Rapport du TerrorBot

Vincent Buisset  
Julien Faidide

# Description du projet

Le TerrorBot avance lentement en ligne droite avec les leds avant allumées et s’arrête dès qu’il touche un obstacle. Il recule jusqu’à sa position de départ en faisant clignoter ses feux arrières. Ensuite, il fonce rapidement tout droit avec les leds avant allumées ! Va-t’il s’écraser contre l’obstacle ? Et non ! Il s’arrête juste avant de toucher l’obstacle. Quel efroi !

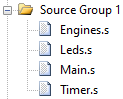
## Vidéo du programme

<https://youtu.be/UHmo6leS6oE>

## Différents scénarios

Le TerrorBot ne fonctionne que dans un seul scénario, il avance jusqu’à toucher un mur, la distance avec le mur peut varier.

# Architecture du projet



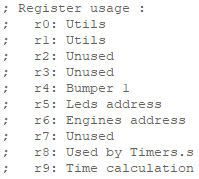
Gère les moteurs

Gère les leds

Programme principal

Gère les délais d'attente

## Utilisaiton des registres



Utilisé pour des calculs

Utilisé pour des calculs

Inutilisé

Inutilisé

Adresse de l'état des bumper

Adresse de l'état des leds

Adresse de l'état des moteurs

Inutilisé

Utilisé pour les délais d'attentes dans Timer.s

Utilisé pour des calculs avec le temps

# Engines.s : Gestion des moteurs

Fichier fourni dans le cours, comporte des sous programmes pour simplifier l'utilisation des moteurs. La seule modification apportée est l'ajout d'un sous programme MOTEUR\_SET\_SPEED\_R0, qui permet d'assigner la vitesse des moteurs avec la valeur du registre r0.

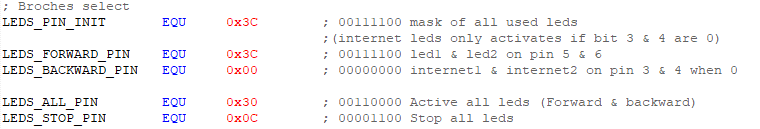
# Timer.s : Gestion des moteurs

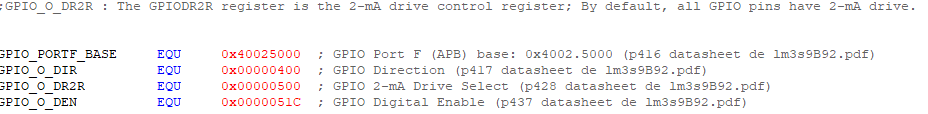
Sous programmes servant à gerer des temps d'attentes au sein du projet. Actuellement, ce fichier ne comporte que 1 seul sous programme : WAIT\_R8. Celui-ci contient une simple boucle, qui décrémente R8 jurqu'a 0 avant de se stopper. Pour generer un temps d'attente, il suffit donc d'assigner R8 avec le temps voulu et d'appeler WAIT\_R8.

# Leds.s : Gestion des leds

Comporte des sous programmes pour faciliter l'utilisation des leds.

## Choix des GPIO





LEDS\_PIN\_INIT : Masque des leds à utiliser

LEDS\_FORWARD\_PIN : valeur d'allumage des leds avant

LEDS\_BACKWARD\_PIN : valeur d'allumage des leds arriere

LEDS\_ALL\_PIN : valeur d'allumage avant & arriere

LEDS\_STOP\_PIN: valeur pour éteindre toute les leds

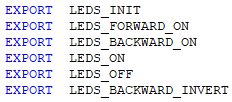
GPIO\_PORTF\_BASE: Adresse du port F

GPIO\_O\_DIR : Adresse du Pin 1 sur le port F

GPIO\_O\_DR2R : Adresse e selection de l'intensité (2mA) sur le port F

GPIO\_O\_DEN : Adresse des fonctions digitales sur le port F

## Sous programmes

Inisitalise les GPIO

Allume les leds avant

Allume les leds arriere

Allume toute les leds

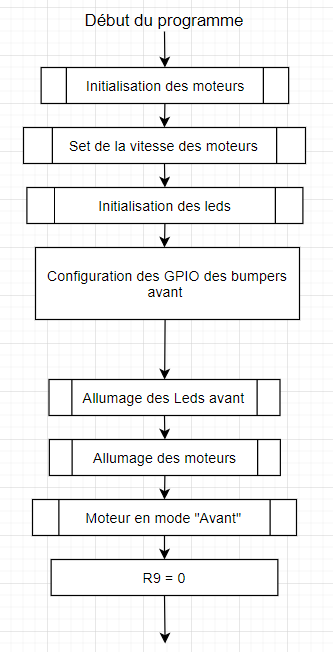
Eteind toute les leds

Inverse l'état des leds arrière, utile pour le clignottement

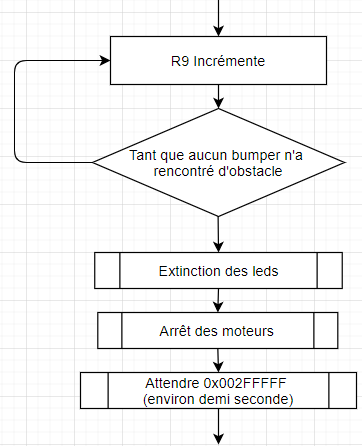
# Main.s : Programme principal

Programme qui utilise Engine.s, Leds.s et Timer.s afin de réaliser le comportement en description du projet. Le programme peut etre découpé en 4 parties :

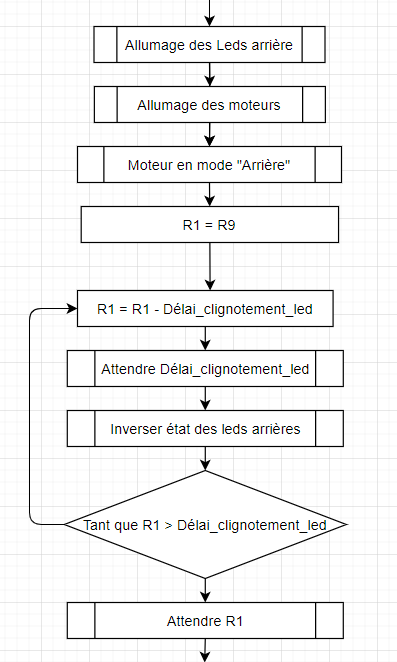
## Initialisation



## Le robot avance



## Le robot recule



## Le robot fonce !

