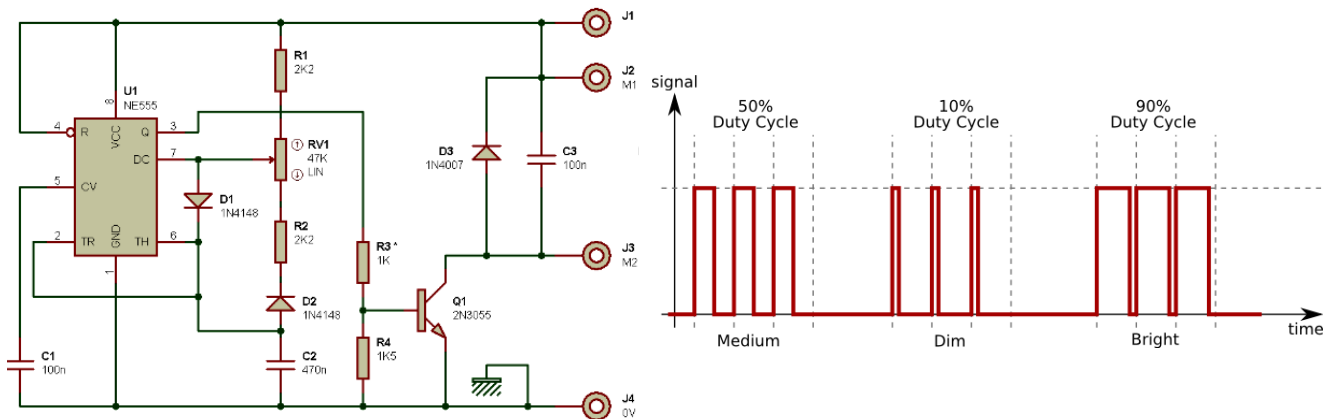


MODULATION À LARGEUR D'IMPULSIONS



Objectifs du cours :

Ce cours traitera essentiellement les points suivants :

- variation de vitesse d'un moteur DC
- PWM :
 - définition
 - intérêt
 - fréquence
 - rapport cyclique
 - obtention du signal

VARIATION DE VITESSE D'UN MOTEUR DC

Quand on veut faire varier la vitesse d'un moteur, la première idée qui vient à l'esprit est de faire varier la tension aux bornes du moteur mais on constate que pour des valeurs faibles de la tension, le moteur ne tourne pas. Le moteur demande une tension minimale pour démarrer. Si cette dernière est trop basse, les forces électromagnétiques ne sont pas suffisantes pour vaincre le frottement. Il devient donc difficile d'ajuster la vitesse de façon précise.

La solution à ce problème est astucieuse. Il suffit de fournir au moteur une tension qui est toujours la même soit la tension maximale ! Par contre, cette tension ne sera appliquée que par très courtes périodes de temps. En ajustant la longueur de ces périodes de temps, on arrive à faire tourner plus ou moins vite les moteurs. Mieux, on remarque que la vitesse des moteurs devient proportionnelle à la longueur des périodes passées à la tension maximale par rapport au temps passé sans application de tension (tension nulle) est

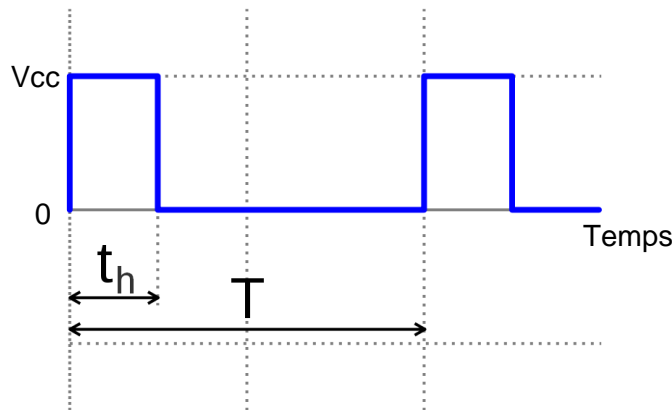
donc le cœur de la solution. En réalité, cette solution est très connue en contrôle des systèmes et en électronique et elle porte le nom de **PWM (Pulse Width Modulation)** ou **Modulation par Largeur d'Impulsions (MLI)**.

PWM (MLI)

DÉFINITION

Le PWM est un signal numérique, donc la tension peut prendre deux valeurs seulement. Dans certains cas très spécifiques (onduleurs à MLI par exemple) on fabrique un troisième niveau en inversant la tension du niveau haut.

Le signal est carré. Le niveau bas correspond généralement à 0 Volt. La période est notée **T** ; la durée de l'impulsion (pour laquelle la tension est celle de l'état haut) est appelée **t_h**.



Si la période change, le signal n'est plus vraiment périodique au sens strict. On appelle alors T **la pseudo période**.

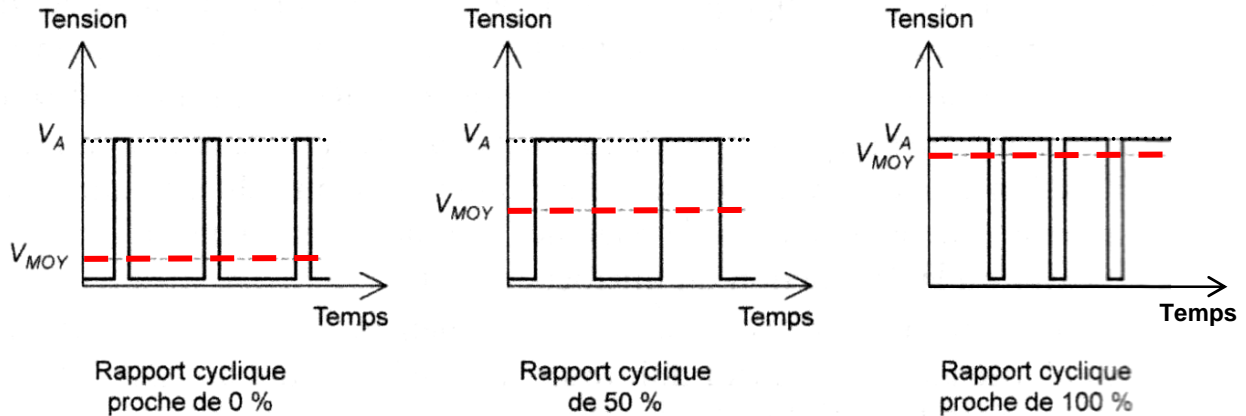
INTÉRÊT

Le principal intérêt de la technique PWM est de limiter la chauffe des composants électroniques. Par exemple : une lampe de 20 Watts allumée au maximum consomme 20 W. Si par une commande de gradation elle est allumée au quart de sa puissance, elle consomme 5 W. Le composant analogique devrait alors dissiper 15 W, ce qui implique un énorme radiateur. En PWM, la puissance fournie est soit maximale, soit nulle. Lorsqu'elle est maximale, pendant un quart du temps par exemple, il n'y a pas besoin de dissiper de puissance résiduelle. Lorsqu'elle est nulle, il n'y a pas besoin de dissiper non plus de puissance, car elle n'est pas fournie du tout. Un autre intérêt du PWM est que la tension appliquée au moteur pendant t_h est Vcc. Celle-ci est suffisante pour vaincre les frottements et faire tourner le moteur. La tension moyenne appliquée au moteur est proportionnelle au rapport cyclique, ce qui permet d'avoir des consignes de vitesse faibles :

$$V_{\text{MOY}} = \frac{t_h \times V_{\text{CC}}}{T}$$

FRÉQUENCE

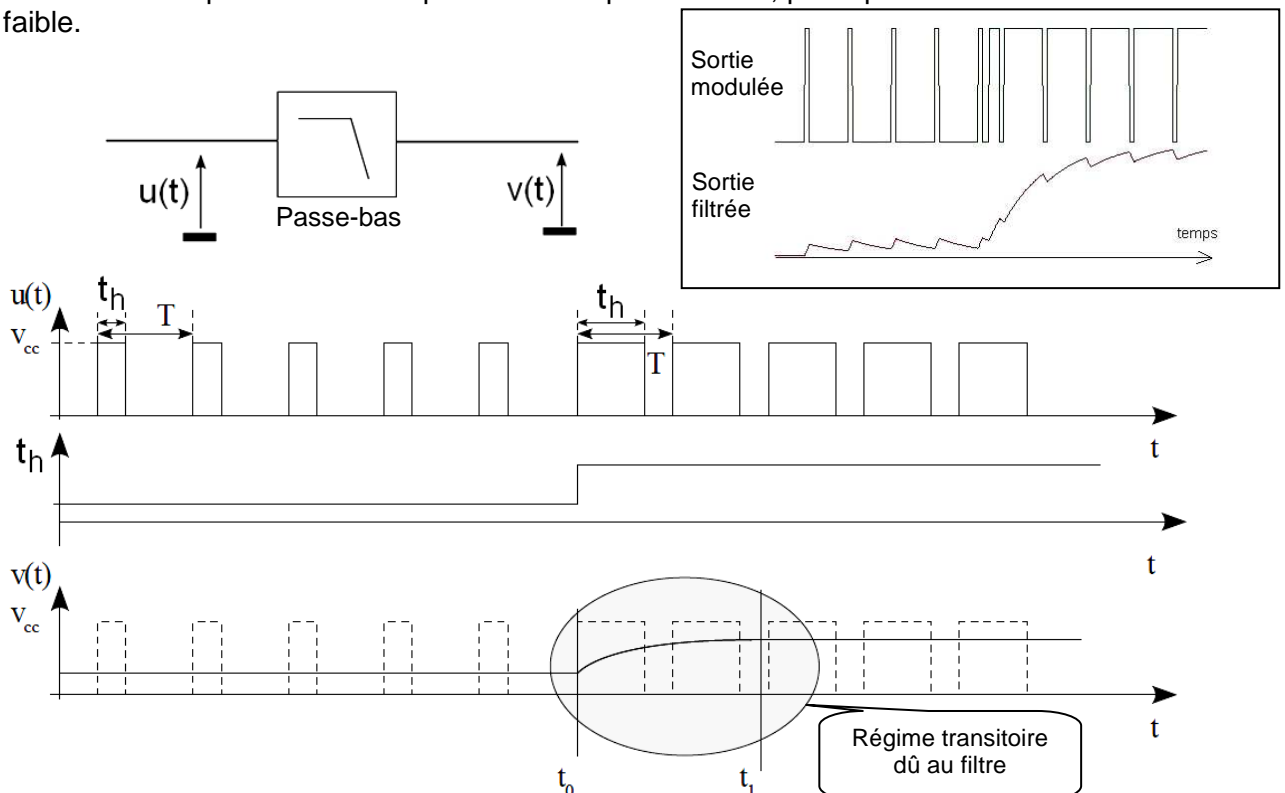
La commande d'actionneurs de puissance par PWM est très liée à la notion de fréquence. Pour que l'impression d'une valeur moyenne constante d'allumage apparaisse, il faut que l'alternance d'allumage/extinction soit suffisamment rapide pour qu'elle ne se remarque pas. Selon les utilisations la fréquence du PWM va de 100 Hz (100 cycles par seconde) à 200 kHz.



Remarques :

Même si le signal n'est nullement une tension analogique, il peut le devenir si on lui fait subir un filtrage de type passe-bas.

Filtrer ce signal carré permet d'obtenir la tension moyenne. La fréquence de coupure du filtre doit être plus faible que la fréquence du signal carré. Autrement dit la fréquence porteuse doit être significativement supérieure à la fréquence de coupure du filtre, pour que l'ondulation résiduelle soit faible.



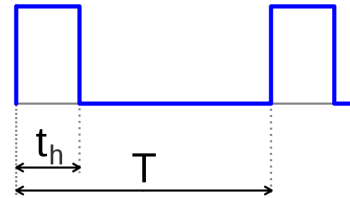
RAPPORT CYCLIQUE

On appelle rapport cyclique le rapport :

$$\alpha = 100 \times \frac{t_h}{T} \quad \text{exprimé en pourcentage.}$$

Si $t_h=0$ alors $\alpha=0\%$ et la tension moyenne de sortie est nulle.

Si $t_h=T$ alors $\alpha=100\%$ et la tension moyenne de sortie est égale à V_{cc} .

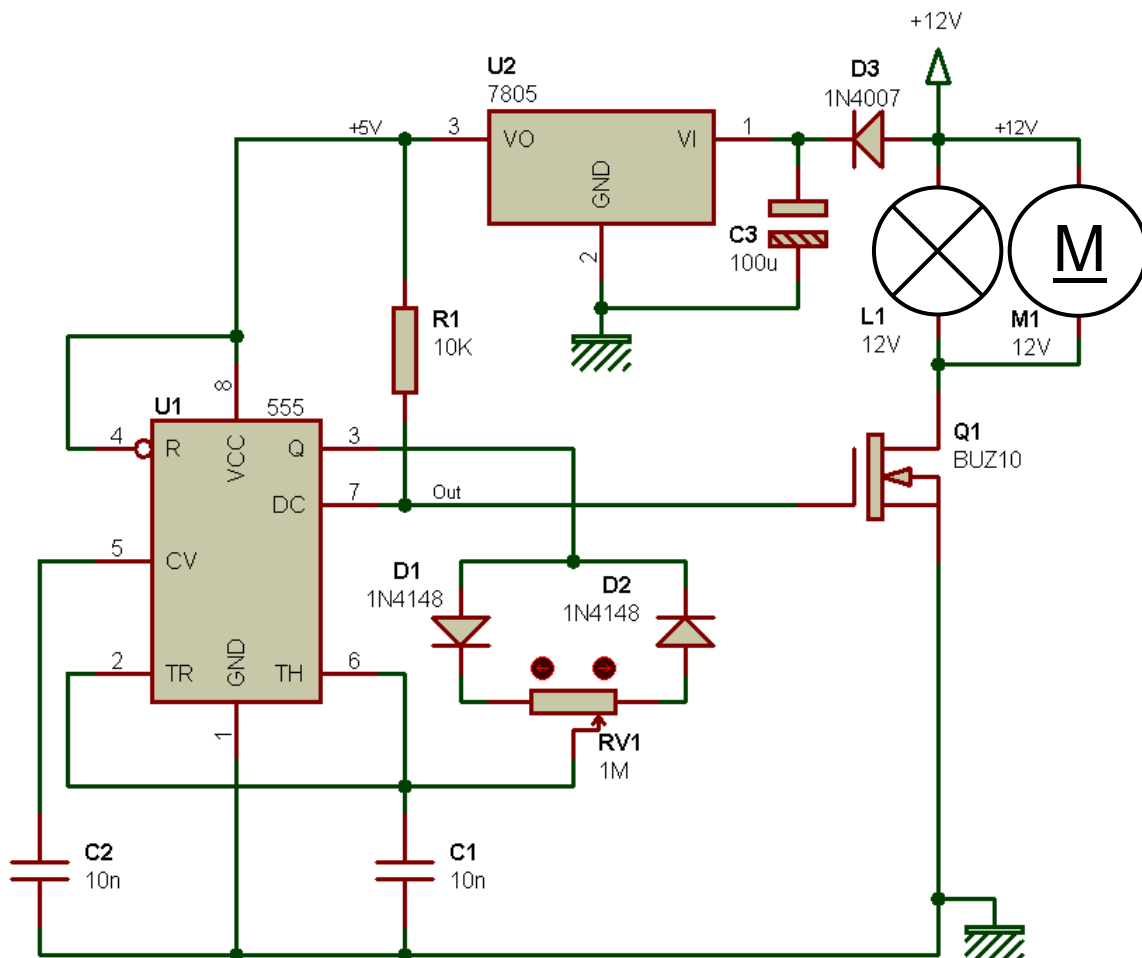


OBTENTION DU SIGNAL

Plusieurs possibilités existent pour obtenir un signal PWM. Un montage avec le CI NE555, ou une logique programmée, peuvent être utilisés.

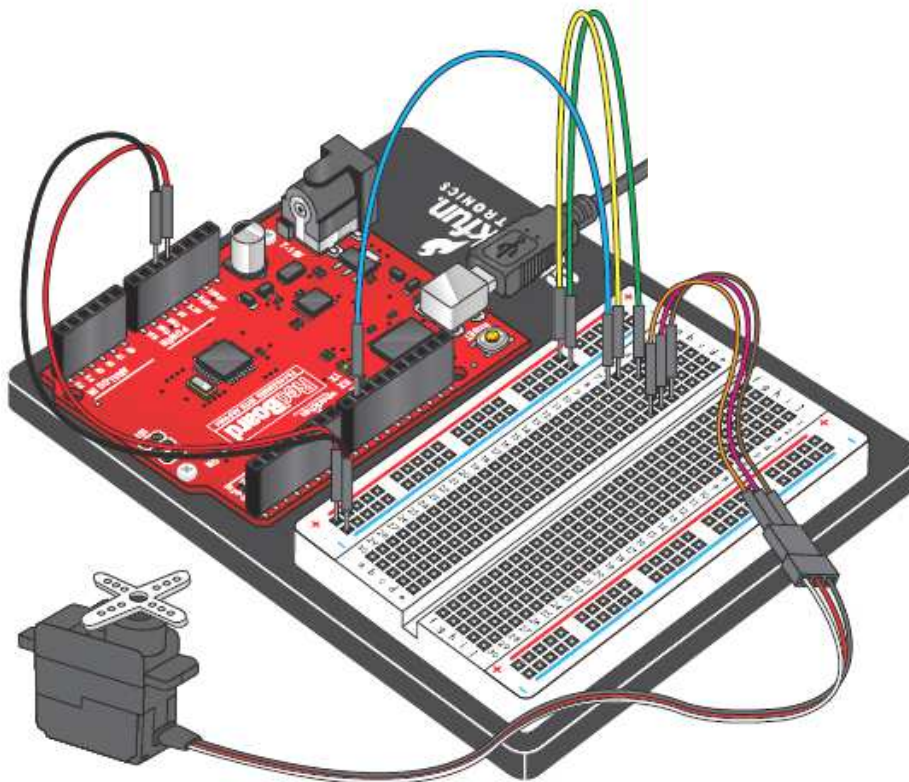
Deux exemples avec le NE555 et la carte RedBoard (Arduino).

PWM/MLI avec le NE555



Le NE555 (U1) est monté en multivibrateur et délivre un signal de fréquence fixe avec un rapport cyclique variable. La fréquence de base est fixée par la valeur du potentiomètre RV1 et du condensateur C1. Avec les valeurs du schéma, la fréquence est de l'ordre de 120 Hz. Descendre la valeur de C1 à 1 nF permet de monter la fréquence à 1,2 KHz. Le rapport cyclique est directement fonction de la position du curseur du potentiomètre RV1, couplé aux deux diodes D1 et D2 qui permettent de bien séparer les cycles de charge et de décharge du condensateur C1. Quand le curseur de RV1 est en position centrale, les cycles de charge et de décharge de C1 prennent autant de temps et le rapport cyclique est de 50 %. Si le curseur de RV1 est du côté de D1, la charge de C1 est plus rapide et sa décharge est plus lente, ce qui conduit à un rapport cyclique faible (inférieur à 50 %), le MOSFET reste plus longtemps bloqué qu'il ne reste passant (la lampe s'éclaire faiblement et/ou le moteur tourne lentement). Si le curseur de RV1 est du côté de D2, la charge de C1 est plus lente et sa décharge est plus rapide, ce qui conduit à un rapport cyclique élevé (supérieur à 50 %), le MOSFET reste plus longtemps passant que bloqué (la lampe s'éclaire fortement et/ou le moteur tourne vite).

PWM/MLI avec la RedBoard (Arduino)



Un petit servomoteur, consommant moins de 40 mA, peut être utilisé sans interface de puissance. Voici par exemple les morceaux de code qu'il faudra inclure dans votre sketch :

```
#include <Servo.h> // bibliothèque permettant de commander un servomoteur

Servo servo1 ; // crée un objet servo1 de la classe Servo
                // (un par servomoteur à commander)

servo1.attach(9) ; // indique quelle broche sera utilisée par servo1

servo1.write(30) ; // fera pivoter le servo1 jusqu'à la valeur 30°
```