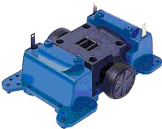




Fiche guide 4	TS SI		P.P.E. Robot suiveur de ligne	
Analyse et synthèse	4h			
 Lycée Polyvalent PIERRE EMILE MARTIN	Analyse et synthèse de la fonction « Traiter »			

Nom(s) :	Classe :	Groupe :
----------	----------	----------

Objectifs :

Calculer les coefficients nécessaires au réglage de la fréquence de rotation des roues du robot à réaliser.

Etablir le graphe des transitions représentatif de la fonction « Traiter » du robot à réaliser.

Etablir l'algorithme de calcul de la variable VisuPosition nécessaire à la fonction « Communiquer » du robot à réaliser.

Matériel

Robot suiveur de ligne MrLineTiny. Oscilloscope 2 voies + sondes. Piste.


Logiciel

CodeVision AVR. Répertoire du projet Stay_On_LineV3E.

Documentation

Dossier technique « Robot suiveur de ligne ».

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

 : Dossier technique « Robot suiveur de ligne ».

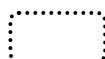
Rappel des conventions utilisées dans les schémas fonctionnels



Fonction matérielle



Energie



Fonction logicielle



Signal porteur d'une information



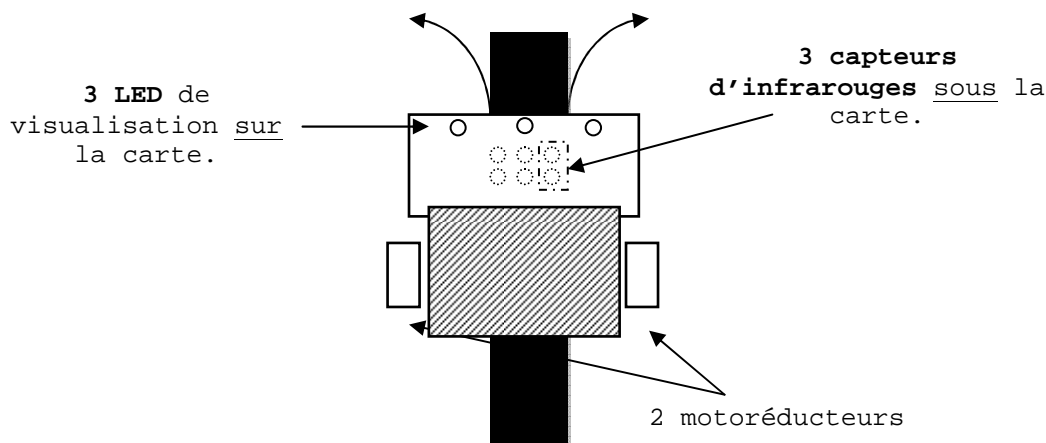
Variable logicielle

A) Mise en situation

Rappels sur le principe de suivi de la ligne adopté par « M^r LineTiny »

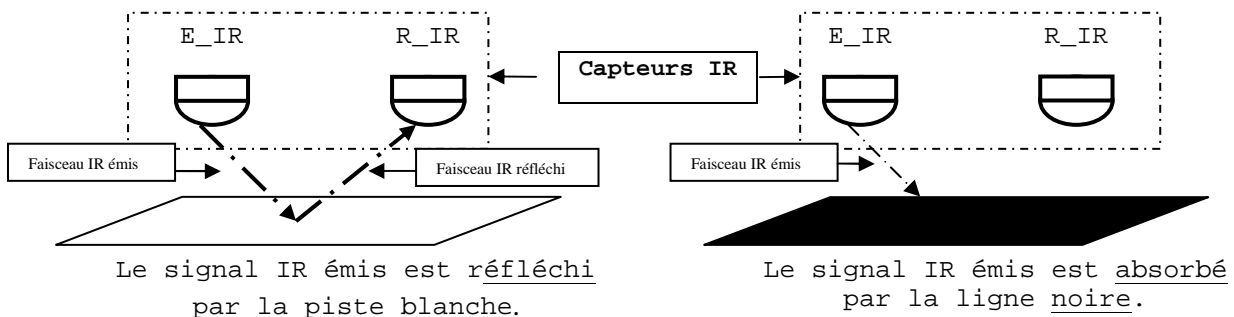
Le robot doit être capable de suivre une ligne noire de 30mm de large tracée sur un fond blanc.

Deux **motoréducteurs** assurent la propulsion et la direction du robot.



Le robot détecte sa position par rapport à la ligne noire tracée sur la piste grâce à **trois capteurs** d'infrarouges (**IR**). Chaque capteur se compose d'un émetteur **E_IR** (diode IR) et d'un récepteur **R_IR** (phototransistor IR).

Le principe de détection de la ligne noire est illustré ci-dessous :

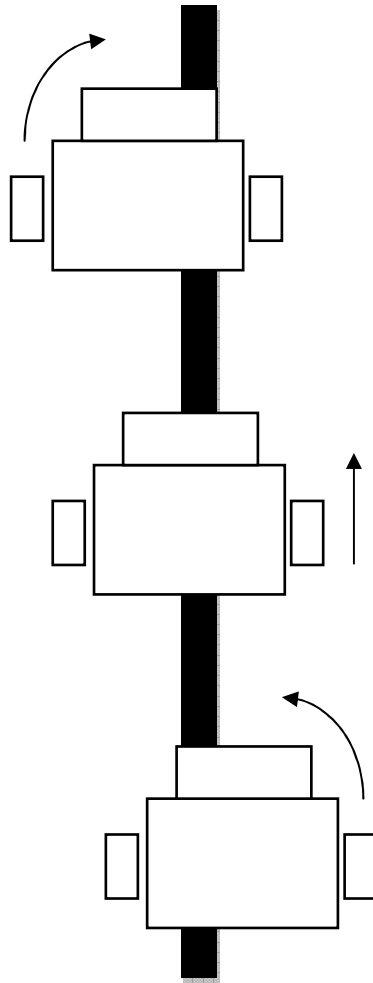


Les informations délivrées par les capteurs sont représentatives de la **position du robot** par rapport à la ligne. Elles sont **traitées** par un programme implanté dans un **microcontrôleur**.

Grâce à une structure appelée « **Timer** », le microcontrôleur adapte la fréquence de rotation de chacun des **motoréducteurs** associés aux roues.

Ceci à pour effet de corriger la trajectoire du robot afin de le maintenir sur la ligne noire.

Robot à gauche =>
 Ralentir moteur droit
 (ou / et)
 Accélérer moteur gauche



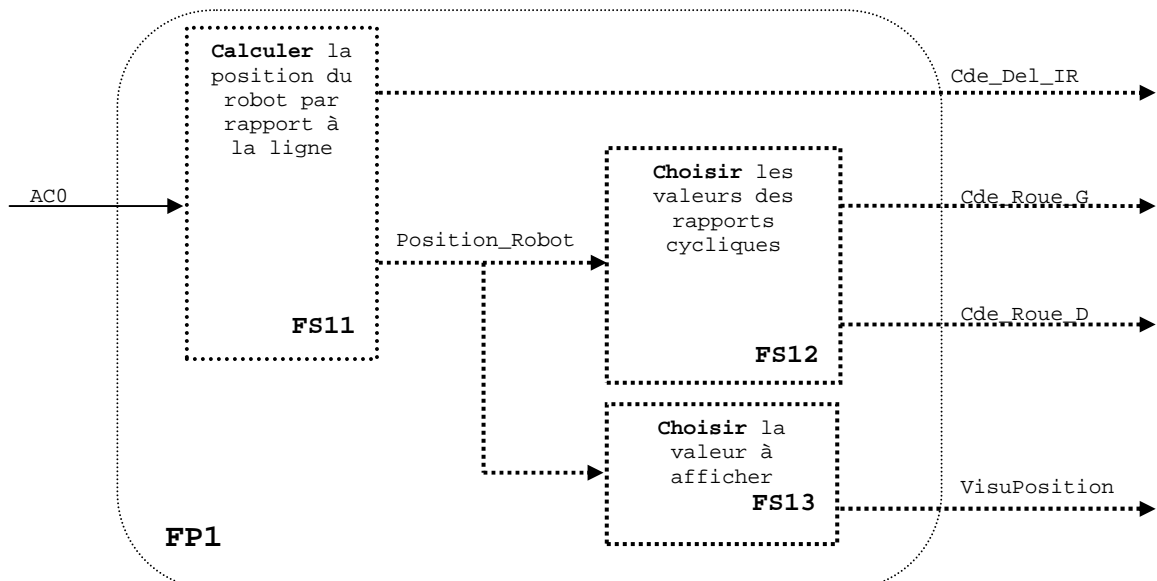
Robot au centre =>
 Commander les moteurs pour obtenir la même fréquence de rotation

Robot à droite =>
 Ralentir moteur gauche
 (ou / et)
 Accélérer moteur droit

B) Analyse de la fonction FP1 « Traiter l'information... »


B1) Présentation

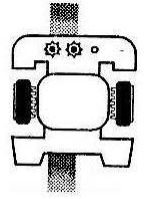
La fonction FP1 « Traiter l'information... » est une fonction logicielle. Elle est décomposée en trois fonctions secondaires FS11, FS12 et FS13. (Voir p6, 7, 10 et 11 du).



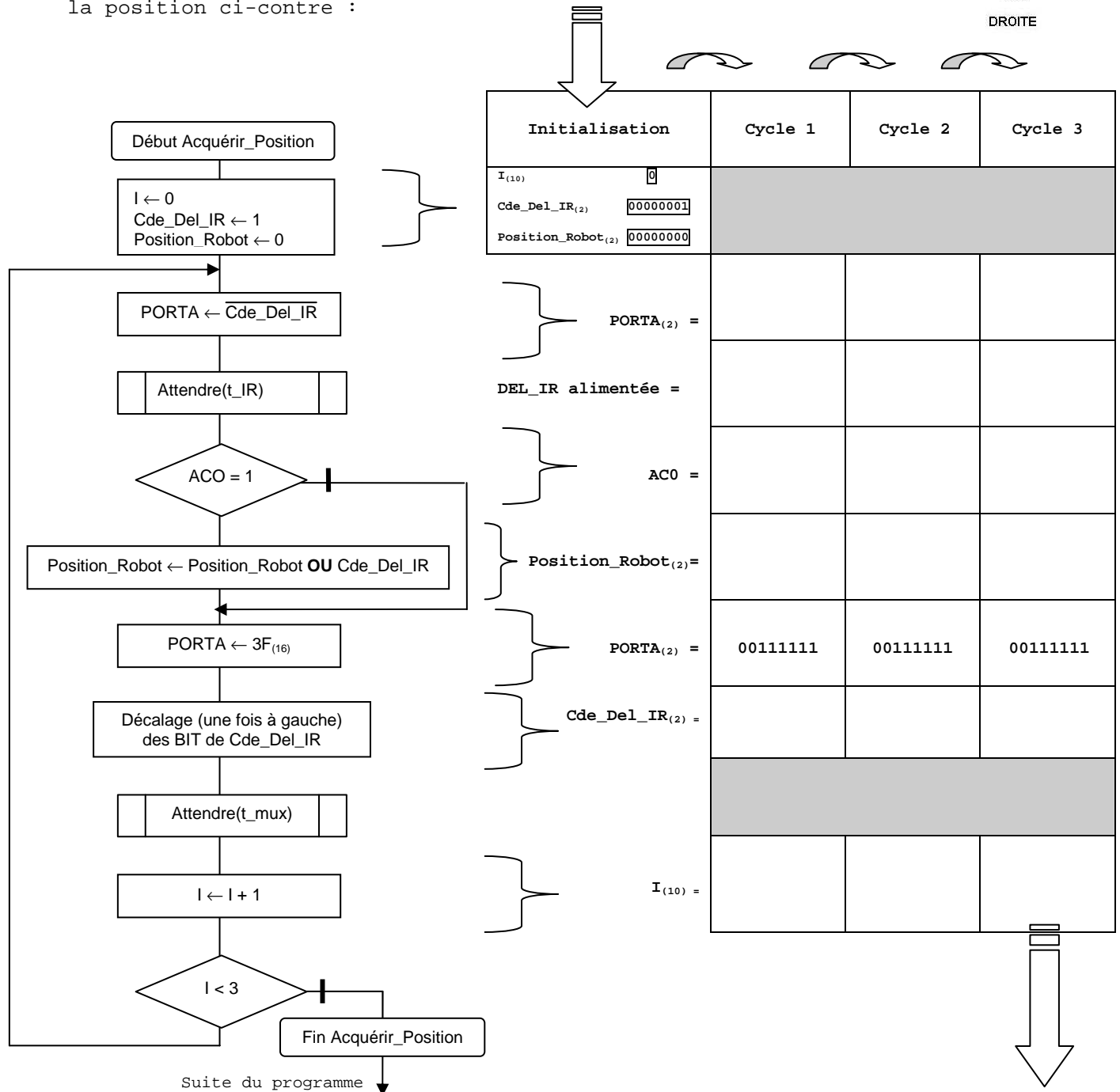
B2) Analyse de FS11 « Calculer la position du robot par rapport à la ligne »

Objectif : Adapter l'algorithme de calcul de la position du robot par rapport à la ligne pour le robot à réaliser.


L'analyse ci-dessous doit être faite avec le **schéma structurel de l'annexe 1** du présent document. La description des variables est fait dans le .



Q1) Complétez le tableau ci-dessous sachant que le robot est dans la position ci-contre :



Q2) Complétez le texte ci-dessous.

A la fin de l'acquisition, la variable Position_Robot₍₁₀₎ = _____, la fonction « Traiter » sait que le robot est dans la position _____. Vérifiez ce résultat en le comparant aux informations données dans le  (Fonction « Traiter »).

B3) Analyse de FS12 « Choisir les valeurs des rapports cycliques »

Objectif : Calculer les coefficients nécessaires au réglage de la fréquence de rotation des roues du robot à réaliser.

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

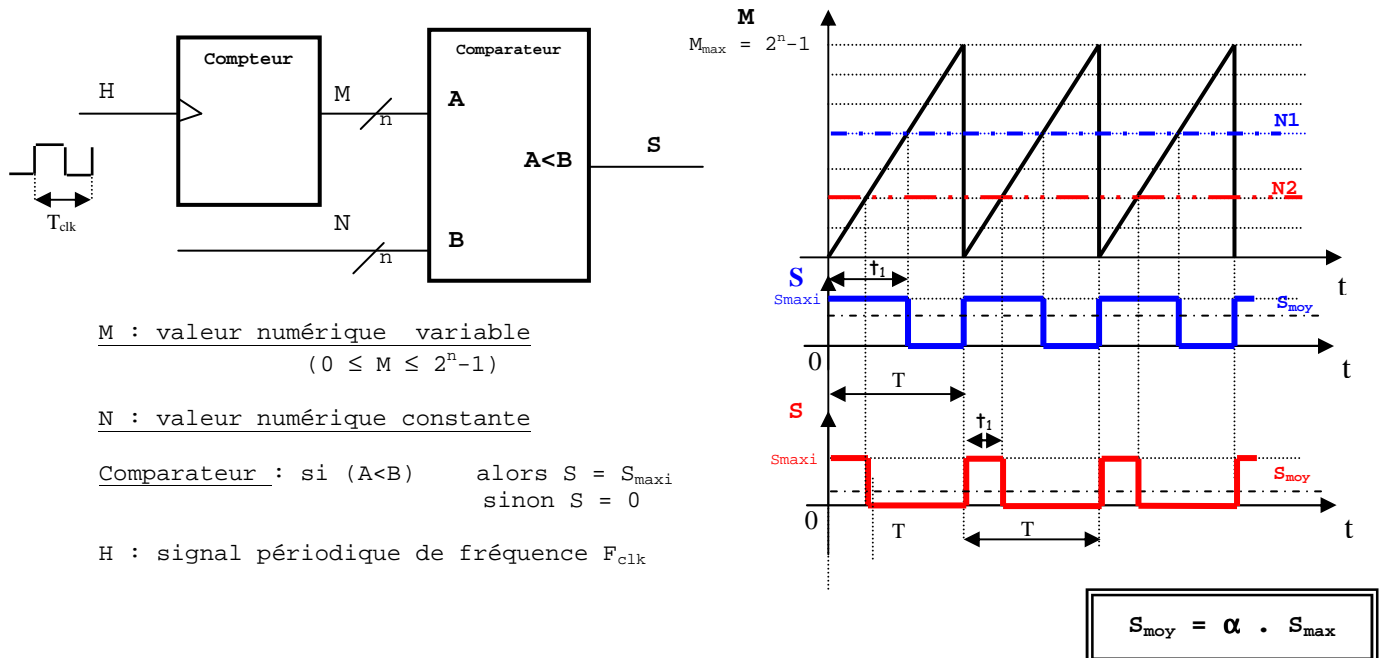


B31) Principe de réglage de la fréquence de rotation des moteurs

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 .



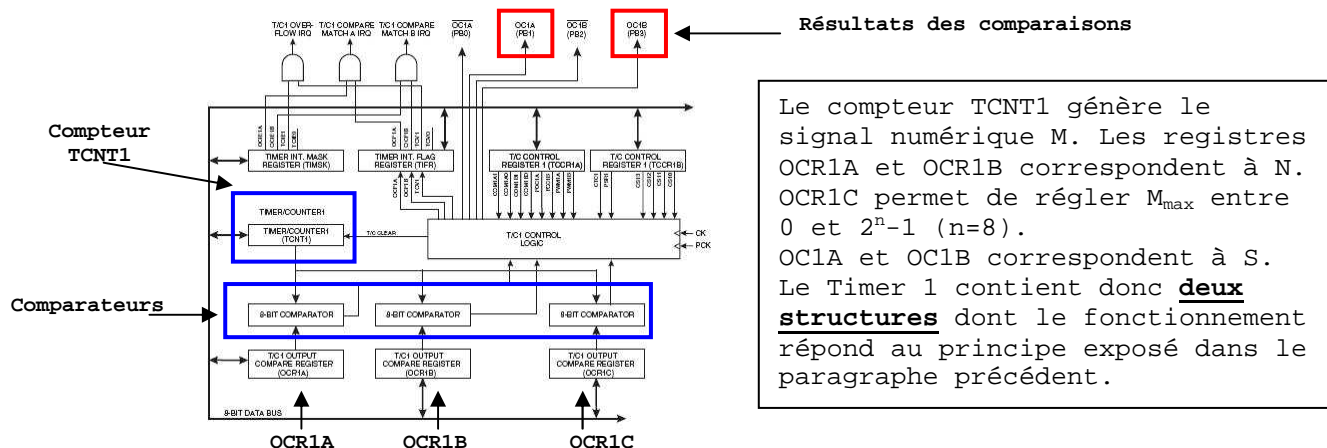
Dans les microcontrôleurs, les signaux modulés en largeur d'impulsion sont générés par une structure appelée **TIMER**. Celle-ci répond au principe développé ci-dessus.

• 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.

Schéma fonctionnel du Timer 1 (documentation ATMEL)



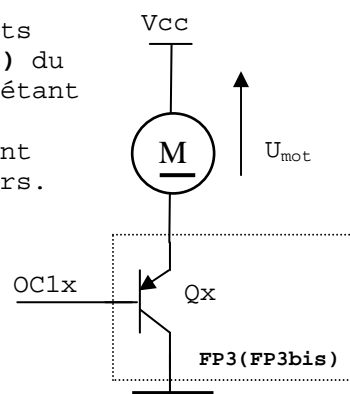
La modification du rapport cyclique α des signaux de commande des moteurs du robot se fait en modifiant les valeurs contenues dans les registres OCR1A et OCR1B.

• Complément : Présentation des fonctions FP3(FP3bis) « Distribuer »

PDIP/SOIC	
(MOSI/SDA/OC1A) PB0	1
(MISO/DO/OC1A) PB1	2
(SCK/SCLO/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
PB0 (ADC0)	20
PA1 (ADC1)	19
PA2 (ADC2)	18
PA3 (AREF)	17
AVCC	15
PA4 (ADC3)	14
PA5 (ADC4)	13
PA6 (ADC5/AIN0)	12
PA7 (ADC6/AIN1)	11

Les signaux de commande des moteurs des robots sont issus des broches **PB1(OC1A)** et **PB3(OC1B)** du microcontrôleur. **L'énergie** de ces signaux n'étant pas suffisante pour entrainer les moteurs à courant continu du robot, des transistors sont placés entre le microcontrôleur et les moteurs. (Voir p13)

Les moteurs sont reliés au circuit des transistors Q1 et Q2 par l'intermédiaire des connecteurs JP2 et JP3. (Annexe 1 du)



Pour calculer les valeurs numériques à placer dans les registres OCR1A et OCR1B, il faut établir une relation entre le rapport cyclique α souhaité, OCR1x ($x = A$ ou B) et OCR1C.

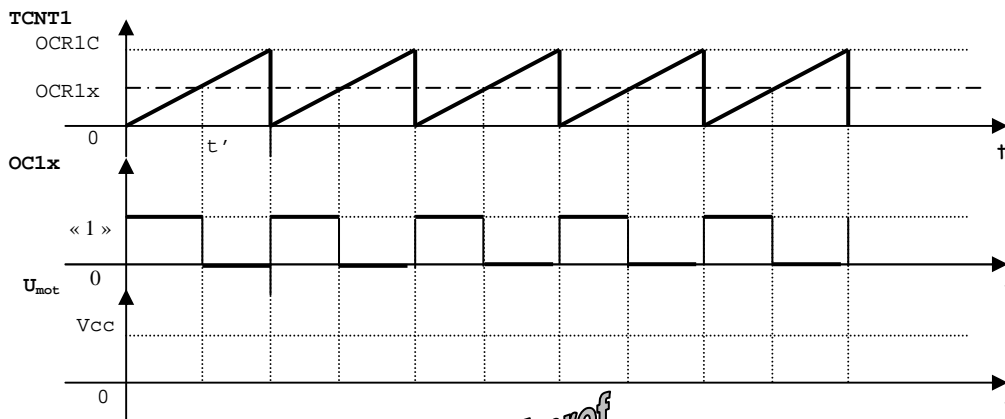
La démarche proposée dans le paragraphe suivant va vous aider à établir cette relation.

B32) Détermination de l'expression du rapport cyclique α en fonction de OCR1x et OCR1C

Les chronogrammes ci-dessous sont extraits des « Datasheets » ATMEL.
Pour établir $\alpha = f(\text{OCR1x}, \text{OCR1C})$, il faut « raisonner » avec le signal U_{mot} présent aux bornes du moteur. Il faut donc tenir compte du transistor Q.

Q3) Tracez $U_{\text{mot}}(t)$ sur les chronogrammes ci-dessous sachant que le transistor Q (considéré parfait) fonctionne en commutation. **Repérez** le temps à l'état haut t_1 et la période T du signal U_{mot} par des flèches.

Indications : « 0 » = 0V, « 1 » = 5V, $V_{\text{cc}} = 6V$. Sur le symbole du transistor, l'intensité du courant circule dans le sens de la flèche.



Appel prof

Pour faire vérifier votre chronogramme

Q4) Exprimez TCNT1 en fonction de t pour $t \in [0, T]$.

Q5) Exprimez t_1 en fonction de T et de t' (1).

Q6) A l'instant $t = t'$, $\text{TCNT1} = \text{OCR1x}$. **Exprimez** t' en fonction de OCR1C et de OCR1x (2).

Q7) Exprimez α en fonction de OCR1C et OCR1x à partir des expressions (1) et (2).

Q8) Complétez le tableau ci-dessous pour OCR1C = 255 (arrondissez à l'entier supérieur).

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

Q9) Complétez la partie correspondant au programme C ci-dessous.

```
#define _0PC          255          // 0%  
#define _10PC _____          // 10%  
#define _20PC _____          // 20%
```

B33) Validation expérimentale de la commande des moteurs

- ① **Copiez** le répertoire « **FicheGuide4** » du PPE sur le bureau de votre PC.
- ② Double-clique sur le fichier projet « **Stay_On_LineV3E.prj** »
- ③ **Complétez** les lignes repérées par :

// à COMPLETER dans le fichier .C

- ④ **Compilez** le projet (icône Make)



- ⑤ **Positionnez** les cavaliers **I2** et **I4** en mode « **Programmation** » conformément à l'annexe 1 du .
- ⑥ **Connectez** la carte « MrLineTiny » au PC. (Connecteur HE10-10)
- ⑦ **Mettez** le commutateur Marche/Arrêt sur Marche.
- ⑧ **Télécharger** le programme dans le microcontrôleur (Bouton « Program »)







Appel prof

Pour utilisation de l'oscilloscope

- ⑨ **Mettez** le commutateur Marche/Arrêt sur Arrêt. Placez le robot sur la feuille représentant la piste (**Annexe 3**). Placez les sondes de l'oscilloscope comme cela est représenté en **annexe 2**.
- ⑩ **Réglez** l'oscilloscope à partir de l'exemple de relevé de l'**annexe 4** de ce document.



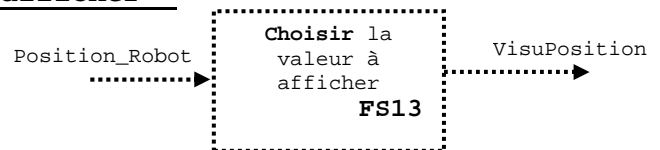
Q10) Pour chacune des positions du robot données dans le tableau ci-dessous, **visualisez** le signal de commande des moteurs et **calculez** leur rapport cyclique. **Complétez** le tableau ci-dessous.

Position_Robot	LED (rouge)	α mesuré avec l'oscilloscope		Conforme au graphe des transitions du 	
		CDE_ROUE_G (%)	CDE_ROUE_D (%)		
Centrel				Oui <input type="checkbox"/>	Non <input type="checkbox"/>
Gauche				Oui <input type="checkbox"/>	Non <input type="checkbox"/>
Gauche+				Oui <input type="checkbox"/>	Non <input type="checkbox"/>
Droite				Oui <input type="checkbox"/>	Non <input type="checkbox"/>
Droite+				Oui <input type="checkbox"/>	Non <input type="checkbox"/>

(○ LED éteinte - ● LED éclairée)

B4) Analyse de FS13 « Choisir la valeur à afficher »

Objectif : Etablir l'algorithme de transcodage nécessaire à la commande des LED de visualisation de la position du robot.



La table des différents codes à affecter à la variable VisuPosition a été établie dans la fiche guide 2. Ces codes commandent l'éclairage des LED en fonction de la position du robot.

Q11) Ecrivez ci-dessous l'algorithme de commande des LED (FS13) en fonction de la position du robot après avoir analysé le programme Stay On LineV3E.C.

This image shows a single sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

C) Synthèse

Le robot à concevoir sera équipé de **cinq capteurs infrarouge**.

C1) Conception de l'algorithme de FS11 pour le robot à réaliser

Q12) Dessinez l'algorithme de commande de ces cinq capteurs.

C2) Conception de FS12 pour le robot à réaliser

C21) Calcul des coefficients α (complément)

Le rapport cyclique α de la commande des moteurs du robot à réaliser doit pouvoir être réglé entre 0 et 90% par pas de 5%.

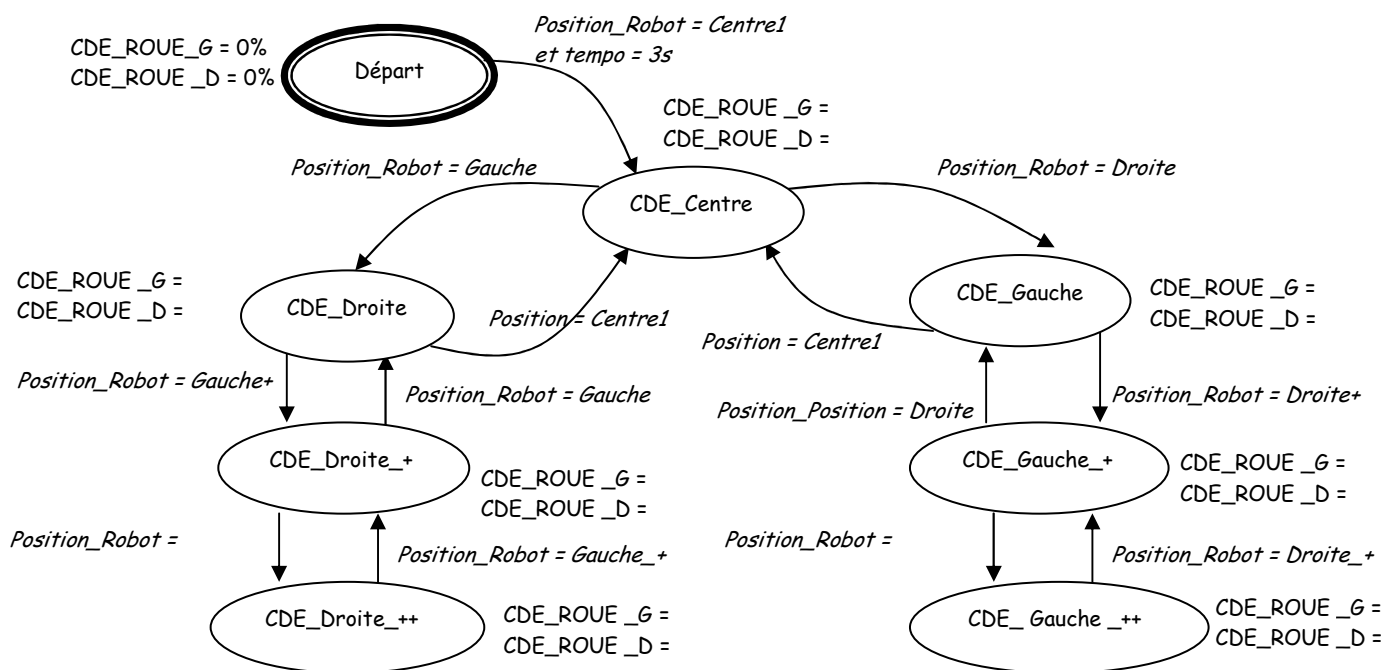
Q13) Complétez le tableau ci-dessous.

$\alpha(\%)$	5	15	25	35	45	55	65	75	85
OCR1x									

C22) Elaboration du graphe des transitions du robot à réaliser

Vous avez abordé la notion de graphe des transitions lors de la mise en œuvre du robot MrLineTiny.

On rappelle ci-dessous le graphe implanté dans le μC ATINY26.



Q14) Dessinez le graphe des transitions du robot à réaliser à partir de celui de MrLineTiny. (Rappel : Le robot à réaliser dispose de 5 capteurs).

Le programme définitif du robot à réaliser sera écrit à partir de votre graphe.

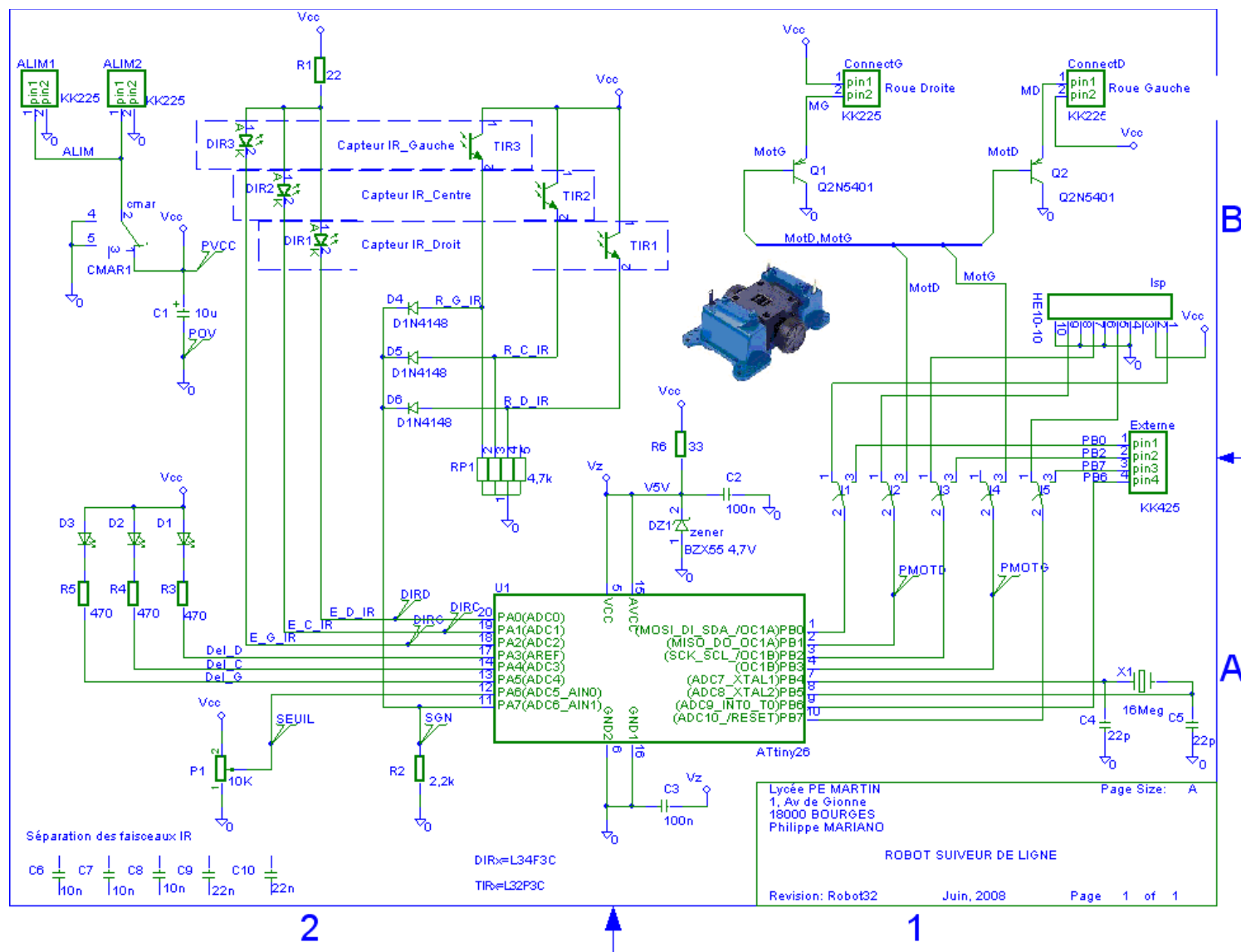
Graphe des transitions du robot à réaliser

C3) Conception de FS13 pour le robot à réaliser

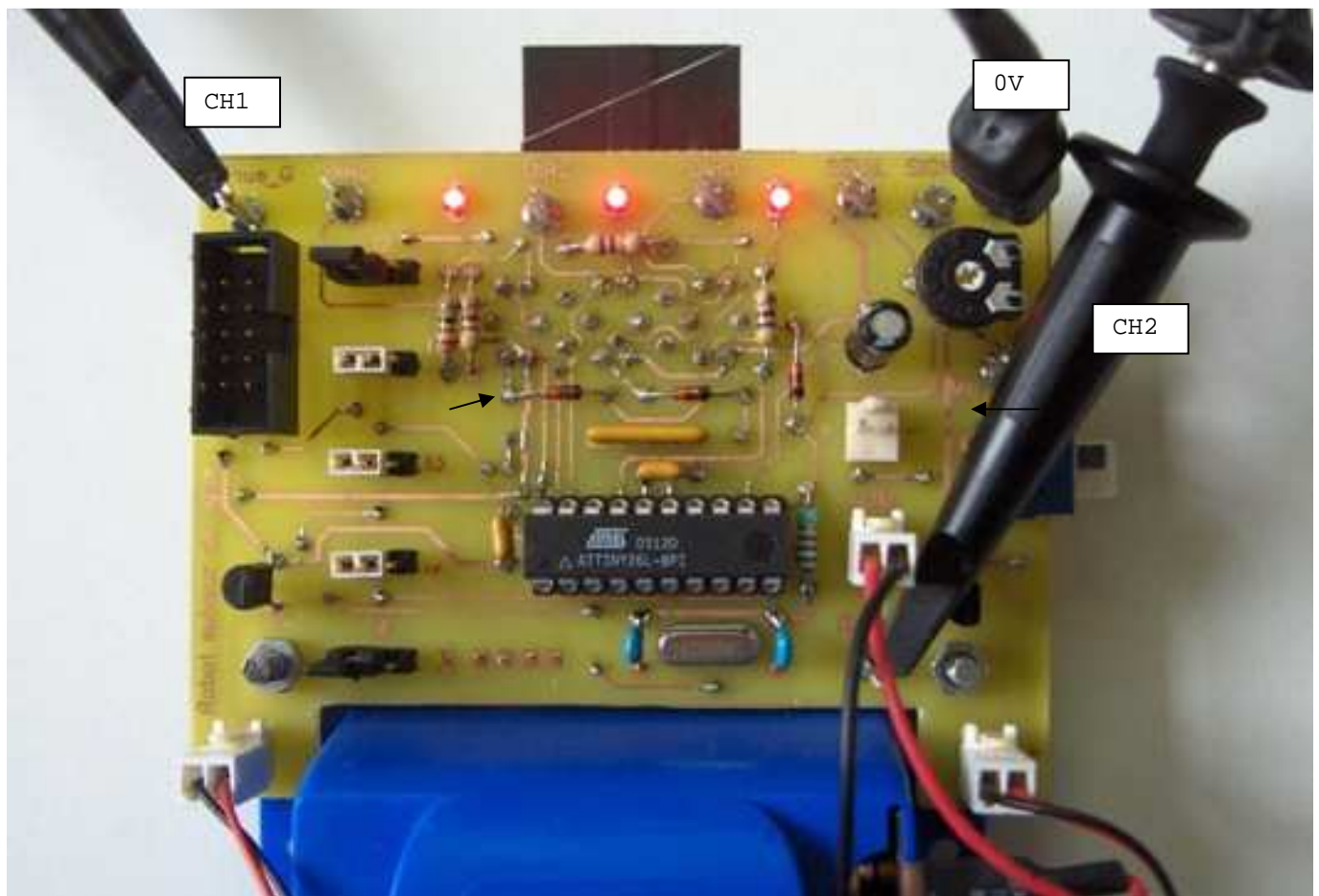
Q15) Ecrivez l'algorithme de FS13 pour le robot à réaliser. Utilisez les valeurs de la variable *VisuPosition* déterminées dans la partie synthèse de la fiche guide 2 et déterminez les valeurs de la variable *Position_Robot* en fonction du schéma de commande des capteurs que vous avez dessiné dans la partie synthèse de la fiche guide 3.

This image shows a single page of white paper with horizontal blue or grey ruling lines. The lines are evenly spaced and run across the width of the page, leaving small margins at the top and bottom. There are no vertical margin lines, text, or other markings on the page.

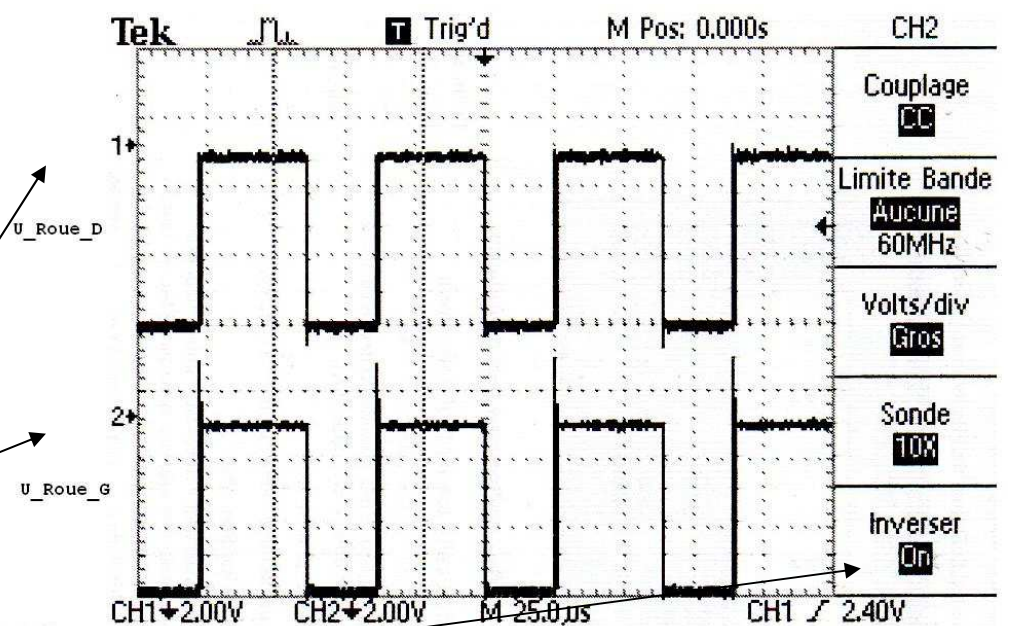
Annexe 1 : Schéma structurel de la carte M^rLineTiny



Position des sondes



Exemple de mesure (Les trois capteurs sur la ligne)



Il faut inverser les signaux pour avoir la représentation de la tension aux bornes des moteurs.

