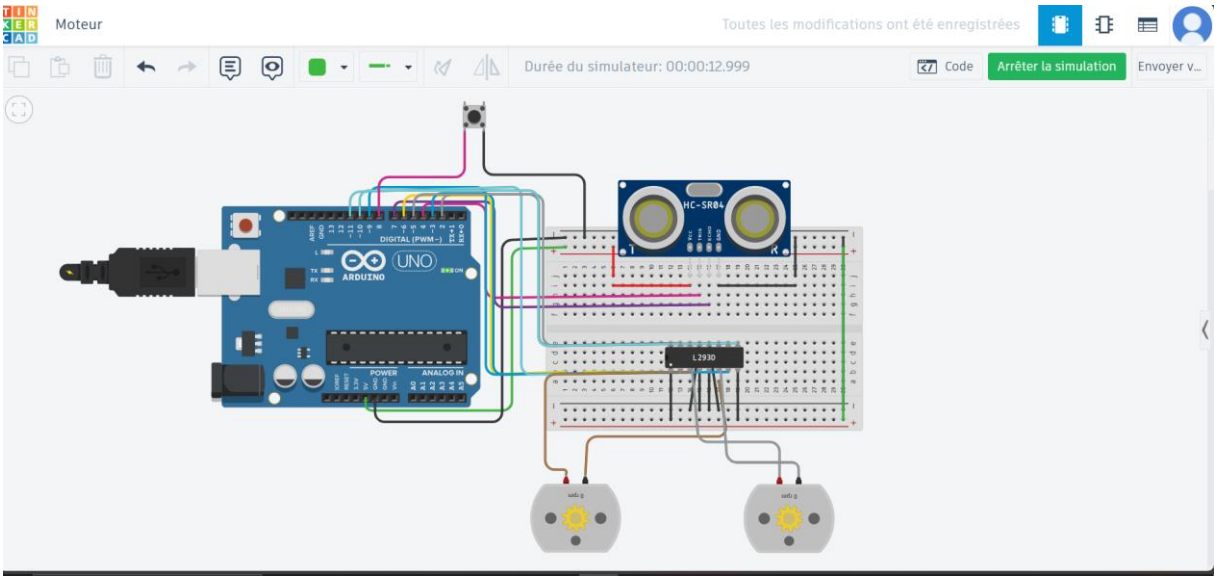
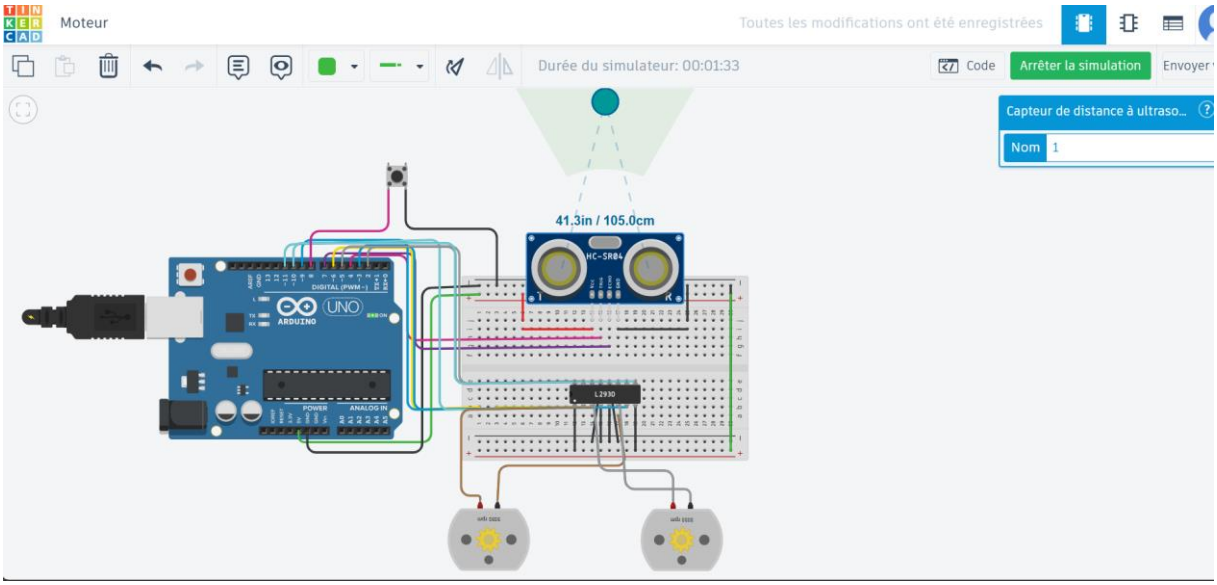
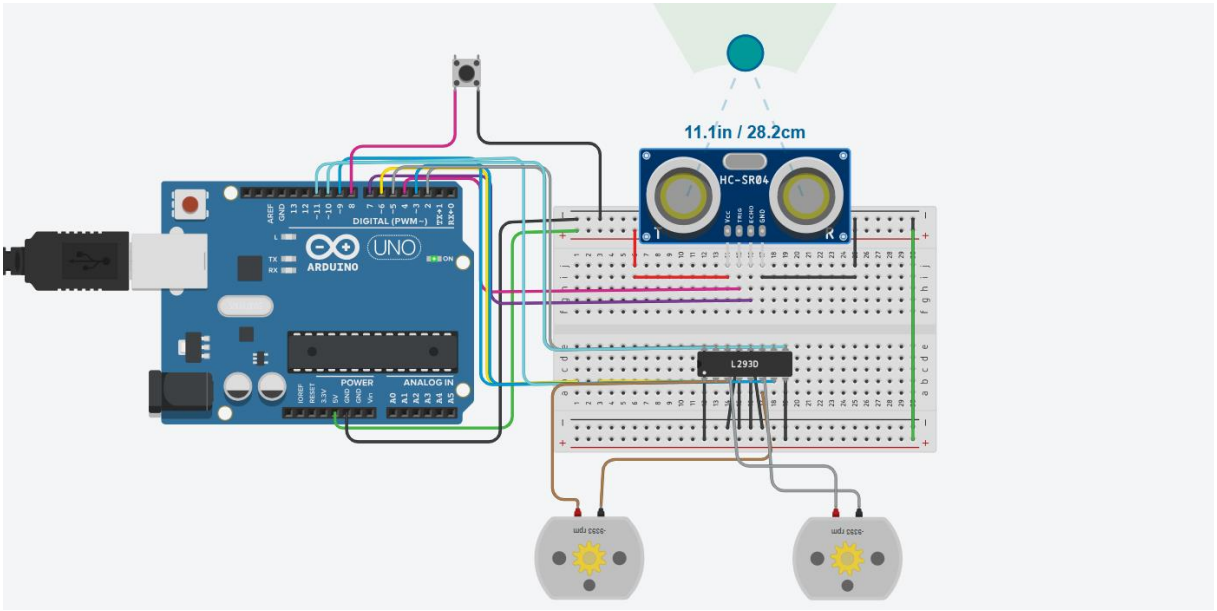
JAONINA Sissie Thephanies








```ino
int trigPin = 4;
int echoPin = 7;
long time;
float distanceCm, distanceInch;
const int bouton = 8;
int btnstate;
bool robotActive = false;

void setup()
{
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
  pinMode(11, OUTPUT);
  pinMode(9, OUTPUT);
  pinMode(3, OUTPUT);
  pinMode(bouton, INPUT_PULLUP);
  Serial.begin(9600);
}

void loop()
{
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);

  // Transmetre le son pour microsecond
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);

```

Moniteur série

```
29
30    // Calcul distance
31    time = pulseIn(echoPin, HIGH);
32    Serial.println("time");
33    Serial.println(time);
34    distanceCm = time * 0.034 / 2;
35    distanceInch = time * 0.0133 / 2;
36
37    // Affichage la valeur finale
38    Serial.println("Distance :");
39    Serial.println(distanceCm);
40    Serial.println("Cm");
41    Serial.println(distanceInch);
42    Serial.println("Inch");
43
44    // Lecture de l'état du bouton
45
46    btnstate = digitalRead(bouton);
47    Serial.print("Bouton état : ");
48    Serial.println(btnstate); // Ajoutez cette ligne pour le diagnostic
49
50    btnstate = digitalRead(bouton);
51
52     if (btnstate == LOW) {
53      delay(100); // Anti-rebond
54      if( btnstate == LOW) { // Vérifiez à nouveau
55        robotActive = !robotActive; // Inverser l'état du robot
56        delay(200); // Anti-rebond supplémentaire
57      }
```

```
    }
  }



  if (robotActive) {
    if (distanceCm < 30) {
      // Moteur 1
      digitalWrite(9, HIGH);
      digitalWrite(3, LOW);
      analogWrite(11, 255);
      delay(2000);
      analogWrite(11, 0);
      delay(100);

      // Moteur 2
      digitalWrite(2, HIGH);
      digitalWrite(5, LOW);
      analogWrite(10, 255);
      delay(2000);
      analogWrite(10, 0);
      delay(100);

      // Changer de direction
      digitalWrite(2, HIGH);
      digitalWrite(5, LOW);
      analogWrite(10, 255);
      delay(100);

```

Moniteur série

```
72        // Moteur 2
73        digitalWrite(2, HIGH);
74        digitalWrite(5, LOW);
75        analogWrite(10, 255);
76        delay(2000);
77        analogWrite(10, 0);
78        delay(100);
79
80        // Changer de direction
81        digitalWrite(2, HIGH);
82        digitalWrite(5, LOW);
83        analogWrite(10, 255);
84        delay(100);
85
86      } else {
87        // Moteur 1
88        digitalWrite(3, HIGH);
89        digitalWrite(9, LOW);
90        analogWrite(11, 255);
91
92        // Moteur 2cap
93        digitalWrite(5, HIGH);
94        digitalWrite(2, LOW);
95        analogWrite(10, 255);
96        delay(1000);
97      }
98    }
99  }
100
```

 Moniteur série