

Proposition de synthèse

L'Informatique et la Programmation

Compétence : Écrire un programme dans lequel des actions sont déclenchées par des événements extérieurs.

Connaissance(s) associée(s) : Notions d'algorithme et de programme.

Les objets connectés sont souvent programmés pour fonctionner automatiquement

Description du programme	
Algorithme	Algorigramme
<p>Un algorithme est une description d'une suite ordonnée d'instructions permettant d'atteindre un objectif.</p>	<p>Un algorigramme est la représentation normalisée graphique d'un algorithme.</p>
<p>Si le capteur à ultrason du robot détecte un obstacle à moins de 5cm, les LED doivent s'allumer en rouge, sinon elles s'allument en vert.</p>	<pre> graph TD Start([Début]) --> Decision{Distance ultrason inférieure à 5?} Decision -- Oui --> Action1[Allumer les LED en rouge] Decision -- Non --> Action2[Allumer les LED en vert] Action1 --> Decision Action2 --> Decision </pre> <p>A partir de cette représentation graphique, nous allons pouvoir réaliser le programme du système.</p>

Proposition de synthèse

L'Informatique et la Programmation

Compétence : Écrire un programme dans lequel des actions sont déclenchées par des événements extérieurs.

Connaissance(s) associée(s) : Notions d'algorithme et de programme.

Le programme

Langage par bloc

Langage par bloc est un langage de programmation graphique simplifié et accessible.



Le code

Le code est une suite d'instructions simples permettant la création du programme et utilisable par la machine.

Remarque : dans l'exemple, le code a été généré à partir du programme en langage graphique.

```

1 #include <Arduino.h>
2 #include <Wire.h>
3 #include <SoftwareSerial.h>
4
5 #include <MeMCore.h>
6
7 MeDCMotor motor_9(9);
8 MeDCMotor motor_10(10);
9 void move(int direction, int speed)
10 {
11     int leftSpeed = 0;
12     int rightSpeed = 0;
13     if(direction == 1){
14         leftSpeed = speed;
15         rightSpeed = speed;
16     }else if(direction == 2){
17         leftSpeed = -speed;
18         rightSpeed = -speed;
19     }else if(direction == 3){
20         leftSpeed = -speed;
21         rightSpeed = speed;
22     }else if(direction == 4){
23         leftSpeed = speed;
24         rightSpeed = -speed;
25     }
26     motor_9.run((9)==M1?-(leftSpeed):(leftSpeed));
27     motor_10.run((10)==M1?-(rightSpeed):(rightSpeed));
28 }
29 double angle_rad = PI/180.0;
30 double angle_deg = 180.0/PI;
31 MeUltrasonicSensor ultrasonic_3(3);
32 MeRGBLed rgbled_7(7, 7==7?2:4);
33
34 void setup(){
35 }
36
37 void loop(){
38     if((ultrasonic_3.distanceCm()) < (5)){
39         rgbled_7.setColor(0, 255, 0, 0);
40         rgbled_7.show();
41     }else{
42         rgbled_7.setColor(0, 0, 255, 0);
43         rgbled_7.show();
44     }
45     _loop();
46 }
47
48 void _delay(float seconds){
49     long endTime = millis() + seconds * 1000;
50     while(millis() < endTime)_loop();
51 }
52
53 void _loop(){
54 }

```