

Le Robot

Embedded

par Mathieu Garon

Retours

- Vous connaissez la différence entre un MCU et un CPU;
- Vous avez fait au moins un circuit simple;
- Nommez moi quelques périphériques?
- Comment fonctionne un interruption?
- Défi de prog;
- Commentaires sur le cours jusqu'à présent?

Sommaire

- Le projet;
- Encore un retour sur Git : équipes!
- Programmation modulaire C;
- Hardware : Le bouton et gestion des I/O digitales;
- Premier module du robot : boucle principale, démarré et stopper le robot. Préparation des autres modules?

1 - Le projet

Vous allez construire un suiveur de ligne qui possède d'autres fonctions spéciales

La première étape consiste à comprendre le problème et tout ces composantes. Pour cette exercice on va prendre pour acquis que nous concevons un suiveur de ligne motorisé tout simplement.

Dans ce cours, peu de design, plus du technique! mais...

D'autres défis sont à venir, vous êtes les bêta testeur du système de formation!



1 - Le projet

Définition des objectifs pour la formation:

- La propulsion du robot se fait avec un moteur DC;
- Le Robot communique à l'aide d'un écran LCD;
- Le Robot doit être activé/stopper par un push bouton à n'importe quel moment;
- Le Robot a un "steering" pour tourné, contrôler par un servomoteur;
- Le Robot a deux capteurs photosensibles pour regarder le sol;

On va avoir besoin d'un diagramme fonctionnel!

1 - Le projet

*Les diagrammes fonctionnels sont la base d'un design, il réduit **considérablement** les chances d'erreur de design tout en proposant une vision globale du problème.*

- Fonctions du système;
- Entrées sorties;
- Communication entre les fonctions;
- Aucun élément de design!

Au final, il s'agit d'un plan pour désigner un robot suiveur de ligne, les types de technologies utilisé peuvent changer, mais pas les fonctions.

1 - Le projet

Fonctions d'un robot suiveur de ligne

Stopper le robot

Propulser le robot

Démarrer le robot

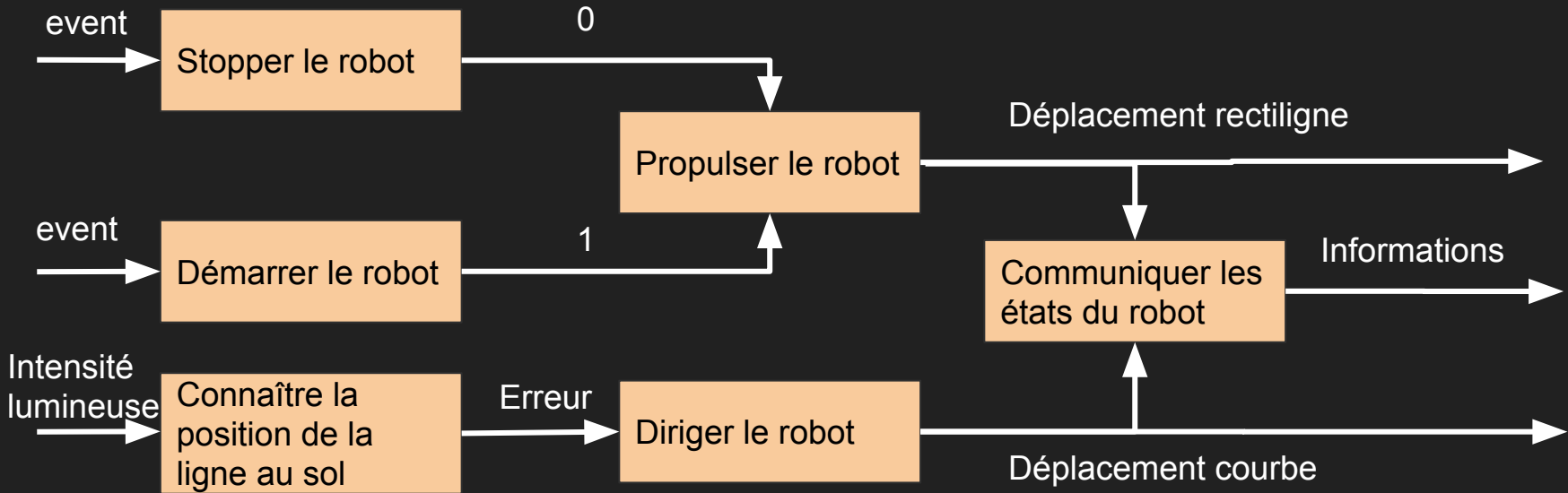
Diriger le robot

Connaître la
position de la
ligne au sol

Communiquer les
états du robot

1 - Le projet

Diagramme fonctionnel d'un robot suiveur de ligne



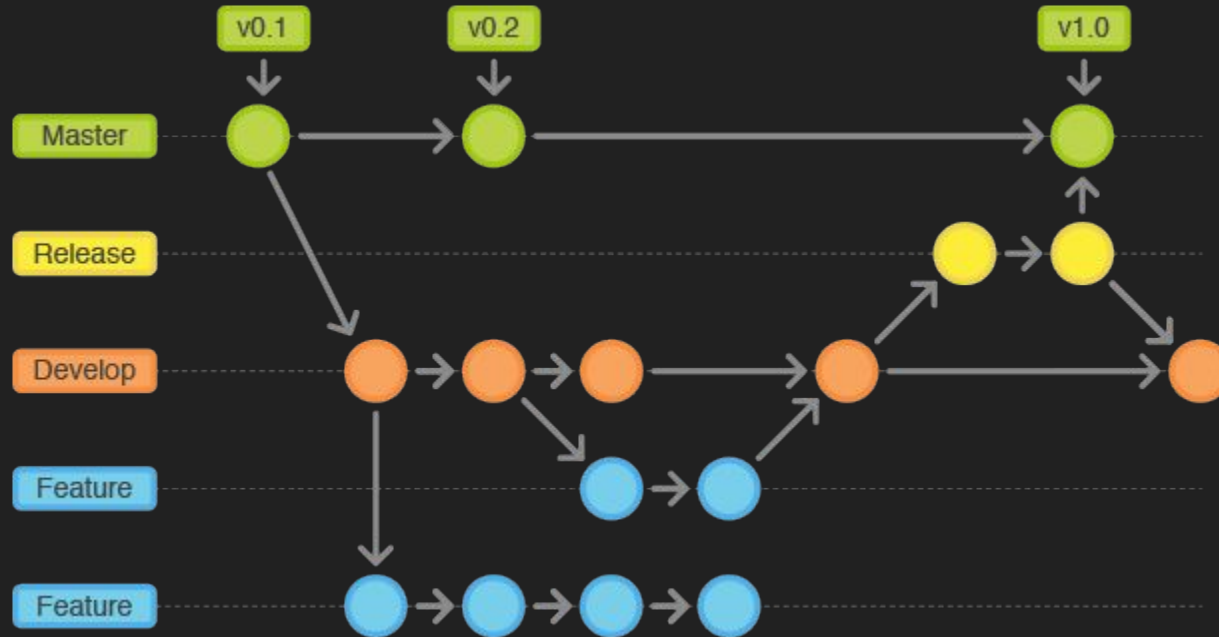
2 - Git

Maintenant, on commence à utiliser git sérieusement.

- Pour le projet, on va définir des équipes, car vous n'aurez pas le temps de tout faire seul.
- Si ce n'est pas fait, cloner le répertoire de nouveau dans un autre dossier;
- Dans le dossier, clique droit et git bash;
- Un des membres de votre équipe va faire les commandes suivantes:
 - `git branch nom_d'équipe;`
 - `git push origin nom_d'équipe.`
- Allez voir sur github.



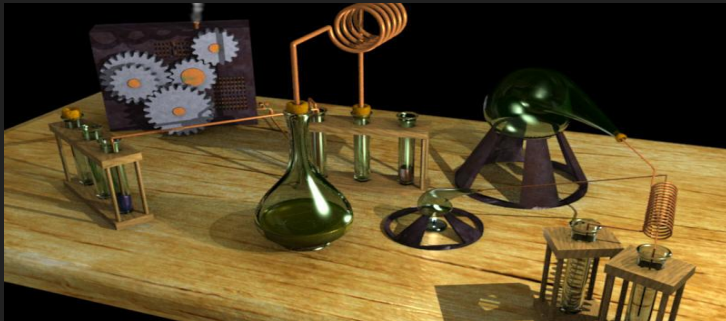
2 - Git



3 - Programmation modulaire

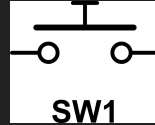
La programmation modulaire est la source de tout code propre et réutilisable

- Facile en Orienté Objet mais malheureusement, peu utile en terme de mémoire et d'efficacité;
- Parfait pour le travail d'équipe;
- Je vous conseil fortement de ne pas coder le projet dans un seul fichier!



4 - Bouton

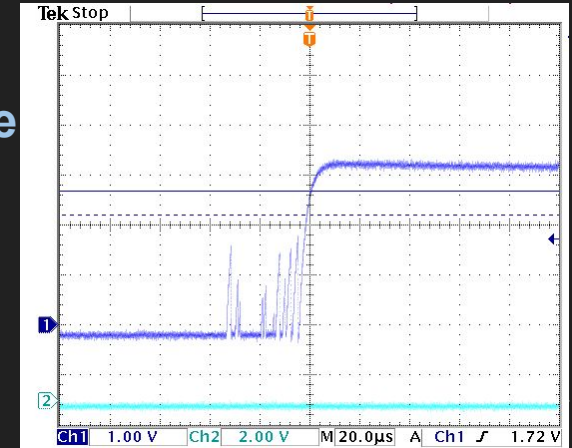
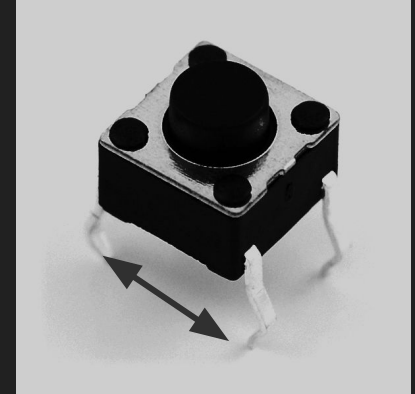
Le “Push button” :



Il s'agit simplement d'un interrupteur qui ferme le circuit seulement au **moment de la pression**.

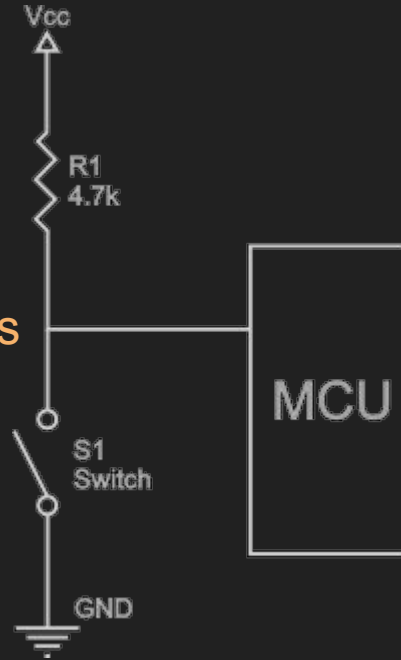
Attention: les broches sont en paires :

Attention: le signal du contact n'est pas une **unique transition**



4 - Circuit de Pull-up / Pull-down

- Une lecture de tension sur du métal flottant donne 0V ou 5V?
- La résistance sert à quoi? (pas d'ailes de poulets pour cette question!)

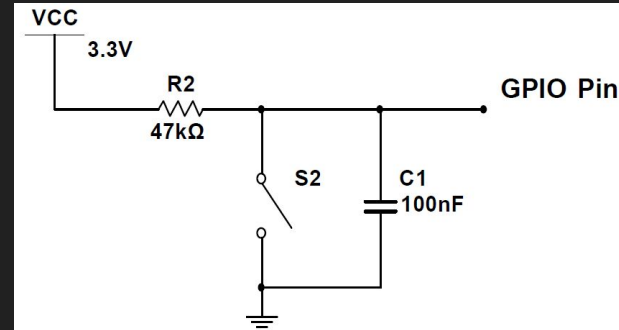


4 - Debounce

Comment assurer un unique transition lors du contacte du bouton?

- Matériel : Filtre passe bas
- Autre méthodes? :

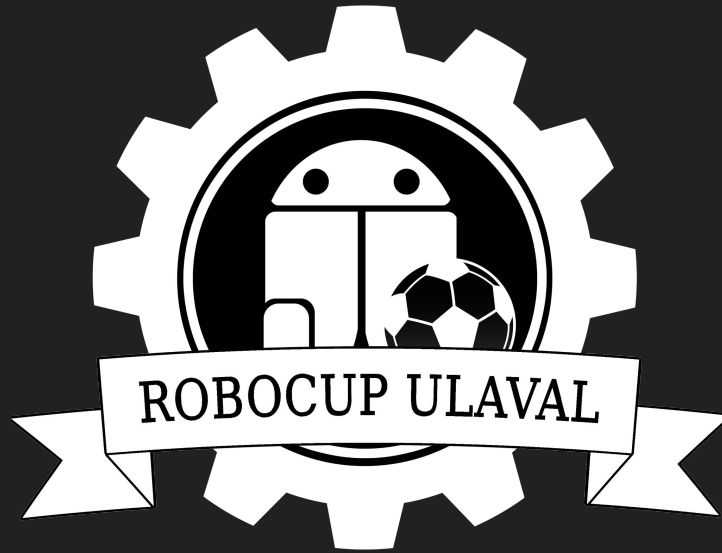
6 ailes pour une méthode ~Parfaite!



5 - Code

Objectifs d'ici la semaine prochaine!

- Avoir une branche git par équipe;
- Avoir un loop principale pour le robot;
- Avoir une led (ça va être notre premier output du robot);
- Avoir un module bouton;
- Pouvoir activer la led et désactivé la led avec un push button (désolé avec une switch c'est trop facile ;)).



Posez vos questions