




Fiche guide	TS SI		P.P.E. Robot suiveur de ligne	
Mise en œuvre 3				
	Etude de la fonction « Traiter » du robot « Mr LineTiny » et commande des moteurs avec un signal modulé en largeur d'impulsion (MLI)			
Nom(s) :		Classe :	Groupe :	

Objectifs

Comprendre comment la fonction traiter effectue le réglage de la fréquence de rotation des moteurs.
Proposer un graphe des transitions pour la fonction traiter du robot à réaliser.

Matériel

Robot suiveur de ligne « Mr LineTiny ». Oscilloscope 2 voies + sondes. Piste.

Documentation

Dossier technique.

Sur le site WebGE à l'adresse <http://p.mariano.free.fr/> (rubrique PPE)
Le présent document et le dossier technique.

A) Présentation du principe de réglage de la fréquence de rotation des moteurs

Chacune des roues du robot suiveur de ligne est équipée d'un moteur à courant continu. La vitesse de chacun des moteurs peut être réglée indépendamment de l'autre.

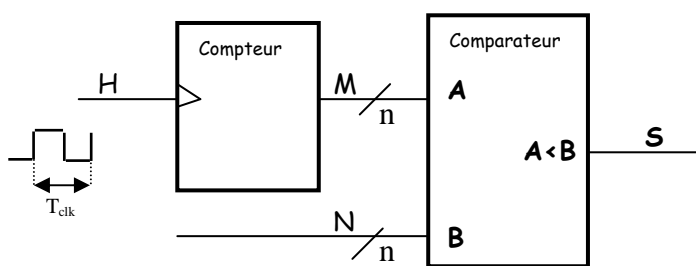
• Présentation

Pour faire varier la vitesse de rotation d'un moteur à courant continu, il suffit de modifier la valeur moyenne de la d.d.p. présente à ses bornes en le commandant par un signal dit Modulé en Largeur d'Impulsion « **MLI (ou PWM)** ».

Génération d'un signal Modulé en Largeur d'Impulsion (principe)

Un signal modulé en largeur d'impulsion peut être obtenu à partir d'un signal périodique H de fréquence fixe $F_{clk} = 1/T_{clk}$. En effet, en appliquant ce signal à l'entrée d'un compteur, on obtient un signal numérique M (codés sur n bits) capable d'évoluer entre 0 et $2^n - 1$. La représentation de $M(t)$ est appelée **rampe numérique**. En appliquant $M(t)$ et un signal constant $N(t)$ (codé sur n bits) à un comparateur numérique, on obtient un signal binaire $S(t)$ de période $T = (2^n - 1) \cdot T_{clk}$ dont le temps t_1 à l'état « 1 » est réglé avec la valeur de N . On appelle $\alpha = t_1/T$ le rapport cyclique du signal $S(t)$. On montre que la valeur moyenne S_{moy} de $S(t)$ est égale au produit de α par S_{maxi} .

On donne ci-dessous le schéma de principe d'une structure générant un signal M.L.I et les chronogrammes de $S(t)$ pour deux valeurs particulières de N .

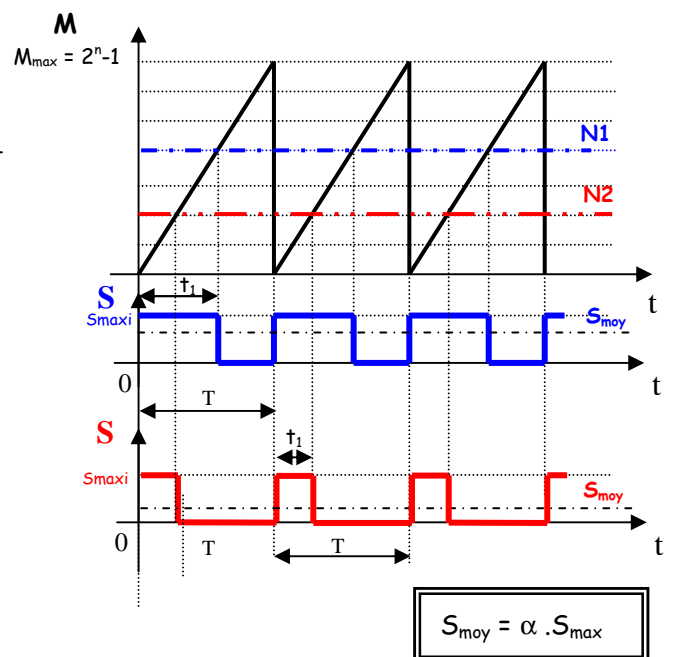


M = valeur numérique variable ($0 \leq M \leq 2^n - 1$)

N = valeur numérique constante

Comparateur : si $A < B$ alors $S = S_{maxi}$
sinon $S = 0$

H : signal périodique de fréquence F_{clk}



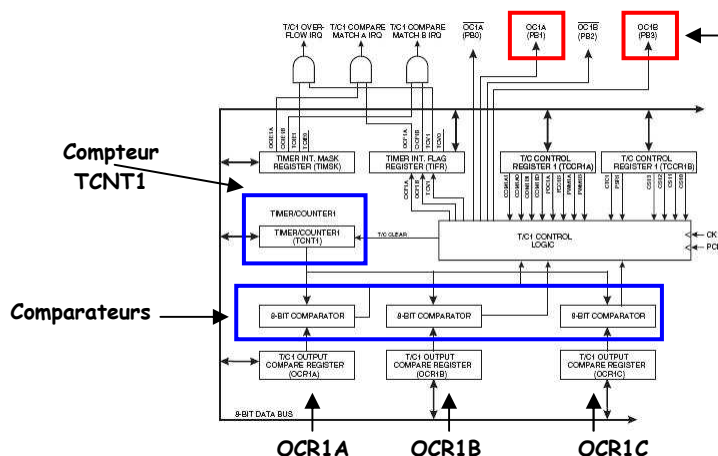
Dans les microcontrôleurs, les signaux modulés en largeur d'impulsion sont générés par une structure appelée **TIMER**. Celle-ci répond au principe développé ci-dessus.

*M.L.I : Modulation de largeur d'impulsion (P.W.M. : Pulse With Modulation)

- **Génération d'un signal MLI avec le microcontrôleur ATINY26**

Le **Timer1** de l'ATINY26 permet de générer quatre signaux modulés en largeur d'impulsion.

Il intègre un **compteur**, des **comparateurs** et divers registres. En mode M.L.I. son fonctionnement répond au principe exposé dans le paragraphe précédent.



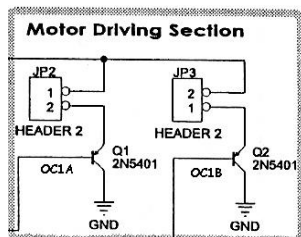
Résultats des comparaisons

Le compteur TCNT1 génère le signal numérique M. Les registres OCR1A et OCR1B correspondent à N. OCR1C permet de régler M_{max} entre 0 et $2^n - 1$ ($n=8$). OC1A et OC1B correspondent à S. Le Timer 1 contient donc deux structures dont le fonctionnement répond au principe exposé dans le paragraphe précédent.

- **Commande des moteurs des robots**

Les signaux de commande des moteurs des robots sont issus des broches PB1 (OC1A) et PB3 (OC1B) du microcontrôleur. Ces signaux commandent des transistors (Q1, Q2).

PDIP/SOIC	
(MOSI/DO/SDA/OC1A) PB0	1
(MISO/DO/OC1A) PB1	2
(SCK/SCL/OC1B) PB2	3
(OC1B) PB3	4
VCC	5
GND	6
(ADC7/XTAL1) PB4	7
(ADC8/XTAL2) PB5	8
(ADC9/INT0/T0) PB6	9
(ADC10/RESET) PB7	10
PA0 (ADC0)	20
PA1 (ADC1)	19
PA2 (ADC2)	18
PA3 (AREF)	17
GND	16
AVCC	15
PA4 (ADC3)	14
PA5 (ADC4)	13
PA6 (ADC5/AIN0)	12
PA7 (ADC6/AIN1)	11



Les transistors inversent les signaux de commandes OC1A et OC1B.

Les moteurs sont reliés au circuit des transistors Q1 et Q2 par l'intermédiaire des connecteurs P2 et JP3.

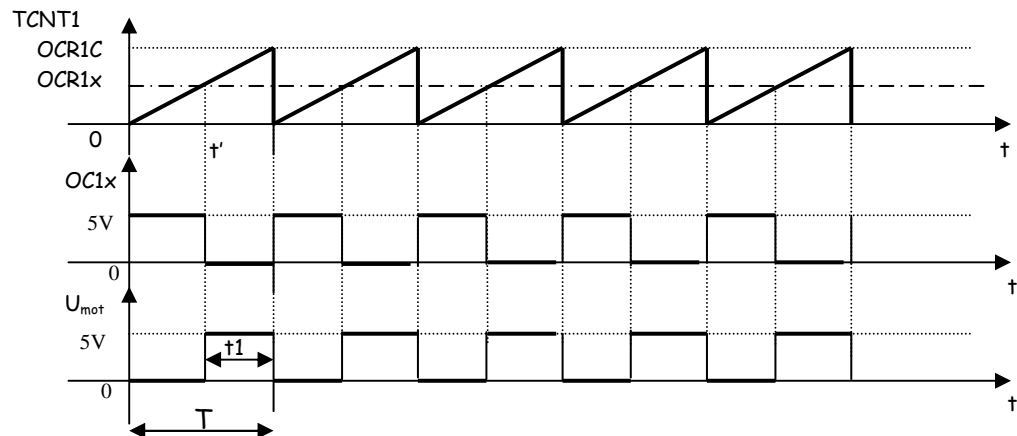
La modification du rapport cyclique α des signaux de commande des moteurs du robot se fait en modifiant les valeurs contenues dans les registres OCR1A et OCR1B.

B) Analyse de la solution existante

B1) Activité 1 : Détermination de l'expression de α en fonction de OCR1x et OCR1C

Objectif : Déterminer les valeurs à placer dans les registres OCR1x pour régler la valeur du rapport cyclique α . Observer le comportement du robot pour les différentes valeurs de α .

On donne les chronogrammes ci-dessous :



Première possibilité pour réaliser ce travail : travail non guidé

Q1) Déterminez $\alpha = t_1/T$ en fonction de OCR1x et OCR1C avec $x = A$ ou B par la méthode de votre choix.

Complétez le tableau ci-dessous pour OCR1C = 255.

$\alpha(\%)$	10	20	30	40	50	60	70	80	90
OCR1x									

Deuxième possibilité pour réaliser ce travail : travail guidé

Q1a) Exprimez $TCNT1 = f(t)$ pour $t \in [0, T]$

Q1b) Exprimez $t_1 = f(T, t')$ (1)

Q1c) A $t = t'$, $TCNT1 = OCR1x$, exprimez $t' = f(OCR1C, OCR1x)$ (2)

Q1d) Exprimez $\alpha = f(OCR1C, OCR1x)$ à partir des expressions (1) et (2)

Q1e) Complétez le tableau ci-dessous si OCR1C = 255 (arrondissez à l'entier supérieur)

$\alpha(\%)$	10	20	30	40	50	60	70	80	90
OCR1x									

Expérimentation : Observation du comportement du robot pour les différentes valeurs de α

Ouvrez le projet « 2_CVAVR_robot\robot3\testPWM.prj. Complétez le fichier avec les valeurs de α et testez-le sur le robot.

ATTENTION : Lors de la programmation, les cavaliers doivent être placés sur « programmation » comme sur l'annexe 1 du dossier technique.

Vous effectuerez les tests suivant :

- (1) visualisation des signaux OC1A et OC1B sur un oscilloscope (à développer avec TP)
- (2) observation de la vitesse du robot pour les différentes valeurs de α .

B2) Activité 2 : élaboration de l'algorithme de commande des moteurs

Objectif : Comprendre le graphe des transitions proposé et ajuster la commande des moteurs pour que le robot suive la ligne le plus rapidement possible.

La fonction traiter est présentée dans le dossier technique.

Ouvrez le projet « 2_CVAVR_robot\robot3\StayOnLine.prj. Modifiez les valeurs des variables CDE_ROUE_D et CDE_ROUE_G et testez le programme sur le robot.

C) CONCEPTION

Proposez le graphe des transitions du robot à réaliser.