérielle utilisée pour les travaux pratiques (TP) comprend :

- Automate/Interface Homme-Machine (IHM) : USP 070-B10,
- Module d'Entrées/Sorties : UIS WCB1,
- Module de communication/CPU : UAC-02RSC.

L'objectif est de fournir une compréhension approfondie de ces composants et de leur programmation, afin de maîtriser les aspects fondamentaux de l'automatisation et de la régulation dans des environnements industriels variés. La suite du document détaille la mise en œuvre pratique de ces concepts à travers des exemples spécifiques, illustrant la puissance et la flexibilité d'Unilogic dans la gestion de scénarios d'automatisation complexes.

# Lecture des Entrées TOR, Mesure de la Température, Codage d'un Grafcet

## 1.1 Configuration du Schéma Système

Repère Schéma	Type d'E/S	Désignation sur le bornier du module UIS-WCB1
V0	Sortie TOR	O2
H2	Sortie TOR	O3
H3	Sortie TOR	O4
PT100	Entree Temp	R <sub>2+</sub> , RT <sub>2-</sub> , RT <sub>2+</sub>
QA	Entree TOR	I0

Figure 1.1 – Schéma de configuration des ports

Le schéma présenté dans la Figure 1.1 offre une vue d'ensemble des configurations de ports nécessaires à l'utilisation des Entrées/Sorties (IO). Cette visualisation permet d'identifier précisément les informations de port requises pour la manipulation efficace des signaux dans les applications visées.

## 1.2 Interface IHM

La conception initiale prévoit la réalisation de deux interfaces graphiques, commutables via un bouton situé en haut à gauche, comme illustré dans l'image de l'interface visuel'.

### 1.2.1 Interface Visuelle

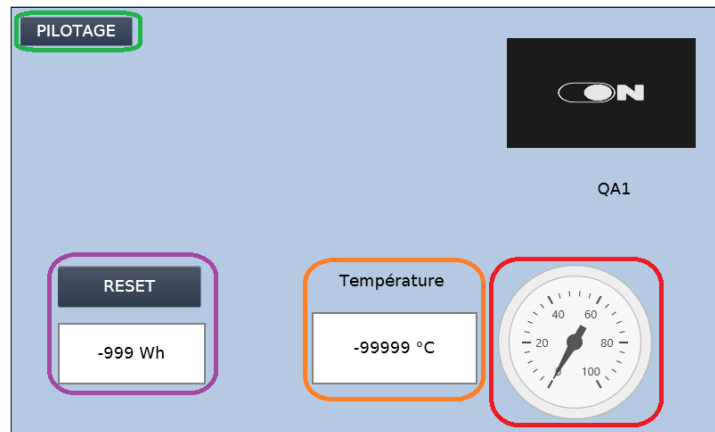


Figure 1.2 – Interface visuelle avec éléments de contrôle et d'affichage.

Un cadran, mis en évidence en rouge sur la figure 1.2, affiche la température de manière graphique. La valeur numérique réelle de la température est également présentée dans un encadré marqué en orange. En complément, l'interface inclut un indicateur de consommation électrique, souligné en violet, relié au compteur électrique.

### 1.2.2 Interface de Pilotage

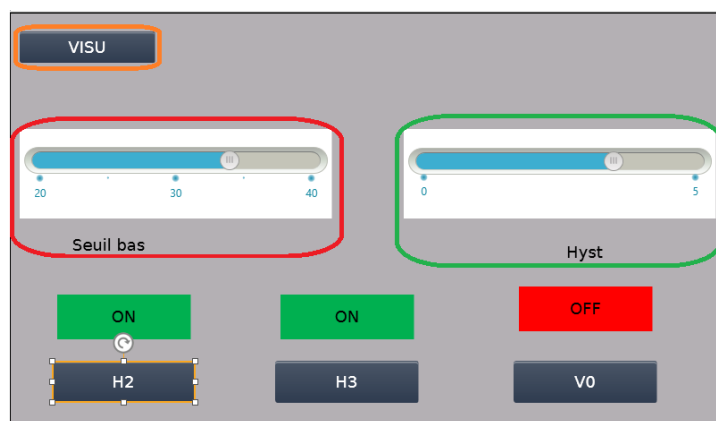


Figure 1.3 – Interface de pilotage avec contrôles avancés.

Cette interface comporte un curseur (slider) ajustant la valeur minimale de l'hystérésis pour l'activation du ventilateur, afin de prévenir les oscillations à l'approche du seuil de température. Cette fonctionnalité est visualisée en rouge dans la figure 1.3. Un second curseur permet de définir la plage de l'hystérésis, située entre 0 et 5, indiquée en vert sur le schéma.

## 1.3 Programme

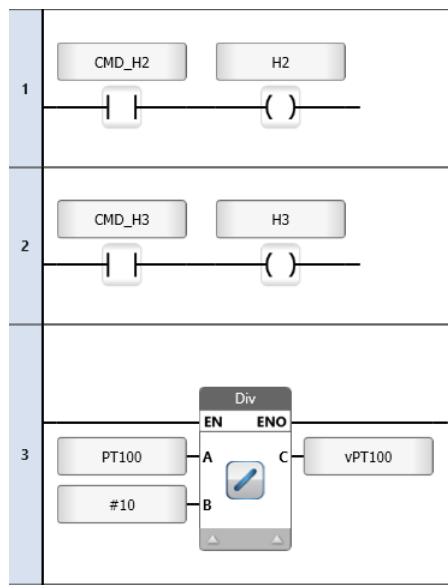


Figure 1.4 – Implémentation des routines H2 et H3 activées par des dispositifs d'interface homme-machine

Les routines H2 et H3 sont configurées pour être activées via les interfaces correspondantes. La valeur de température est ajustée par un facteur de division par dix pour une représentation adaptée dans l'interface utilisateur, comme illustré dans la Figure 1.2.

$$\text{Ligne 1 : Si } CMD\_H2 = \text{ON} \rightarrow H2 = \text{activé} \quad (1.1)$$

$$\text{Ligne 2 : Si } CMD\_H3 = \text{ON} \rightarrow H3 = \text{activé} \quad (1.2)$$

$$\text{Ligne 3 : } vPT100 = \frac{PT100}{10} \quad (\text{Division de la valeur de PT100 par 10}) \quad (1.3)$$

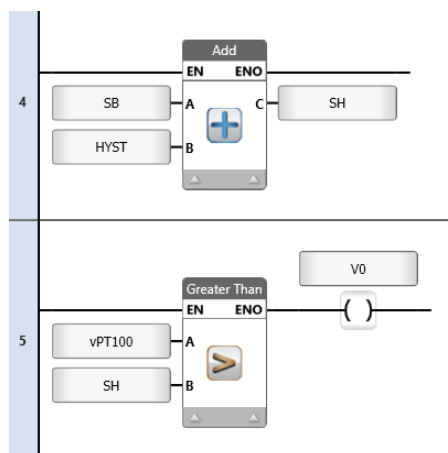


Figure 1.5 – Gestion de l'hystérésis et activation du système de refroidissement

La commande 4 intègre un algorithme d'hystérésis pour stabiliser l'activation du ventilateur en fonction des seuils définis dans l'interface de pilotage (Figure 1.3). La commande 5 active le système

de refroidissement basé sur la température mesurée par le capteur PT100, régulée via l'hystérésis.

Ligne 4 :  $SH = SB + HYST$  (Addition de la base de consigne SB avec l'hystérésis HYST) (1.4)

Ligne 5 : Si  $v_{PT100} > SH \rightarrow VO = ON$  (Activation du signal VO si vPT100 dépasse SH) (1.5)

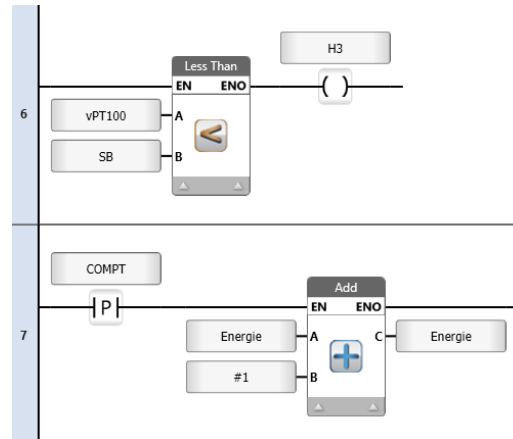


Figure 1.6 – Désactivation du refroidissement et comptage de la consommation énergétique

La ligne 6 désactive le système de refroidissement lorsque l'hystérésis indique une température inférieure au seuil minimal, évitant ainsi les oscillations inutiles. La ligne 7 implémente un compteur d'énergie, enregistrant chaque watt-heure consommé par le système, à travers un dispositif de mesure impulsif.

Ligne 6 : Si  $v_{PT100} < SB \rightarrow H3 = OFF$  (Désac de H3 si vPT100 est inférieur à SB) (1.6)

Ligne 7 :  $Energie = Energie + 1$  (Incrémentation du compteur d'énergie) (1.7)

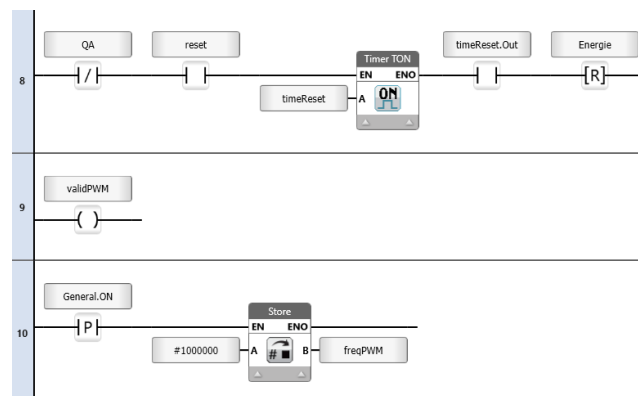


Figure 1.7 – Diagramme fonctionnel illustrant la logique de réinitialisation de l'énergie et de gestion du signal PWM

Au niveau de la ligne 8, la condition pour l'exécution de la réinitialisation énergétique est définie par l'état activé du disjoncteur QA et la pression soutenue pendant trois secondes sur le bouton de

réinitialisation. Lorsque ces conditions sont remplies, le temporisateur 'timeReset' atteint la valeur de trois secondes et active le relais de réinitialisation, qui efface le compteur d'énergie accumulée, comme indiqué par la variable 'Energie'.

$$\text{Si } QA = \text{ON} \wedge \text{reset pressé pendant } 3s \rightarrow \text{Réinitialisation de } \textit{Energie} \quad (1.8)$$

La ligne 9, quant à elle, implique un circuit logique où le signal valide PWM 'validPWM' est conditionné à la présence d'un signal opérationnel de général 'General.ON'. Ce dernier assure la validité du signal PWM pour le processus de modulation en largeur d'impulsion.

$$\text{Validité de PWM} \leftarrow \text{General.ON} \quad (1.9)$$

Enfin, la ligne 10 introduit une instruction de stockage où la valeur fixe '#1000000' est stockée dans la variable 'freqPWM' lorsque le signal général 'General.ON' est actif. Ceci établit la fréquence du signal de modulation en largeur d'impulsion (PWM) pour les opérations subséquentes.

$$\text{Si } \textit{General.ON} = \text{actif} \rightarrow \textit{freqPWM} \leftarrow \#1000000 \quad (1.10)$$

Chaque étape de ce diagramme représente une partie cruciale du contrôle de processus, permettant une gestion fine et une réinitialisation contrôlée de la consommation énergétique ainsi que la génération de signaux PWM pour des applications nécessitant une modulation précise.