# Comptage et émission

BUT :

Dans le cadre de notre projet on a besoin de faire le comptage des impulsions à la sortie de notre système de mesure de puissance consommée puisque le nombre des impulsions va représenter la puissance moyenne consommée.

Problèmes :

L'Arduino est une carte mono-cœur avec un code procédural, donc le vrai multithreading est impossible. C’est ici ou apparait notre problème puisqu’on va dans notre projet utiliser la carte comme un émetteur cyclique d’information et comme un compteur qui ne doit pas perdre aucune impulsion lors de cette émission alors on doit rechercher des solutions pour rendre notre système capable de faire deux taches simultanément .

Les solutions proposées :

1. Les interruptions :

Une interruption est un déclenchement qui arrête l’exécution d’un programme ou d’un bout de code pour exécuter une autre fonction. Une interruption permet ainsi de stopper ou d’effectuer un programme puis d’en reprendre l’exécution à partir de l’instant où il a été stoppé.

Don pour garantir la certitude de notre comptage on doit associer à chaque arrivé d’impulsion une interruption pour qu’elle l’ajoute à son total compté et qu’elle continue son fonctionnement.

Cependant cela peut bien fonctionner pour une certaine fréquence limitée qui ne dépasse pas le 1Khz et puisque la fréquence représentant la consommation peut dépasser cette valeur on risque ici de perdre la bonne valeur de consommation. Enfaite lors de la réalisation du notre test on a remarqué une différence entre les valeurs calculées par la carte en utilisant les interruptions et celles lues sur un équipement de comptage des impulsions.

Donc on doit penser à séparer le hardware faisant le comptage de celui responsable à l’émission.

1. L’utilisation des temporisateurs :

Un temporisateur ou « Timer » est un élément hardware intégré au contrôleur Arduino. C'est comme une horloge et peut être utilisé pour mesurer des événements temporels. La minuterie peut être programmée par certains registres spéciaux. Vous pouvez configurer le pré-calculateur pour le temporisateur, le mode de fonctionnement et bien d’autres choses.

Donc l’utilisation des temporisateurs garantie la séparation des deux fonctionnements et par suite la valeur exacte de la consommation. Dans ce cadre, nous allons utiliser une bibliothèque prédéfinie qui va utiliser les temporisateurs de la carte Arduino et par suite calculer les impulsions. Cette bibliothèque est appelée FreqCount .

FreqCount mesure la fréquence d'un signal en comptant le nombre d'impulsions pendant une durée déterminée de 1Khz jusqu’à 8Mhz.

Code d’utilisation :

