

Robot suiveur de ligne - Contrôle PID - Configuration Android

1- Introduction

Line Follower Robot with PID control, using an Android device to easily setup the main control parameters for better and fast loop tuning.

2- Choses utilisées dans ce projet

a. Composants matériels

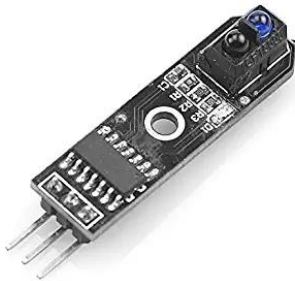
i. Arduino mega (1)



iv. Appareil Android (1)



ii. TCRT5000 Module de capteur de suiveur de ligne infrarouge 4CH (5)



v. Support de batterie 2xAA



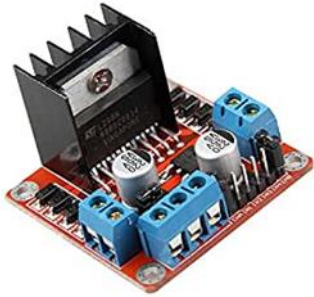
iii. Module Bluetooth HC-05 (1)



vi. Moteur électrique Arduino (2)



vii. Moteur L298



viii. Fils Arduino



b. Applications logicielles et services en ligne

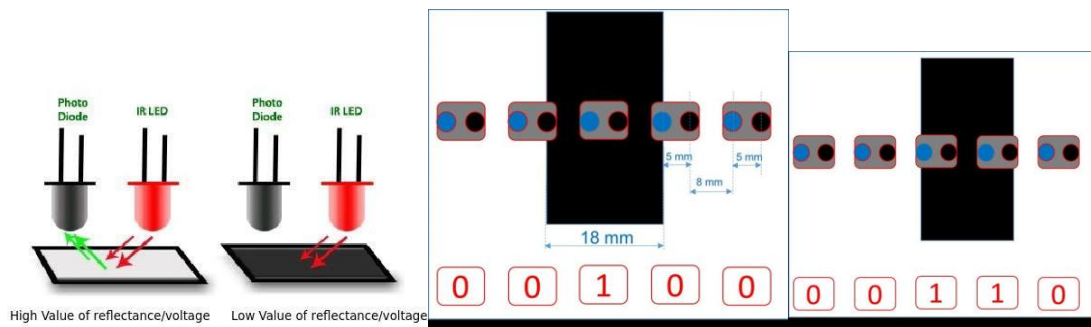
i. Arduino IDE



ii. MIT App Inventor 2



3- Implémentation de la logique du capteur infrarouge



The IR sensor consists of an individual IR LED and an IR Photodiode. The IR light emitted by the LED strikes the surface and is reflected back to the IR Photodiode. The photodiode then generates an output voltage proportional to the reflectance level of the surface (higher values for "light surfaces" and lower for "black/dark surfaces").

Dans le cas des capteurs utilisés, un circuit intégré au niveau du module génère en sortie un simple signal numérique (HIGH: Dark; LOW: Light). Un potentiomètre installé sur le module (voir photo) ajustera le niveau de lumière correct pour qu'il soit considéré comme "sombre" ou "clair". Il fonctionne de manière à ce que lorsque la couleur de la lumière réfléchiée est noire / sombre, un niveau numérique ÉLEVÉ ("1") soit généré à sa sortie et un niveau BAS ("0") pour une autre couleur plus claire. J'ai utilisé ici un module intégré avec 4 capteurs et un module supplémentaire avec un seul capteur (forme différente, mais même logique). La combinaison est un tableau de 5 capteurs que j'ai trouvé bon pour un contrôle agréable et fluide, comme expliqué ci-dessous.

Le réseau de 5 capteurs est monté de manière à ce que si un seul capteur est centré par rapport à la ligne noire, seul ce capteur spécifique produira un HIGH. De l'autre côté, l'espace entre les capteurs doit être calculé pour permettre à 2 capteurs de couvrir simultanément toute la largeur de la ligne noire, produisant

également un HIGH sur les deux capteurs (voir les images ci-dessus).

Les sorties de réseau de capteurs possibles lors du suivi d'une ligne sont :

- 0 0 0 0 1
- 0 0 0 1 1
- 0 0 0 1 0
- 0 0 1 1 0
- 0 0 1 0 0
- 0 1 1 0 0
- 0 1 0 0 0
- 1 1 0 0 0
- 1 0 0 0 0

Le fait d'avoir 5 capteurs permet de générer une "variable d'erreur" qui aidera à contrôler la position du robot sur la ligne, comme indiqué ci-dessous.

Considérons que la condition optimale est lorsque le robot est centré, avec la ligne juste en dessous du "capteur central" (capteur 2). La sortie du tableau sera : 0 0 1 0 0 et dans cette situation, "l'erreur" sera "zéro". Si le robot commence à rouler vers la gauche (la ligne "semble bouger" à droite ") l'erreur doit augmenter avec un signal positif. Si le robot commence à se déplacer vers la droite (la ligne" semble bouger "à gauche"), dans le de la même manière, l'erreur doit augmenter, mais maintenant avec un signal négatif.

La variable d'erreur, liée à l'état du capteur sera :

0 0 1 0 0 ==> Error = 0

- 0 0 0 0 1 ==> Error = 4
- 0 0 0 1 1 ==> Error = 3

- $0\ 0\ 0\ 1\ 0 \implies \text{Error} = 2$
- $0\ 0\ 1\ 1\ 0 \implies \text{Error} = 1$
- $0\ 1\ 1\ 0\ 0 \implies \text{Error} = -1$
- $0\ 1\ 0\ 0\ 0 \implies \text{Error} = -2$
- $1\ 1\ 0\ 0\ 0 \implies \text{Error} = -3$
- $1\ 0\ 0\ 0\ 0 \implies \text{Error} = -4$