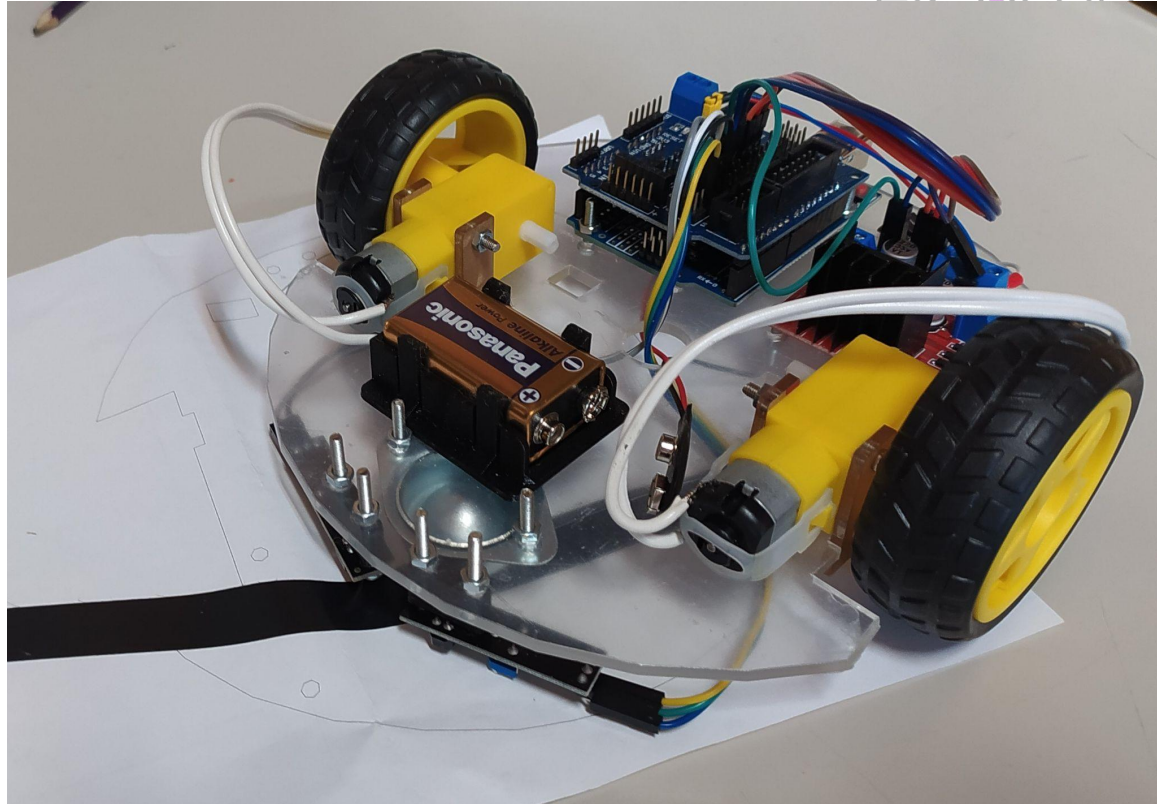




atelier robots suiveurs de lignes

Programme

- Visite de l'ACoLab
- Montage mécanique
- Détection des lignes
- Suivi des lignes



ACoLab : atelier collaboratif

- Un lieu pour réaliser vos projets
- Un atelier avec des machines à dispositions des adhérents
- Des formations à l'usage des machines et des technologies
- Fonctionnement uniquement bénévole



CONNEXIONS, cultivons le numérique



Description du robot



Robot 'differential drive' : deux roues centrales indépendantes + 1 roulette

- base plexiglas, vis, écrous, roue libre
- 2 moteur + supports + roue
- 2 capteurs infrarouge + support
- carte L298N pour l'alimentation de l'arduino et le pilotage des moteurs
- carte arduino pour la programmation
- carte sensor shield v5
- pile 9V + support + connecteur
- interrupteur, câbles dupont

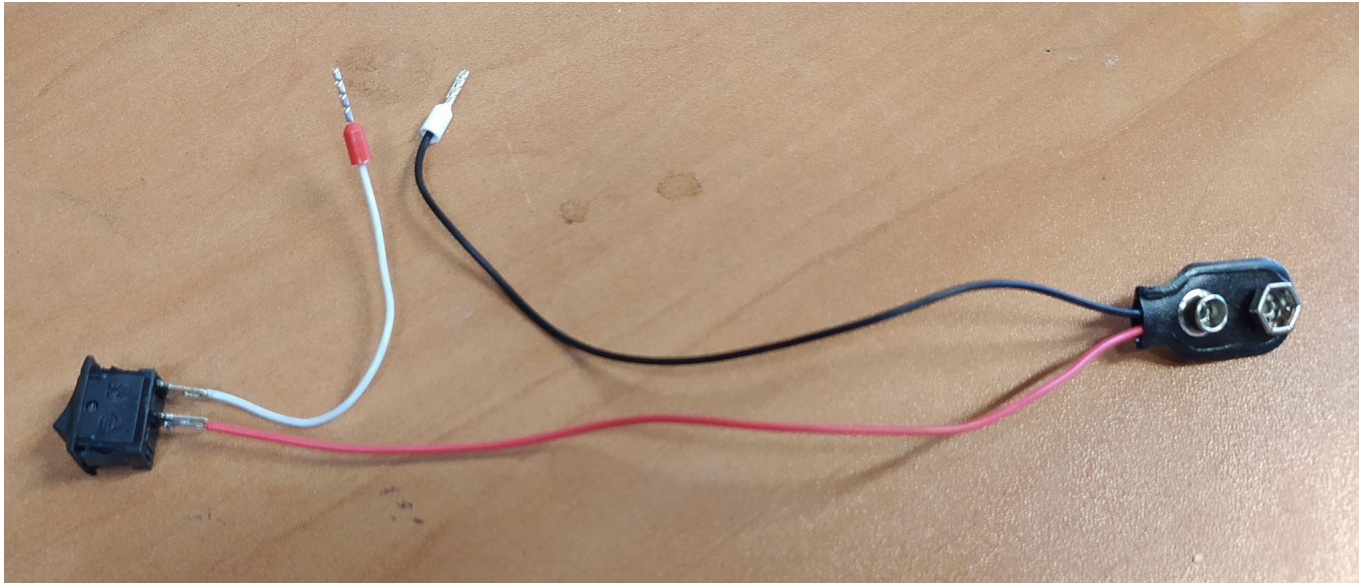
Soudure

Couper et dénuder 5 à 10 cm de câble

Souder le câble rouge du connecteur de pile sur le I de l'interrupteur

Souder le petit câble sur le O de l'interrupteur

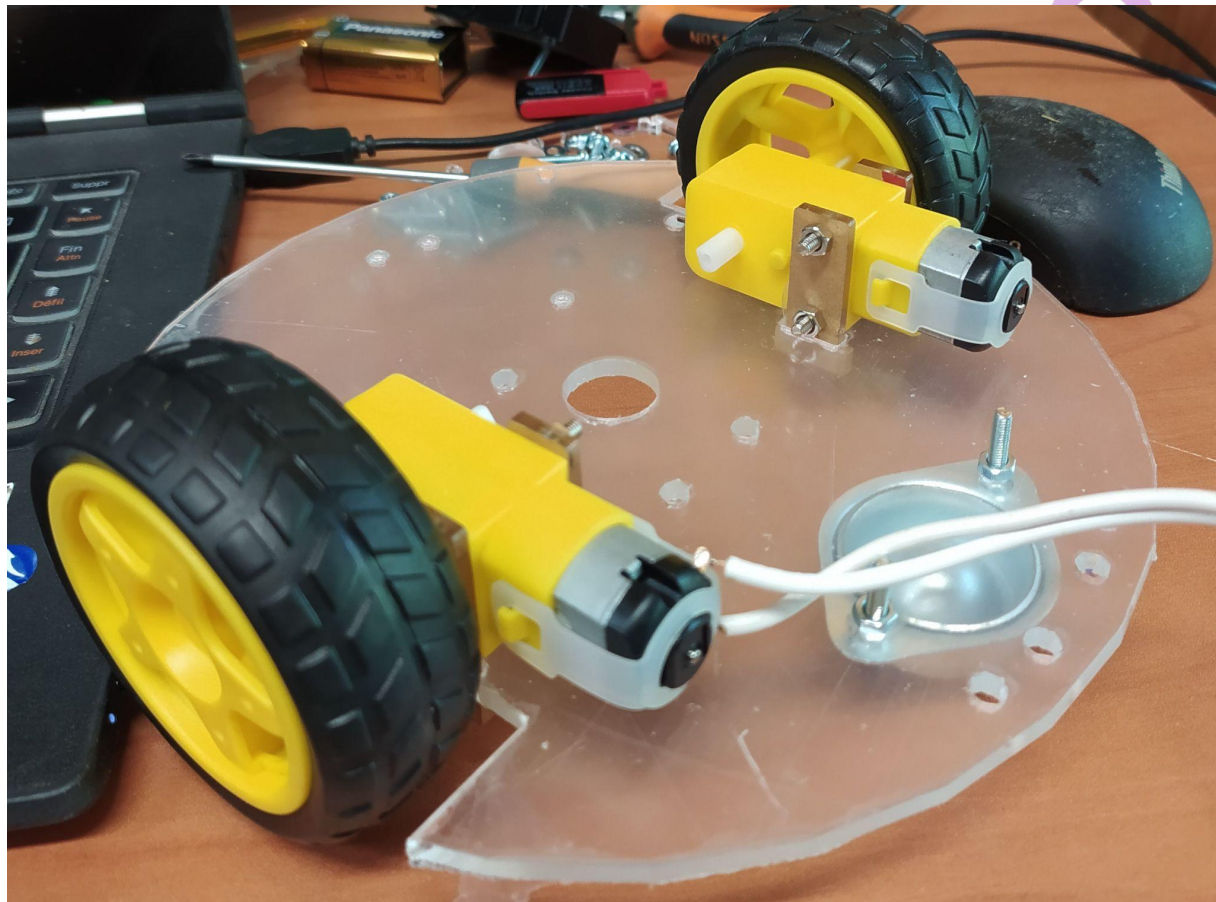
Mettre une cosse sur le petit câble et une sur le câble noir du connecteur de pile



Montage mécanique

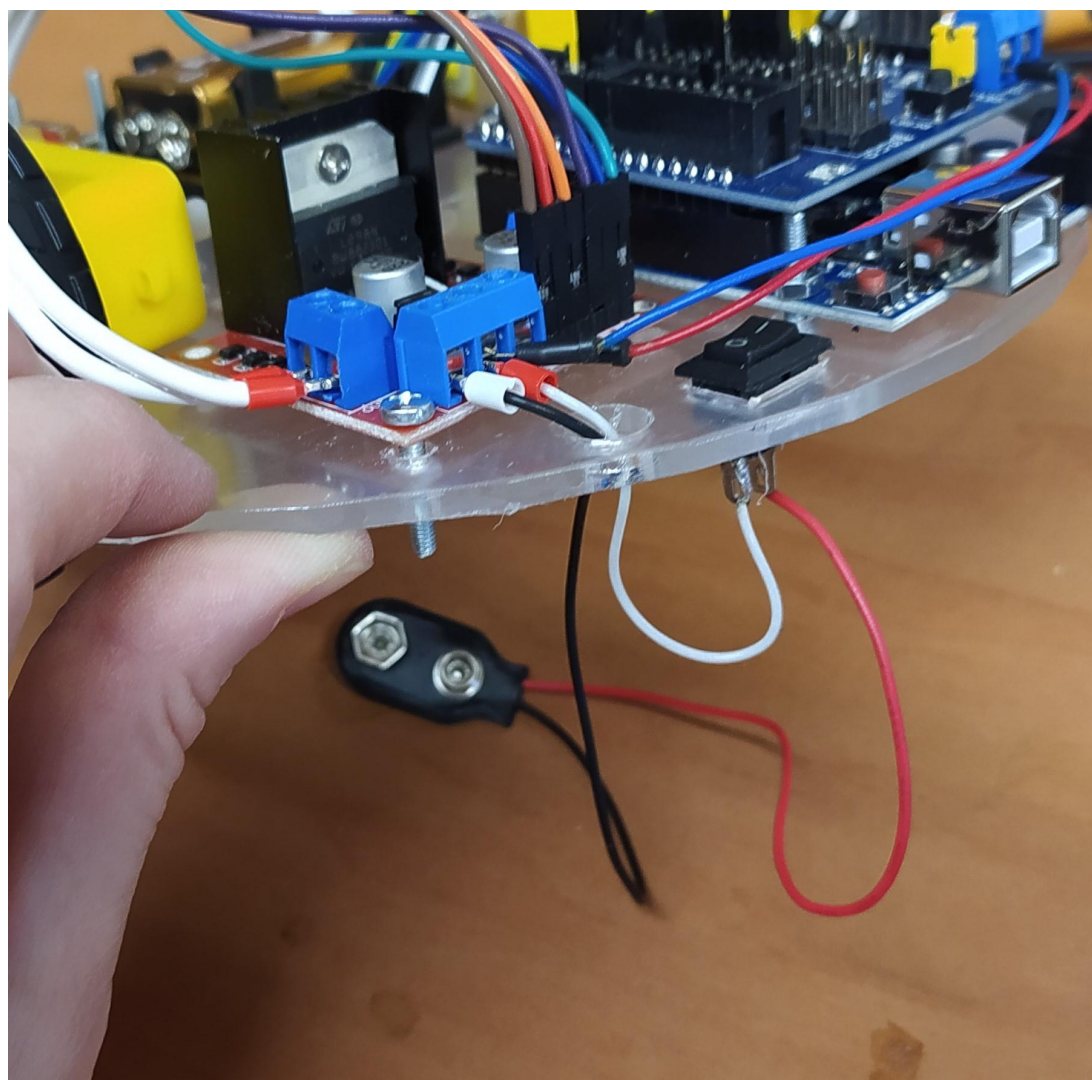
moteurs et roues

roue libre



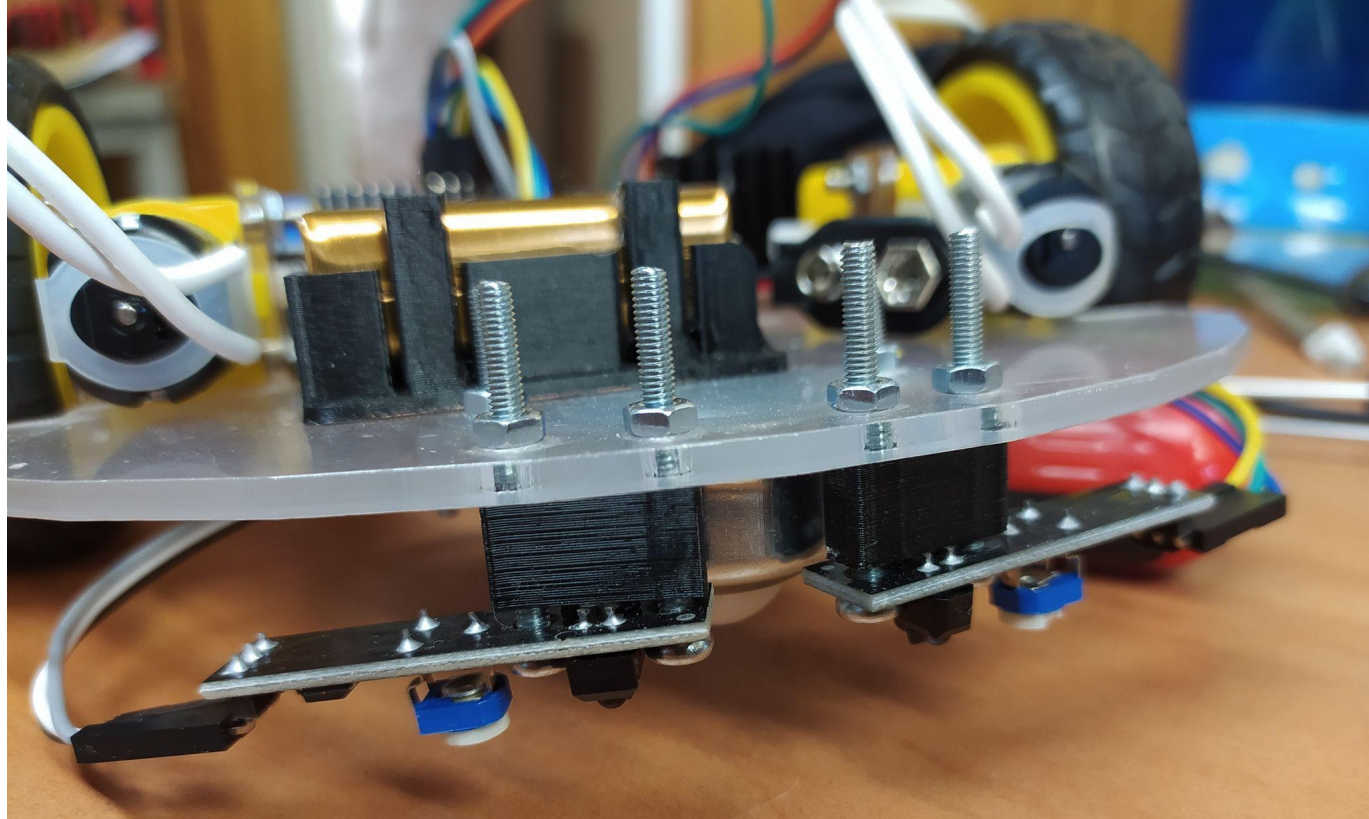
Montage mécanique

support de pile + interrupteur

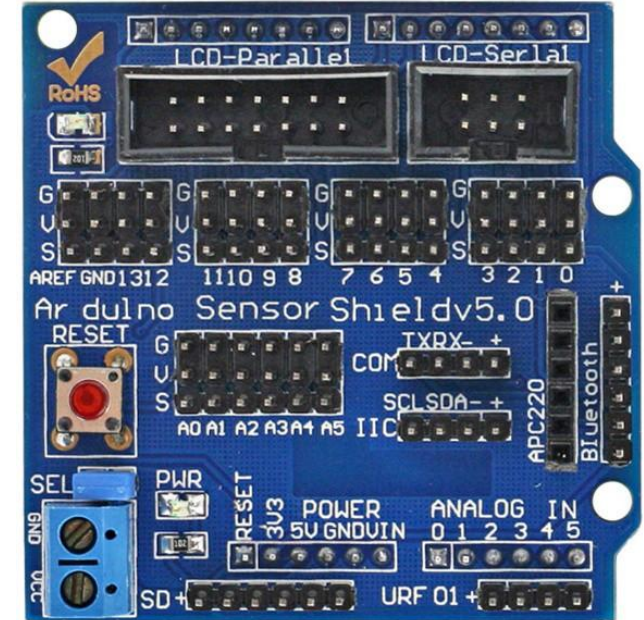
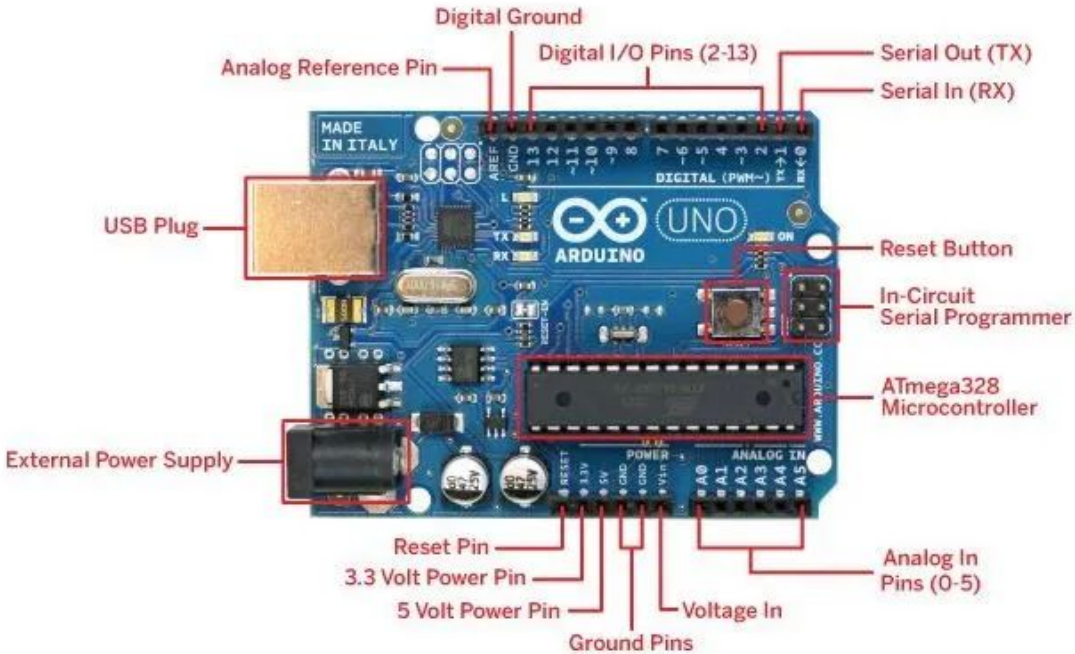


Montage mécanique

capteurs infrarouge



Carte arduino

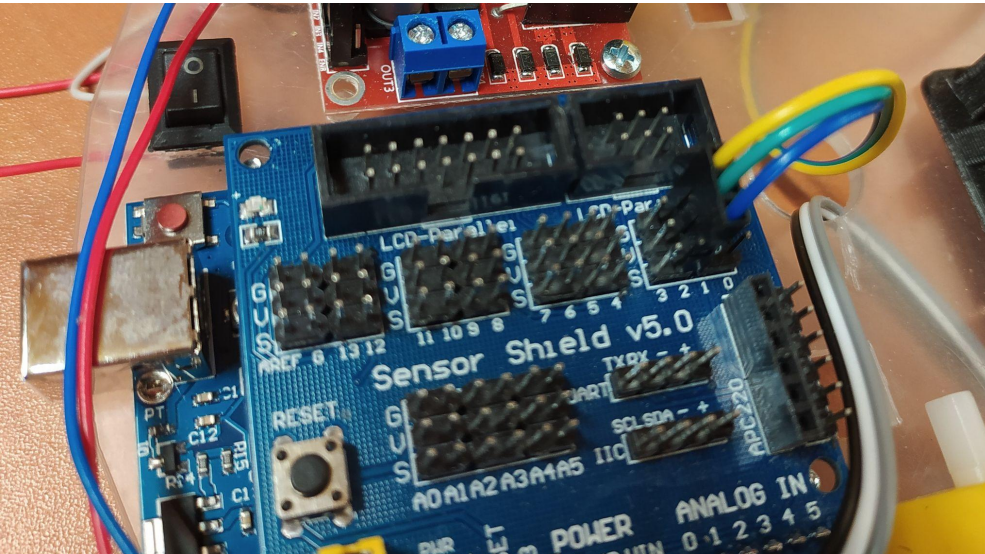


Sensor shield V5.0

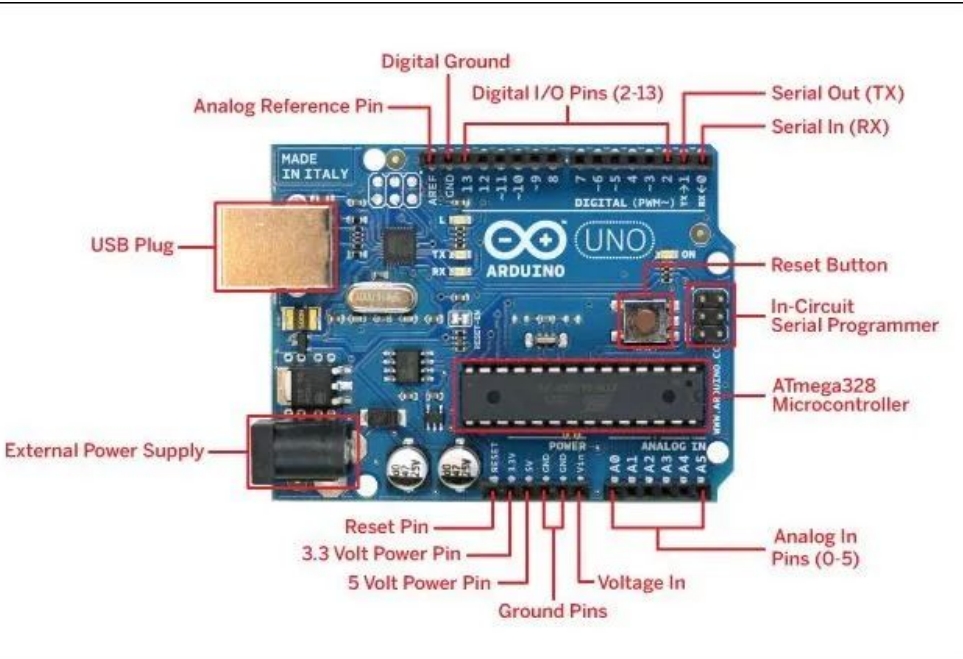
Montage mécanique

carte arduino, carte sensor shield V5

câblage capteur infrarouge sur le 2 et le 3



Carte arduino



```
Blink | Arduino 1.0

/*
 * Blink
 * Turns on an LED on for one second, then off for one second, repeatedly.
 *
 * This example code is in the public domain.
 */

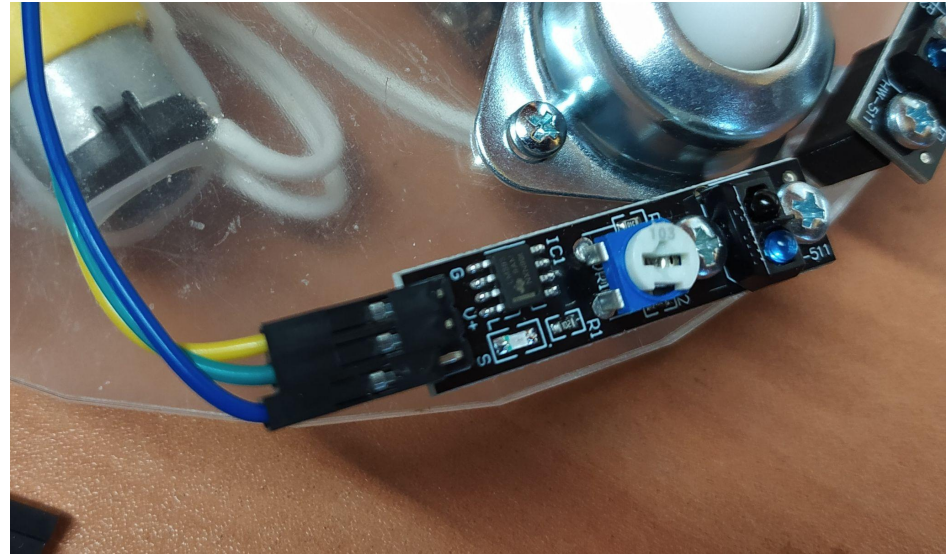
void setup() {
  // initialize the digital pin as an output.
  // Pin 13 has an LED connected on most Arduino boards:
  pinMode(13, OUTPUT);
}

void loop() {
  digitalWrite(13, HIGH); // set the LED on
  delay(1000);            // wait for a second
  digitalWrite(13, LOW);  // set the LED off
  delay(1000);            // wait for a second
}
```

Examples/01.Basics/Blink

Capteurs infrarouge

- alimenter l'arduino avec le cable USB
- Voir que le capteur infrarouge s'allume/s'éteint en fonction de la présence du noir/blanc
- régler les potentiomètres sur le capteurs pour la détection de la ligne



Capteurs infrarouge

- code arduino
- Examples/01.Basics/

DigitalReadSerial

```
#define pin_ir_droit  2
#define pin_ir_gauche 3

void setup() {
  Serial.begin(9600);

  pinMode(pin_ir_droit, INPUT);
  pinMode(pin_ir_gauche, INPUT);
}

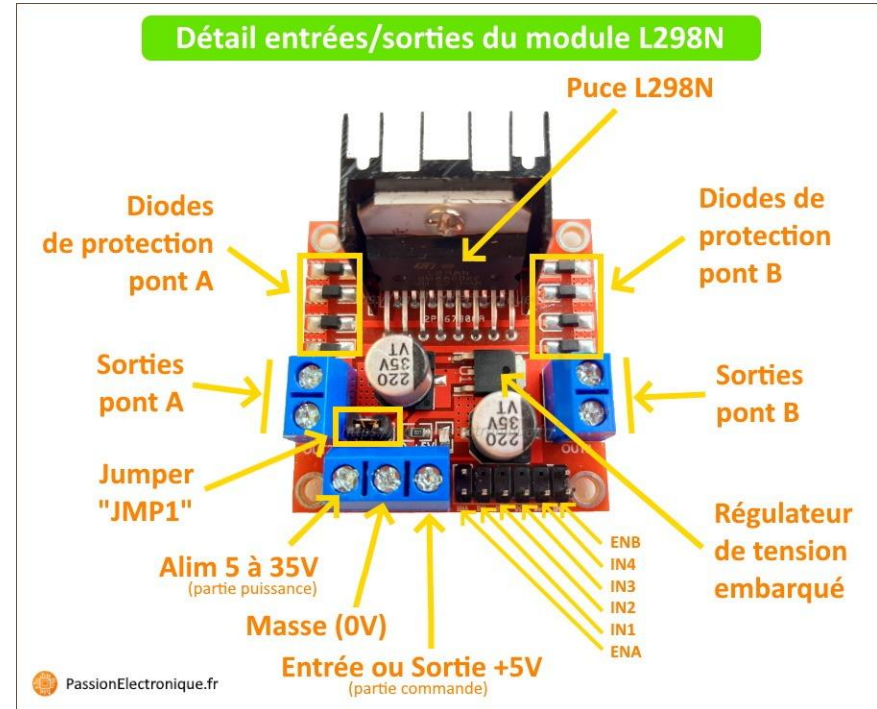
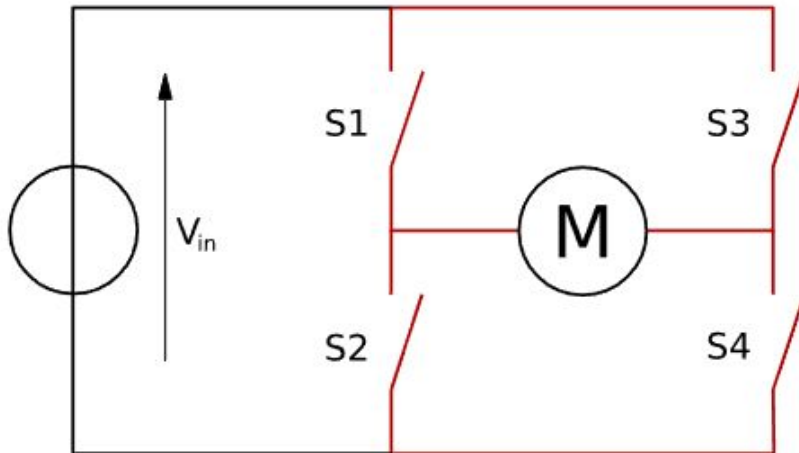
void loop() {
  int noir_a_droite = digitalRead(pin_ir_droit);
  int noir_a_gauche = digitalRead(pin_ir_gauche);

  if (noir_a_droite==1){
    Serial.println("je vois du noir à droite");
  }
  else{
    Serial.println("pas de noir à droite");
  }

  delay(100);
}
```

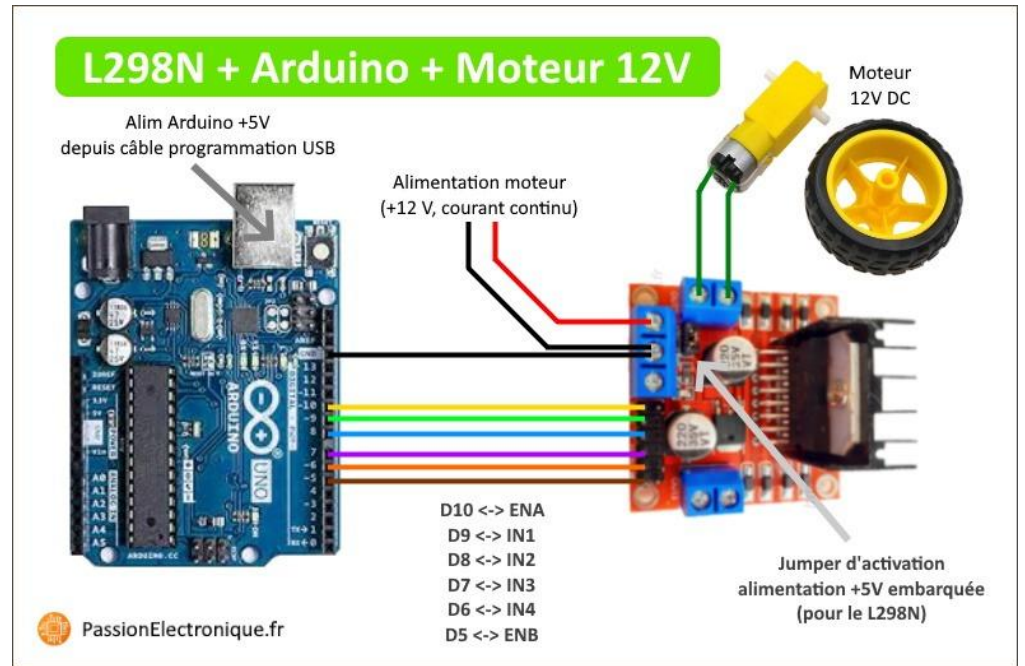
Carte L298N

- alimentation 5V pour la carte arduino
- 2 ponts en H pour la commande des moteurs



Carte L298N

- moteur gauche vers out2
- moteur droit vers out3
- interrupteur vers +12V
- câble noir pile + arduino vers GND et GND sur la carte capteurs
- Câble rouge de +5V vers VCC sur la carte capteurs
- 6 câbles ENA à ENB vers entrées 5 à 10 (pin V) sur la carte capteurs



Carte L298N

PWM : Pulse Width Modulation

25% Duty Cycle



50% Duty Cycle



75% Duty Cycle



← T →

Table de vérité L298N - Fonctionnement

Entrées de sélection				
ENA	IN1	IN2	Résultat, en sortie	1er pont (A) 2ème pont (B)
ENB	IN3	IN4	Résultat, en sortie	
L	X	X	Moteur en roue libre (à l'arrêt, sans frein)	
H	L	L	Blocage du moteur (arrêt rapide, freinage fort)	
	L	H	Marche arrière	
	H	L	Marche avant	
	H	H	Blocage du moteur (arrêt rapide, freinage fort)	

X = peu importe

L = état bas (0V, par exemple)

H = état haut (+5V, par exemple)

Code



- Faire tourner un moteur

```
#define pin_ir_droit    2
#define pin_ir_gauche   3
#define borneENA        10
#define borneIN1        9
#define borneIN2        8

int vitesse = 80; //entre 0 et 255

void setup() {
  Serial.begin(9600);

  pinMode(pin_ir_droit, INPUT);
  pinMode(pin_ir_gauche, INPUT);

  pinMode(borneENA, OUTPUT);
  pinMode(borneIN1, OUTPUT);
  pinMode(borneIN2, OUTPUT);
}
```

```
void loop() {

  int noir_a_droite = digitalRead(pin_ir_droit);
  int noir_a_gauche = digitalRead(pin_ir_gauche);

  if(noir_a_droite){
    // Configuration du L298N en "marche avant"
    digitalWrite(borneIN1, LOW); // L'entrée IN1 doit être au niveau haut
    digitalWrite(borneIN2, HIGH); // L'entrée IN2 doit être au niveau bas
    analogWrite(borneENA, vitesse);
    Serial.println("avance !");
  }
  else{
    analogWrite(borneENA, 0);
    Serial.println("stop !");
  }
  delay(100);
}
```


Programme de suivi de ligne



A vous de jouer



Merci d'avoir suivi cet atelier

