Ngày 15 tháng 5 năm 2023

Buổi sáng

Buổi chiều

1. Code chương trình

```
#include <AFMotor.h>

AF_DCMotor motor1(5); // Create motor object for motor 1
AF_DCMotor motor2(3); // Create motor object for motor 2
float pTerm, iTerm, dTerm;
int error;
int previousError;
float kp = 10;
float ki = 0;
float kd = 10;
float output;
int integral,  derivative;

int R1 = A1;
int R2 = A2;
int M  = A3;
int L2 = A4;
int L1 = A5;

int motor1Forward  = 11;
int motor2Forward  = 6;
int motor1Backward = 10;
int motor2Backward = 9;
//int motor1pwmPin   = 5;
//int motor2pwmPin   = 3;
int motor1newSpeed;
int motor2newSpeed;
int s              = 100;
int motor1Speed    = s; //Default r
int motor2Speed    = s; //Default l

void setup() {
  //Declare all IR sensors  as inputs
  pinMode(L1, INPUT);   // L1
  pinMode(L2, INPUT);   // L2
  pinMode(M, INPUT);   // M
  pinMode(R2, INPUT);   // R2
  pinMode(R1, INPUT);   // R1
  pinMode(motor1Forward, OUTPUT);
  pinMode(motor1Backward,  OUTPUT);
  pinMode(motor2Forward, OUTPUT);
  pinMode(motor2Backward, OUTPUT);
  Serial.begin(9600);
```

```
}

void  loop() {
  //Put all of our functions here
  if ((digitalRead(L1) == 1) && (digitalRead(L2) == 1) && (digitalRead(M) == 1) && (digitalRead(R2) == 1)
&& (digitalRead(R1) == 1)) {
    stopMotors();
  }
  calculateError();
  pidCalculations();
  changeMotorSpeed();

  Serial.print("R1 : ");Serial.println(digitalRead(R1));
  Serial.print("R2 : ");Serial.println(digitalRead(R2));
  Serial.print("M  : ");Serial.println(digitalRead(M));
  Serial.print("L2 : ");Serial.println(digitalRead(L2));
  Serial.print("L1 : ");Serial.println(digitalRead(L1));

  Serial.print("motor1newSpeed: ");
  Serial.println(motor1newSpeed);
  Serial.print("motor2newSpeed: ");
  Serial.println(motor2newSpeed);

  delay(1000);
}

//đổi digital thành analog
void calculateError() {
  //Determine  an error based on the readings
  if       ((digitalRead(L1)== 1) && (digitalRead(L2) == 0) && (digitalRead(M) == 0) &&
(digitalRead(R2)== 0) && (digitalRead(R1) == 0)) {
    error = 4; // extreme right sensor
  } else if ((digitalRead(L1)== 1) && (digitalRead(L2) == 1) && (digitalRead(M) == 0) &&
(digitalRead(R2)== 0) && (digitalRead(R1) == 0)) {
    error = 3;// right
  } else if  ((digitalRead(L1)== 1) && (digitalRead(L2) == 1) && (digitalRead(M) == 1) &&
(digitalRead(R2)== 0) && (digitalRead(R1) == 0)) {
    error = 2;
  } else if  ((digitalRead(L1)== 1) && (digitalRead(L2) == 1) && (digitalRead(M) == 1) &&
(digitalRead(R2)== 0) && (digitalRead(R1) == 1)) {
    error = 1;
  } else if ((digitalRead(L1)== 1) && (digitalRead(L2) == 1) && (digitalRead(M) == 0) &&
(digitalRead(R2)== 0) && (digitalRead(R1) == 1)) {
    error = 1;
  } else if ((digitalRead(L1)== 1) && (digitalRead(L2) == 0) && (digitalRead(M) == 0) &&
(digitalRead(R2)== 0) && (digitalRead(R1) == 1)) {
    error = 0;
  } else if ((digitalRead(L1)== 1) && (digitalRead(L2) == 1) && (digitalRead(M) == 0) &&
(digitalRead(R2)== 1) && (digitalRead(R1) == 1)) {
```

```
    error = 0;
  } else if ((digitalRead(L1)== 1) && (digitalRead(L2) == 0) && (digitalRead(M) == 1) &&
(digitalRead(R2)== 1) && (digitalRead(R1) == 1)) {
    error = -1;
  } else if ((digitalRead(L1)== 1) && (digitalRead(L2) == 0) && (digitalRead(M) == 0) &&
(digitalRead(R2)== 1) && (digitalRead(R1) == 1)) {
    error = -1;
  } else if ((digitalRead(L1)== 0) && (digitalRead(L2) == 0) && (digitalRead(M) == 1) &&
(digitalRead(R2)== 1) && (digitalRead(R1) == 1)) {
    error = -4;
  } else if ((digitalRead(L1)== 0) && (digitalRead(L2) == 0) && (digitalRead(M) == 0) &&
(digitalRead(R2)== 1) && (digitalRead(R1) == 1)) {
    error  = -6;
  } else if ((digitalRead(L1)== 0) && (digitalRead(L2) == 0) && (digitalRead(M) == 0) &&
(digitalRead(R2)== 0) && (digitalRead(R1) == 1)) {
    error = -8;//extreme left sensor
  } else  if ((digitalRead(L1)== 0) && (digitalRead(L2) == 0) && (digitalRead(M) == 0) &&
(digitalRead(R2)== 0) && (digitalRead(R1) == 0)) {
    error = 0;
  }

}

void pidCalculations()  {
  pTerm = kp * error;
  integral += error;
  iTerm = ki * integral;
  derivative = error - previousError;
  dTerm = kd * derivative;
  output  = pTerm + iTerm + dTerm;
  previousError = error;
}

void stopMotors() {
  motor1.run(RELEASE); // set PWM to 0 to stop motor 1
  motor2.run(RELEASE); // set PWM to 0 to stop motor 2
  digitalWrite(motor1Forward, LOW);
  digitalWrite(motor1Backward, LOW);
  digitalWrite(motor2Forward, LOW);
  digitalWrite(motor2Backward, LOW);
}

void changeMotorSpeed() {
```

2. Kết luận

Chạy thử kiểm tra lỗi xung PWM nguyên ban đầu xác định là do L298n không nhận được xung

Ngày 16 tháng 5 năm 2023

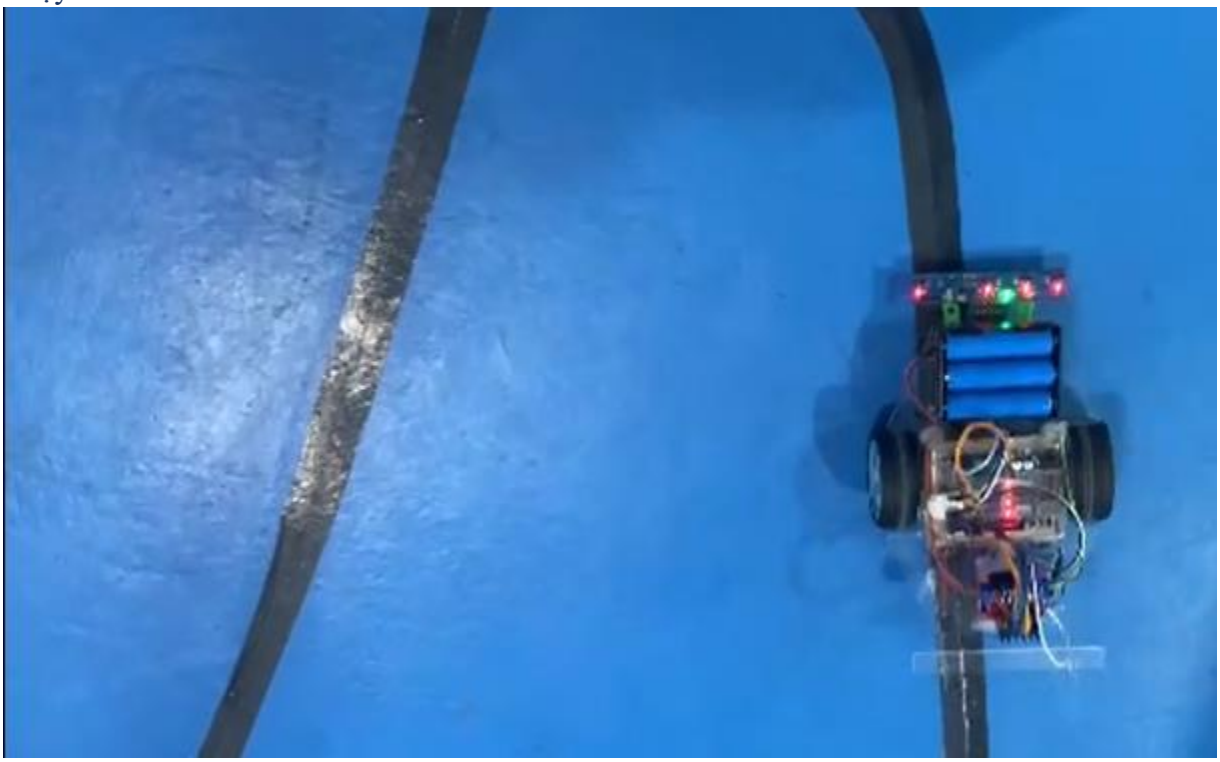
Buổi sáng

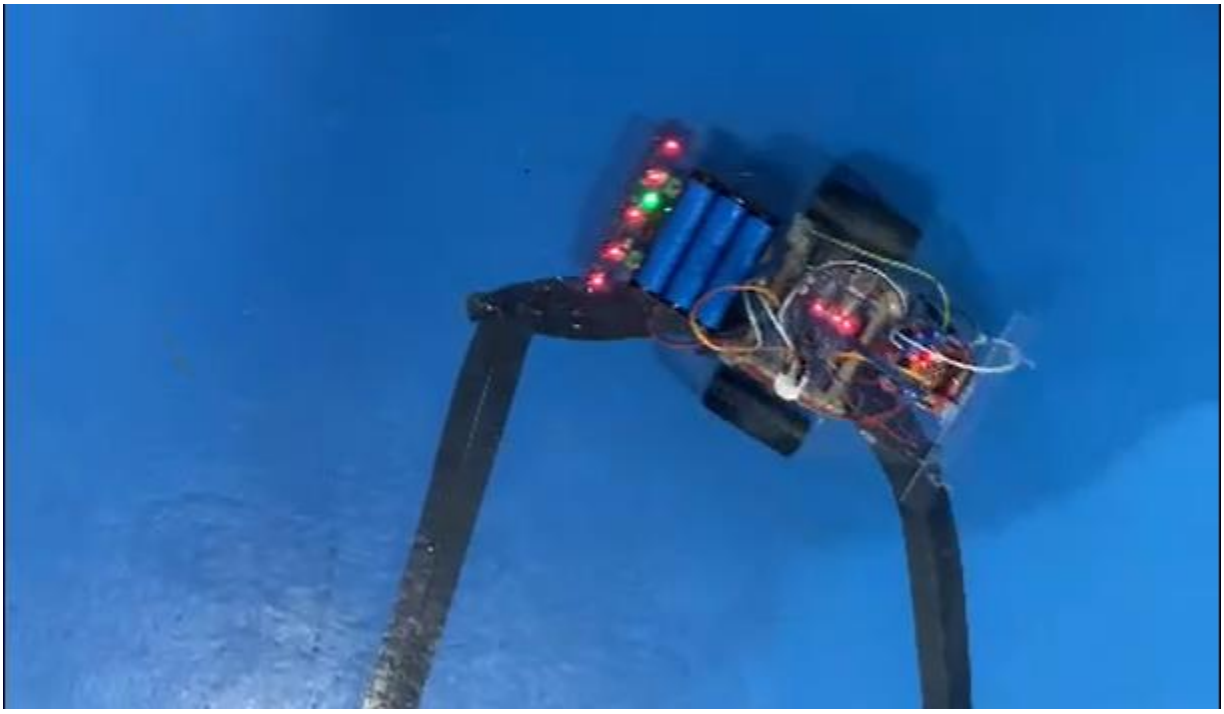
Thiết kế mô phỏng cam demo1

Buổi chiều

- Đã mua mới module L298n xe vẫn không bắt line
- Lỗi xác định do Arduino không phát xung PWM
- Thay Arduino và đã sửa được lỗi xe chạy bám theo line

Chạy thử lần 1

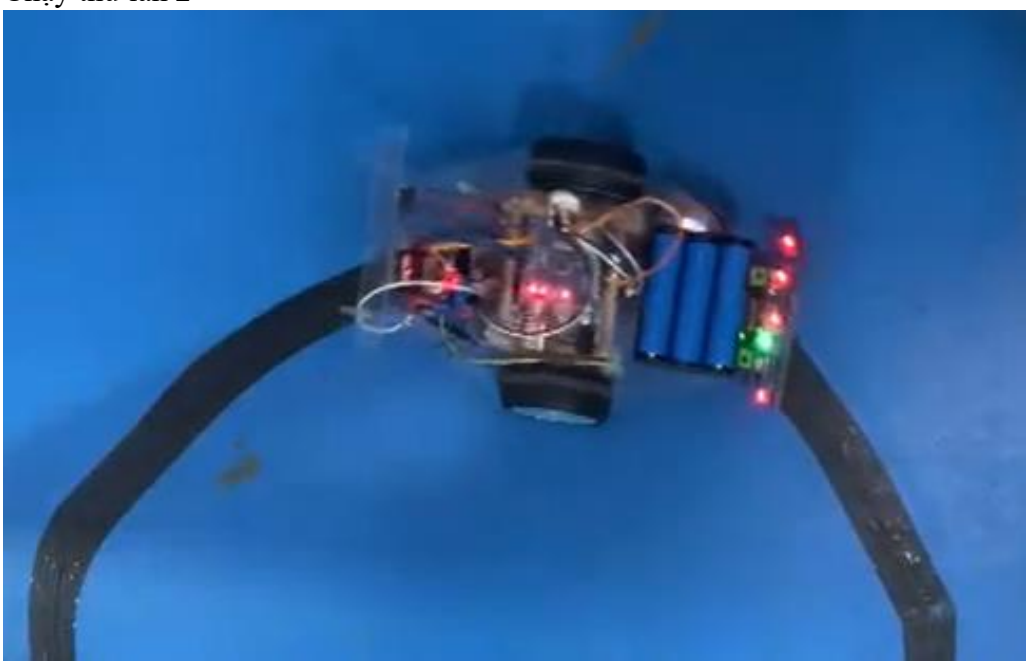


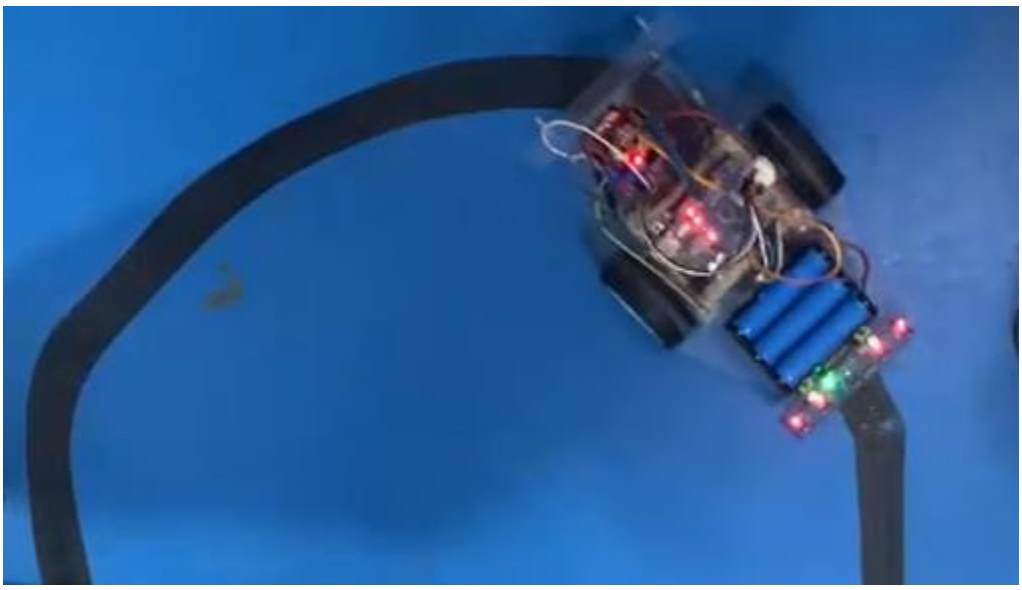


Kết luận

Lỗi do vòng lặp loop có delay(1000) – dừng lại 0.5s

Chạy thử lần 2





Kết luận

Xe chạy đúng theo yêu cầu

Ngày 17 tháng 5 năm 2023

Buổi sáng

Code đã được tối ưu

```
float pTerm, iTerm, dTerm;
int error;
int previousError;
float kp = 5;
float ki = 0;
float kd = 5;
float output;
int integral,  derivative;

int R1 = A1;
int R2 = A2;
int M  = A3;
int L2 = A4;
int L1 = A5;

int motor1Forward = 12;
int motor2Forward = 6;
int motor1Backward = 13;
int motor2Backward = 11;
int motor1pwmPin   = 3;
int motor2pwmPin   = 5;
int motor1newSpeed;
int motor2newSpeed;
int s              = 85;
int motor1Speed    = s; //Default r
int motor2Speed    = s; //Default l

void setup() {
  //Declare all IR sensors  as inputs
  pinMode(L1, INPUT);   // L1
  pinMode(L2, INPUT);   // L2
  pinMode(M, INPUT);   // M
  pinMode(R2, INPUT);   // R2
  pinMode(R1, INPUT);   // R1
  pinMode(motor1Forward, OUTPUT);
  pinMode(motor1Backward,  OUTPUT);
  pinMode(motor1pwmPin, OUTPUT);
  pinMode(motor2Forward, OUTPUT);
  pinMode(motor2Backward, OUTPUT);
  pinMode(motor2pwmPin, OUTPUT);

  Serial.begin(9600);
}

void  loop() {
```

```
  //Put all of our functions here
  calculateError();
  pidCalculations();
  changeMotorSpeed();

  Serial.print("R1 : ");Serial.println(digitalRead(R1));
  Serial.print("R2 : ");Serial.println(digitalRead(R2));
  Serial.print("M  : ");Serial.println(digitalRead(M));
  Serial.print("L2 : ");Serial.println(digitalRead(L2));
  Serial.print("L1 : ");Serial.println(digitalRead(L1));

  Serial.print("motor1newSpeed: ");
  Serial.println(motor1newSpeed);
  Serial.print("motor2newSpeed: ");
  Serial.println(motor2newSpeed);

  //delay(1000);
}
void calculateError() {
  int sensorValues[5];
  sensorValues[0] = digitalRead(R1);
  sensorValues[1] = digitalRead(R2);
  sensorValues[2] = digitalRead(M);
  sensorValues[3] = digitalRead(L2);
  sensorValues[4] = digitalRead(L1);

  int errorValues[13][5] = {
    {0, 1, 1, 1, 1},   // 6
    {0, 0, 1, 1, 1},   // 5
    {0, 0, 0, 1, 1},   // 4
    {0, 0, 0, 0, 1},   // 3
    {1, 0, 1, 1, 1},   // 2
    {1, 0, 0, 1, 1},   // 1
    {1, 1, 0, 1, 1},   // 0
    {1, 1, 0, 0, 1},   // -1
    {1, 1, 1, 0, 1},   // -2
    {1, 0, 0, 0, 0},   // -3
    {1, 1, 0, 0, 0},   // -4
    {1, 1, 1, 0, 0},   // -5
    {1, 1, 1, 1, 0},   // -6
  };

  for (int i = 0; i < 13; i++) {
    bool match = true;
    for (int j = 0; j < 5; j++) {
      if (sensorValues[j] != errorValues[i][j]) {
        match = false;
        break;
      }
```

```
    }
    if (match) {
      error = i - 6;
      break;
    }
  }
}
void pidCalculations()  {
  pTerm = kp * error;
  integral += error;
  iTerm = ki * integral;
  derivative = error - previousError;
  dTerm = kd * derivative;
  output  = pTerm + iTerm + dTerm;
  previousError = error;
}


void changeMotorSpeed() {
  // Check if all IR readings are 0
  if ((digitalRead(L1) == 0) && (digitalRead(L2) == 0) && (digitalRead(M) == 0) && (digitalRead(R2) == 0)
&& (digitalRead(R1) == 0)) {
    // Perform specific action when all IR readings are 0
    // For example, stop the motors
    analogWrite(motor2pwmPin, 0);
    analogWrite(motor1pwmPin, 0);
    digitalWrite(motor1Forward, LOW);
    digitalWrite(motor2Forward, LOW);
    digitalWrite(motor1Backward, LOW);
    digitalWrite(motor2Backward, LOW);
  } else if ((digitalRead(L1) == 1) && (digitalRead(L2) == 1) && (digitalRead(M) == 1) && (digitalRead(R2)
== 1) && (digitalRead(R1) == 1)) {
    // Perform specific action when all IR readings are 1
    // For example, move backward
    analogWrite(motor2pwmPin, 70);
    analogWrite(motor1pwmPin, 70);
    digitalWrite(motor1Forward, LOW);
    digitalWrite(motor2Forward, LOW);
    digitalWrite(motor1Backward, HIGH);
    digitalWrite(motor2Backward, HIGH);
  } else {
    // Change motor speed of both motors accordingly
    motor2newSpeed = motor2Speed + output;
    motor1newSpeed = motor1Speed - output;
    // Constrain the new speed of motors to be between the range 0-255
    motor2newSpeed = constrain(motor2newSpeed, 0, 255);
    motor1newSpeed = constrain(motor1newSpeed, 0, 255);
    // Set new speed and run motors in the forward direction
    analogWrite(motor2pwmPin, motor2newSpeed);
```

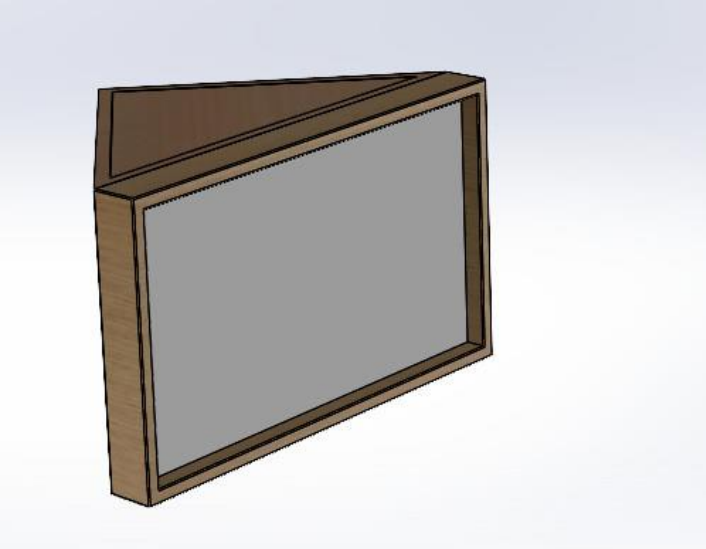
```
    analogWrite(motor1pwmPin, motor1newSpeed);
    digitalWrite(motor1Forward, HIGH);
    digitalWrite(motor2Forward, HIGH);
    digitalWrite(motor1Backward, LOW);
    digitalWrite(motor2Backward, LOW);
  }
}
```

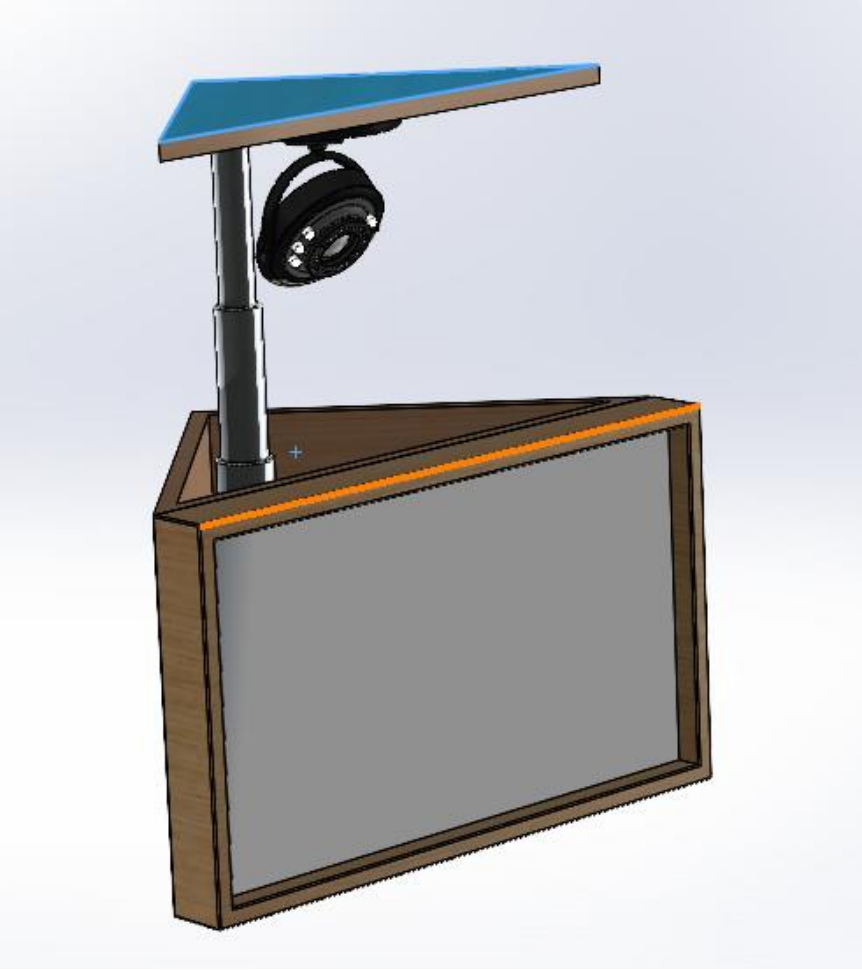
Buổi chiều

Mặc đồng phục lên hội trường theo hướng dẫn của cô Hạnh.

Ngày 18 tháng 5 năm 2023

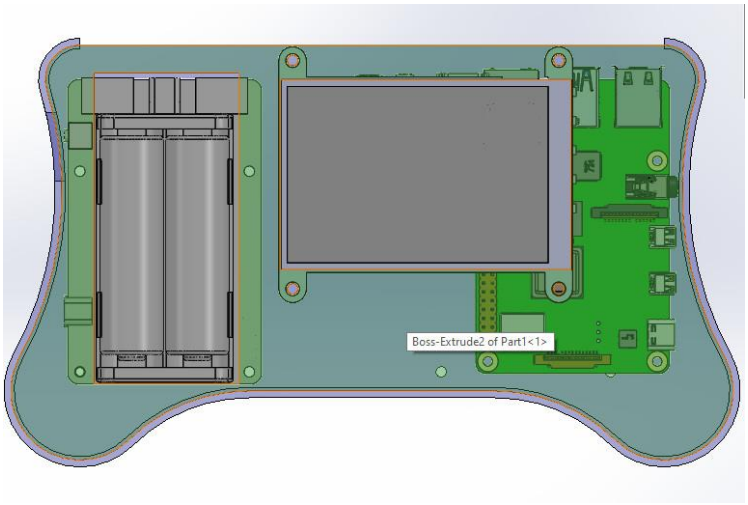
Buổi sáng

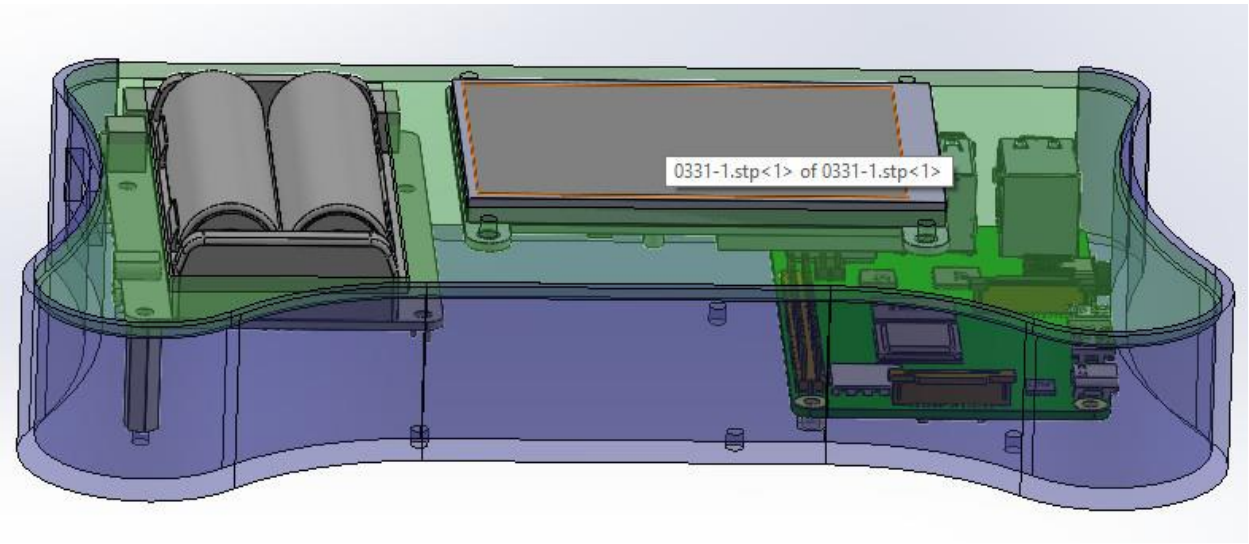


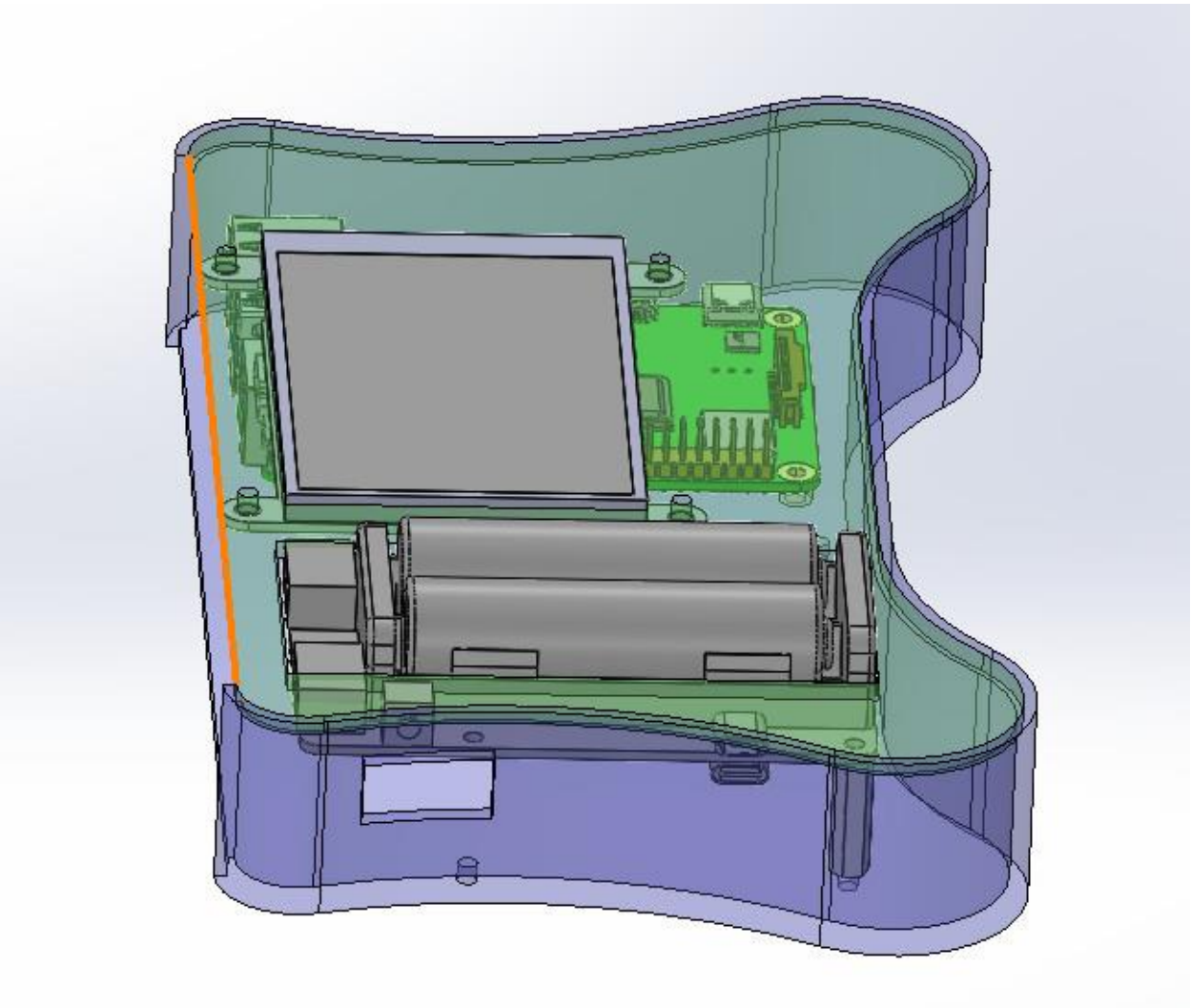


Buổi chiều

Thiết kế tay cầm điều khiển cho robot nhện



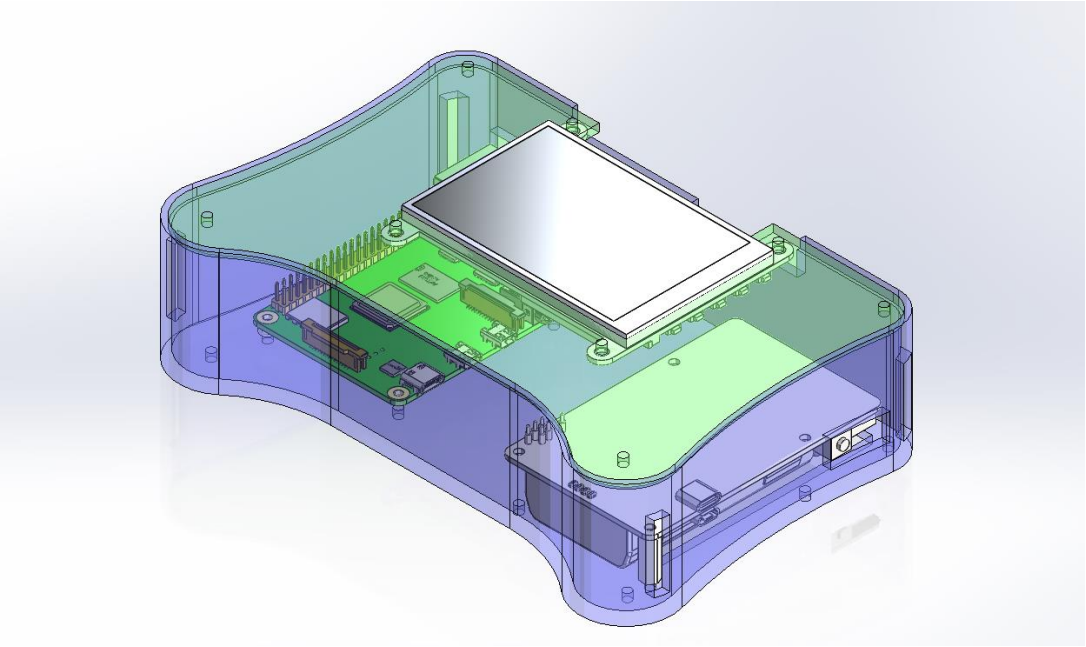

0331-1.stp<1> of 0331-1.stp<1>

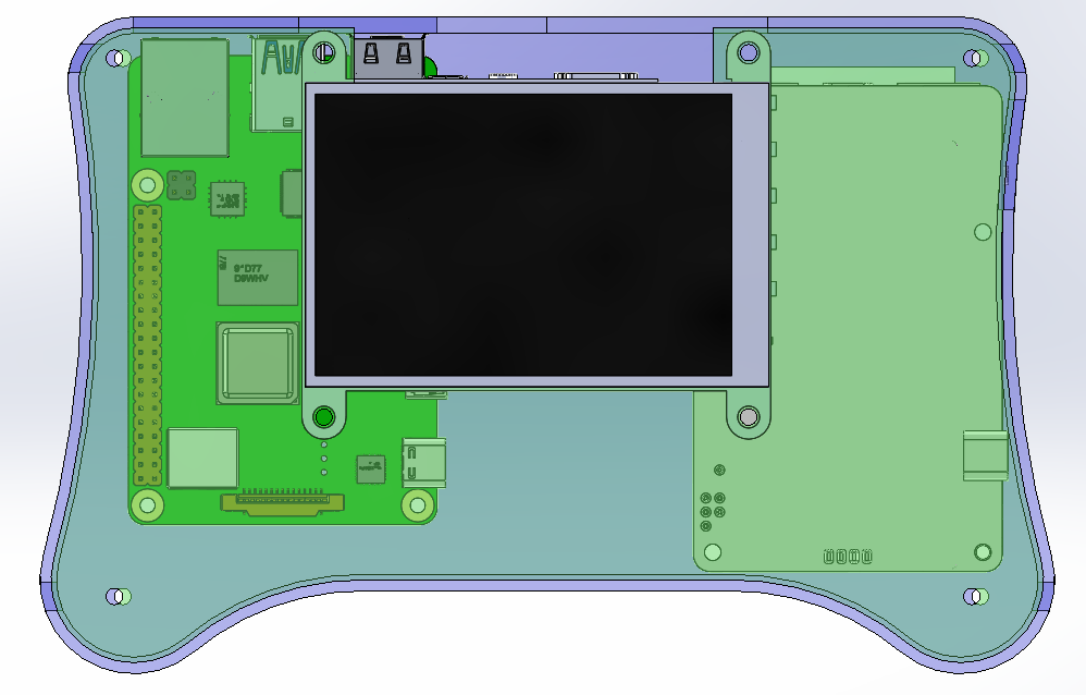


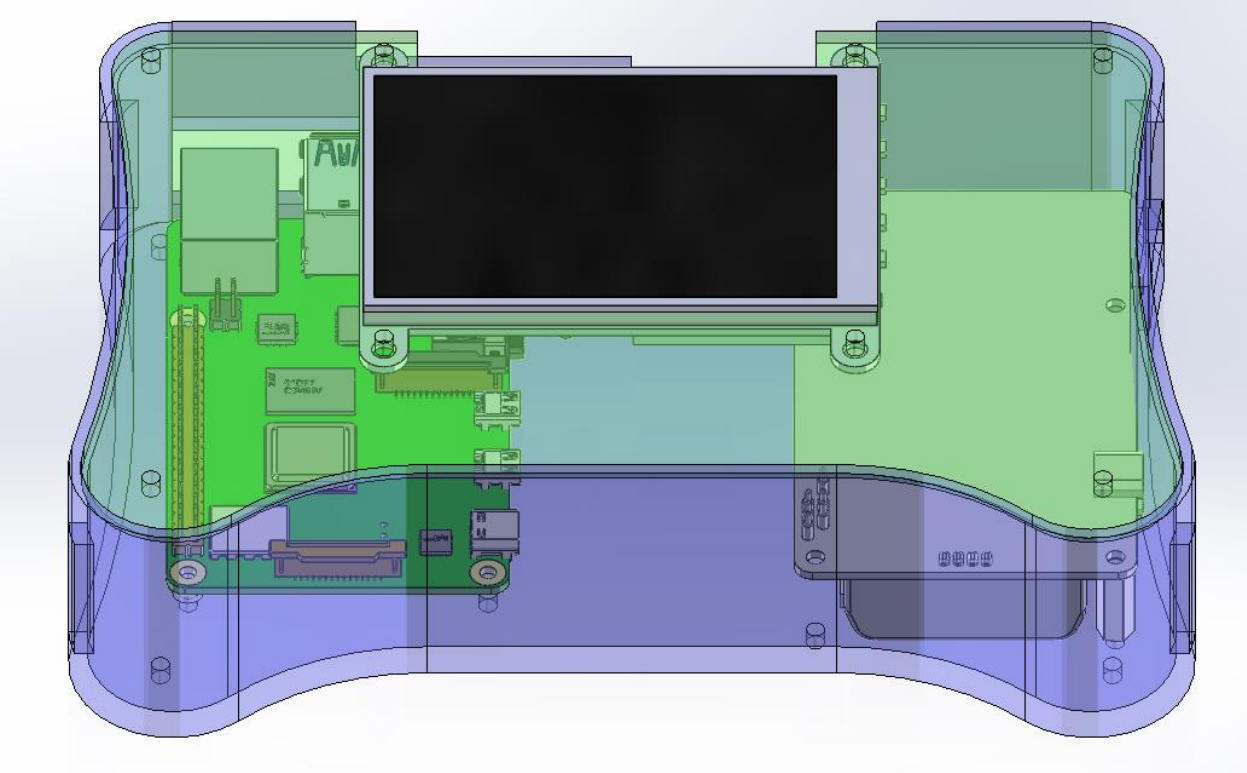
Ngày 19 tháng 5 năm 2023

Buổi sáng

Hoàn thiện tay cầm nhện







Buổi chiều

Hoàn thiện tay camera đấu kín