

Université de Toulon

IUT de Toulon

Département Génie Electrique et Informatique Industrielle (GEII)

# SAE

## Recette Final

écrit le 19 juin 2024

par

Bruno HANNA

*Encadrant universitaire :* Stephane PIGNOL

---



# Table des matières

<b>Introduction</b>	<b>1</b>
1 Explication du Premier Bout de Programme LabVIEW . . . . .	2
1.1 Surveillance de la Puissance Secteur et Solaire . . . . .	2
1.2 Blocs de Mesure et de Contrôle . . . . .	2
1.3 Fonctionnalité et Flux de Données . . . . .	3
2 Explication du Deuxième Bout de Programme LabVIEW . . . . .	3
2.1 Calcul de la Puissance Réactive (PR) . . . . .	4
2.2 Gestion du Temps en Fonction de la Puissance Secteur (PS) . . . . .	4
3 Explication du Troisième Bout de Programme LabVIEW . . . . .	5
3.1 Envoi de la Requête HTTP . . . . .	5
3.2 Contrôle de l'Arrêt du Programme . . . . .	6
<b>Conclusion</b>	<b>7</b>

# Introduction

Dans le cadre de cette SAE, nous allons présenter la recette pour le test final du système de gradation pour un panneau solaire. Ce système intègre une résistance fictive afin de simuler la consommation d'électricité. L'objectif principal est de comprendre et de simuler le comportement du panneau solaire sous différentes conditions de consommation électrique.

Le rapport est structuré en plusieurs sections, chacune décrivant une partie spécifique du programme LabVIEW utilisé pour ce projet. Nous commencerons par la surveillance et le contrôle de la puissance électrique provenant des sources solaires et secteur. Ensuite, nous aborderons le calcul de la puissance réactive et la gestion du temps en fonction de la puissance secteur. Enfin, nous détaillerons l'envoi de requêtes HTTP pour ajuster dynamiquement un délai et assurer un arrêt contrôlé du programme.

Ce travail vise à démontrer l'efficacité de l'intégration des technologies de communication et de contrôle dans les systèmes énergétiques, en particulier pour l'optimisation de l'utilisation des panneaux solaires dans des conditions réelles d'utilisation.

# 1 Explication du Premier Bout de Programme LabVIEW

Le programme LabVIEW présenté ci-dessus est conçu pour surveiller et contrôler la puissance électrique provenant d'un panneau solaire et d'une source secteur. Voici une explication détaillée de ce programme :

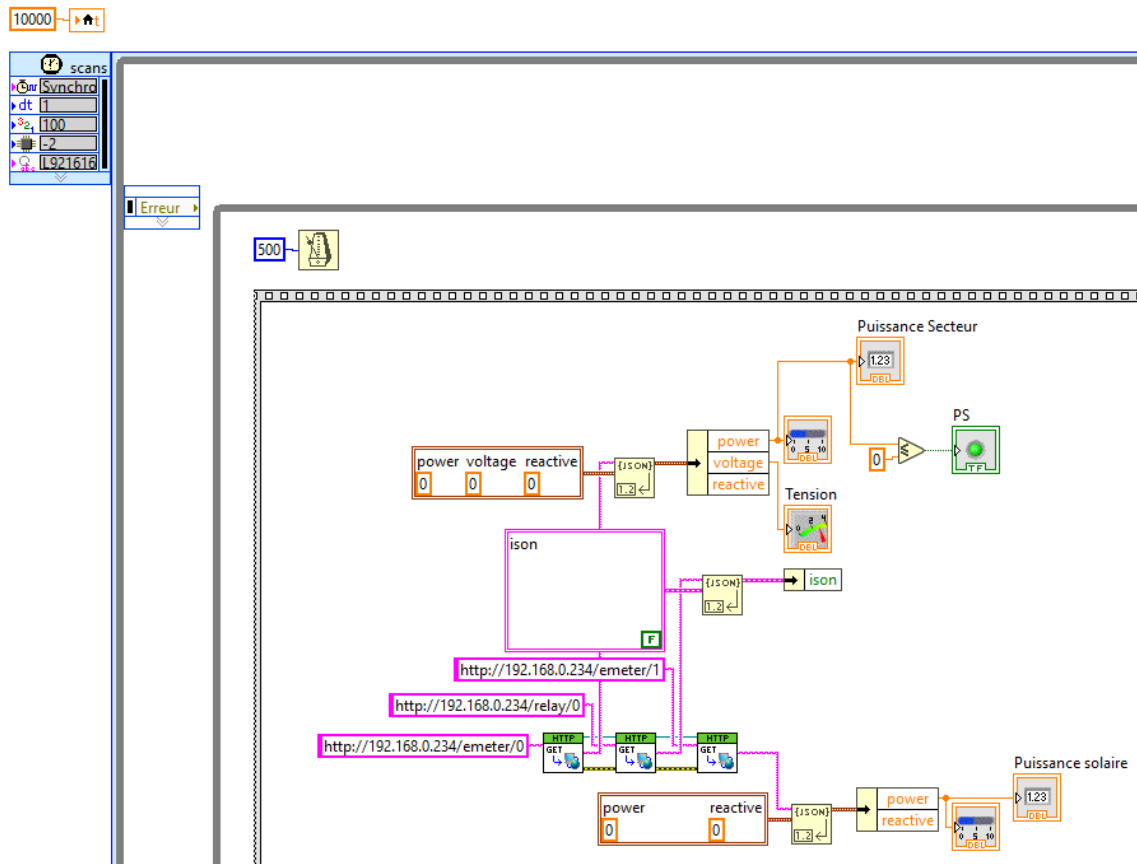


Figure 1 – Premier Bloc

## 1.1 Surveillance de la Puissance Secteur et Solaire

Le programme commence par la surveillance des paramètres de puissance, de tension et de puissance réactive provenant de la source secteur et du panneau solaire. Les données de ces paramètres sont obtenues via des capteurs connectés au réseau local.

## 1.2 Blocs de Mesure et de Contrôle

### 1. Mesure des Paramètres Électriques :

- Les blocs de mesure capturent les valeurs de puissance, de tension et de puissance réactive. Ces valeurs sont affichées en temps réel sur l'interface utilisateur.

### 2. Contrôle des Relais par API SHELLY HTTP :

- Le programme utilise des appels d'API pour contrôler les relais et recevoir les valeurs de consommation électrique. Par exemple, l'URL `http://192.168.0.234/emeter/0` est uti-

lisée pour obtenir les données de consommation et `http://192.168.0.234/relay/0` pour activer ou désactiver le relais.

### 3. Affichage et Interface Utilisateur :

- Les valeurs mesurées sont affichées à l'aide d'indicateurs numériques pour la puissance, la tension et la puissance réactive. Un indicateur de lumière montre également l'état de la source de puissance.

## 1.3 Fonctionnalité et Flux de Données

Le programme fonctionne en boucle continue, scannant les valeurs des capteurs à des intervalles réguliers (configurés à 500 ms). Les données sont ensuite transmises et affichées sur l'interface utilisateur pour fournir une surveillance en temps réel.

## 2 Explication du Deuxième Bout de Programme LabVIEW

Ce deuxième bloc de programme LabVIEW est responsable du calcul de la puissance réactive (PR) et de la gestion du temps en fonction de la puissance secteur (PS).

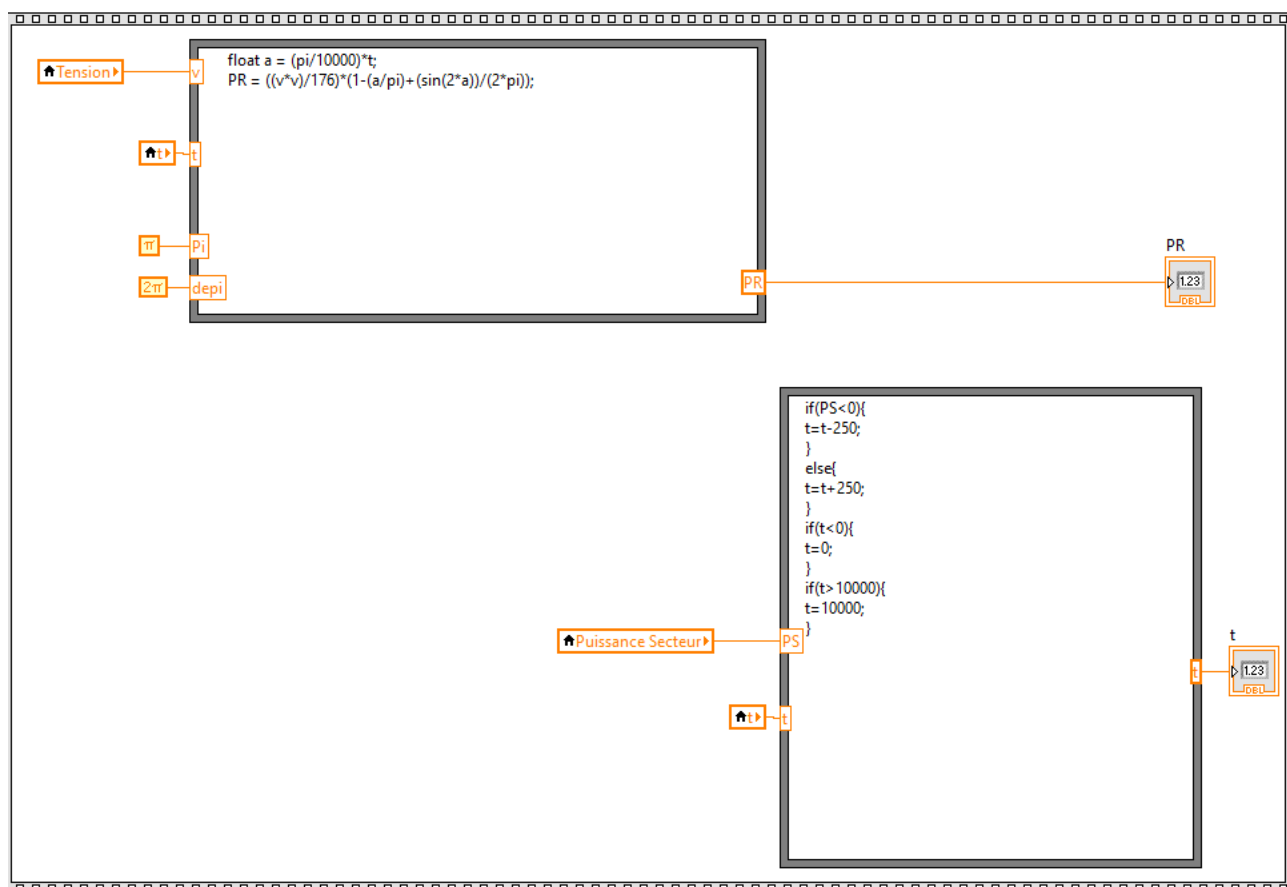


Figure 2 – Deuxième Bloc

Voici une explication détaillée de son fonctionnement :

## 2.1 Calcul de la Puissance Réactive (PR)

Le premier sous-système du bloc calcule la puissance réactive à l'aide de la formule suivante :

$$a = \frac{\pi}{10000} \cdot t \quad (1)$$

$$PR = \frac{V^2}{176} \left( 1 - \frac{a}{\pi} + \frac{\sin(2a)}{2\pi} \right) \quad (2)$$

— **Variables :**

- $V$  : La tension mesurée.
- $t$  : Le temps, ajusté dans le deuxième sous-système.
- $a$  : Une variable intermédiaire calculée en fonction du temps  $t$  et de la constante  $\pi$ .

- **Formule :** La puissance réactive  $PR$  est calculée en fonction de la tension  $V$  et de la variable  $a$ . Cette formule permet de modéliser la variation de la puissance réactive au cours du temps.

## 2.2 Gestion du Temps en Fonction de la Puissance Secteur (PS)

Le deuxième sous-système ajuste la variable de temps  $t$  en fonction de la puissance secteur  $PS$  avec la logique suivante :

```

1 if (PS < 0) {
2     t += 250;
3 } else {
4     t -= 250;
5 }
6 if (t < 0) {
7     t = 0;
8 }
9 if (t > 10000) {
10    t = 10000;
11 }
```

— **Variables :**

- $PS$  : La puissance secteur mesurée.
- $t$  : La variable de temps, ajustée en fonction de  $PS$ .

— **Logique de Contrôle :**

- Si  $PS$  est négatif,  $t$  est incrémenté de 250.
- Si  $PS$  est positif ou nul,  $t$  est décrémenté de 250.
- Le temps  $t$  est contraint entre 0 et 10000 pour éviter des valeurs hors de la plage acceptable.

Ce sous-système permet d'ajuster dynamiquement la variable de temps en fonction de la puissance secteur, ce qui influence directement le calcul de la puissance réactive (PR) dans le premier sous-système.

En combinant ces deux sous-systèmes, le programme peut simuler et analyser la puissance réactive en réponse aux variations de la puissance secteur et des conditions de charge.

### 3 Explication du Troisième Bout de Programme LabVIEW

Le troisième bloc de programme LabVIEW est responsable de l'envoi d'une requête HTTP à un serveur pour ajuster un délai, et d'assurer l'arrêt contrôlé du programme.

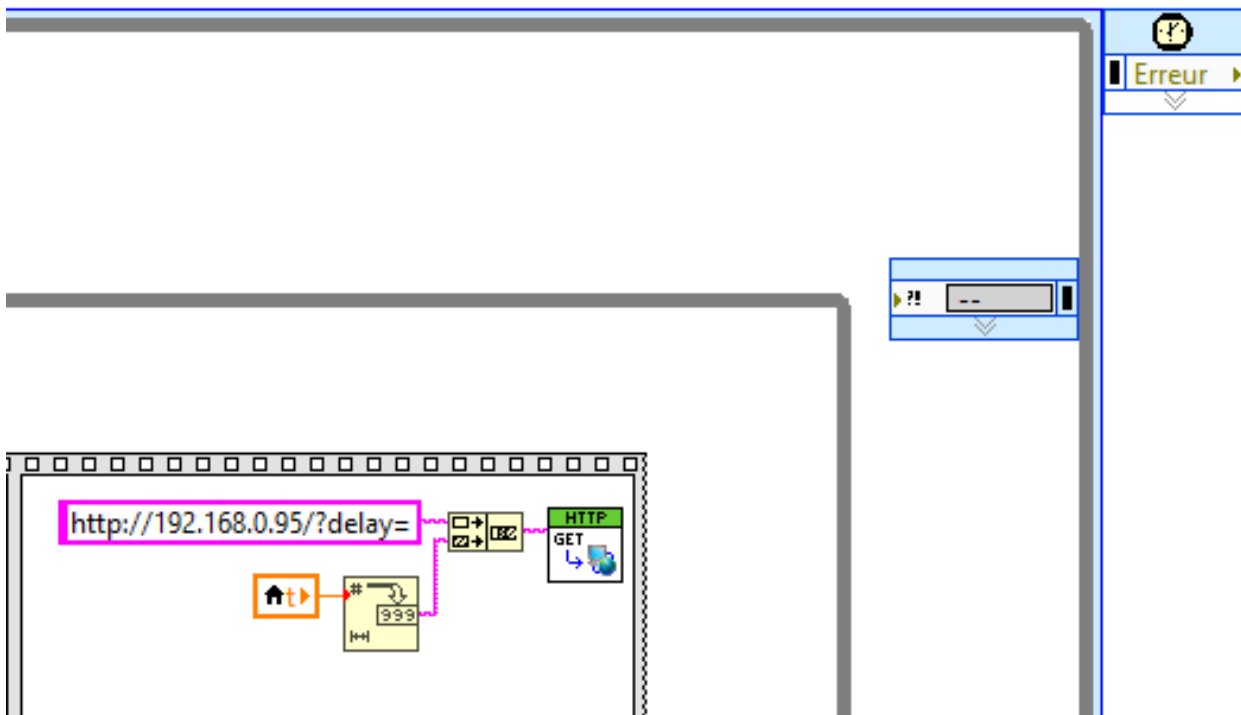


Figure 3 – Troisième Bloc

Voici une explication détaillée de son fonctionnement :

#### 3.1 Envoi de la Requête HTTP

Le programme envoie une requête HTTP pour ajuster un délai en fonction des paramètres spécifiés. L'URL utilisée pour cette requête est `http://192.168.0.95/?delay=`, suivie de la valeur du délai calculée.

- **Construction de l'URL :**

- L'URL de base est `http://192.168.0.95/?delay=`.
- La valeur du délai, représentée par la variable *delay*, est concaténée à l'URL de base pour former la requête complète.

- **Envoi de la Requête :**

- La requête HTTP GET est envoyée à l'URL construite, ce qui permet d'ajuster le délai sur le serveur.

### 3.2 Contrôle de l'Arrêt du Programme

Le programme inclut également un mécanisme pour arrêter proprement l'exécution du code. Un bouton d'arrêt (stop) est utilisé pour interrompre le programme lorsque nécessaire.

— **Bouton d'Arrêt :**

- Le bouton stop permet à l'utilisateur de terminer l'exécution du programme de manière contrôlée. Lorsque ce bouton est activé, le programme cesse d'envoyer des requêtes HTTP et arrête toutes les opérations en cours.

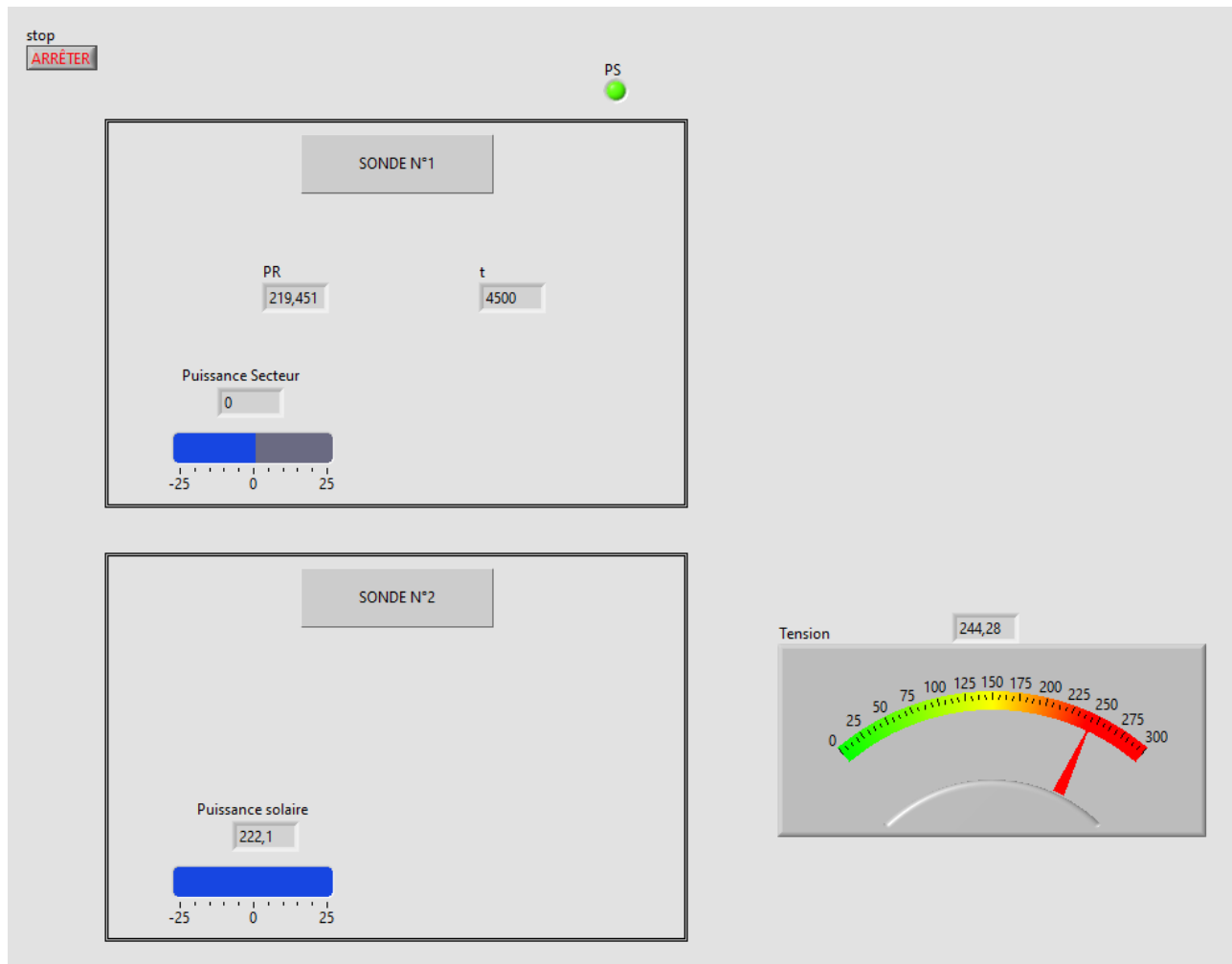


Figure 4 – Interface Utilisateur



# Conclusion

Ce travail pratique a permis de développer et de tester un système de gradation pour un panneau solaire, intégrant une résistance fictive pour simuler la consommation d'électricité.

Nous avons d'abord détaillé le fonctionnement du premier bloc de programme LabVIEW, qui se concentre sur la surveillance et le contrôle de la puissance électrique provenant de sources solaires et secteur. Cette partie inclut l'utilisation d'API SHELLY HTTP pour gérer les relais et recevoir les données de consommation électrique, avec une interface utilisateur affichant les paramètres en temps réel.

Ensuite, le deuxième bloc a été expliqué, montrant comment la puissance réactive est calculée et comment le temps est géré en fonction de la puissance secteur. Ce sous-système permet une modélisation précise de la variation de la puissance réactive.

Enfin, le troisième bloc a été présenté, démontrant l'envoi de requêtes HTTP pour ajuster dynamiquement un délai et assurer un arrêt contrôlé du programme. Cette section met en avant l'importance de la communication avec un serveur pour le contrôle en temps réel.