

R3.07 : INFORMATIQUE EMBARQUEE

TP7 : Mise en œuvre de la PWM

1 Objectifs du TP et présentation de la PWM

Ce TP a pour but de mettre en œuvre le signal de commande d'une PWM (*Pulse Width Modulation*, en français Modulation de Largeur d'Impulsion).

La PWM est un moyen de faire délivrer à un générateur (par exemple une source de tension) une quantité variable d'énergie à un récepteur (par exemple une source de courant tel un moteur), avec un rendement énergétique théorique de 100%.

Dans la **Figure 1** ci-dessous, on trouve un « demi-pont » où K_1 et K_2 sont des interrupteurs électroniques de type « transistor-diode », qui peuvent être soit ouverts, soit fermés. K_1 et K_2 sont toujours commandés en opposition pour éviter un court-circuit de la source de tension E .

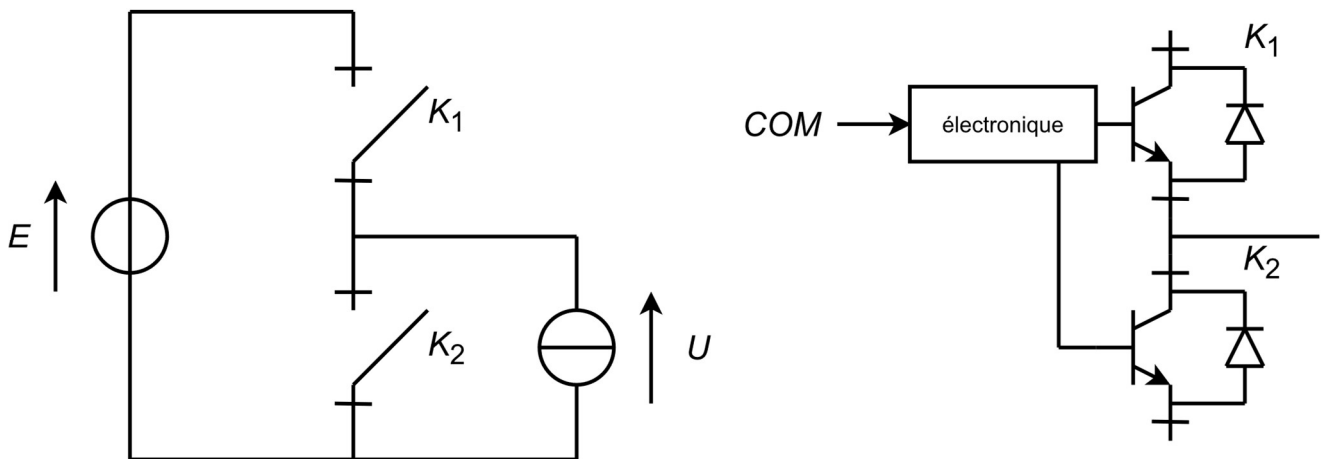


Figure 1. Demi-pont et interrupteur "transistor-diode"

Un signal logique de nom COM permet quand il est à 1 de fermer K_1 et d'ouvrir K_2 . Inversement, quand COM vaut 0, K_1 est ouvert et K_2 fermé.

- Compte tenu de l'allure temporelle du signal COM représenté **Figure 2**, représentez sur la même figure l'allure temporelle de la tension U aux bornes du récepteur. Sur cette figure, T est la période de la PWM et t_H le temps où K_1 est fermé sur cette période.
- Si $\alpha = \frac{t_H}{T}$, quelle est l'expression de U_{MOYEN} ?

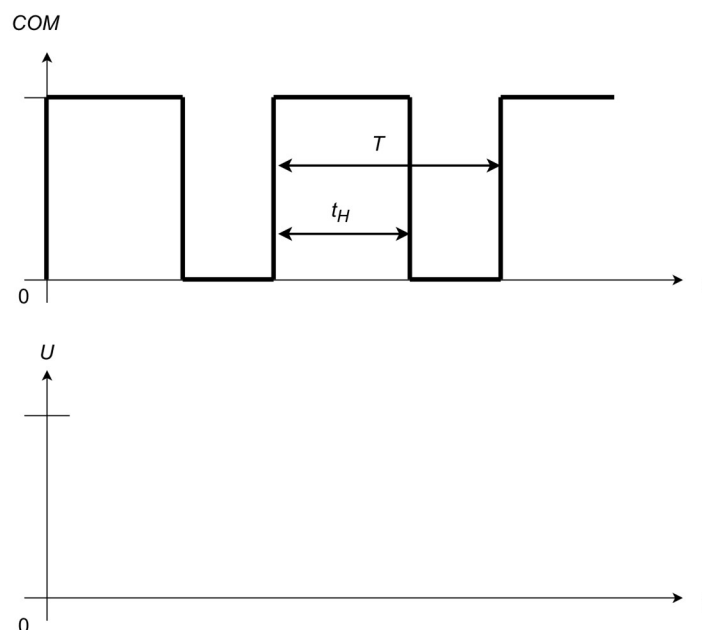


Figure 2. Signal de commande des interrupteurs et tension résultante du récepteur

2 Projet TP7A : Initialisation d'une PWM statique

2.1 Cahier des charges

Il s'agit de présenter sur la patte RC2 d'un PIC18F4520 doté d'un oscillateur à 8MHz un signal PWM de fréquence $F = \frac{1}{T} = 8 \text{ kHz}$ avec un rapport cyclique $\alpha = 40\%$.

2.2 Préparation

2.2.1 PWMs du PIC / carte EasyPIC v7

Avertissement : Nous précisons ici que, par souci de simplicité, nous avons utilisé les PWMs en mode 8 bits alors qu'il est possible de les exploiter en mode 10 bits pour obtenir une meilleure résolution temporelle.

La documentation des modules PWM est donnée dans **39631E.pdf** (*datasheet* du PIC18F4520, cf. Moodle), chapitre 15. Les modules PWM ont besoin pour fonctionner du timer 2, détaillé chapitre 13.

Sur le PIC18F4520 on peut générer deux PWM distinctes. Elles sont gérées par les modules de nom CCP1 (première PWM) et CCP2 (deuxième PWM).

« Module CCP » signifie *Capture/Compare/PWM module*. Il s'agit d'un module 3-en-1 mais seule la PWM nous intéresse ici. Dans la documentation, ne vous intéressez pas aux aspects *Capture* et *Compare*.

Dans ce TP, nous utiliserons la première PWM dont la sortie correspondante est la patte RC2. On pourrait typiquement la connecter à l'entrée de commande *COM* d'un demi-pont. Sur la carte EasyPIC v7, RC2 est utilisé pour faire varier la luminosité d'arrière-plan de l'afficheur LCD comme sur indiqué sur la **Figure 3**. Certes, ce n'est pas un demi-pont à proprement parler, vous l'aurez remarqué, mais le principe reste le même.

3. Vous vérifierez que les 8 *switches* orange de SW4 sont comme sur la figure avant de commencer.

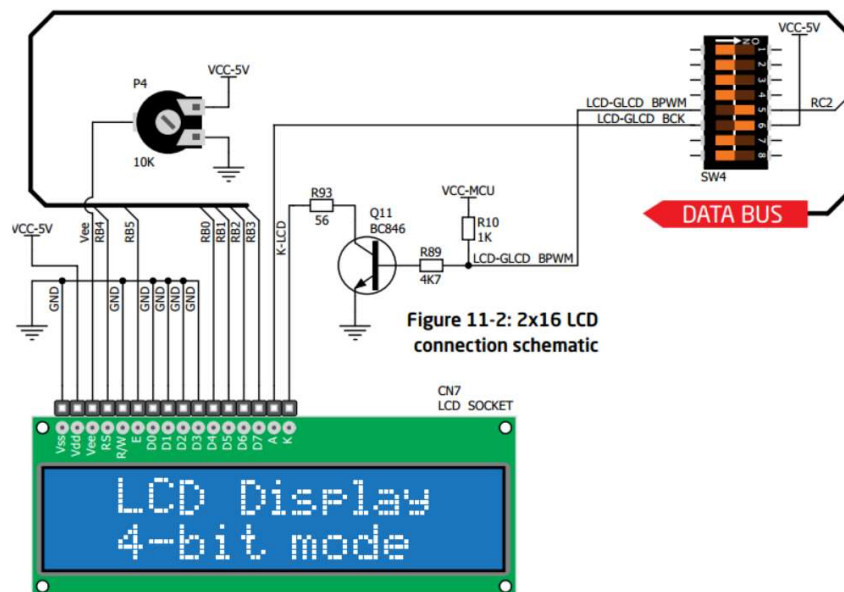


Figure 3. Réglage de la luminosité d'arrière-plan de l'afficheur LCD de la carte EasyPIC v7

2.2.2 Réglage de la période T et du rapport cyclique α

Un signal PWM peut-être caractérisé par deux grandeurs : sa période T (*PWM Period*) et son rapport cyclique α . Nous allons les initialiser dans cet ordre.

2.2.2.1 Réglage de la période T

Le réglage de T est confié au timer 2, via ses registres TMR2 et PR2.

2.2.2.1.1 Fonctionnement de base

TMR2 est la sortie d'un compteur 8 bits qui s'incrémente au rythme de l'oscillateur principal (période $4 \times T_{OSC}$). L'oscillateur est éventuellement connecté à une *prescaler* de valeur « TMR2 PreScale Value » (nous l'appellerons PS2) avant d'attaquer le compteur.

4. Il n'y a que trois valeurs possibles de PS2, quelles sont-elles ? Par quels bits de quel registre les règle-t-on ?

Dans son fonctionnement basique, le compteur du timer 2 part de 0, compte au rythme indiqué précédemment et repasse à 0 après 255.

5. La sortie du compteur peut être vue comme une dent de scie de période T . En fonction de T_{OSC} , quelle période présente-t-elle quand $PS2 = 1$? On appellera cette période T_{MAX} car on laisse compter le compteur jusqu'à son maximum de 255. Faites un graphe.
6. Donnez la valeur numérique de T_{MAX} quand $F_{OSC} = 8$ MHz (fréquence de l'oscillateur du PIC en salle de TP) et $PS2 = 1$. Quelle est la fréquence PWM F_{MIN} correspondante ?

2.2.2.1.2 Fonctionnement complet

En réalité, il est possible de faire cesser prématurément la montée de la dent de scie. Effectivement, il existe un registre 8 bits de nom PR2 tel que, quand TMR2 devient égal à PR2, alors TMR2 repasse à 0.

Ainsi, T est réglable de 0 à $(PR2 + 1) \times (4 \times T_{OSC}) \times PS2$. Dans le fonctionnement de base, on avait $PR2 = 255$. C'est d'ailleurs la valeur de PR2 au *reset*.

7. Quand $PR2 = 127$ (et $PS2 = 1$), quelle est la valeur de la période T' obtenue ? celle de F' correspondant ? Quelle est la valeur maximum de TMR2 avant de retomber à 0 ? Faites un graphe.
8. Êtes-vous d'accord avec **Equation 15.1** (page 144 de **DS39631E.pdf**) ?

EQUATION 15-1:

$$\text{PWM Period} = [(PR2) + 1] \cdot 4 \cdot T_{OSC} \cdot (\text{TMR2 Prescale Value})$$

9. Quelle valeur de PR2 retenir pour obtenir la fréquence PWM du cahier des charges (on retiendra $PS2 = 1$) ? Donnez alors la valeur du registre T2CON. Faites un graphe.

2.2.2.2 Réglage de α

Le niveau de la dent de scie obtenue est comparé à la valeur d'un registre de nom CCPR1L. Quand CCPR1L est supérieur ou égal à TMR2 la sortie PWM est à 1 et à 0 sinon.

10. Quelle est la valeur de CCPR1L nécessaire pour obtenir le α du cahier des charges ? Complétez votre graphe précédent avec la droite horizontale d'équation $y = CCPR1L$ et l'état de la patte RC2, c'est-à-dire le signal de commande *COM* du demi-pont.
11. Quelle valeur limite de CCPR1L amène un α égal à 100% ?

2.3 Validation du travail

12. Écrivez un court programme réalisant le cahier des charges. La boucle **while (1)** est vide. Vous pouvez vous aider de la rubrique 15.4.4 SETUP FOR PWM OPERATION, recopié **Figure 4** :

15.4.4 SETUP FOR PWM OPERATION

The following steps should be taken when configuring the CCP module for PWM operation:

1. Set the PWM period by writing to the PR2 register.
2. Set the PWM duty cycle by writing to the CCPRxL register ~~and CCPxCON <5:4> bits.~~
3. Make the CCPx pin an output by clearing the appropriate TRIS bit.
4. Set the TMR2 prescale value, then enable Timer2 by writing to T2CON.
5. Configure the CCPx module for PWM operation.

Figure 4. Mise en œuvre de la PWM

13. Vérifiez à l'oscilloscope que RC2 présente bien la fréquence et le rapport cyclique demandés dans le cahier des charges.

3 Projet TP7B : Réglage du rapport cyclique

3.1 Cahier des charges

La PWM doit avoir les mêmes valeurs initiales que précédemment.

En outre, quand on appuie sur le bouton RE0, CCPR1L est incrémenté sans toutefois dépasser sa valeur limite. Quand on appuie sur le bouton RE1, CCPR1L est décrémenté sans toutefois passer sous 0.

3.2 Préparation

14. N'oubliez pas que RE0 et RE1 sont des entrées analogiques au *reset* et qu'il faut les rendre numériques pour s'en servir comme entrée Tout ou Rien.

3.3 Validation du travail

15. Vérifiez que le cahier des charges est respecté. Vous pouvez visualiser les signaux à l'oscilloscope et utiliser un terminal série.

4 Culture générale

16. Avec $PS2 = 1$, que devient PR2 si on veut que la fréquence de la PWM devienne égale à 40 kHz ?

17. Quelle est alors la valeur maximum que l'on peut donner à CCPR1L ? Quel désavantage y voyez-vous par rapport aux situations des deux cahiers des charges précédemment traités ?

18. Que vaut CCPR1L pour un α de 80% ?

19. Quel intérêt y'a-t-il à augmenter la fréquence de la PWM ? Qu'est-ce qui limite cette augmentation ?