

# TP1

## Prise en main de l'IDE PSoC Creator

### Interrupton Hardware, Timer, PWM, ADC

#### Compétences attendues :

#### Outil de developpement

- Être capable d'utiliser un IDE (chaîne de compilation, programmation, debugger)
- Savoir mettre en place un mécanisme d'interruption
- Savoir paramétrer un Timer (génération de signaux rectangulaire et PWM unipolaire)

#### Ressources :

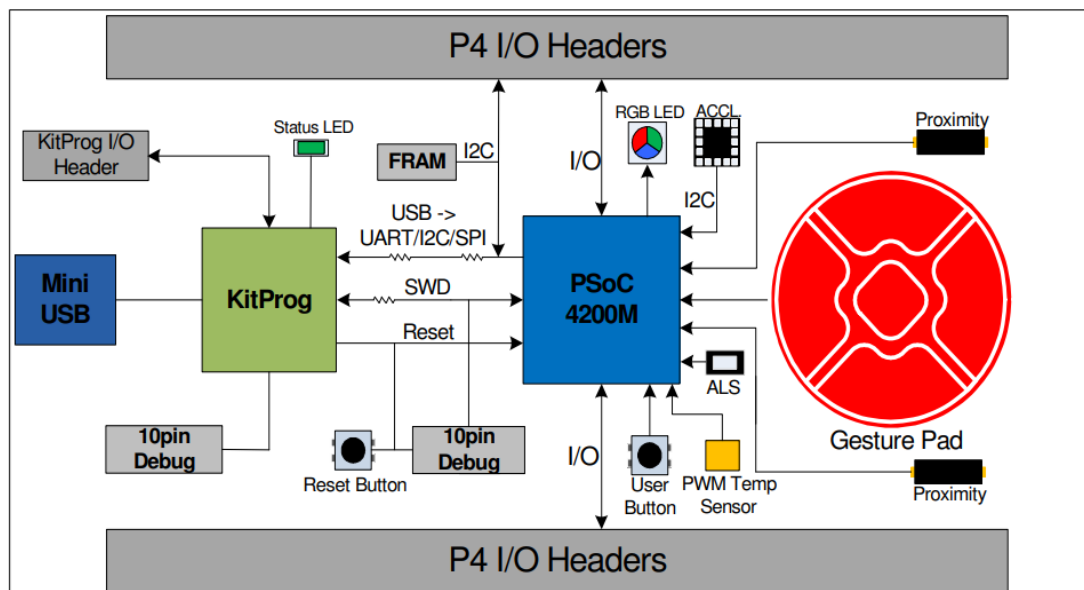
- MC60\_ressource-1, MC60\_ressource\_IT\_Timer\_ADC
- Notes d'application Infineon
- Structure détaillée de la carte (Infineon-CY8CKIT-044\_Schematic)

#### Consignes :

- Fournir un compte rendu (schéma, programmes indentés et commentés, résultats et interprétations) en format pdf (6 pages max.)

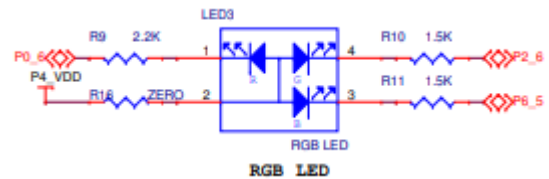
Rappel : Les interruptions sont une partie importante de toute application embarquée. Une interruption libère le processeur de l'obligation d'interroger en permanence la survenance d'un événement (polling).

#### Synoptique de la carte de développement



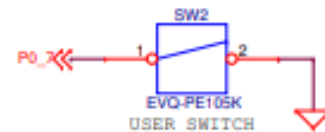
### Partie 1 : LED RGB contrôlée par PWM

Faire clignoter la LED RGB en vert (câblée en P2.6 : cf. schématique) à une fréquence de 2 Hz (rapport cyclique 0.5) à l'aide d'un bloc PWM. Faites en sorte d'activer la LED rouge lorsque la verte est éteinte et inversement.



Transposez la fréquence du signal PWM à 200 Hz et restez sur la couleur rouge uniquement. Relevez et sauvegardez le signal obtenu à l'oscilloscope. Validez le fonctionnement.

A présent, on souhaite simuler le fonctionnement d'un allumage de feux stop d'une auto. On rappelle que depuis une bonne dizaine d'années, les feux utilisent la technologie PWM. Pour cela, on simule l'appui sur la pédale de frein par un interrupteur (prendre SW2 « user button » sur la carte). On doit piloter la LED RGB en rouge (cf. schématique).



La technique utilisée consiste à faire passer le rapport cyclique de la PWM de 5% à 95% lors d'un appui sur SW2.

Fournir une solution informatique permettant de satisfaire à ce fonctionnement. (On veillera à placer une résistance de « pull-up » de quelques kΩ sur SW2) Validez le fonctionnement et relevez les signaux dans les deux cas de figure (SW2 off et SW2 on)

Periode à 10ms pour éviter l'effet clignotement

### Partie 2 : contrôle d'une LED par interruption

Fournir une solution informatique de telle sorte que la LED RGB (couleur bleue) clignote avec une fréquence de 2/3 Hz et un rapport cyclique de 2/3.

Attention LED bleu dispo uniquement sur line\_n  
Rapport cyclique géré dans le main.c  
A refaire avec une interruption

### Partie 3 : conversion analogique numérique

On souhaite échantillonner une grandeur analogique (tension issue d'un potentiomètre) toutes les 500ms et afficher la valeur de la tension sur un moniteur série (TeraTerm).

Pour la communication série, un bloc UART sera utilisé dans le design.

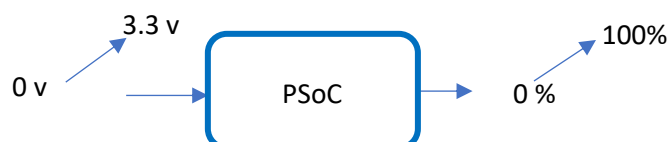
Ecrire un programme permettant de satisfaire à ce fonctionnement.

Rem : Le potentiomètre doit être polarisé à 3.3V et la tension de référence du convertisseur sera VDDA (3.3V).

Faire ensuite correspondre l'image de cette tension au rapport cyclique d'un signal PWM à 20kHz destiné à piloter les drivers d'un convertisseur statique d'énergie.

A  $v_{\text{analog}} = 0V$  correspond  $RC=0\%$

A  $v_{\text{analog}} = 3.3V$  correspond  $RC = 100\%$   
(RC : Rapport Cyclique)



Prévoir également l'affichage du rapport cyclique sur le moniteur série (TeraTerm).

Validez le fonctionnement et vérifiez que le rapport cyclique affiché et le même que celui fourni par l'oscilloscope.