



FENERBAHÇE ÜNİVERSİTESİ

Engel Kaldırıcı Sumo Robot

Evrım Arda Kalafat, Arda Alhan, Cüneyt Balcı

Fenerbahçe Üniversitesi

Bilgisayar Mühendisliği

İstanbul, Türkiye

e-mail: evrim.kalafat@stu.fbu.edu.tr, arda.alhan@stu.fbu.edu.tr ,
cuneyt.balci@stu.fbu.edu.tr

Proje özeti: Sumo robot gereklenip, verilecek 5 adet hedefin 60 saniye ierisinde platformdan atabilecek bir sumo robot algoritması geliřtirilecektir. Robot'un diğeri bir modu ise izgi izlemektir. Bu modda ise kensine baėlı olan sensörleri kullanarak izgi üzerinde takip etmesi beklenmektedir.

Anahtar Kelimeler: Sumo Robot.

Abstract: A sumo robot algorithm will be developed that will implement the sumo robot and throw 5 targets from the platform within 60 seconds. Another mode of Robot is to follow lines. In this mode, it is expected to follow the line using the sensors connected to it.

Keywords: Sumo Robot.

I. Giriř

Bu projede bizden savařacak ve önündeki 6 cismi devirecek aynı zamanda da izgi takibi yapacak iki farklı sumo robot algoritması istenmiřtir. Biz de bu yönergelere göre iki farklı algoritma yazdık ve robotlarımızla alıřtırdık. Aynı zamanda derste yapılan yarıřmada da robotumuz tüm rakiplerini başarı ile yenerek birinci olmuřtur.

II. Sistem Mimarisi ve Kullanılan Yazılım

Savaş ve Cisim Devirme Algoritması

```
//SETUP-----  
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); // Buzzer Declared as Output  
    pinMode(Button, INPUT);   // Buzzer Declared as Output  
  
    pinMode(RPwm, OUTPUT);    // Four PWM Channel Declared as Output  
    pinMode(RDir, OUTPUT);  
    pinMode(LPwm, OUTPUT);  
    pinMode(LDir, OUTPUT);  
  
    digitalWrite(buzzer, LOW); // Buzzer Pin Made Low for Silence :)  
    digitalWrite(ArduLed, LOW); // Arduino Mode Led Made Low  
    digitalWrite(DS1, HIGH); // 3 Dipswitch Pin Pullups Made  
    digitalWrite(DS2, HIGH);  
    digitalWrite(DS3, HIGH);  
  
    digitalWrite(RFSens, HIGH);  
    digitalWrite(MSens, HIGH);  
  
    Serial.begin(9600);  
  
    tone(9, 523, 300);  
    delay(300);  
    noTone(9);  
}
```

Yukarıdaki kodda Setup() fonksiyonu gözükmemektedir. Bu fonksiyon arduino programı çalıştığında ilk başta bir kere çalışacaktır. Bu fonksiyonun içinde pinMode()'lar ile input ve output giriş çıkışlarımız belirlenmiştir. Ve bazı pinlerin ilk değerleri atanmıştır. Ayrıca en sonda buzzerden setup fonksiyonun çalıştığının geri bildirimi olarak bir ses çıkmaktadır.

```
//setMotor-----  
void setMotor(float Lval, float Rval, int timex){  
    Lval = Lval*2.5;  
    Rval = Rval*2.5;  
  
    if (Lval >=0) {  
        analogWrite(LPwm, Lval);  
        digitalWrite(LDir, HIGH);  
    }  
    else {  
        Lval=abs(Lval);  
        digitalWrite(LDir, LOW);  
        analogWrite(LPwm, Lval);  
    }  
  
    if (Rval >=0) {  
        analogWrite(RPwm, Rval);  
        digitalWrite(RDir, HIGH);  
    }  
    else {  
        Rval=abs(Rval);  
        digitalWrite(RDir, LOW);  
        analogWrite(RPwm, Rval);  
    }  
    delay(timex);  
}
```

Yukarıdaki setMotor fonksiyonu sayesinde loop() fonksiyonumuzun içinde sumo robotumuzun motorlarının gücünü ve yönünü ayarlayabiliyoruz. Örneğin setMotor(100, -50, 300) yazarsak sol motor ileri yönde en hızlı, sağ motor ise geri yönde yarı hızda 300 milisaniyelik bir hareket yapar.

```

//LOOP-----
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);
      setMotor(-100, -100,35); // Geri
      setMotor(-100,100, EdgeTurn); // Left f, Right b,
      LastValue=5;
    }
    else if (analogRead(Ledge)< 100 && analogRead(Redge)> 100) {
      digitalWrite(buzzer, LOW);
      digitalWrite(ArduLed, HIGH);
      setMotor(-100, -100,35); // Back 35 Milliseconds
      setMotor(100, -100, EdgeTurn); // Right f, Left b,
      LastValue=5;
    }
    else if (analogRead(Ledge)> 100 && analogRead(Redge)> 100) {
      digitalWrite(buzzer, LOW);
      digitalWrite(ArduLed, HIGH);
      setMotor(-100, -100,35); // Back 35 Milliseconds
      setMotor(100, -100, EdgeTurn); // Right b, Left f,
      LastValue=5;
    }
  }else
    /// Opponent Sensor Control Routine ///
    if (digitalRead(MSens)==HIGH) {setMotor(100, 100,1); digitalWrite(buzzer, HIGH); LastValue=5;} else
    if (digitalRead(LSens)== HIGH) {setMotor(-50, 50,1); digitalWrite(buzzer, HIGH); LastValue=7;} else
    if (digitalRead(RSens)==HIGH) {setMotor(50, -50,1); digitalWrite(buzzer, HIGH); LastValue=3;} else
    {
      digitalWrite(buzzer, LOW);

      if (LastValue==5) { setMotor(40, 40,1);} else
      if (LastValue==7) { setMotor(-20, 80,2);} else
      if (LastValue==3) { setMotor(80, -20,2);}
    }
  goto Start;
}

```

Yanda Loop fonksiyonumuzu görüyoruz. Bu fonksiyon bir nevi bizim main fonksiyonumuz program boyunca durmadan çalışıyor. Edge Sensor Control Routine kısmında sumo robotumuz, sağ ve sol alt tarafında bulunan kontrast sensörleri sayesinde ringden çıkıp çıkmadığını kontrol ediyor. Opponent Sensor Control Routine kısmında ise karşısında, sağında veya solunda bir cisim olup olmadığını kontrol ederek o yöne doğru bir atılım yapıyor.

Çizgi Takibi Algoritması

```
//MOTOR Kontrol
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;
```

Yukarıda motor kontrol ve kontrast sensörlerimizin tanımlamalarını yaptık.

```
void setup() {
pinMode(RPwm, OUTPUT); // 4 tane kanal çıktı olarak atandı
pinMode(RDir, OUTPUT);
pinMode(LPwm, OUTPUT);
pinMode(LDir, OUTPUT);
}
```

Yukarıda setup fonksiyonu içinde motor kontrollerini OUTPUT olarak pinMode ile ayarladık.

```

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);
    }
}

```

Yukarıdaki fonksiyonda yine önceki algoritmadaki gibi setMotor fonksiyonumuz var ama bunda yukarıdakinden farklı olarak, içine bir zaman değeri almıyor. Yani motor tanımlanan hareketi aksi bir hareket söylenmedikçe yapıyor.

```

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

```

Yukarıdaki Loop() fonksiyonumuzda motorun kontrast sensörlerine göre yine setMotor fonksiyonumuz ile bu sefer bir süre kısıtlaması olmadan hareket etmesini sağladık. Örneğin eğer sol sensör beyaz, sağ sensör siyah görüyor ise sağa doğru bir hareket yaparak çizgiden çıkması engelleniyor.

III. Sonular

Sonu olarak, iki adet sumo robotu algoritmasını gerekleřtirmiş olduk. Biri karřısındaki rakipleri ve cisimleri deviren bir algoritma diğeri ise siyah bir çizgi üzerinde dıřarı tařmadan robotun yol almasını saėlayan bir algoritmadır.

IV. Proje Ekibi

Evrin Arda KALAFAT, 25.09.2001 yılında İstanbul'da doğdu. 2019 yılında Kadıköy Final Temel Lisesi'nden mezun oldu. řu anda Fenerbahe Üniversitesi Bilgisayar Mühendisliėi bölümünde lisans eğitimi almakta. Bilgisayar Mühendisliėi Bölüm Temsilciėi yapmaktadır. C, C++ ve Python dillerinde bilgili. Programlama, yapay zeka ve siber güvenlik ile ilgileniyor.

Cüneyt BALCI, 28.08.2000 yılında İstanbul'da doğdum. 2018 yılında Final Temel Lisesi'nden mezun oldum. řu anda Fenerbahe Üniversitesi - Bilgisayar Mühendisliėi bölümünde ve çift ana dal programı kapsamında Ekonomi (İngilizce) bölümünde lisans eğitimi almaktayım. C, C++ ve Python dillerinde bilgili. Network ve blockchain alanlarıyla ilgileniyor.

Arda ALHAN, 18.05.2001 yılında İstanbul'da doğdu. 2019 yılında Eyüp Anadolu Lisesi'nden mezun oldu. řu anda Fenerbahe Üniversitesi Bilgisayar Mühendisliėi bölümünde lisans eğitimi almakta. C, C++, C#, Selenium ve Python dillerinde bilgili. Programlama ile ilgileniyor.

V. Referans Dosyalar

<https://youtu.be/N0Izup5oB34>

<https://github.com/rhgod/MikrokontrollerProjesi>

VI. Kaynaklar

[1] http://www.levent.tc/files/courses/microcontrollers/project/BLM302_proje_spesifikasyonlari.pdf

[2] <http://www.levent.tc/courses/microcontrollers>