Noms du groupe : BARBER Theo, GOMEZ Antoine et GAUTHIER Tristan

Titre du projet : Prototype robot explorateur

Description :

Le projet consiste au pilotage (automatique) du robot Maqueen auquel est programmé le radar à ultrasons de telle sorte à ce que lorsqu'il détecte un obstacle à une distance x, le robot tourne aléatoirement à droite ou à gauche de manière à l'esquiver. De plus, un autre paramètre est pris en considération : l'éclairage des LED, en guise de phares. Lorsque la lumière est à une intensité i, la lumière émise par les LED est plus ou moins importante et varie en fonction de l'intensité lumineuse environnante.

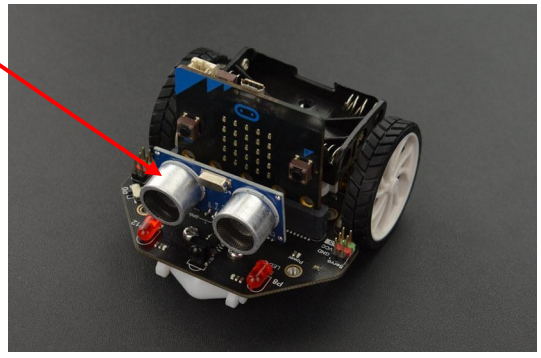
Liste du matériel :

Robot Maqueen (moteur, roues, …), radar à ultrasons, 2 LED, capteur de lumière/light sensor et carte micro-bit.

Les capteurs et actionneurs utilisés :

Radar à ultrasons :

Le robot Maqueen dispose d'une capacité à établir un lien, par le biais du code, entre le moteur et le radar à ultrasons. Le code qui permet ce lien peut être écrit en Python et lié à la carte Micro : bit du robot. Le radar à ultrasons est un capteur qui mesure une distance, on peut le programmer pour qu'il transmette à un actionneur une valeur lorsqu'il atteint une certaine distance déterminée dans le corps du code. Il transmet une information du type analogique, numérisée par une horloge (le radar à ultrasons mesure en réalité un temps) et convertie en distance. Ici, l'actionneur est le moteur (les deux moteurs du robot) qui reçoit l'information du radar à ultrasons, passant d'abord par la carte micro : bit et qui fait une action, ici il fait tourner les roues, en faisant varier l'intensité selon le programme.



Le radar à ultrasons fonctionne selon un principe spécifique :
Il est divisé en deux parties : l'émetteur et le récepteur
Tout d'abord, l'émetteur envoie une impulsion sonore, d'où « radar à ultrasons » qui est renvoyée par une surface réfléchissante (au son) jusqu'au récepteur. Pendant cet échange, le temps est mesuré et ensuite envoyé à la carte micro : bit qui se charge de le convertir en distance, exploitable plus facilement dès lors qu'il est question d'un programme qui fonctionne comme un radar de voiture par exemple.
Sans la carte micro : bit ou autre carte, rien n'est possible : elle transfère et supervise le déroulement du programme (elle reçoit les informations et les transfère aux actionneurs qui font partie du code).

Exemple de commande présente dans le programme :

```
mq.distance()
```

Light Sensor :

La technologie LightSensor déployée dans certains moniteurs Philips est destinée à limiter la consommation d'énergie. Il mesure la luminosité de l'environnement d'utilisation et ajuste en fonction, automatiquement, la brillance de l'écran afin de proposer un rendu optimal, tout en limitant la consommation d'énergie lorsque possible. D'où le nom LightSensor traduit par "capteur de lumière" en français.

Exemple de commande présente dans le programme :

display.read_light_level()

LED :

La LED ou DEL en français est un actionneur programmable à l'aide d'une carte de programmation (Arduino ou micro : bit) que l'on peut utiliser et accompagner d'autres actionneurs mais aussi d'autres capteurs. La LED fonctionne selon l'instruction que le programme lui envoie. Si le programme envoie 1 à la LED, alors elle s'éclaire. A l'inverse, lorsqu'on lui envoie 0, la LED s'éteint. Son utilisation peut servir lors de programmes simulant une porte de garage électrique par exemple : lorsqu'un capteur reçoit l'information d'ouvrir le portail, l'instruction est transmise aux moteurs chargés de lever le portail, mais à une LED qui peut s'éclairer simplement ou également clignoter, comme on a l'habitude de l'apercevoir au quotidien. La LED fonctionne selon un système binaire (0 et 1) mais l'on peut à l'aide d'un code précis régler son intensité en fonction de la lumière ambiante par exemple, en faisant intervenir d'autres capteurs et actionneurs tels que le light sensor.

Exemple de commande présente dans le programme :

pin12.write_digital(1)

pin8.write_digital(1)

Robot MAQUEEN :

Description :

Le robot Maqueen est un petit robot qui fonctionne avec une carte micro : bit. Elle permet de contrôler ce robot en le programmant. La programmation peut se faire dans différents langages mais nous utiliseront le langage python.

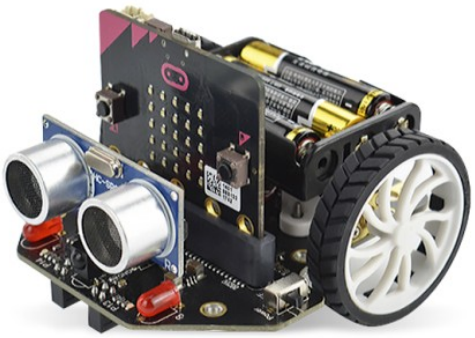
Cette carte permet le contrôle de deux moteurs, d'un détecteur à ultra son, de deux suiveurs de lignes, de 4 LED RGB, un buzzer, de deux LED rouges et d'un récepteur infra rouge pour le pilotage via une télécommande infra rouge qui sont intégrés dans le châssis.

Pilotage de robot :

Les principales actions que l'on peut demander au moteur sont :

Avancer : mq.avance()

Tourner à droite : mq.moteurGauche(50)



mq.moteurDroite(0)

Tourner à gauche : mq.moteurDroite(50)

mq.moteurGauche(0)

Code commenté :

```
from microbit import *                  #on importe la librairie microbit au programme
from maqueen import Maqueen             #on importe depuis la librairie maqueen la fonction  Maqueen au programme
from random import randint              #on importe depuis la librairie random la fonction randint au programme
pin12.write_digital(0)                  #on éteint la LED rouge avant droite
pin8.write_digital(0)                   #on éteint la LED rouge avant gauche
mq=Maqueen()
mq.setVitesse(50)                       #on définit la vitesse du robot
while True:                             #on crée une boucle infinie
    mq.avance()                         #on fait avancer le robot
    d=mq.distance()                     #on associe la variable d à la distance captée par le capteur à       ultrason
    if d<20:                            #obstacle proche
        r=randint(1,2)                  #tire un nombre aléatoirement, soit 1, soit 2
        if r==1:                        #si le nombre tiré est 1
            mq.stop()                   #on stoppe le robot
            sleep(1000)                 #on attend une seconde
            mq.moteurGauche(50)         #on allume le moteur gauche pour faire tourner à droite le robot
            mq.moteurDroit(0)           #on éteint le moteur droit pour faire tourner à droite le robot
            sleep(1000)                 #on attends une seconde
            mq.avance()                 #on fait avancer le robot
        else:                           #si le nombre tiré est 2
            mq.stop()                   #on stoppe le robot
            sleep(1000)                 #on attends une seconde
            mq.moteurDroit(50)          #on allume le moteur droit pour faire tourner le robot à gauche
            mq.moteurGauche(0)          #on éteint le moteur gauche pour faire tourner le robot à gauche
            sleep(1000)                 #on attend une seconde
            mq.avance()                 #on fait avancer le robot
    if display.read_light_level() <= 75:    #on mesure la luminosité, si elle est inférieur à 75,
        pin12.write_digital(1)          #on allume la LED avant droite
        pin8.write_digital(1)           #on allume la LED avant gauche
    else:                               #sinon,
        pin12.write_digital(0)          #la LED avant droite reste éteinte
        pin8.write_digital(0)           #la LED avant gauche reste éteinte
```

Extensions ou développements possibles du projet :

Nous pourrions développer le code du robot explorateur de manière à ce qu'il soit autonome et que l'on puisse lui attribuer les fonctionnalités d'un rover d'exploration réel, en l'améliorant et programmant un pilotage automatique à une destination dont les paramètres sont communiquées au robot, et en parallèle, un pilotage manuel à distance à l'aide par exemple d'une manette à

joysticks ou à touches analogiques en vue de l'exploration de zones plus ou moins denses, rurales, tropicales ou encore des zones sous-terraines. Pour une amélioration optimale, d'autres éléments pourront être modifiés pour faciliter l'exploration du robot : roues type tout-terrain, moteurs plus puissants, … Nous pourrons de même revoir la taille du rover, qui n'est pour l'instant qu'un simple prototype miniature, mais qui peut être utilisé dès maintenant pour circuler dans des espaces où l'Homme ne peut accéder. Nous pourrons rajouter une caméra qui renvoie en direct les images qu'elle perçoit.