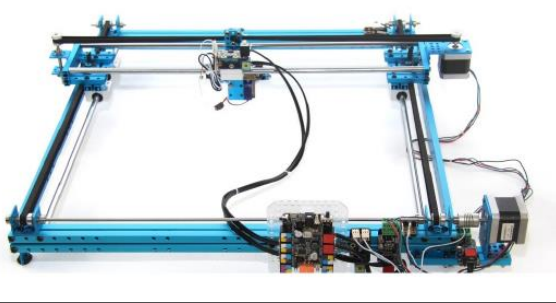


Apprenons la programmation !

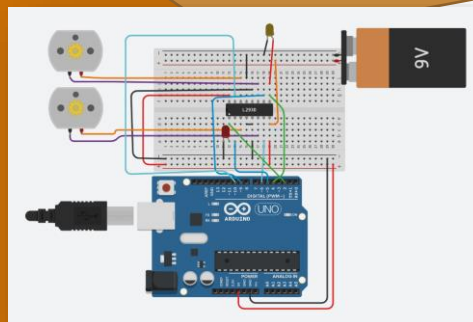


Présentation du projet

L'objectif de ce projet est d'initier un jeune public à la programmation à travers des jeux permettant de diriger un bras robotique.

L'interfacage a été fait sous Python avec Tkinter pour plus de modularité.

L'objet se dirige par une platine 2 axes par 2 moteurs. Les 2 moteurs pas à pas sont pilotés par la carte Raspberry qui rend ses positions précisément. La carte Raspberry utilise Python. Elle récupère les consignes du algorithme directement .



Compromis: Circuits sont adaptés à la carte Arduino.

Un labyrinthe



Apprenons la programmation !

Labyrinthe

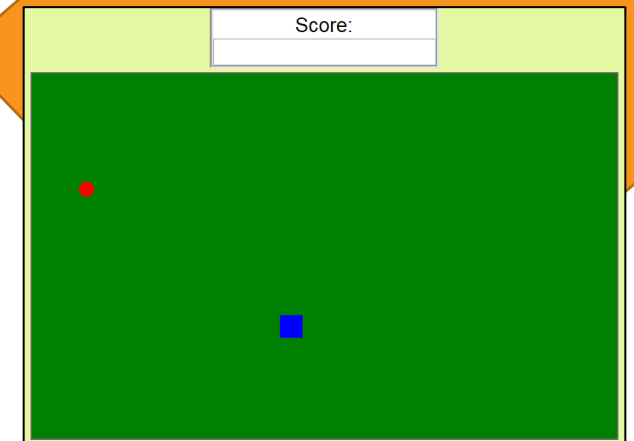
Ce jeu a pour but de sortir d'un labyrinthe

Snake

Ce jeu a pour but de récupérer différents fruits sur un parcours



Snake



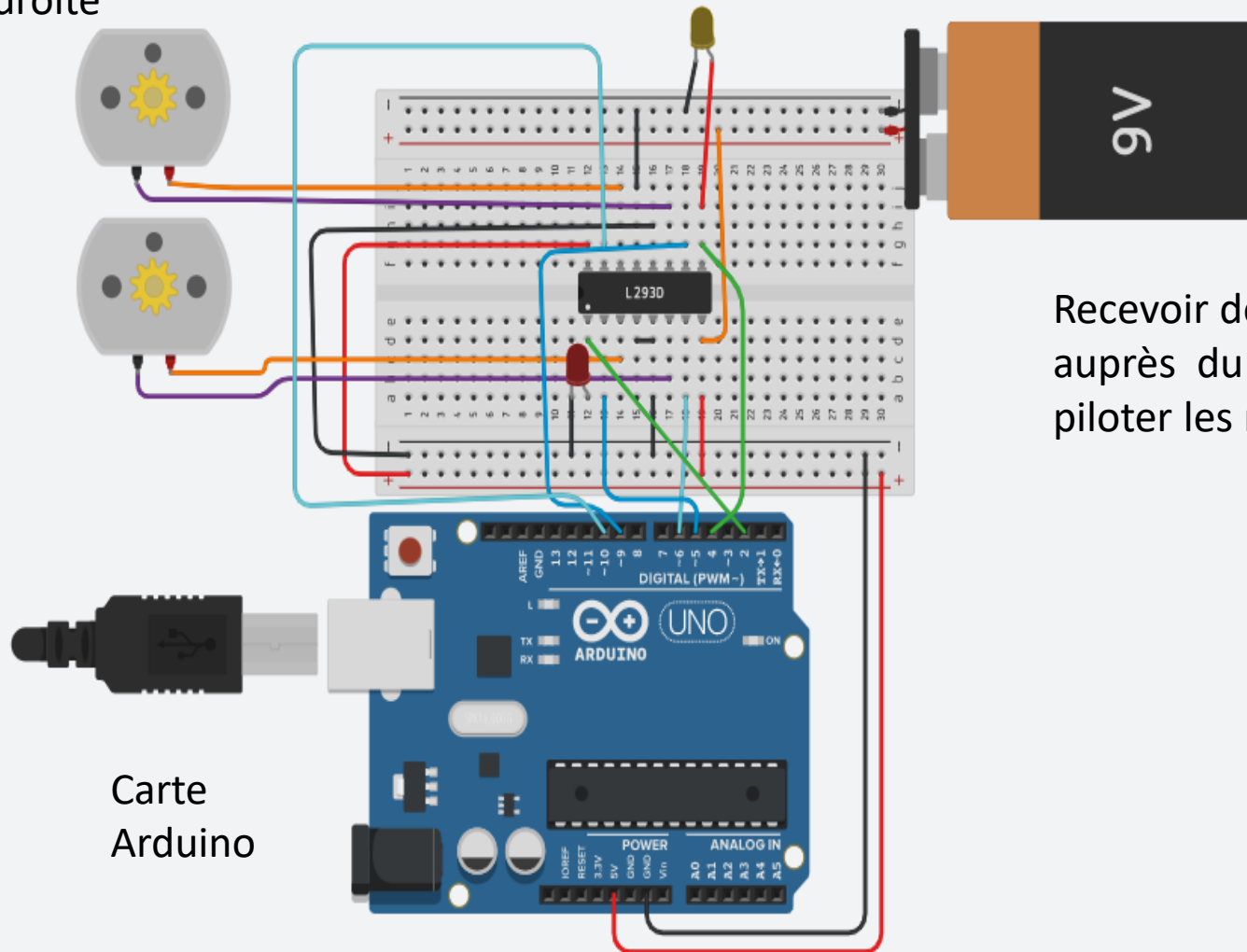
Transmission :

- > pilotage des moteurs
- > commandables via l'interface

Problèmes rencontrés:

- > travail à distance et organisation
- > choix du logiciel de simulation
- > modularité des fonctions tkinter et de la gestion des évènements (menu)
- > Simulateur ne convient pas au cas réel

2 DC moteur
Avant(haut),
arrière(bas), gauche,
droite



Carte
Arduino

Recevoir des consignes
auprès du itinéraire et
piloter les moteurs

文本

1 (Arduino Uno R3)

```
1 //Init Moto
2 //This program is to control the movement of forward, backward,
3 //go right and go left according to the consign given.
4 //Red LED refers to movement of motor1
5 //Yellow LED refers to movement of motor2
6 const int active1 = 2;// to activate motor 1
7 const int active2 = 4;// to activate motor 2
8
9 const int motor1_a = 5;
10 const int motor1_b = 6;
11 const int motor2_a = 9;
12 const int motor2_b = 10;
13 int speed = 127; //50% duty cycle,max=255
14 int time = 500; //default 1000ms
15
16 char movement = 'F';//F:forward B:backward R:right L:left
17 //motor 1 will control 'forward' and 'backward'
18 //motor 2 will control 'right' and 'left'
19 int step = 2;// number of steps in this direction
20
21 void setup()
22 {
23   pinMode(active1, OUTPUT);
24   pinMode(active2, OUTPUT);
25   pinMode(motor1_a, OUTPUT);
26   pinMode(motor1_b, OUTPUT);
27   pinMode(motor2_a, OUTPUT);
28   pinMode(motor2_b, OUTPUT);
29 }
30
31 void loop()
32 { if (step>0)
33   {
34     switch(movement)
35     {
36       case 'F':
37         forward_motor1(time);
38         step=step-1;
39         break;
40       case 'B':
41         backward_motor1(time);
42         step=step-1;
43         break;
44       case 'R':
45         forward_motor2(time);
46         step=step-1;
```

串行监视器