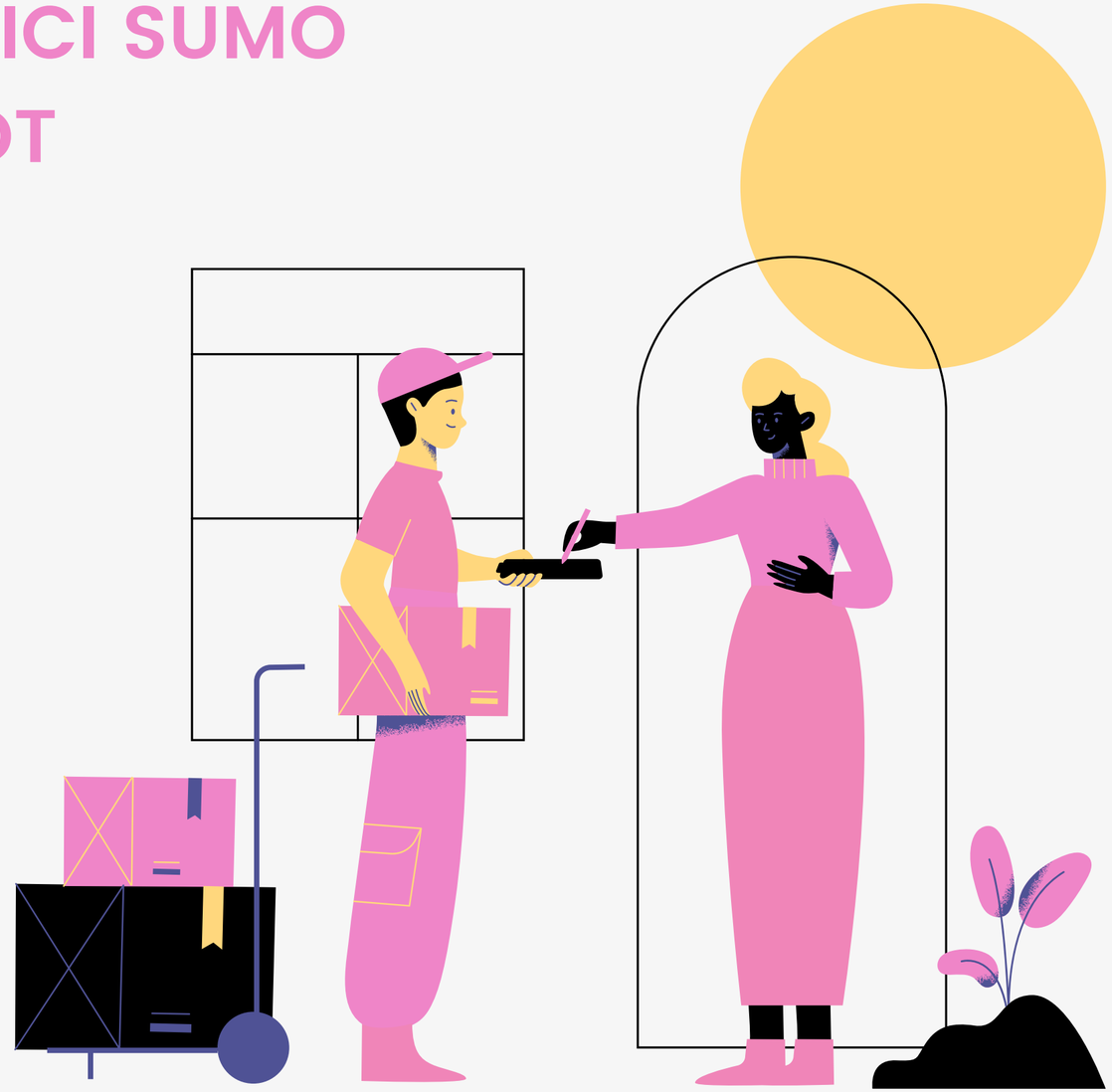
ENGEL KALDIRICI SUMO ROBOT

DENİZ UZUN
İREM KALKANLI
ÖZLEM ÇALI
İREM BOZKURT
CEYDA UYMAZ



```
//MOTOR CONTROL
int RPwm = 11;//sağ motorun hızı
int RDir = 13;//sağ motorun yönü
int LPwm = 3;//sol motorun hızı
int LDir = 12;//sol motorun yönü


int ArduLed = 8;

//EDGE & CONTRAST SENSORS
int Redge = A1;//saha dışına çıkmaması için gereken sensörler
int Ledge = A0;
```

```
//TRIMPOTS//potasyometre
int SPD = A7;//direnci değiştirerek motorun hızını ayarlar.
int TRN = A6;//direnç açıya göre değiştirilir.

int Button = 10;
int ArduLed = 8;
int Buzzer = 9;

int Speed =50;
int MaxSpeed = 50; // Idle Speed while no sensor giving data.
int TurnSpeed = 55; // Left and Right Forward Turning Speed
int EdgeTurn = 15; // Turning Time variable when minisumo sees white line  ->190
int Duration; // Turning Time at minisumo starting.
int LastValue = 5; // Last Value Variable for remembering last Opponent sensor sense.
```

```
void setup() {
pinMode(LSens, INPUT);    // Left Opponent Sensor Input
pinMode(RSens, INPUT);    // Right Opponent Sensor Input
pinMode(MSens, INPUT);    // Middle Opponent Sensor Input
pinMode(Buzzer, OUTPUT);  // Buzzer Declared as Output
pinMode(ArduLed, OUTPUT); // ArduLed Declared as Output
pinMode(Button, INPUT); // Button Input

pinMode(RPwm, OUTPUT);  // Four PWM Channel Declared as Output
pinMode(RDir, OUTPUT);
pinMode(LPwm, OUTPUT);
pinMode(LDir, OUTPUT);

digitalWrite(RFSens, HIGH);
digitalWrite(MSens, HIGH);

digitalWrite(Buzzer, LOW);
digitalWrite(ArduLed, LOW);  // Arduino Mode Led Made Low

Serial.begin(9600);
 tone(9, 523, 300);
  delay(300);
   noTone(9);

}
```

```
void Set_Motor (float Lval, float Rval, int timex){
  Lval = Lval*2.5;
  Rval = Rval*2.5;
```

```
if (Lval >=0) {
```

```
        analogWrite(LPwm, Lval);
        digitalWrite(LDir, LOW);
        } else {
        Lval=abs(Lval);
        digitalWrite(LDir, HIGH);
        analogWrite(LPwm, Lval);
        }
    if (Rval >=0) {
        analogWrite(RPwm, Rval);
        digitalWrite(RDir, HIGH);
        } else {
        Rval=abs(Rval);
        digitalWrite(RDir, LOW);
        analogWrite(RPwm, Rval);
        }

    delay(timex);
}
```

```
void loop() {
  digitalWrite(RPwm, LOW);
  digitalWrite(LPwm, LOW);
  tone(Buzzer, 18, 100);
```

```
 Start:
 /// Edge Sensor Control Routine ///
 digitalWrite(ArduLed, LOW);
if (analogRead(Ledge)>100 && analogRead(Redge)< 100) {
  digitalWrite(Buzzer, LOW);
  digitalWrite(ArduLed, HIGH);
  Set_Motor(-50, -50,350); // Geri
  Set_Motor(50, -50, EdgeTurn); // Left Backward, Right Forward, Turning Time Based on ETRN Trimpot
  LastValue=5;
}
  else  if (analogRead(Ledge)< 100 && analogRead(Redge)> 100) {
   digitalWrite(Buzzer, LOW);
    digitalWrite(ArduLed, HIGH);
  Set_Motor(-50, -50,350); // Back 35 Milliseconds
  Set_Motor(-50, 50, EdgeTurn); // Right Backward, Left Forward, Turning Time Based on ETRN Trimpot
  LastValue=5;
 }
  else  if (analogRead(Ledge)>100 && analogRead(Redge)> 100) {
    digitalWrite(Buzzer, LOW);
    digitalWrite(ArduLed, HIGH);
```

```
Set_Motor(-50, -50,15); // Back 35 Milliseconds
Set_Motor(50, -50, EdgeTurn); // Right Backward, Left Forward, Turning Time Based on ETRN Trimpot
```

```
    LastValue=5;

 }else
/// Opponent Sensor Control Routine ///
//while (digitalRead(Button)==LOW) {Set_Motor(0, 0, 20); digitalWrite(Buzzer, LOW); LastValue=3;} digitalWrite(Buzzer, LOW);
if (digitalRead(MSens)==LOW) {Set_Motor(50, 50,1); LastValue=5;} else
if (digitalRead(LSens)== LOW) {Set_Motor(-50, 50,1); LastValue=7;} else
if (digitalRead(RSens)==LOW) {Set_Motor(50, -50,1); LastValue=3;} else
{

//Speed=(analogRead(SPD)/10.3); Speed=100-Speed;
if (LastValue==5) { Set_Motor(20, 20,1);} else // Forward, Based on SPD (A7) Trimpot
if (LastValue==7) { Set_Motor(-20, 40,2);} else  // Left Turning Based on SPD (A7) Trimpot
if (LastValue==3) { Set_Motor(40,-20,2);}  // Right Turning Based on SPD (A7) Trimpot
}
goto Start;
}
```

```
//MOTOR CONTROL
int RPwm = 11;//sağ motorun hızı
int RDir = 13;//sağ motorun yönü
int LPwm = 3;//sol motorun hızı
int LDir = 12;//sol motorun yönü

//EDGE & CONTRAST SENSORS
int Redge = A1;
int Ledge = A0;
```

```
void Set_Motor (float Lval, float Rval){
  Lval = Lval*2.5;
  Rval = Rval*2.5;
```

```
if (Lval >=0) {
    analogWrite(LPwm, Lval);
    digitalWrite(LDir, LOW);
    } else {
    Lval=abs(Lval);
    digitalWrite(LDir, HIGH);
    analogWrite(LPwm, Lval);
    }
 if (Rval >=0) {
```

```
        analogWrite(RPwm, Rval);
        digitalWrite(RDir, HIGH);
        } else {
        Rval=abs(Rval);
        digitalWrite(RDir, LOW);
        analogWrite(RPwm, Rval);
        }

}
```

```
void setup() {

pinMode(RPwm, OUTPUT);
pinMode(RDir, OUTPUT);
pinMode(LPwm, OUTPUT);
pinMode(LDir, OUTPUT);

}
```

```
void loop() {
  Start:
 if(analogRead(Ledge)>100 &&analogRead(Redge)>100){//Sol ve sağ sensör siyah zeminde olduğu için motorun düz gitmesi istenir.
 forward();
 }if(analogRead(Ledge)<100&&analogRead(Redge)>100){//Sol sensör beyaz zeminde, sağ sensör siyah zeminde olduğu için motorun sağa gitmesi istenir.
 right();
 } if(analogRead(Ledge)>100&&analogRead(Redge)<100){//Sol sensör siyah zeminde, sağ sensör beyaz zeminde olduğu için motorun sola gitmesi istenir.
 left();
 }

 if(analogRead(Ledge)<100&&analogRead(Redge)<100){//Sol ve sağ sensör beyaz zeminde olduğu için motorun geri gitmesi istenir.
 back();
 }
 goto Start;
}
```

```
void forward(){
   Set_Motor(30,30);
   }
```

```
void forward(){
  Set_Motor(30,30);
  }


void left(){
  Set_Motor(0,30);
  }
 void right(){
  Set_Motor(30,0);
  }
  void back(){
  Set_Motor(-30,-30);
  }
```

TEŞEKKÜR EDERİZ