

2021-2022



PROJET ARDUINO MECHA SKARNER

OLIVIERI Rémi
& SAMIEZ Maxime

Groupe 4

<https://github.com/Jokrem/MechaSkarner-Remi-Maxime>

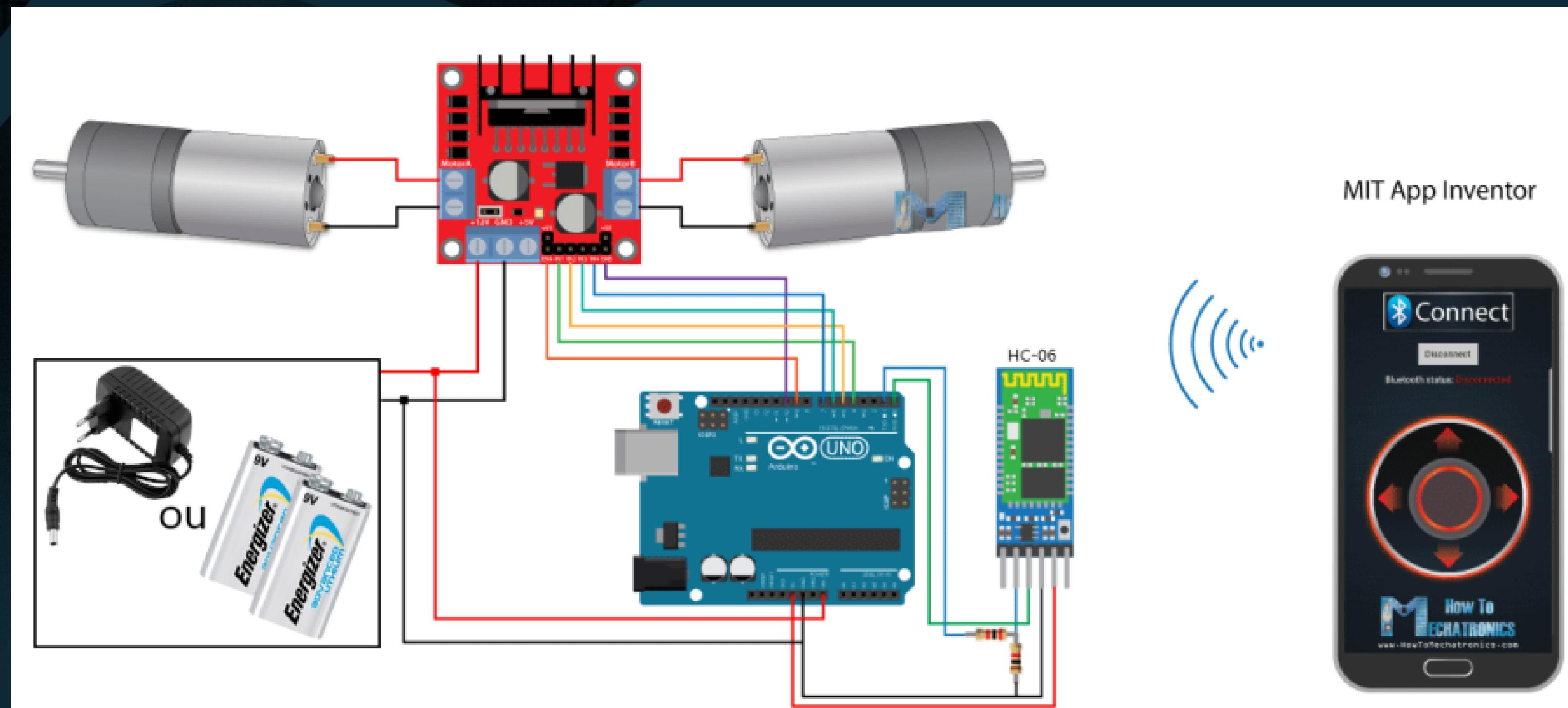


BUT DU PROJET

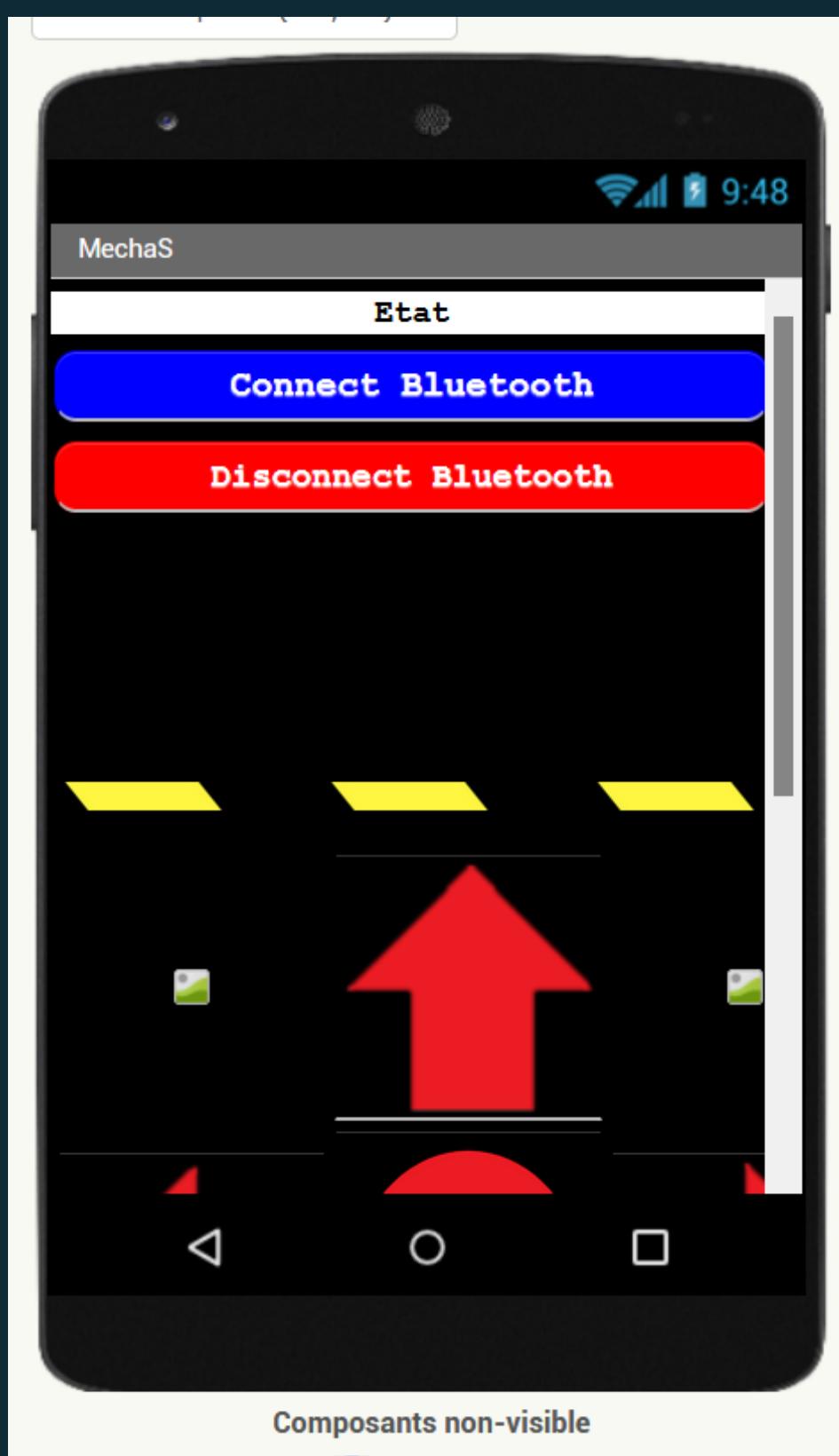
ROUGE / ORANGE / JAUNE / VERT /
BLEU / VIOLET / MARRON

DÉPLACEMENT

- Chenilles/moteurs
- Application Bluetooth
- Boite



MIT APP INVENTOR 2



Composants

- Screen1
 - Etat
 - On
 - Off
- Arrangement_tableau1
 - Bouton1
 - Bouton3
 - Bouton4
- Arrangement_tableau3
 - Image1
 - Image2
 - Image3
 - Image4
 - Bouton5
 - Bouton2
- Arrangement_tableau4
 - Rouge
 - Orange
 - Jaune
 - Vert
 - Bleu
 - Violet
 - Marron
 - Choisir_une_couleur
 - Bouton6
 - Image6
- Bluetooth
- Notification

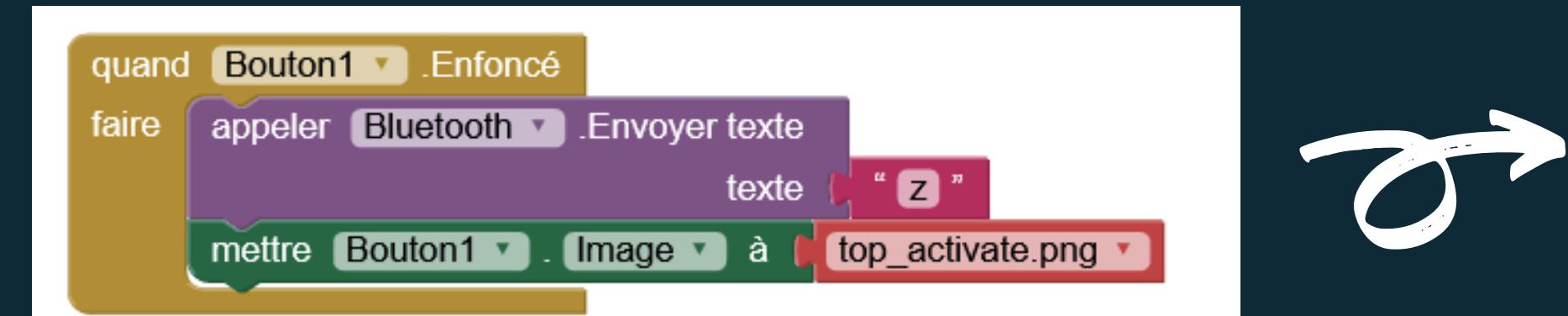
Renommer Supprimer

```
quand Screen1 .Initialise
faire si pas Bluetooth .Activé
alors appeler Notification .Afficher Alerte
    notice "Bluetooth disabled"
    mettre Etat .Texte à "Disconnected"
    mettre Etat .Couleur texte à noir

quand On .Avant prise
faire mettre On .Éléments à Bluetooth .Adresses et noms

quand On .Après prise
faire si appeler Bluetooth .Se connecter
    adresse On .Sélection
alors mettre On .Éléments à Bluetooth .Adresses et noms
    mettre Etat .Texte à "Connected"
    mettre Etat .Couleur texte à vert

quand Off .Clic
faire appeler Bluetooth .Déconnecter
    mettre Etat .Texte à "Disconnected"
    mettre Etat .Couleur texte à noir
```



```

void loop() {
    if (hc06.available()) {
        message = hc06.read();
        if (message != -1) {
            if (message == 'z') {
                GoTop();
            }
        }
    }
}

```

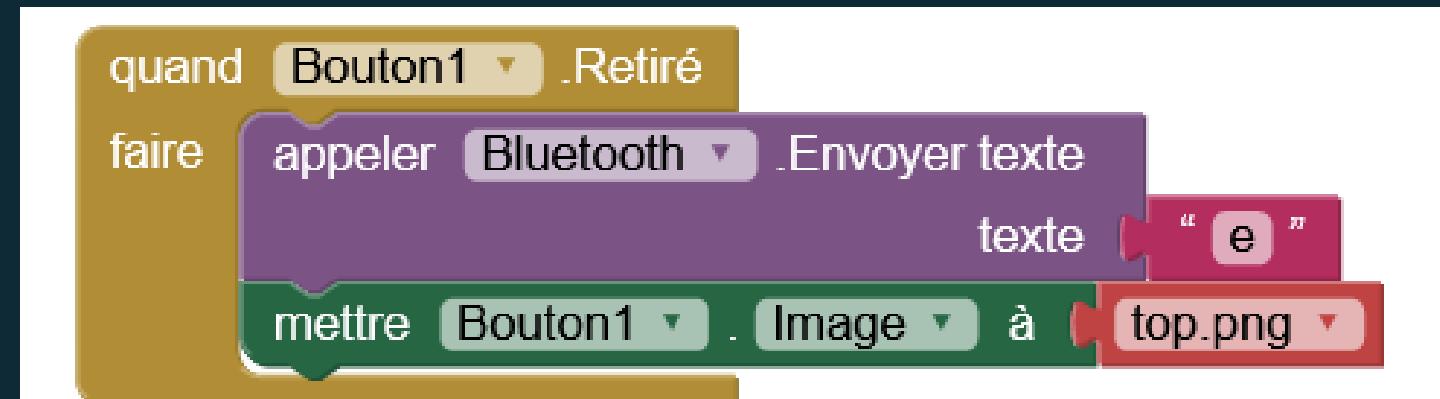
Exemple : avancer

```

void Stop()
{
    Serial.println("Stop");

    //Speed
    digitalWrite(ENA, 0);
    digitalWrite(ENB, 0);
}

```



```

void GoTop() {
    Serial.println("forward");

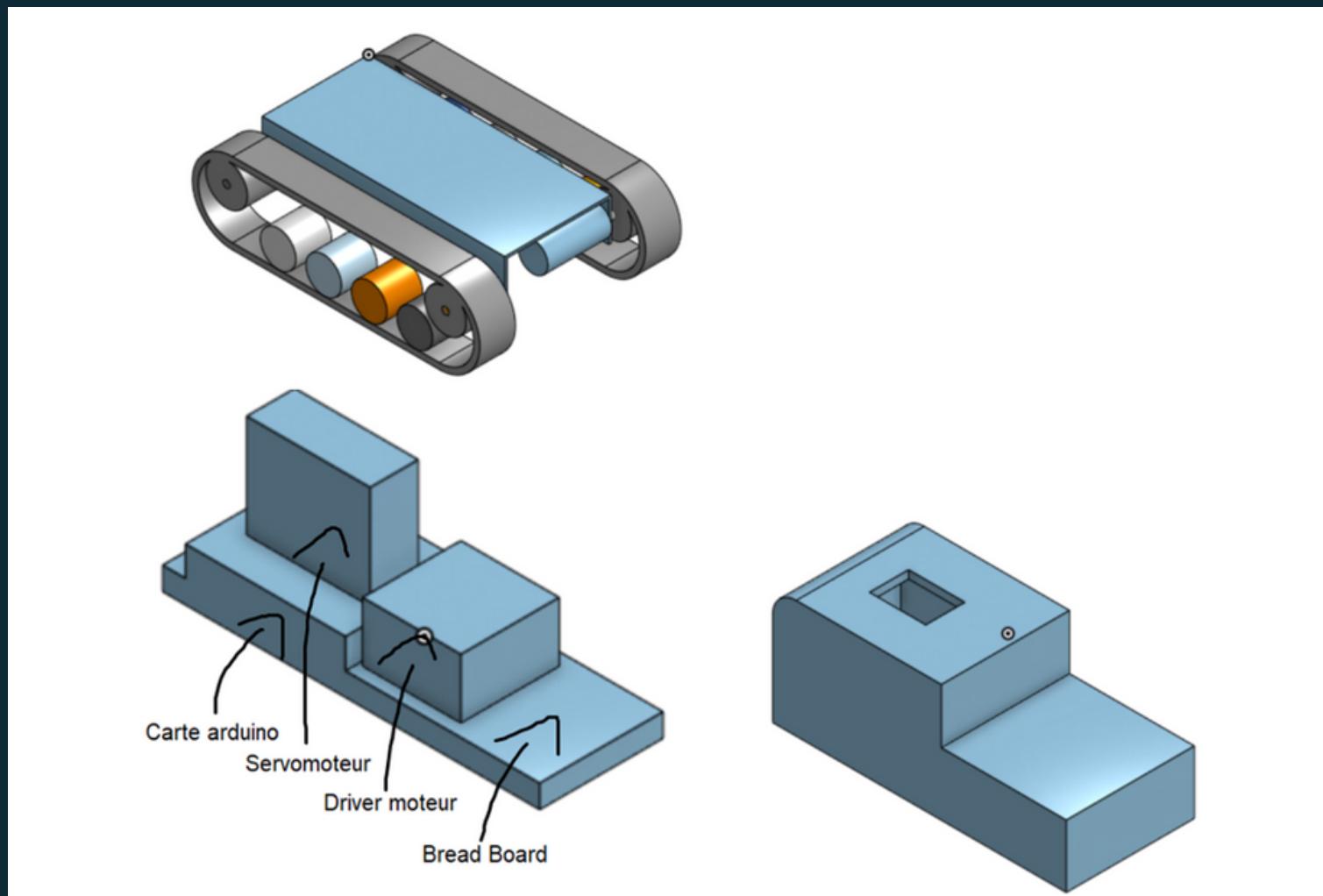
    //Motor A direction
    digitalWrite(IN1, LOW);
    digitalWrite(IN2, HIGH);

    //Motor B direction
    digitalWrite(IN3, LOW);
    digitalWrite(IN4, HIGH);

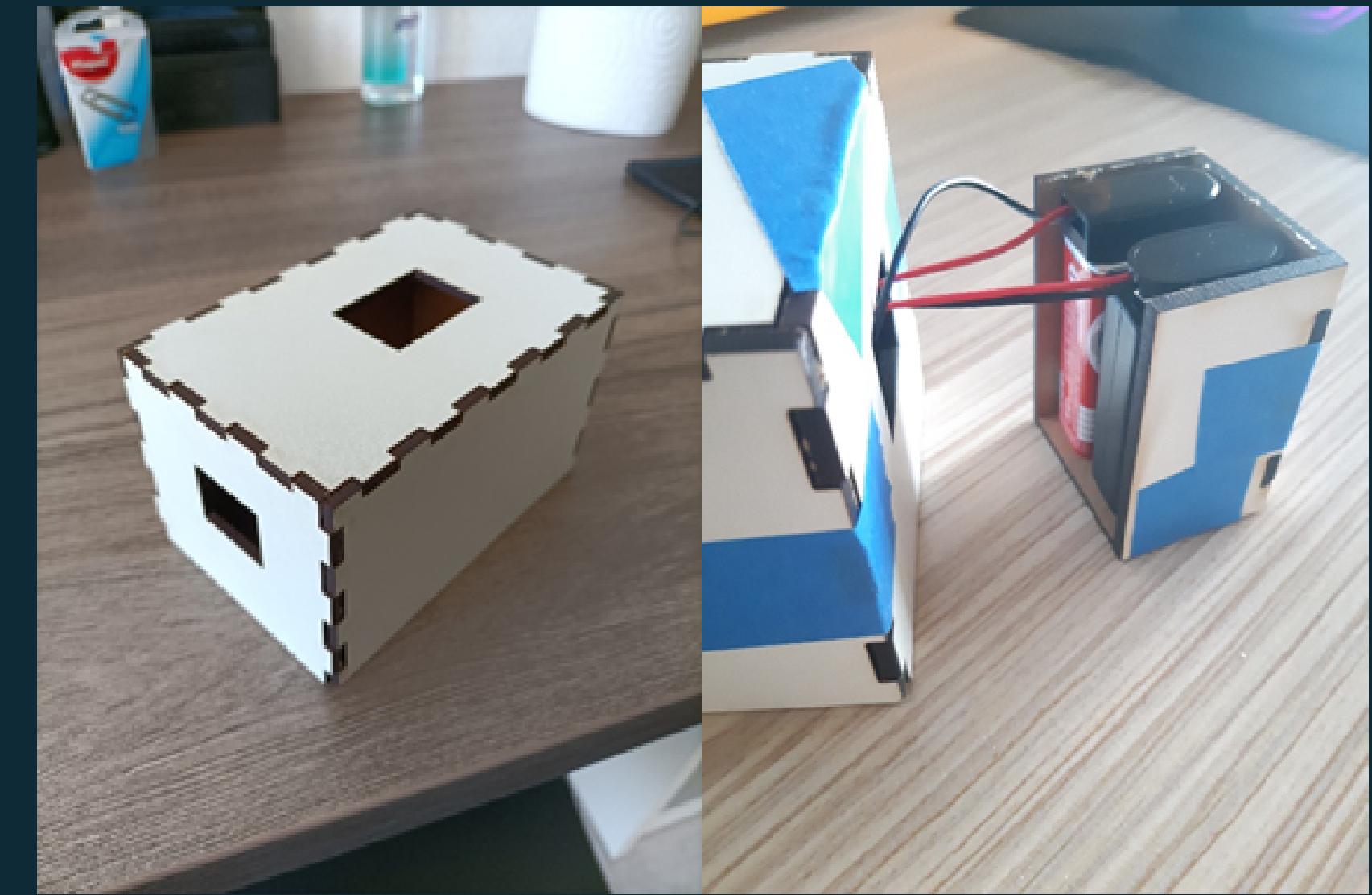
    //Speed
    analogWrite(ENA, 255);
    analogWrite(ENB, 255);
}

```

DESSUS DU TANK



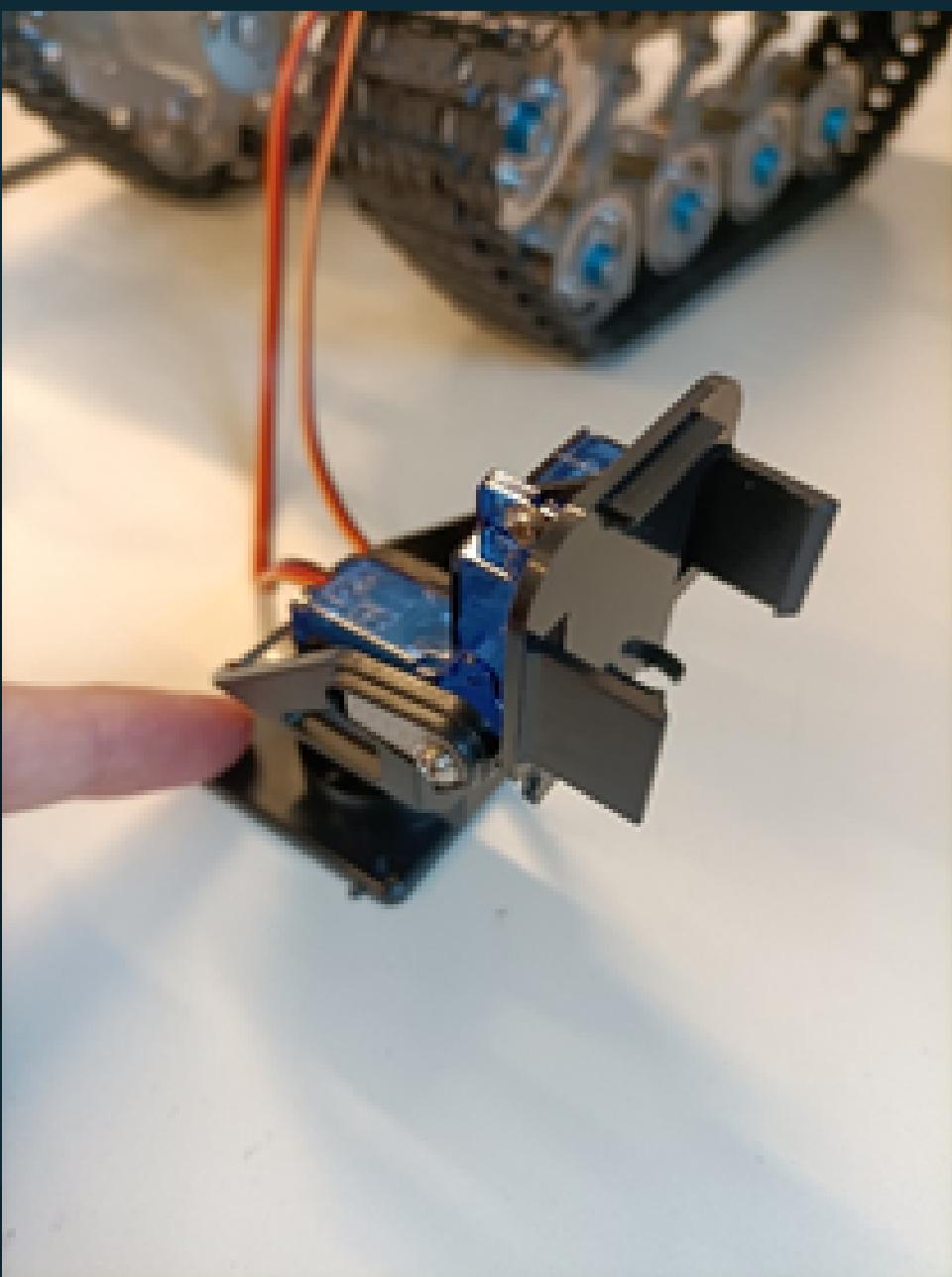
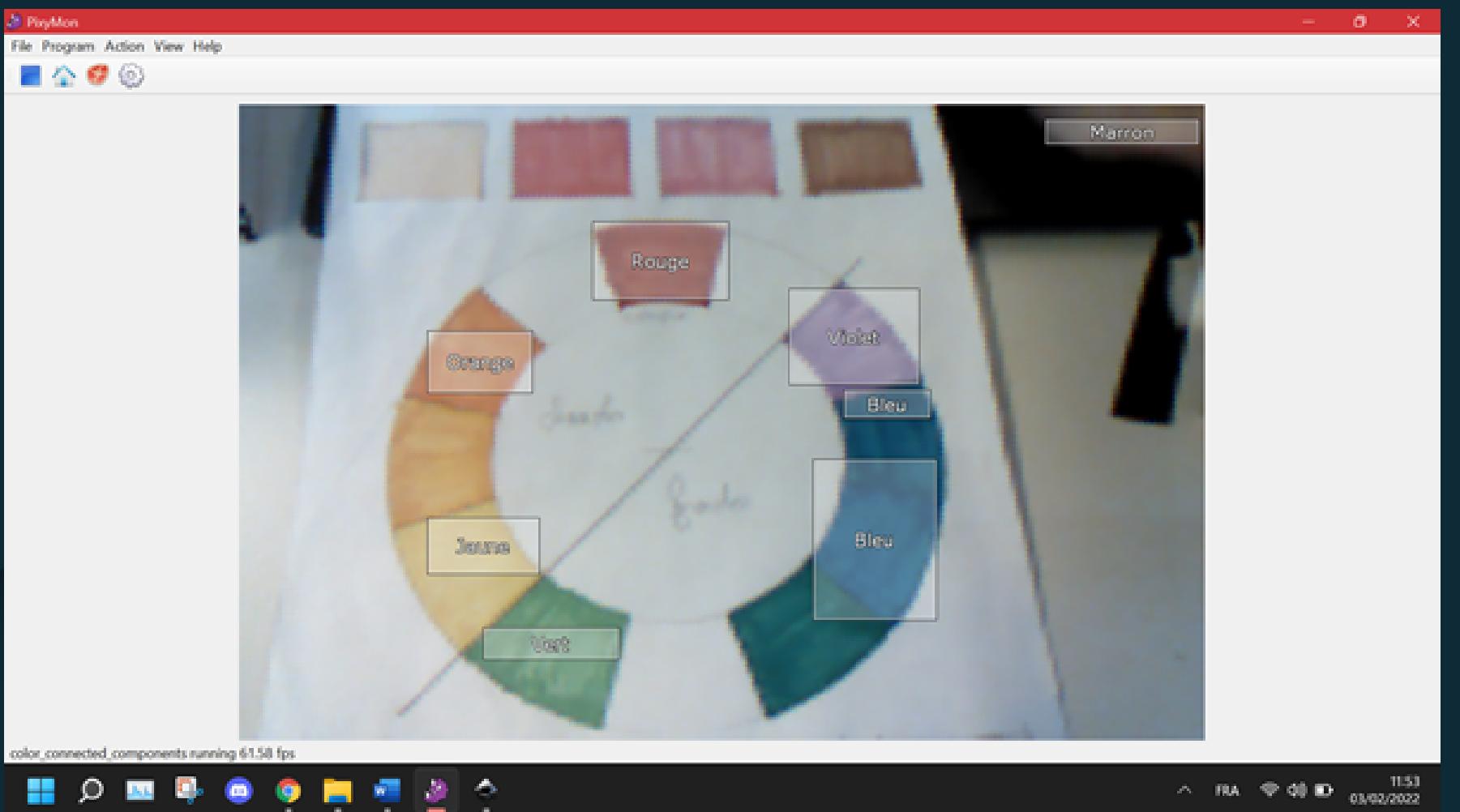
Avant (onshape)



Après (découpe laser)

TIR LASER

- Servomoteurs/laser
- Détection des couleurs avec *Pixy2*



CODE DES SERVO

```
#include <Servo.h>           //Servo
#include <Pixy2.h>            //Camera

Servo turret_bas;
Servo turret_haut;

const int laser = 8;
String couleur = "bleu";

const String liste_coul[] = {"rouge", "orange", "jaune", "vert", "bleu", "violet", "marron"};

int pos_bas = 0;
int pos_haut = 135;

int x = 0;
int y = 135;
```

```
void loop()
{
    turret_haut.write(pos_haut);

    for (int j = 0; j < 150; j++)
    {
        turret_bas.write(j);
        turret_haut.write(135);
        pos_haut = 135;
        pos_bas++;
        delay(25);

        pixy.ccc.getBlocks();

        for (int i = 0; i < pixy.ccc.numBlocks; i++)
        {
            Serial.println(liste_coul[pixy.ccc.blocks[i].m_signature - 1]);
            if (liste_coul[pixy.ccc.blocks[i].m_signature - 1] == couleur)
            {
                int x = pixy.ccc.blocks[i].m_x;
                Serial.println(x);
            }
        }
    }
}
```

```
void setup() {
    Serial.begin(9600);
    turret_bas.attach(A0);
    turret_haut.attach(A1);
    pinMode(laser, OUTPUT);
    pixy.init();
}
```

```
int y = pixy.ccc.blocks[i].m_y;
while (y > 165 or y < 135)
{
    Serial.println(y);
    if (y < 135)
    {
        pos_haut++;
        y += 3;
        turret_haut.write(pos_haut);
        delay(15);
    }
    else
    {
        pos_haut--;
        y -= 3;
        turret_haut.write(pos_haut);
        delay(15);
    }
}
```

```
Serial.println("Tir");
digitalWrite(laser, HIGH);
delay(1000);
digitalWrite(laser, LOW);
}
}

for (int i=pos_bas; i>0; i--)
{
    turret_bas.write(i);
    pos_bas--;
    delay(15);
}
```



Jokrem/MechaSkarner-Remi-Maxime

Contribute to Jokrem/MechaSkarner-Remi-Maxime development by creating an account on GitHub.



MERCI !