

Eğitim Kurumları İçin Robot Platform Tasarımı ve Uygulaması

Ahmet SEVİM¹, Alaa ELEYAN^{2*}, Gülden ELEYAN³

¹ Elektrik-Elektronik Mühendisliği, Karadeniz Teknik Üniversitesi, Trabzon, Türkiye

² Elektrik-Elektronik Mühendisliği, Ankara Bilim Üniversitesi, Ankara, Türkiye, ORCID: [0000-0002-0644-8039](https://orcid.org/0000-0002-0644-8039)

³ Fen Bilimleri Enstitüsü, Hacettepe Üniversitesi, Ankara, Türkiye, ORCID: [0000-0002-9539-9899](https://orcid.org/0000-0002-9539-9899)

* Sorumlu Yazar: alaa.eleyan@ankarabilim.edu.tr

Received: 18 May 2021; Accepted: 8 July 2021

Öz

Robot platformlarını programlanmaya hazır bir donanım olarak tanımlayabiliriz. Günümüzde örnekleri giderek artan bu robot platformları birçok eğitim kurumu tarafından araştırma ve eğitim amaçlı kullanılmaktadır. Robot platformları hazır donanımları ile araştırma süreçlerini hızlandırdığı gibi eğitim için de interaktif bir eğitim imkânı sunmaktadır. Bu nedenle Initio robot platformu eğitimde dijitalleşmeyi hızlandırmak ve araştırma projelerinde kullanılmak amacıyla tasarlanmıştır. Sistem kontrolü, Raspberry Pi Zero W bilgisayarı kullanılarak sağlanan Initio robot platformunun Python Flask kütüphaneleri ile internet üzerinden kontrolü sağlanmıştır. Bünyesinde bulunan kamera ve uzaklık sensörü sayesinde farklı projelerde kullanılabilir. Ayrıca, platformun OpenCV Kütüphanesi ile birlikte kullanımında nesne tanıma ve takibi gibi uygulamalar da gerçekleştirilebilir.

Anahtar Kelimeler: nesnelerin interneti, raspberry pi, robot platformları, robotik

Robot Platform Design and Application for Educational Institutions

Abstract

Robot platforms can be defined as off-the-shelf hardware ready for programming. Nowadays, the drastically increasing number of programmable mobile robot platforms has become accessible to many educational institutions for research and education purposes. Robot platforms offer an interactive educational environment by teaching students computer science in a tangible way where they can transform the codes they write into actions. The Initio robot platform is designed as a mobile robot platform for this purpose. The Raspberry Pi Zero W computer-controlled Initio robot platform is suitable to be used in different projects with its built-in camera and distance sensor. Python Flask library is used for control mechanism of the platform over the internet. Moreover, OpenCV library can be utilized by Initio robot platform to build different applications such as object recognition and tracking.

Keywords: Internet of things, raspberry pi, robot platforms, robotics.

1. Giriş

Güncel tanımı ile robotlar, elektronik ve mekanik birimlerden oluşan, algılama yeteneğine sahip olan ve programlanabilen cihazlardır. Robotlar kullanıldıkları amaçlara ve ortamlara göre tasarlanıp sınıflandırılmaktadır. İlerleyen teknoloji ile daha sık karşımıza çıkmaya başlayan robotlar hayatı kolaylaştırmaktadır. Özellikle Endüstri 4.0 ile adından sıkça söz ettiren nesnelerin interneti kavramı bu teknolojik dönüşümü hızlandırmaktadır.

Nesnelerin İnterneti kavramını inceleyecek olursak, ilk kez 1999 yılında Kevin Ashton tarafından, P&G isimli şirket için hazırlanan sunumda bahsi geçmiş bir ifadedir. Bu sunumun içeriğinde şirketin tedarik zincirinde RFID teknolojisi uygulamasının firmaya faydaları sıralanmış ve kullanımı önerilmiştir. Nesnelerin interneti hayatımızı daha güvenli, akıllı, kolay ve üretken hale getirerek yaşama biçimimizi değiştirecek bir teknolojidir. Nesnelerin interneti terimi, benzersiz bir şekilde adreslenebilen ve aklımıza gelen tüm cihazların internet veya herhangi belirli bir protokol ile birbirlerine bağlı olarak diğer

cihazlarla iletişim halinde olmasını ifade etmektedir [1,2]. Initio robot platformu, internet protokolü üzerinden diğer cihazlarla iletişim halinde çalışarak kontrol ve haberleşme imkânı sunmaktadır.

Gelişen robot teknolojisi diğer alanları da etkileyerek bazı teknolojik dönüşümlerin önünü açmaktadır. Bu dönüşümün gerçekleştiği alanlardan biri de eğitimidir. Robotların eğitimde kullanılması fikri, 1980'li yılların başında ortaya çıkmıştır [3]. Eğitimin giderek dijitalleştiği günümüzde, eğitim için kullanılan robot platformu örneklerinin çoğaldığı görülmektedir [4-13]. Bu platformlar sundukları özellikler ile birçok eğitim uygulamasında kullanılmaktadır. Bu sayede daha etkileşimli bir eğitim gerçekleştirilmektedir. Öğrenciler bu eğitimler sırasında soyut kavramları daha rahat anlayıp, yaptıkları uygulamalar ile öğrendiklerini pekiştirmektedirler. Fakat bu robotlar sundukları birçok avantajın yanında yüksek maliyeti ile büyük bir dezavantaja da sahiptirler. Bu maliyetler üniversite gibi büyük eğitim kurumları dışında karşılanması güç rakamlar olmaktadır. Ancak eğitimde dijitalleşmenin yaygınlaşması ve tüm eğitim kurumları için ulaşılabilirlik amacı adına düşük maliyet büyük önem arz etmektedir.

Initio robot platformu eğitim kurumlarında kullanımın yanı sıra maliyet ve performans kriterlerine öncelik verilerek tasarlanmıştır. Genel amaçlı mobil robot platformu olarak tasarlanan Initio, maliyet konusundaki uygunluğuyla öğrencilerin ilköğretim çağlarında bile bu platformu kullanabilmesine olanak tanımaktadır. Bu sayede öğrencilerin, eğitici teknolojik cihazlara ilgi ve alakalarının artırılması aynı zamanda bu gibi cihazların eğitimin bir parçası haline getirilmesi hedeflenmiştir. Diğer bir kriter olan yüksek performans konusunda ise platform performansını Raspberry Pi Zero W bilgisayarından almaktadır[14]. Bu sayede görüntü işleme gibi karmaşık uygulamalarda maliyetin azaltılması amaçlanmıştır. Bunun yanı sıra kullanım kolaylığı, şarj edilebilirlik, batarya yönetim sistemi, kamera ve hareketli dış gövde gibi ek donanımsal özellikler de sistem tasarımında kullanılmıştır.

Makale şu şekilde yazılmıştır; ilk olarak 2. Bölümde, yapılan literatür çalışmasından bahsedilmiştir. 3. Bölüm'de, sistemin donanımı üzerine yapılan çalışmalardan ve tasarımlardan bahsedilmiştir. 4. Bölüm'de ise, tasarlanan donanımın yazılım ile denenmesinden bahsedilmiştir. Son olarak 5.Bölüm'de sistem tasarımıyla ilgili çıkarımlar yapılarak sistemin eksiklerinden bahsedilmiştir.

2. Literatür

Yapılan literatür çalışması sonucunda piyasada bulunan bazı robot platformları incelenmiştir bunlar; Pololu 3pi, mBot 1.1, Waveshare Alphanot 2, Pioneer 3-AT, E-puck, Robotino, Koala II gibi genel amaçlı mobil robot platformlarıdır.

Üstte isimleri verilen mobil robot platformlarının avantaj ve dezavantajları üzerine kısaca değinecek olursak:

Pololu 3pi: Mikrodenetleyici tabanlı olan bu platform Atmel firmasının 8bit AVR serisi mikrodenetleyicilerinden birini kullanmaktadır [15,16]. Bu yapı düşük maliyet ve kolay bir kullanım sunmasına karşın Wi-Fi, Bluetooth gibi haberleşmeler için ayrı modüller gerektirmesi, bunun yanısıra kamera kullanımı konusunda işlem kapasitesinin yetersiz kalmasından dolayı çoğu proje için uygun olmadığının göstergesidir.

mBot 1.1: Ortaöğretim ve Lise eğitim robotları için güzel bir örnek olan mBot 1.1 mikrodenetleyici tabanlı platformlara örnektir. Üzerinde Pololu 3pi gibi 8 bit AVR serisi bir mikrodenetleyici bulundurmaktadır. Scratch yapısına benzeyen programlama ile çok basit bir kullanım sumaktadır. Bunun yanında Bluetooth veya 2.4 GHz kablosuz haberleşme seçeneklerini düşük maliyette sunan bu platform sunduğu bu özellikler ve karşın mikrodenetleyici tabanlı olmasından dolayı çoğu projeye uygun değildir.

Waveshare Alphanot 2: Initio robot platformuna en yakın tasarıma sahip robot platformudur. Farklı olarak üzerinde Raspberry Pi 3 bilgisayarını barındırmaktadır bu nedenle birçok projede kullanılabilir. Dezavantajlarına bakılırsa; bunlardan ilki şarj ve batarya yönetim sisteminin bulunmamasıdır. Ayrıca Raspberry Pi ve kameranın platform içinde satılmaması maliyeti arttıran nedenler içindedir.

Pioneer 3-AT: Adept Technology şirketi tarafından üretilen Pioneer 3-AT, üniversite araştırmaları için tasarlanan mobil robot platformlarına örnektir. Yüksek maliyete ve karışık bir yapıya sahiptir. Platform Pioneer ROS kütüphanelerini çalışmaktadır. Ayrıca platforma kişisel bilgisayarlar eklenip böylece daha karmaşık projelerde kullanılabilir.

E-puck: Clearpath Robotics şirketi tarafından üretilen E-puck'ta üniversite araştırmalarına yönelik tasarlanmıştır. Bu nedenle yüksek maliyet ve karmaşık bir yapıya sahiptir. Üzerinde ROS'un yanı sıra OpenCV ve PLC kütüphaneleri ile projeler geliştirilebilir.

Robotino: Festo-Didactic şirketi tarafından üretilen Robotino üzerinde mikroişlemci tabanlı bir bilgisayar bulundurmaktadır. Karmaşık projelerde kullanılacak bu platform yüksek maliyete sahiptir. Ayrıca platform ROS ile uyumlu olup ROS ile geliştirilen projelerde kullanılabilir.

Koala II: K-TEAM şirketi tarafından üretilen Koala II'de yüksek maliyetli karmaşık robot platformlarına örnektir. Bu platform üzerinde de ROS kütüphaneleri çalıştırılıp karmaşık projelerde kullanılabilir.

Yukarıda bahsedilen yüksek maliyette sınır 300 dolardır. Üniversite için tasarlanan robotlarda maliyet bazen 4000 dolar gibi yüksek rakamları bulmaktadır. Yaptıkları karmaşık işler bu maliyetlerin ana nedeni olmasına rağmen yeni gelişen teknoloji ile bilgisayarların artık bir kredi kartı büyüklüğünde olması, bu platformların tekrardan gözden geçirilmesi ve daha iyi alternatiflerinin tasarlanması gerektiğini göstermektedir.

Initio Robot platformunu yukarıda verilen genel amaçlı robot platformlarından ayıran noktaların en başında ise maliyet, kolay kullanım ve performans gelmektedir. Bunun yanı sıra Initio hem lise hem üniversite projeleri için bir köprü görevi üstlenmektedir. Bu ifadeyi açarsak hem lise projeleri için basit bir şekilde kullanılabilir hem de üniversitelerde küçük ve orta seviyeli projelerde kullanılabilir. Bu sayede düzeyi fark etmeksizin birçok eğitim kurumunun kullanacağı açık kaynak kod bir donanım platformu oluşturulmuş olacaktır.

3. Sistemin Tasarımı

Platformun tasarımından bahsetmek gerekirse öncelikli olarak sistemi oluşturan donanımlar ve yazılımlar kısmına değinmek gerekir. Sistem bu iki ana başlık altında toplanırsa daha iyi anlaşılacaktır. Sistemin donanım kısmında çeşitli malzemeler kullanılmıştır. Platformun tasarımında kullanılan genel donanımlar aşağıda belirtilmiştir:

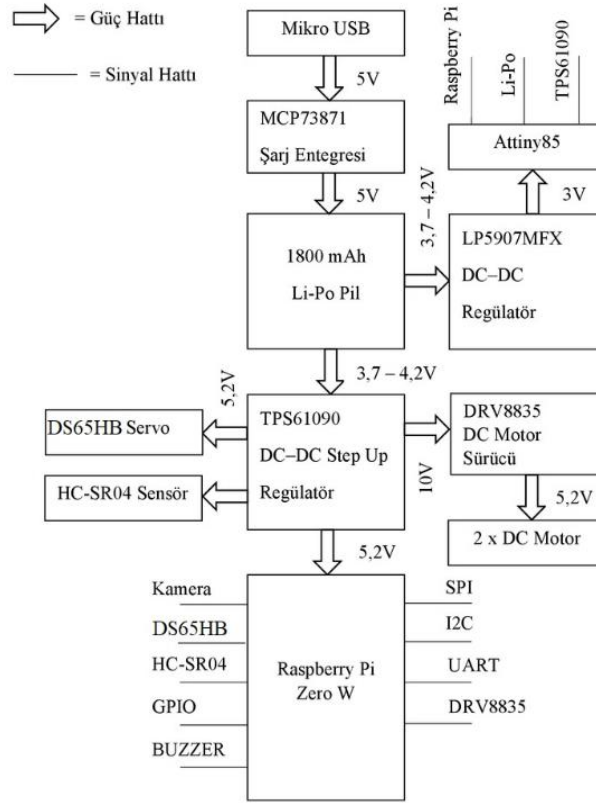
- Raspberry Pi Zero W Bilgisayarı
- Ana PCB kart
- Raspberry Pi Kamera Modülü V2
- HC-SR04 Ultrasonik Mesafe Sensör
- DS65HB Servo Motor
- Redüktörlü DC Motor
- Li-Po Pil

Sistemin yazılım kısmında ise platformu geliştirmede kullanılan programlama, işaretleme ve betik dilleri aşağıda belirtilmiştir:

- C, C++, Python (Programlama)
- HTML, CSS (İşaretleme)
- JavaScript (Betik)

Bu bahsedilenlerin dışında ana PCB kart üzerine konumlandırılacak 3 boyutlu gövde ve yine ana PCB kartı taşıyacak tekerler de platformda kullanılmıştır. Sistem tasarımına ilk olarak ana PCB karttan başlanmıştır. Ana PCB kart üzerinde şarj, batarya kontrol, motor sürücü ve sistem açma kapama kontrol elektronik devrelerini bulundurmaktadır [17]. Bu elektronik devrelerin birbiri ile uyumlu çalışması için farklı programlar kullanılarak birçok simülasyon yapılmıştır. Yapılan bu simülasyonlar sonrasında projede kullanılması en uygun entegreler seçilmiştir. Yine Ana PCB kart üzerine konumlandırılan

konektörlere Raspberry Pi Zero W bilgisayarı ve diğer donanımların bağlantısı sağlanmaktadır. Ayrıca bu Ana PCB kart motorlar, pil ve ana gövdeyi taşımaktadır.



Şekil 1: Genel devre tasarım blok diyagramı.

Ana PCB kart tasarımında kullanılan devre elemanları Şekil 1’de verilmiştir. Bunlara kısaca değinecek olursak:

Mikro USB: Sistemin uygun şarj kablosu ve adaptör ile şarj edilmesi için kullanılmıştır. Sistem 5V 1A’lık şarj adaptörü ile şarj edilmektedir.

MCP73871: Li-Po pil şarj entegresidir. Sistemin enerji kaynağı olan 1800 mAh’lık Li-Po pili 1A değeriyle şarj edebilmek ve şarj kontrolünü sağlamak için kullanılmıştır.

LP5907MFX: DC-DC düşük gürültülü regülasyon entegresidir. Attiny85’e gelen gerilimi 3V’a regüle etmek için kullanılmıştır.

Attiny 85: AVR serisi 8 bit mikrodenetleyicidir. Batarya kontrol ve açma kapama kontrolünü sağlamak için kullanılmıştır.

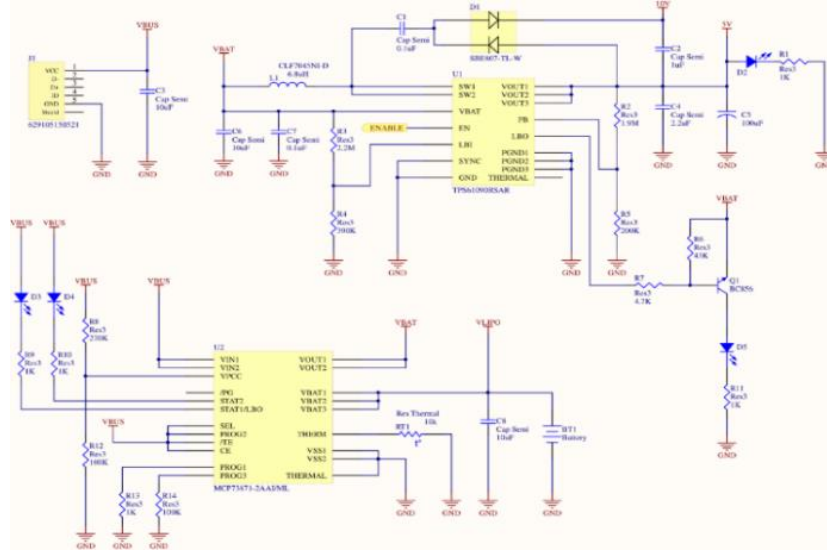
TPS61090: Step-up DC-DC regülasyon entegresidir. Li-Po pilin sağladığı 3,7V’u Raspberry Pi Zero W’nun çalışma gerilimi olan 5V seviyelerine taşımak için kullanılmıştır.

DRV8835: DC motor sürücüsü entegresidir. Üzerinde bulundurduğu çift H köprüsü ile 2 adet DC motoru sürebilmektedir. Platformun hareketinde kullanılan 2 adet DC motor bu entegre yardımıyla sürülmüştür.

Üst tarafta bahsedilen tüm elemanların tipik uygulamaları ve özellikleri dikkate alınarak elektronik devre çizimi yapılmıştır. PCB kart tasarımı ölçü ve yerleşim olarak aşağıda bahsedilen 3 boyutlu ana gövde tasarıma uygun olacak şekilde tasarlanmıştır.

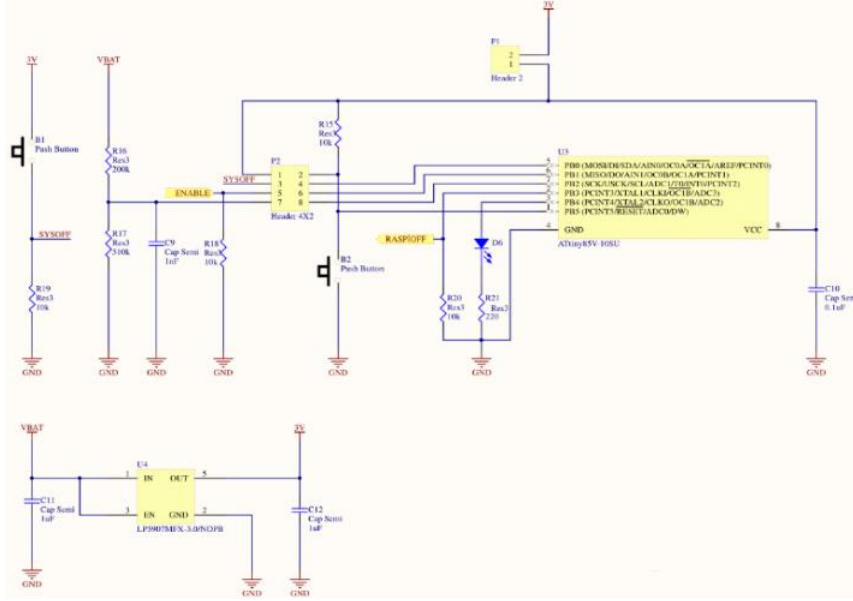
İlk önce devrenin şematik çizimi yapılmıştır. Şekil 2, 3 ve 4’te bu şematik çizimler görülmektedir. Şekil 2’deki şematik çizimden bahsetmek gerekirse: Li-Po pili şarj etmek için adaptör takılacak J1 mikro USB

soketi kullanılmıştır. Şarj kontrolü için ise U2 yani MCP73871 entegresi eklenmiştir. Entegrenin özellikleri ve tipik uygulaması dikkate alınarak diğer pasif devre elemanları da gerekli yerlere eklenmiştir. Li-Po pil şarj olurken kırmızı, şarj bittiğinde ise yeşil LED yanması için sırasıyla, D3 ve D4 ledi eklenmiştir. Şekil 2’deki bir diğer entegre de U1 yani TPS61090 step up DC-DC regülatör entegresidir. Bu entegre 1,8 ila 5,5 V arası gerilimleri 5 V 500 mA çıkış değerlerine sabitlemektedir. Entegrenin üstte belirtilen özellikleri ve tipik uygulaması dikkate alınarak diğer pasif devre elemanları gerekli yerlere eklenmiştir. Regülasyonun gerçekleştiğinin anlaşılması için yeşil D2 LED’i eklenmiştir. Pilin geriliminin düştüğünü yani şarjının azaldığını anlamak için ise kırmızı D5 LED’i eklenmiştir.

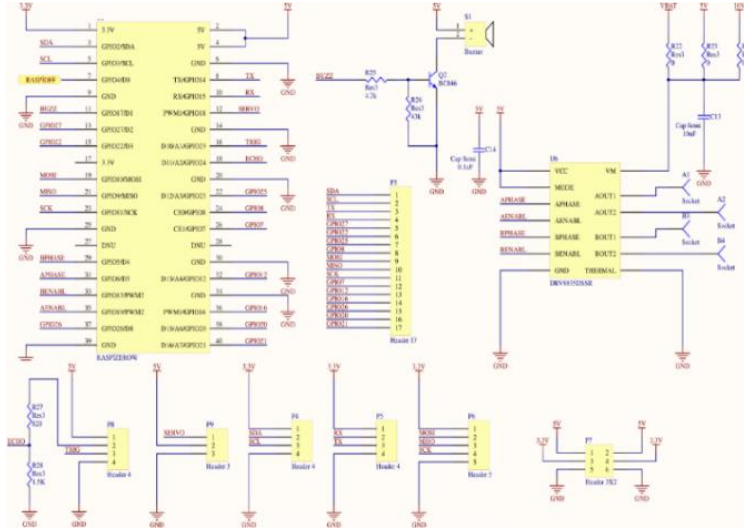


Şekil 2: Elektronik devre şeması- 1. kısım.

Şekil 3’te ise sistemin açılıp kapanmasını kontrol etmek için U3 yani ATtiny85 mikrodenetleyicisi kullanılmıştır. Attiny85 mikrodenetleyicisinin PB1 nolu pini ile TPS61090 step up DC-DC regülatör entegresinin Enable pini birleştirilerek açma kapama kontrolü sağlanmıştır. Şekil 3’teki bir diğer entegre ise LP5907MFX DC-DC regülatör entegresidir. Bu entegre Attiny85 mikrodenetleyicisini 3V ile beslemek için kullanılmıştır. Ayrıca P2 konektörü de Attiny85 mikrodenetleyicisinin SPI pinlerinden kolayca programlanması için bırakılmıştır. Şekil 3’te görüldüğü üzere PB0 ve PB5 girişlerine push butonlar bağlanmıştır. Bunlardan B1 butonu sistemin kapa komutunu almasını sağlamaktadır. B2 ise tekrar başlatmak için kullanılır. Buton PB5, yani ATtiny85 mikrodenetleyicisinin Reset girişine bağlı olduğundan butona basıldığında mikrodenetleyici kendini Resetler ve sistem yeniden açılır. Sistem ilk açıldığında voltaj değerlerini okumak için Li-Po pil, gerilim bölücü devre ile ATtiny85 mikrodenetleyicisinin ADC özellikli PB2 pinine bağlanmıştır. ATtiny85 mikrodenetleyicisi sadece sistem çalışırken aktif durumdadır, diğer durumlarda derin uyku modunda çalışmaktadır. ATtiny85 mikrodenetleyicisinin PB3 pin’ine ile Raspberry Pi bilgisayarının GPIO4 pini birbirine bağlanmıştır. Bu sayede kapama kontrolü için haberleşme hattı oluşturulmuştur. Batarya ve açma kapama kontrolü için oluşturulan senaryo Şekil 10’da ayrıntılı olarak verilmiştir.



Şekil 3: Elektronik devre şeması – 2. kısım.

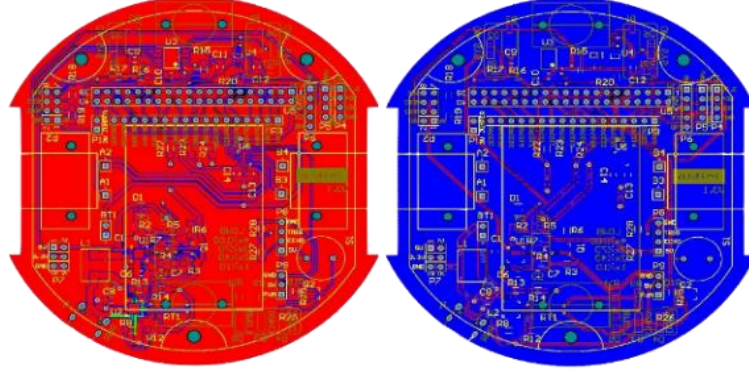


Şekil 4: Elektronik devre şeması – 3. kısım.

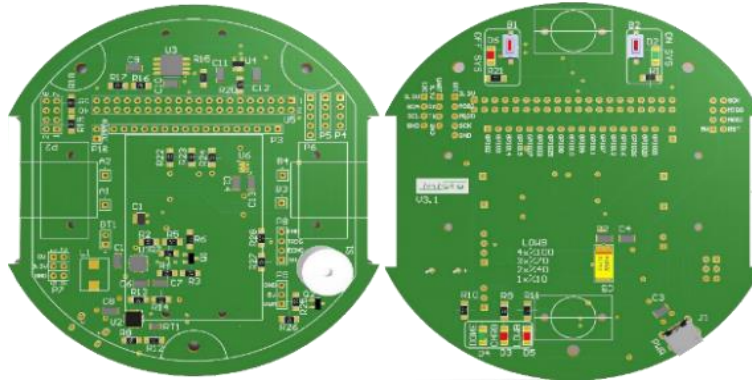
Şekil 4'teki şematik çizimde Raspberry Pi bilgisayarı U5 olarak isimlendirilmiştir. Raspberry Pi bilgisayarının pinlerinin özelliklerine göre P4, P5, P6 ve P3 olarak konektörlere dağıtılmıştır. Bunlardan P4 I2C haberleşmesi için P5 UART haberleşmesi için P6 ise SPI haberleşmesi için ayrılmıştır. P3 ise geriye kalan tüm GPIO çıkışları için ayrılmıştır. Platformda bulunacak DS65HB servo motoru için P9, HC-SR04 sensörü için P8 konektörü kullanılmıştır. HC-SR04 sensörü TTL 5V seviyesinde çalıştığı için, R27 ve R28 gerilim bölücü dirençleriyle gerilim 3,3V'a çekilmiştir. Ayrıca S1 buzzer'ının GPIO pinlerinin sağlayacağı akımdan daha fazla akım çekmesinden dolayı Q2 yani BC846 N kanal transistör kullanılmıştır.

DRV8835 ise platformda bulunan 2 adet DC motoru sürmek için kullanılmıştır. Tipik uygulama ve özellikleri dikkate alınarak Raspberry Pi bilgisayarından her bir motor için lojik(A/BPHASE) ve PWM(A/BENABLE) bilgisi alınmıştır. Motor beslemesi ise 5V veya 10V olmak üzere kullanıcının seçimine bırakılmıştır. Son olarak P7 ile 3,3V ve 5V alınabilecek konektörler bırakılmıştır.

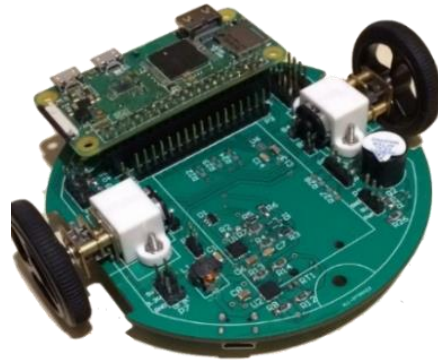
Şematik çizim yapıldıktan sonra PCB kart çizimi yapılmıştır. Çizim şematikteki iletken yol bağlantılarına göre ve bazı elemanların tipik PCB kart uygulamaları dikkate alınarak, 2 katlı olarak yapılmıştır. Şekil 5'te ana PCB kart tasarımının iletken yolları görünümü, Şekil 6'da ise 3 boyutlu görünümünün program çıktıları verilmiştir.



Şekil 5: Ana PCB kart iletken yolları görünümü.



Şekil 6: Ana PCB kart 3B görünümü.

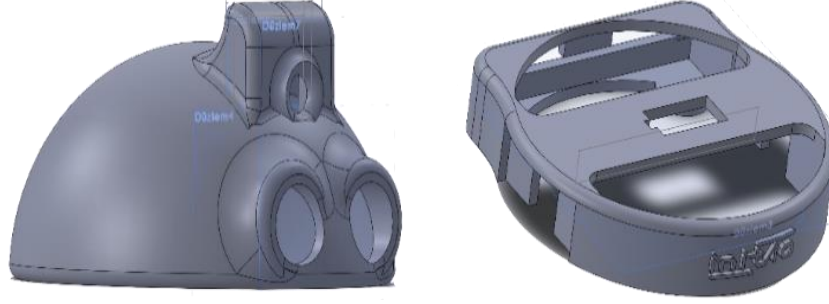


Şekil 7: Ana PCB kart görünümü.

Daha sonra tasarlanan ana PCB kart çizim programı ile GBR formatına çevrilmiş ve PCB kart basımı yapan bir şirkette basılmıştır. Basım sürecinin ardından üzerindeki entegrelerin lehimleri yapılmış ve test edilmiştir. Ardından ana PCB kartta önceden bırakılmış deliklere DC motorlar ve sarhoş tekerler vidalar yardımıyla sabitlenmiştir. Ana PCB kart son olarak Şekil 7'deki haline getirilmiştir.

PCB tasarımı tamamlandıktan sonra 3 Boyutlu ana gövde tasarımı yapıp 3 boyutlu yazıcı yardımıyla basılmıştır. Üst ana gövdede Raspberry Pi bilgisayar, Kamera Modülü ve HC-SR04 ultrasonik mesafe sensörü konumlandırılmıştır. Alt ana gövdede ise DS65HB servo motoru konumlandırılmıştır. Servo motor ile üst ana gövdenin alt ana gövdeden ayrı hareket etmesi sağlanmıştır. Bu sayede üst gövdenin,

180 derecelik alanda istenilen konuma gelmesi sağlanmıştır. Bu yöntem kameraya gövdenin hareket edemeyeceği yerlerde belli görüntüleme olanakları sunmuştur. Şekil 8’de üst ve alt ana gövdenin 3 boyutlu program çıktıları verilmiştir.



Şekil 8: 3B üst ana gövde (sol) ve alt ana gövde (sağ).

Platformun son halini oluşturmak için Ana PCB kart ve Ana gövdeler üstte belirtilen donanımlar ile birleştirilmiştir. Ardından platform Şekil 9’da görülmekte olan son haline getirilmiştir.



Şekil 9: Robot platformun son hali.

4. Yazılım ve Donanımın Denenmesi

Programlanmaya ilk olarak Arduino UNO kullanılarak, Attiny 85 mikrodnetleyicisinden başlanmıştır. Öncelikle Arduino UNO ile Attiny 85 mikrodnetleyicisi arasında SPI pinlerinin karşılıklı bağlantıları yapılmıştır. Bağlantılar yapıldıktan sonra Attiny 85 mikrodnetleyicisi üzerine önyükleyici yazdırılmıştır. Ardından Arduino IDE kullanılarak batarya ve açma kapama kontrol yazılımı yazılmıştır [20]. Batarya kontrolden kısaca bahsetmek gerekirse mikrodnetleyicinin ADC özelliği kullanılarak sürekli olarak batarya denetlenmektedir. Şekil 10’da Attiny85 mikrodnetleyicisine yazılan batarya ve açma kapama kontrol programın akış diyagramı verilmiştir.



Şekil 10: Attiny 85 için yazılan programın akış diyagramı.

Attiny 85 mikrodenetliycisinin programlanması bittikten sonra Raspberry Pi bilgisayarı programlanmıştır. Platformun IOT cihazlar gibi web üzerinden kontrol edilmesi için Flask framework yapısı kullanılarak Python kodları yazılmıştır [21,22]. İlk önce Raspberry Pi üzerine Flask framework yapısının kütüphanelerini yüklemek için terminal üzerine sırasıyla Şekil 11'de görülen kodlar yazılmıştır.

```
$ sudo apt-get update
$ sudo apt-get upgrade
$ pip install Flask
```

Şekil 11: Flask yükleme adımları.

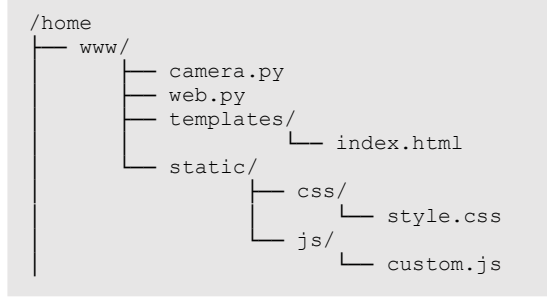
Kütüphane yüklemesi tamamlandıktan sonra Flask ile ilk olarak yerel ağ üzerinden web video yayını yapılmıştır. Şekil 12'de görülen kodlar terminale yazılarak yayım için gerekli kaynak kodlara ulaşılmıştır.

```
$ git clone
https://github.com/miguelgrinberg/flask-
video-streaming.git
```

Şekil 12: Video yayını için kaynak kodlar.

Kod yazıldıktan sonra Home/pi klasörü üzerinde flask-video-streaming klasörü oluşturulmuştur. Bu kodlar Python 3 ile çalıştırılmış ve bu sayede yerel ağ üzerinden web video yayını yapılmıştır. Web video yayını yapıldıktan sonra web üzerinden sensör verilerinin gösterileceği ve GPIO kontrolünün yapılacağı bir web arayüz oluşturulmuştur. Bunun için HTML, CSS ve JavaScript kodları kullanılmıştır [18][19]. Bu

oluşturulan web arayüzün Flask ile uyumlu çalışması için Flask framework yapısında daima kullanılması gereken klasör hiyerarşisine uygun olması gerekmektedir. Şekil 13'te bu projede kullanılan klasör hiyerarşisi verilmiştir.



Şekil 13: Flask framework klasör hiyerarşisi.

HTML kodu Şekil 13'te görüldüğü gibi templates klasörü altına index.html dosya adı ile klasörlenmiştir. Siteyi görsel açıdan zenginleştirmek için CSS kodları css klasörü altına style.css dosyası olarak klasörlenmiştir. Klasörleme yapıldıktan sonra yazılan Python kodu Python 3 kullanılarak çalıştırılmıştır. Daha sonra terminale “cd www/” yazılarak klasörün içine ulaşılmıştır. Sonra terminale “python3 web.py” yazılmış ve Python kodu Raspberry Pi bilgisayarı üzerinde çalıştırılmıştır.

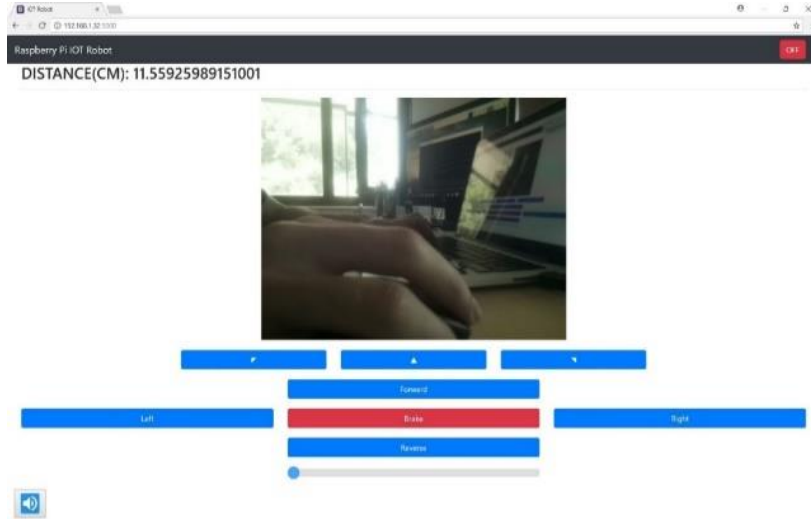
```

pi@pi:~ $ cd www/
pi@pi:~/www $ python3 web.py
* Running on http://0.0.0.0:5000/ (Press CTRL+C to quit)
* Restarting with stat

```

Şekil 14: Terminal çıktısı.

Programın çalışmasından bahsetmek gerekirse Şekil 14'te alınan terminal çıktısı, Raspberry Pi bilgisayarının o anda sahip olduğu yerel IP numarası ve 5000 numaralı portu kullanarak web server yayınına başladığını ifade etmektedir. HTML kodlar artık aynı yerel ağa bağlı olan ve bir web tarayıcısı çalıştırabilen herhangi bir cihazdan erişilebilir durumdadır. Şekil 15'te görüldüğü gibi web tarayıcısının link kısmına Raspberry Pi bilgisayarının IP adresi yazılarak kontrol paneline ulaşılabilir ve robotun kontrolü gerçekleştirilebilir. Şekil 15'te görülen arayüz HTML, CSS ve JavaScript kodları ile oluşturulmuştur. Şekil 15'te verilen bu web arayüzde en üstte canlı web yayını yapan ekran görülmektedir. Ayrıca DC motor kontrolü için ileri, geri, sağ, sol, dur gibi kontrol butonları konumlandırılmıştır. Servo motor için ise üç adet yön butonu kullanılmıştır. Bunların yanı sıra buzzer ve sistemi kapat butonları da kullanılmıştır. Arayüzde görülen slider ise DC motor hızını ayarlamak için kullanılmıştır. En üstte solda görülen uzaklık verisi ise HC-SR 04 sensöründen gelen veriler kullanılarak dinamik olarak arayüzde gösterilmiştir. Yazılan Python ve web arayüz kodlarında temel yapılara bakılırsa; Şekil 16'da bir kısmı görülen JavaScript kodları, control URL'si üzerinden HTTP GET metodunu kullanarak Python programına yön ve hız bilgisini aktarır.



Şekil 15: Web arayüz.

```
function Forward(){
$.ajax({
  type: "GET",
  url: $SCRIPT_ROOT + "/control",
  contentType: "application/json",
  charset="utf-8",
  data: { direction: 1, speed:
$("#Speed").val() }
});
}
```

Şekil 16: Kontrol- JavaScript kısmı.

Daha sonra Flask framework yapısı kullanılarak, Şekil 17’de görüldüğü gibi control URL’si üzerinde motor kontrolleri için yazılan program Python üzerinde çalıştırılır. Arayüzde bulunan tüm butonlar için benzer şekilde programlama yapılmıştır. Uzaklık verisinin web arayüze aktarılması ve dinamik bir yapıda çalışması için yazılan JavaScript kodları Şekil 18’de belirtilmiştir. Sensör verisinin Python kısmında okunup web arayüze aktaran Python kodu Şekil 19’da verilmiştir.

```
@app.route('/control', methods=['GET'])
def Control():
    direction =
    request.args.get('direction', type=int)
    speed = request.args.get('speed',
    type=int)
    #print(speed)
    Motor_Control(direction, speed)
    return 'control'
```

Şekil 17: Kontrol- Python kısmı.

```
$(document).ready(function(){
  function updateThing(thingState){
    $('#distance').text(thingState.distance)
  };
  var thingSource = new EventSource("{{
url_for('thing')}}");
  thingSource.onmessage = function(e) {
    var data = $.parseJSON(e.data);
    updateThing($.parseJSON(e.data));
  }
});
```

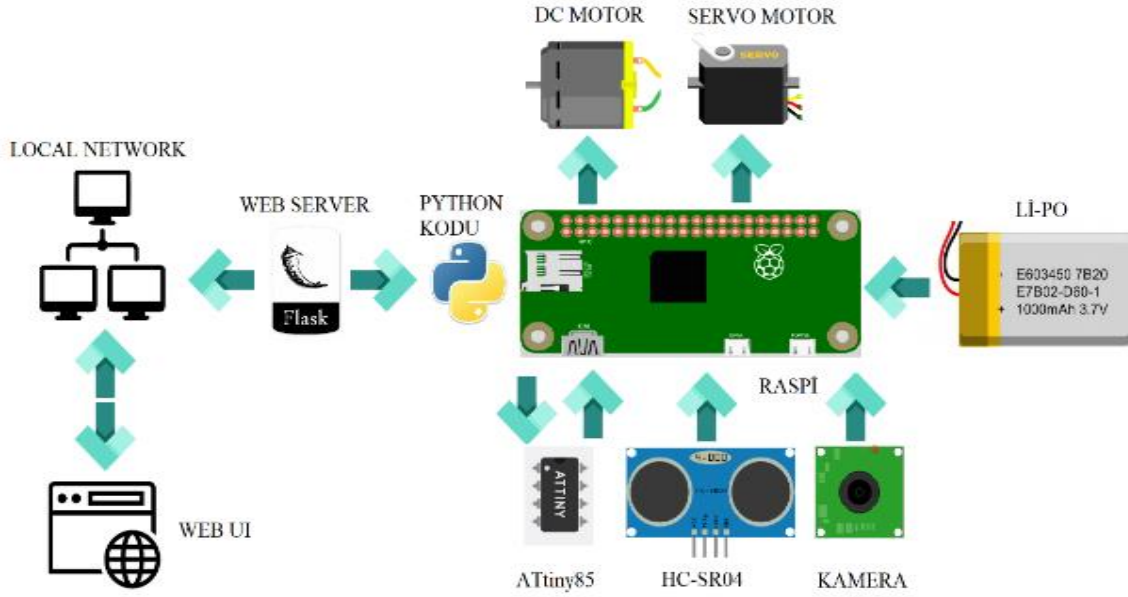
Şekil 18: Dinamik veri aktarımı JavaScript kısmı.

```

@app.route('/thing')
def thing():
    def read_thing_state():
        while True:
            distance_now = distance()
            time.sleep(1)
            thing_state = {'distance' : distance_now }
        yield
    'data:{0}\n\n'.format(json.dumps(thing_state
    ))
    return Response(read_thing_state(),

```

Şekil 19: Dinamik veri aktarımı Python kısmı.



Şekil 20: Platformun genel diyagramı.

Üstte bahsedilen adımların daha iyi anlaşılması için Şekil 20’de genel bir diyagram verilmiştir. Diyagramda da görüldüğü gibi Raspberry Pi bilgisayarı üzerinde çalışan Python kodu Flask web server yardımıyla yerel ağımızda bir web arayüz sayesinde kontrol edilebilir. Bu web arayüz de Şekil 15’te görülmektedir. Ayrıca sistemin genel girdileri Şekil 20’de belirtildiği gibi HC-SR04 ve kameradır, sistemin genel çıktıları ise DC ve servo motorlardır. ATtiny85 mikrodenetleyicisi ise hem girdi hem çıktı olarak çalışmaktadır. Oluşturulan bu yapı yine Şekil 20’de görüldüğü gibi 3,7 V 1800 mAh’lık Li-Po pil ile beslenmektedir.

5. Sonuçlar ve Öneriler

Initio robot platformu donanım ve yazılım olarak başarıya ulaşmıştır. Projenin geliştirme sürecinde karşılaşılan sorunlar çözülerek kararlı bir yapıya getirilmesi sağlanmıştır. Proje planlanan şekilde sonlandırıldıktan sonra edinilen deneyimlere dayalı çıkarımlar yapılmıştır. Bunlardan ilki tek çekirdekli ARM11 mimarili SoC’a sahip Raspberry Pi Zero W bilgisayarının görüntü işleme gibi bazı performans gerektiren uygulamalarda zayıf kalmasıdır. Buna çözüm olarak da dört çekirdekli ARM A53 mimarili SoC’a sahip Raspberry Pi 3 modeline geçilmesi önerilmektedir. Ancak model değişiminin maliyeti ve enerji tüketimi daha fazla olacağından sistem tasarımının tekrar gözden geçirilmesi de gerekecektir. Diğer çıkarımlar ise; gelecek olursak platformda açma kapama ve batarya yönetimi için Attiny85 mikrodenetleyici kullanılmasına rağmen motor sürüş kontrolleri Raspberry Pi Zero W bilgisayarından yapılmış olmasıdır. ARM’ın A serisi mimarilerinin daha çok işletim sistemi koşturmak için kullanılmasından ötürü motor sürüşü PWM dalga üretimi gibi konularda kararsızlıklar oluşmuştur. Ayrıca Raspberry Pi Zero W’nün üzerinde ADC ve DAC bulundurmaması platformda ARM’ın M

mimarili mikrodeneleyici yapısına duyulan ihtiyacı ortaya çıkarmıştır. Bu yapı iki şekilde oluşturulabilir: Bunlardan biri ST Microelectronics firmasının yakın zamanda piyasaya sunduğu, içinde hem A hem M mimarisini barındıran STM32MP1 serisi MPU'lar ile veya ikinci bir yöntem olarak Raspberry Pi 3 bilgisayarının yanında gömülü yine ST Microelectronics firmasına ait STM32F407 mikrodeneleyicisi kullanılabilir. Her iki yöntemde de mimarileri haberleştirmek için SPI, I2C, UART gibi haberleşme yöntemleri kullanılabilir.

Bir diğer konu ise herhangi bir ağa bağlı değilseniz veya herhangi bir monitöre sahip olmadığınızda Raspberry Pi'ye erişiminizin olmaması sorunudur. Sorunun çözümü için donanım kısmına yerleştirilecek FT232 seri haberleşme entegresi yardımıyla Raspberry Pi ve ana bilgisayarınız arasında UART protokolü üzerinden haberleşme yapılabilir. Bu sayede GUI yani grafiksel kullanıcı arayüzünü göremesek de teminal üzerinden Linux komutlarıyla Raspberry Pi kullanabilmektedir.

Platform birçok yazılım projesinde kullanılacak donanıma sahiptir. Bu çalışmada, robot ile yapılabilecek projelerden biri olan internet kontrollü bir yapıdan bahsedilmiştir. İlerideki çalışmalarda yazılımda ROS yapısı kullanılabilir ve bunun yanısıra eğitim kurumlarında Scratch programı ile (blok diyagramları şeklinde) robot hareketleri programlanabilir.

Son olarak platformun tasarım sürecinde kullanılan programlardan ve çıktılardan bahsetmek gerekirse 3 boyutlu dış gövde çizimleri SolidWorks ile yapılmış ve yazdırılmıştır. Sonuçlar gayet tatmin edici olup platforma dış görünüş olarak ürün görüntüsü vermiştir. Devre şemaları ise Altium'da çizilerek çift kat PCB kart olarak bastırılmıştır. Altium diğer PCB kart çizimi programlarına göre daha ayrıntılı olduğu için bu projede tercih edilmiştir ve tüm ihtiyaçları karşılamıştır. Devrenin simülasyonlarında ise Proteus ve Multisim gibi programlardan yararlanılmıştır. Baskısı yapılmış devre, simülasyonların birebir aynısıdır.

Teşekkür

Bu proje 1139B411702840 no'lu başvuru numarası ile 2209-B Sanayiye Yönelik Lisans Araştırma Projeleri Destekleme Programı 2017/12 kapsamında TÜBİTAK tarafından desteklenmiştir.

Referanslar

- [1] K. Karabulut, Z. Güneş, "System Design For Internet Of Things And Network Coding Applications In The Wireless Personal Area Networks" *Yüksek Lisans Tezi, İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü*, İstanbul, 2015
- [2] U. Çabuk, O. Dağdeviren, Y. Yiğit, M. Süvari, "Gömülü Sistemler için Android Tabanlı Bir Mikroişlemci Programlama Yazılımı ve Arayüzü". *Bilişim Teknolojileri Dergisi*, Cilt 11, No 4, Ankara, ss. 321-332, 2018
- [3] B. Şişman, "Eğitimde Robot Kullanımı" *Eğitim Teknolojileri Okumaları* 2016, A. İşman, H.F. Odabaşı, B. Akkoyunlu, Eds, *TOJET*. Ankara, ss.299-314, 2016
- [4] M. Akçakoca, "Eğitim ve Araştırma Amaçlı Gezgin Robot Geliştirilmesi" *Yüksek Lisans Tezi, Eskişehir Osmangazi Üniversitesi, Fen Bilimleri Enstitüsü*, Eskişehir, 2017
- [5] O. Yazgan, "Genel Amaçlı Bir Mobil Robot Platformun Tasarımı ve Gerçeklenmesi" *Yüksek Lisans Tezi, İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü*, İstanbul, 2008
- [6] M. K. Gönüllü, "Mobil Robot Araştırmalarında Kullanılmak Üzere Bir Mobil Robot Platformu Geliştirmesi" *Yüksek Lisans Tezi, Orta Doğu Teknik Üniversitesi, Fen Bilimleri Enstitüsü*, Ankara, 2013
- [7] N. Yılmaz, Ş. Sağıroğlu, M. Bayrak, "Genel Amaçlı Web Tabanlı Mobil Robot: Sunar" *Gazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi*, Cilt 21, No 4, Ankara, ss. 745-752, 2006
- [8] S. F. Barrett, et al. "Using Robots to Teach Complex Real Time Embedded Systems Concepts", *Computers in Education Journal*, Cilt 16, No.4, ss. 58-58, 2006
- [9] Z. Fang, F. Yanhua, C. Tianyou, "A low-cost modular robot for research and education of control systems, mechatronics and robotics", *4th IEEE Conference on Industrial Electronics and Applications*, 2009
- [10] J. McLurkin, et al. "Using multi-robot systems for engineering education: Teaching and outreach with large numbers of an advanced, low-cost robot", *IEEE Transactions on Education* cilt 56 No. 1, ss. 24-33, 2013
- [11] T. Bräunl, "Embedded Robotics: Mobile Robot Design and Applications with Embedded Systems", Springer Science & Business Media, 2008

- [12] E. Küçükülahlı, R. Güler, “Open Source Mobile Robot with Raspberry Pi”, *Balkan Journal Of Electrical & Computer Engineering (BAJECE)*, Turkey, ss. 242-247, 2015
- [13] H-S. Juang, K-Y. Lum, “Design and Control of a Two-Wheel Self-Balancing Robot using the Arduino Microcontroller Board”, *10th IEEE International Conference on Control and Automation (ICCA)*, Hangzhou, China, ss.634-639, 2013
- [14] A. Bradbury, B. Everard, *Learning Python with Raspberry Pi*, Wiley, New Jersey, 978-1-118-71705-9, 2014
- [15] Y. Zhu, “*Embedded Systems with ARM Cortex-M Microcontrollers in Assembly Language and C: Third Edition*”, E-Man Press LLC, 978-0982692660, 2017
- [16] E. Williams, “*Make: AVR Programming*”, Maker Media, Inc, 978-1449355784, 2014
- [17] L. Nashelsky, R. L. Boylestad, “*Electronic Devices and Circuit Theory*”, Prentice Hall, New Jersey, 978-0135026496, 2009
- [18] C. Murphy, R. Clark, O. Studholme, D. Manian, “*Beginning HTML5 and CSS3*”, Apress, New York, 978-1-4302-2875-2, 2012
- [19] B. J. D'mello, S. S. Sriparasa, “*JavaScript and JSON Essentials*”, Packt, Birmingham, 978-1783286034, 2018
- [20] S. Monk, *Programming Arduino Getting Started with Sketches*, McGraw-Hill Publishing, New York, 9780071784221, 2012
- [21] M. Grinberg, “*Flask Web Development*”, O'Reilly Media, California, 9781491991725, 2014
- [22] J. Spurlock, “*Bootstrap: responsive web development*”, O'Reilly Media, California, 9781449343910, 2013