

BLM 302 – Mikrokontrolörler ve Robotik

Engel Kaldırıcı Sumo Robot

Recep Gemalmaz, Berk Tunç, Ogün Berat Gürses, Damla Su Karadoğan

Fenerbahçe Üniversitesi

Bilgisayar Mühendisliği

İstanbul, Türkiye

e-mail: [damla.karadogan@stu.fbu.edu.tr](mailto:damla.karadogan@stu.fbu.edu.tr), berk.tunc@stu.fbu.edu.tr, ogun.gurses@stu.fbu.edu.tr, recep.gemalmaz@stu.fbu.edu.tr,

1. **Giriş**

**Proje özeti:** Sumo robot gerçeklenip, verilecek 5 adet hedefin 60 saniye içerisinde platformdan atabilecek bir sumo robot algoritması geliştirilecektir. Robot’un diğer bir modu ise çizgi izlemektir. Bu modda ise kensine bağlı olan sensörleri kullanarak çizgi üzerinde takip etmesi beklenmektedir.

**Anahtar Kelimeler:** Sumo, robot, Ardunio, Nano, Sensör.

**Abstract:** Sumo robot is a sumo robot that can be implemented and thrown from the platform within 60 seconds of 5 targets to be given. algorithm will be developed. Another mode of Robot is to follow lines. In this mode, the sensors connected to it. It is expected to follow the line using.

**Keywords:** Sumo, robot, Ardunio, Nano, Sensör.

1. **Sistem Mimarisi**

Kullanılacak Araçlar: Proje kapsamında 1 araç kullanılacaktır.

Ardunio IDE: Arduino için Entegre Geliştirme Ortamı, C ve C ++ dilleri ile yazılmış bir platformlar arası uygulamadır.Arduino uyumlu kartlara program yazmak ve yüklemek için kullanılır, aynı zamanda 3. taraf çekirdekler vesatıcıların geliştirme kartları içinde kullanılabilir.

ÇİZGİ TAKİBİ

Graphical user interface, text, application, email

Description automatically generated

Yukarıda çizgi takibi kodunun giriş kısmı gösterilmektedir. Başlangıçta sensörler, ledler, buzzer ve motor bilgileri ilgili pinlere tanımlanmıştır. Bu tanımlamalar sonucu robot ile kod arasında bağlantılar yapılmıştır. Set\_Motor komutu hız değişkenlerinin nasıl çalışacağını ayarlamak için kullanılan bir fonksiyondur. Bu kısımda Set\_Motor fonksiyonu 2 değişken alarak hız ayarı yapmaktadır. Alınan değişkenler sol ve sağ motorlara verilerek dönüşler ve robotun yapacağı manevralarda alacağı hız durumu burada ayarlanmaktadır. Hangi durumlarda motorlara elektrik gideceği hangi durumlarda gitmeyeceği burada ayarlanmıştır.

Table

Description automatically generated with low confidence

Yukarıdaki resimde bulunan kodlarda setup fonksiyonu ve loop fonksiyonu bulunmaktadır. Setup fonksiyonunda pinMode komutuyla değişkenlere input-output bilgileri yollanmıştır ve sensörlere bilgilerin girişi buradan sağlanmaktadır. Ayrıca uart haberleşmesi sağlanmıştır.

Loop fonksiyonunda ise contrast sensörüne gelen bilgiler okunarak zeminin siyah mı yoksa beyaz mı olduğu kontrol edilmektedir. Sensörler biri ya da ikisi beyaz alana girerse farklı, ikisi de siyahta devam ederse farklı hareketler yaparak siyah bant takibi sağlanmıştır.

**Şişe devirme algoritması**

Graphical user interface, text, application

Description automatically generated

Yukarıda gösterilen kısımda önceki kodda olduğu gibi sensörler, ledler, buzzer ve motor bilgileri ilgili pinlere tanımlanmıştır. Bu tanımlamalar sonucu robot ile kod arasında bağlantılar yapılmıştır.

Table

Description automatically generated

Setup fonksiyonunun içerisinde pinMode ile değişkenlerin input output bilgileri ayarlanmış ve digital Write ile sensörlerin çalışma şekilleri belirlenmiştir. Ardından robot başlatıldığında kısa süreli bi ses çalınarak işlemin yapıldığı gösterilmiştir.

Text

Description automatically generated

Loop fonksiyonu içerisinde robotun ayarlanmış alandan çıkmaması için contrast sensörleri ile birkaç çeşit kontrol yapılmış ve robotun belirlenen alandan çıkmaması sağlanmıştır. Ardından opponent sensor control routine kısmında robotun bir cisim gördüğünde nasıl tepkiler vereceği ayarlanmıştır.

**Kullanılan Yazılım**

Bu proje için ardunio idesi kullanılmıştır. Ardunio IDE, Arduino için Entegre Geliştirme Ortamı( IDE ), C ve C ++ dilleri ile yazılmış bir platformlar arası uygulamadır ( Linux, macOS, Windows için,). Arduino uyumlu kartlara program yazmak ve yüklemek için kullanılır, aynı zamanda 3. taraf çekirdekler ve satıcıların geliştirme kartları içinde kullanılabilir.

**Sonuçlar**

Bu proje kapsamında bir sumo robotunun, sensörlerin, mikrokontrolcülerin ve ardunionun nasıl çalıştığını, ne ile çalıştığını, hangi şartlarda çalıştığını öğrenmiş olduk. Ardunio ortamında kod geliştirerek bir robota gereken sensörlerle istediğimiz işlemleri nasıl gerçekleştirebileceğimizi öğrendik. Robot Contest haftasında kodlarımız ile robot içerisinde denemeler yaparak yaptığımız kodları test ettik ayrıca kodlarımızın çalıştığını göstermiş olduk.

**Proje Ekibi**

**Damla Su KARADOĞAN**, 11.02.2001 yılında doğdu. 2019 yılında Özel Envar Anadolu Lisesinden mezun oldu. Şu anda Fenerbahçe Üniversitesinde Endüstri Mühendisliği bölümünde lisans eğitimini almakta ve Bilgisayar Mühendisliğinde ÇAP eğitimi alıyor. Öğrenci numarası, 190302016.

**Ogün Berat GÜRSES**: 10.11.2000 Sakarya ili Adapazarı ilçesinde doğdu. 2018 yılında Sakarya Anadolu Lisesi’nden mezun oldu. Şu anda Fenerbahçe Üniversitesi’nde Bilgisayar Mühendisliği bölümünde eğitim almaktadır. C, C++, mySQL, Verilog, Sys-Verilog ve Python ile ilgilenmektedir.

**Berk TUNÇ:** 04.02.2000 yılında Yalova ili Merkez ilçesinde doğdu. 2018 yılında Şehit Osman Altınkuyu Anadolu Lisesi'nden mezun oldu. Şu anda Fenerbahçe Üniversitesi’nde Bilgisayar Mühendisliği bölümünde bölümünde eğitim almaktadır. mySQL, HTML, C, C++, Verilog, Sys-Verilog ve Python ile ilgilenmektedir.

**RECEP GEMALMAZ:** 16.10.2000 tarihinde Kadıköy’de dünyaya geldi. 2018 yılında Alparslan Anadolu Lisesi’nden Mezun oldu. Şu anda Fenerbahçe Üniversitesi Bilgisayar Mühendisliği bölümünde lisans eğitimi almakta. Java, C, C++ ve Pyhton dillerinde bilgili. Android Programlama, Verilog ve Sys-Verilog ile ilgilenmektedir.

1. **Referans Dosyalar**

**Github Dosyası:** [Engel-Kald-r-c-ve-izgi-Takibi-Yapan-Sumo-Robotu (github.com)](https://github.com/brktnc/Engel-Kald-r-c-ve-izgi-Takibi-Yapan-Sumo-Robotu)

**Youtube Adresi:** [Engel Kaldırıcı ve Çizgi Takibi Yapan Sumo Robotu - Fenerbahçe Üniversitesi - YouTube](https://www.youtube.com/watch?v=UM6MG6os3YY)

1. **Kaynaklar**