

**YÜK TAŞIMAYI OTONOM OLARAK  
GERÇEKLEŞTİREBİLEN ÇOKLU MOBİL ROBOT  
UYGULAMASI**

**2015  
YÜKSEK LİSANS TEZİ  
ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ**

**Emrah BUDAK**

**YÜK TAŞIMAYI OTONOM OLARAK GERÇEKLEŞTİREBİLEN  
ÇOKLU MOBİL ROBOT UYGULAMASI**

**Emrah BUDAK**

**Karabük Üniversitesi  
Fen Bilimleri Enstitüsü  
Elektrik Elektronik Mühendisliği Anabilim Dalında  
Yüksek Lisans Tezi  
Olarak Hazırlanmıştır**

**KARABÜK  
Haziran 2015**

Emrah BUDAK tarafından hazırlanan “YÜK TAŞIMAYI OTONOM OLARAK GERÇEKLEŞTİREBİLEN MOBİL ROBOT UYGULAMASI” başlıklı bu tezin Yüksek Lisans Tezi olarak uygun olduğunu onaylarım.

Doç. Dr. Raif BAYIR

Tez Danışmanı, Mekatronik Mühendisliği Anabilim Dalı

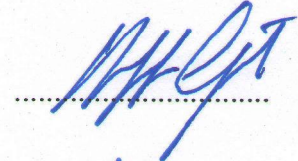


Bu çalışma, jürimiz tarafından oy birliği ile Elektrik-Elektronik Mühendisliği Anabilim Dalında Yüksek Lisans Tezi olarak kabul edilmiştir. 19/06/2015


Ünvanı, Adı SOYADI (Kurumu)

İmzası

Başkan : Yrd. Doç. Dr. Rifat HACIOĞLU(BEÜ)



Üye : Doç. Dr. Raif BAYIR(KBÜ)



Üye : Yrd. Doç. Dr. Ali UYSAL (KBÜ)

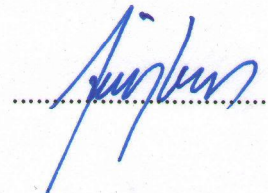


...../...../2015

KBÜ Fen Bilimleri Enstitüsü Yönetim Kurulu, bu tez ile, Yüksek Lisans derecesini onamıştır.

Prof. Dr. Nevin AYTEMİZ

Fen Bilimleri Enstitüsü Müdürü



*“Bu tezdeki tüm bilgilerin akademik kurallara ve etik ilkelere uygun olarak elde edildiğini ve sunulduğunu; ayrıca bu kuralların ve ilkelerin gerektirdiği şekilde, bu çalışmadan kaynaklanmayan bütün atıfları yaptığımı beyan ederim.”*

Emrah BUDAK



## **ÖZET**

**Yüksek Lisans Tezi**

### **YÜK TAŞIMAYI OTONOM OLARAK GERÇEKLEŞTİREBİLEN ÇOKLU MOBİL ROBOT UYGULAMASI**

**Emrah BUDAK**

**Karabük Üniversitesi**

**Fen Bilimleri Enstitüsü**

**Elektrik-Elektronik Mühendisliği Anabilim Dalı**

**Tez Danışmanı:**

**Doç. Dr. Raif BAYIR**

**Haziran 2015, 63 sayfa**

Robotların endüstride otonom olarak kullanılması iş güvenliğinin artırılması, insan gücünün azaltılması ve malzeme güvenliğinin sağlanması açısından önemlidir. Gelişen teknoloji robotlar ve kullanım alanlarının genişletilmesini de sağlamaktadır. Çoklu ve sürü robot sistemleri çeşitli görevleri yerine getirmek amacıyla geliştirilen ve en az iki robottan oluşan sistemlerdir. Bu tez çalışmasında, otonom hareket edebilen, kablosuz olarak birbiriyle ve merkezle haberleşen, üzerinde yer alan yük hücresi algılayıcısı ile ağırlığını algıladığı yükü Öncü ve Destek isimli iki adet robot ile platform dışına taşıyan çoklu robot sistemi geliştirilmiştir. Çoklu robot sisteminde, eğri uydurma yöntemi ile en kısa yol bulunarak yükleri taşıma görevi gerçekleştirilmektedir. Robotun yükleri algılamasında karşılaşılan problemlerin giderilmesinde uygunluk değerinin hesaplanması yöntemi kullanılarak, optik yayımlı engel algılayıcılarının etkili kullanımı sağlanmaktadır. Taşınacak yükün ağırlığı yük hücresiyle belirlendikten sonra, Öncü robotun taşıyabileceği sınırlar

içerisinde ise tek başına taşımakta ve Destek robot hazır durumda beklemektedir. Öncü robotunun taşıyamayacağı bir ağırlıkta ise yükün taşınması için Destek robotu çağırılmakta ve her ikisi yükü platform dışına itmektir. Gerçekleştirilen bu çoklu robot sisteminde çizgi takibi yöntemi yerine eğri uydurma yöntemi kullanıldığında %11 oranında yolun kısaltılması sağlanmış ve görev başarıyla tamamlanmaktadır. Öncü ve Destek robotunun birlikte çalışmasını sağlayacak bir yükün platform dışına taşınması görevi %70 doğrulukla sağlanmaktadır. Bu çoklu robot mobil sistemleri endüstride ürünlerin depo düzeninde ve zeki taşımacılıkta rahatlıkla kullanılabilir.

**Anahtar Sözcükler :** Mekatronik, elektrik-elektronik mühendisliği, mekatronik teknolojileri eğitimi.

**Bilim Kodu** : 905.1.096

## **ABSTRACT**

**M. Sc. Thesis**

### **APPLICATION OF MULTIPLE MOBILE ROBOTS CARRYING LOADS AUTONOMOUSLY**

**Emrah BUDAK**

**Karabük University**

**Graduate School of Natural and Applied Sciences**

**Department of Electrical-Electronics Engineering**

**Thesis Advisor:**

**Assoc. Prof. Dr. Raif BAYIR**

**June 2015, 63 pages**

Using robots autonomously in industrial is important in terms of increasing job security, decreasing human power and material security. Developing technology ensures extension of the robots and the usage areas. Multiple robot system is the system which consists of at least two robots and is developed for performing various tasks. In this thesis study, a multiple robot system which can act autonomously, communicates wirelessly with each other and the center, carries the cargo which is sensed with the load cells on two robots called as Oncu and Destek to output of the platform has been implemented. Carrying cargos task has been performed by finding the shortest way with curve fitting algorithm in multiple robot system. Optical emission distance sensor is used effectively by improving a method similar to the calculating fitness value used also in genetic algorithm for removal of the inference encountered while sensing the cargos. After determining the cargo weight with the load cell, if the cargo weight is in the limits of the weights for Oncu robot, Oncu

pushes the cargo by itself and Destek robot waits on stand by mode. If the cargo is not in the limits for Oncu, Destek is called and both of them push the cargo to the platform output. As a result, reducing the route by 11% could be provided by using curve fitting algorithm instead of line tracking method and the task could be performed successfully for the implemented multiple robot system. The task of carrying the cargo to the platform output with Oncu and Destek robots together has been ensured with 70% success.

**Key Words** : Mechatronics, electric-electronic engineering, mechatronics technology education.

**Science Code** : 905.1.096



## TEŐEKKÜR

Bu tez alıőmasının araőtırılmasında ve yürütülmesinde ilgi ve desteęini esirgemeyen, engin bilgi ve tecrübelerinden yararlandıęım, yönlendirme ve bilgilendirmeleriyle alıőmamı bilimsel temeller ışığında őekillendiren danıőman hocam sayın Do. Dr. Raif BAYIR'a sonsuz teőekkürlerimi sunarım.

Tez alıőmam boyunca laboratuvar ortamlarından faydalandıęım Gazi Üniversitesi Bilgisayar Mühendislięine ve tez alıőmamda büyük desteklerini esirgemeyen Yrd. Do. Dr. Fecir DURAN ve Dr. Meral ÖZARSLAN YATAK'a sonsuz teőekkürlerimi sunarım.

Sevgili aileme maddi manevi her zaman yanımda oldukları için tüm kalbimle teőekkür ederim.

## İÇİNDEKİLER

	<u>Sayfa</u>
KABUL .....	ii
ÖZET .....	iv
ABSTRACT .....	vi
TEŞEKKÜR .....	viii
İÇİNDEKİLER.....	ix
ŞEKİLLER DİZİNİ .....	xii
ÇİZELGELER DİZİNİ .....	xiv
SİMGELER VE KISALTMALAR DİZİNİ.....	xv
BÖLÜM 1 .....	1
GİRİŞ .....	1
BÖLÜM 2 .....	5
ROBOTİK SİSTEMLER .....	5
2.1. ROBOTİK.....	5
2.2. ÇOKLU ROBOT SİSTEMLERİ.....	9
2.2.1. Çoklu Robot Sistemlerinde Haberleşme .....	11
BÖLÜM 3 .....	13
MOBİL ROBOTLARIN MEKANİK VE ELEKTRONİK TASARIMIN GERÇEKLEŞTİRİLMESİ.....	13
3.1. MEKANİK DONANIM .....	14
3.1.1. Deney Düzeneği.....	16
3.1.2. Motorlar.....	17
3.1.3. Tekerlekler.....	18
3.2. ELEKTRONİK DEVRE TASARIM.....	19
3.2.1. Mikro Denetleyici .....	19
3.2.2. INA125 Enstrümantasyon Yükseltici ve Yük Hücresi.....	19

	<u>Sayfa</u>
3.2.3. Kontrast Sensörü.....	20
3.2.4. Mesafe Sensörü.....	21
3.2.5. Batarya .....	23
3.2.6. Regülatör Kartı.....	23
3.2.7. Motor Sürücü.....	24
3.2.8. Robot Denetim Kartı.....	24
BÖLÜM 4 .....	27
YAZILIM ALGORİTMALARI.....	27
4.1. ÖNCÜ ROBOTUNUN ALGORİTMASI.....	27
4.2. DESTEK ROBOTUNUN ALGORİTMASI.....	28
4.3. PID DENETLEYİCİSİ.....	28
4.3.1. Oransal Terim .....	29
4.3.2. İntegral Terimi .....	30
4.3.3. Türev Terimi.....	30
4.3.4. PID Katsayılarını Hesaplama Yöntemi.....	30
4.4. PID DENETLEYİCİSİNİN UYGULANMASI.....	31
4.5. YÖNELME ALGORİTMASI.....	37
4.6. ENGEL ALGILAMA ALGORİTMASI .....	38
4.7. GÜÇ YÖNETİM SİSTEMİ .....	41
BÖLÜM 5 .....	43
HAREKET FONKSİYONU VE HABERLEŞME PROTOKOLÜ .....	43
5.1. EĞRİ UYDURMA .....	43
5.2. ROBOTUN KABLOSUZ İLETİŞİMİ .....	48
BÖLÜM 6 .....	52
DENEYSEL ÇALIŞMALAR .....	52
6.1. YÜKÜN ROBOTA YAKIN KONUMDA OLMASI .....	53
6.2. YÜKÜN ROBOTA UZAK KONUMDA OLMASI.....	54
6.3. YÜKÜ YARDIMLAŞARAK TAŞIMA .....	55
6.4. DENEY SONUÇLARI.....	57

	<b><u>Sayfa</u></b>
6.5.1. Yüke Yaklaşamama Hatası.....	57
6.5.2. Yüğü Taşıma Görevini Tamamlaması .....	58
6.5.3. Koordinat Güncelleme Görevi.....	58
BÖLÜM 7 .....	60
SONUÇ VE DEĞERLENDİRME .....	60
KAYNAKLAR.....	61
ÖZGEÇMİŞ.....	63

## ŞEKİLLER DİZİNİ

	<u>Sayfa</u>
Şekil 2.1. El-Cezeri 'nin eserleri .....	6
Şekil 2.2. İlk endüstriyel robot Unimate .....	7
Şekil 2.3. Endüstride sıkça kullanılan robot kol .....	7
Şekil 2.4. TUSAŞ'ın yapmış olduğu insansız hava aracı Anka.....	8
Şekil 2.5. Da Vinci robotu .....	9
Şekil 2.6. Kilobots sürü robotları .....	10
Şekil 2.7. Kutu itme görevi gerçekleştiren sürü robot uygulaması.....	11
Şekil 3.1. Sistemin bloş Şeması .....	13
Şekil 3.2. Tasarlanan Öncü robotu.....	14
Şekil 3.3. Tasarlanan sürü robot'un Solid Works çizimi.....	15
Şekil 3.4. Lazerde kesilen pleksiglass malzemeler .....	15
Şekil 3.5. Yük hücresi ve cisim itme aparatı .....	16
Şekil 3.6. Motor, motor tutucu ve sensör tutucular .....	18
Şekil 3.7. POLOLU 42x19 mm tekerlek .....	18
Şekil 3.8. Deney alanı.....	16
Şekil 3.9. Deney düzeneğinin koordinatsal gösterimi .....	17
Şekil 3.10. INA125 yük hücresi yükselteci .....	20
Şekil 3.11. CNY70 elektronik bağlantı şeması.....	21
Şekil 3.12. Kontrast sensörlerinin yerleşimi.....	21
Şekil 3.13. GP2D120 4-30 cm mesafe sensörü .....	21
Şekil 3.14. GP2D120X sensörünün çıkış karakteristiği .....	22
Şekil 3.15. Sensör'ün optik yayılımı .....	22
Şekil 3.16. 7.4 V Li-Po batarya.....	23
Şekil 3.17. Regülatör katı .....	23
Şekil 3.18. TB6612FNG motor sürücü katı.....	24
Şekil 3.19. ANKAKIT V2.2 geliştirme kartı.....	25
Şekil 3.20. ANKAKIT V2.2 PCB'si alltan ve üstten görünüş.....	26
Şekil 4.1. Öncü robotunun algoritması.....	27

	<b><u>Sayfa</u></b>
Şekil 4.2. Destek robotunun algoritması .....	28
Şekil 4.3. PID denetleyicinin temel yapısı .....	29
Şekil 4.4. Oransal denetleyici çıkışı .....	31
Şekil 4.5. Enkoder'in elektronik yapısı ve çıkış şekilleri.....	37
Şekil 4.6. Enkoder destekli ilerleme ve açısal dönüş algoritması.....	37
Şekil 4.7. Deney düzeneği .....	38
Şekil 4.8. Sensör çıkış değerleri grafiği.....	39
Şekil 4.9. Sensör çıkış değerlerinin incelenmesi.....	40
Şekil 4.10. Kullanılan Li-Po bataryanın deşarj eğrisi .....	42
Şekil 5.1. Tasarlanan robot'un evrensel düzlemde konumlandırılması .....	43
Şekil 5.2. Eğri uydurma yöntemiyle robotun ile yük arasında çizilen doğru .....	45
Şekil 5.3. Robot'un yapacağı açı ve gideceği mesafenin hesaplanması .....	46
Şekil 5.4. Zigbee topolojileri .....	49
Şekil 5.5. Xbee modül .....	50
Şekil 5.6. Protokolün yapısı.....	50
Şekil 5.7. Protokol örneği .....	51
Şekil 6.1. Robotun çizgileri takip etmesi ve eğri uydurma yöntemini kullanması ..	52
Şekil 6.2. Yük'ün robot'a yakın koordinatta olması deneyi.....	53
Şekil 6.3. Yük'ün robot'a uzak koordinatta olması deneyi .....	54
Şekil 6.4. Destek robotunun tarama işlemini bitirip yük'e yönelmesi .....	56
Şekil 6.5. Öncü ve Destek robotunun birlikte yük'ü itmesi .....	56
Şekil 6.6. Tarama görevinden sonraki hatalı açısal duruş .....	57
Şekil 6.7. Yük'e yaklaşma hatası .....	58
Şekil 6.8. Koordinat güncelleme hatası.....	59

## ÇİZELGELER DİZİNİ

	<u>Sayfa</u>
Çizelge 4.1. Ziegler Nichols'ün 2. metodu katsayıları .....	31
Çizelge 4.2. Yüklerin ölçüm hataları .....	40
Çizelge 5.1. Zigbee ve diğer RF çözümlerin karşılaştırılması .....	48
Çizelge 6.1. Yük'ün Y eksenindeki değişimlerine göre aldığı mesafeler.....	52
Çizelge 6.2. Yük'ün robota yakın koordinatta olma durumu.....	53
Çizelge 6.3. Yük'ün robota uzak koordinatta olma durumu .....	54
Çizelge 6.4. Öncü robotunun yardımlaşarak çalışma durumu .....	55
Çizelge 6.5. Destek robotunun yardımlaşarak çalışma durumu .....	55

## SİMGELER VE KISALTMALAR DİZİNİ

### SİMGELER

RPM	:	Revolutions Per Minute (Bir birim dakikada ki devir)
Mb/s	:	Mega Bit / Second (Mega Bit / saniye)
kHz	:	Kilohertz
MHz	:	Megahertz
GHz	:	Gigahertz
mA	:	miliamper
mAh	:	miliamper hour (miliamper saat)
mm	:	milimetre
cm	:	santimetre
ms	:	milisaniye
kg/cm	:	kilogram / santimetre

### KISALTMALAR

GKF	:	Genişletilmiş Kalman Filtresi
EKH	:	Eşzamanlı Konumlama ve Haritalama
SLAM	:	Simultaneous Localization And Mapping (Eşzamanlı Lokasyon ve Haritalama)
SOM	:	Self Organizing Map (Kendi kendine Harita Organizasyonu)
R.U.R.	:	Rossum's Universal Robots (Rossum'un Evrensel Robotları)
ADC	:	Analog Digital Converter (Analog Dijital Çevirici)
IMU	:	Inertial Measurement Unit (Atalet Ölçü Ünitesi)
PCB	:	Printed Circuits Board (Baskılı Devre Kartı)
ICSP	:	In Circuits Serial Programming (Dahili Seri Programlayıcı)
LDO	:	Low Drop Out (Düşük Gerilim Düşümlü)
PID	:	Proportional-İntegral-Derivative (Oransal-İntegral-Türev)



PWM	:	Pulse Width Modulation (Darbe Genişlik Modülasyonu)
DGM	:	Darbe Genişlik Modülasyonu
GPS	:	Global Positioning System (Küresel Konumlama Sistemi)
HVAC	:	Heating, Ventilating and Air Conditioning (Isıtma, Havalandırma ve İklimlendirme)
UART	:	Universal Asynchronous Receiver/Transmitter (Evrensel Asenkron Alıcı/Verici)
LED	:	Light Emitting Diode (Işık yayan Diyot)

## BÖLÜM 1

### GİRİŞ

Günümüzde robotlar insan hayatını kolaylaştırmak için tıp bilimi, savunma sanayi, endüstriyel üretim, uzay bilimleri ve yük taşıma gibi birçok alanda özellikle güvenlik ve hassasiyet gerektiren uygulamalarda yaygın olarak kullanılmaktadır. Robotlar, kullanıldıkları yerlerde ürün kalitesini arttırmakta ve üretim maliyetlerini düşürmektedirler. Robotların kullanılması ile birlikte, daha hassas ve daha işlevsel olabilmesi için yapılan çalışmalarla robotik teknolojisi her geçen gün gelişmektedir.

Robotların insan hayatını kolaylaştıran yönü bazen kendileri içinde söz konusu olabilmektedir. Bunun için günümüzde kullanılan çoklu robot sistemi, bir robotun tek başına yapamadığı görevlerin yerine getirilmesinde kullanılan daha etkili ve kararlı bir yoldur. Bunların gerçekleştirilmesinde ve senaryoların yerine getirilmesinde çeşitli yöntemler kullanılmaktadır. Yapılan çalışmalar, haritalama, yol bulma, robotların bireysel denetimi, otonomluğun sağlanması, robotların birbiriyle etkileşimi ve enerji denetimi gibi konular üzerinde yoğunlaşmaktadır.

Robotikte otonomluğu sağlamanın yollarından biri robotun bir çevreyi doğru bir şekilde haritalaması ve o çevrede kendini eş zamanlı olarak konumlandırma yeteneğidir. Bu sorun (Simultaneous Localization And Mapping - SLAM) Eşzamanlı Konumlama ve Haritalama (EKH) olarak adlandırılmaktadır. Saeedi ve diğerleri yaptıkları çalışmada EKH için çoklu robotlarla bir merkezi platform gerçekleştirmişlerdir [1]. Genişletilmiş Kalman Filtresi (GKF) veri füzyonu kullanarak önce tek robot EKH daha sonra da yeni bir konumlandırma grid haritalama füzyon algoritması kullanarak çoklu robot EKH geliştirmişlerdir. Haritalama füzyonu, görüntü ön-işleme, segmentasyon, çapraz korelasyon, gerçel dönüşüm matris yaklaşımı, dönüşüm matrisinin ayarlanması ve sonucun doğrulanması gibi çoklu basamak süreçleriyle gerçekleştirilmiştir. Saeedi ve

arkadaşlarının yaptıkları çalışmada, EKH işlemi için çoklu robotlarını yapay sinir ağlarını kullanarak geliştirmişlerdir [2]. Önerdikleri harita öğrenme yöntemi, kendi kendini düzenleyen haritalar yöntemine (Self Organizing Map), (SOM) dayalı öğrenmedir. Öğrenme safhasında haritadaki engeller, hücreleri kümeler halinde gruplandırarak öğrenilmektedir. Bu öğrenme danışmansız olup çıkışı, eğitme patternlerine ihtiyaç olmadan yapılabilmektedir. Gerçek çevrede çoklu robotlarla yapılan testler sonucunda önerilen çözümün etkinliği ortaya çıkmıştır. Yasuda gerçekleştirdiği çalışmada, çoklu robot sisteminde koordinasyon ve senkronizasyonu için otonom denetim mimarisi gerçekleştirmiştir [3]. Gelişmiş akıllı robot denetiminin en büyük engeli denetim sisteminin tasarımıdır. Çalışmada senkronize edilmiş, etkileşimli inter-robot işbirliğini gösteren net modeller sunulmuştur.

Çoklu robotlar çeşitli görevleri yerine getirmek için tasarlanmaktadır. Cheng ve diğerleri ev servisi için yaptıkları çalışmada, çoklu robotların işbirliği için görsel geri beslemeli denetim tekniklerinin uygulanması üzerine yoğunlaşmışlardır [4]. Anne robot ve çocuk robot olarak adlandırdıkları robotlardan her biri, kendi üzerindeki video kamera ve görüntüsünü, karşılıklı etkileşim ve geri beslemeli denetim için kullanılmıştır. Robotlar arasındaki iletişimi hızlandırmak ve bağlantıyı sadeleştirmek için merkezi bir hesaplama birimi gerçekleştirmişlerdir. Robotların görevleri bilgisayar görüntüleri yoluyla belirlenmiştir.

Yol belirleme için yapılan çalışmalarda çeşitli yöntemler kullanılmakta olup çoklu robot sisteminin işlemlerini kısa ve etkili bir şekilde gerçekleştirmeleri sağlanmıştır. Dinamik çevrelerde görev ve yol planlaması, dağıtılmış coğrafi görevlerde çoklu robotlar kullanılacağı zaman istenmektedir. İşbirliği içinde, hedef noktalara ulaşma görevinin gerçekleştirilmesi için görevlerin ve yolların atanması Abdi ve arkadaşları tarafından çalışılmıştır [5-7]. Önerdikleri yöntemde, dinamik çevrelerde kullanılmak üzere ayarlanabilir kuvvet alanı kullanmıştır. Bu analizden kuvvete göre her bir robota görev ataması yapılmaktadır. Aynı zamanda bu yolla çarpışmasız yol belirlenmesi de yapılmaktadır.

Zayıf koordinasyon mekanizmasından dolayı robotlar arasında görev ve kaynak paylaşımı sıkıntılı olmaktadır. Nagarajan ve diğerleri yaptıkları çalışmayla sezgisel

arama tabanlı görev dağılımını formüle etmişlerdir. Deneysel sonuçlar, önerdikleri algoritmanın farklı sorunlarda etkili sonuçlar verdiğini ortaya koymuştur [8].

Statik ve dinamik senaryolarda, yolda engel algılama ve bundan kaçınma gibi işlevleri olan çoklu robotların yol planlaması Genetik Algoritma yaklaşımı kullanılarak Cabreira ve diğerleri tarafından gerçekleştirilmiştir [9]. Netlogo yazılımı kullanarak, sorun için bir model geliştirmişlerdir. Çoklu robotlar ile çeşitli statik ve dinamik engeller içeren modelde, çarpışmalardan sakınıp en kısa adımları kullanarak amacın gerçekleştirilmesi sağlanmıştır. Aynı zamanda çalışmalarının kıyaslanması için A\* (A Star) arama algoritmasının da kullanmışlardır [10]. Çalışmalarının sonucunda önerdikleri genişletilmiş genetik algoritma yaklaşımı geçerli bir alternatif olmuştur.

Çoklu robot sistemlerinin işbirliği denetimi, çevre gözetimi ve güvenliğinden arama-kurtarma, yangın söndürmeye kadar birçok pratik uygulamada kullanılan önemli bir konudur. Goryca ve Hill yaptıkları çalışmada, işbirliği içinde çalışan çoklu robotlar için gerçek zamanlı denetim yazılımı üretmekte bir dizi kuramsal araç geliştirmişlerdir [11]. Biçimsel özellikleri ile tanımlanan çoklu robotların davranışlarını oluşturmayı garanti eden yüksek seviyeli danışmanlı denetim sağlamak için özellikle ayırık olay sistem teorisini uygulamışlardır. Biçimsel özellikleri kadar, robotların kendi çevrelerindeki yüksek seviyeli davranışlarının modeli detaylı olarak tanımlanmıştır. Değiştirilmiş Dijkstra's algoritması kullanılarak izin verilen robot davranışları belirlenerek bir yol belirlenmiştir. Bu değiştirilmiş algoritma çoklu robotların adreslenmesinde kullanılmıştır. Çalışmanın sonucu basit ama etkili bir senaryoyla sunulmuştur.

Bu tez çalışmasında, yük taşıma sistemlerinde kullanılmak üzere bir biriyle yardımlaşan çoklu robot sistemi uygulaması geliştirilmiştir. Bu çalışma kapsamında Öncü ve Destek olmak üzere 2 adet robot geliştirilmiştir. Robotlar, en kısa mesafeyi seçmeleri ve en az güç harcaması ile görevi tamamlamaları için eğri uydurma yöntemi kullanılmaktadır. Robot, alanı tarayıp yüklerin koordinatlarını bulduktan sonra kendisine en yakın yüke eğri uydurma yöntemi kullanılarak gitmektedir. Eğer yük taşıyabileceği kadar ağırlığa sahipse yükü matrisin dışına taşımaktadır.

Taşıyacağından fazla olan yük için ise karar algoritması kullanılarak diğer robottan yardım istemektedir. Giriş bölümünde bu konuda yapılan çalışmalar ve literatür taraması verilmektedir.

Çalışmanın ikinci bölümünde robotların gelişimi ve çoklu robot sistemleri incelenmektedir. Üçüncü bölümde çoklu robot uygulamasında kullanılan materyallere ve metotlara yer verilmektedir. Dördüncü bölümde ise deneysel çalışmalar ve son olarak beşinci bölümde sonuç ve değerlendirmeler yer almaktadır.

## BÖLÜM 2

### ROBOTİK SİSTEMLER

Robotların ve uygulama alanlarının gelişimi incelendiğinde, gittikçe insanlar için ve insanlarla beraber daha sık çalışmalarının kaçınılmaz olduğu görülür. Robotlar, sağlık hizmetlerinde, rehabilitasyon çalışmalarında ve terapilerde yön gösterecek, ofislerde asistanlık yapacak veya ev işleri ile ilgilenecek, bazen de bizi eğlendirecektir. Tüm uygulamalarda, robotların insanlarla etkileşimleri değişik seviyelerde olacak, insanların karmaşık davranışlarını anlamaya çalışmak zorunda kalacaklardır. Tasarımcılar, sosyal etkileşimler kurabilen robot geliştirmeye mecbur olacaklardır.

Mobil robotların gelişmesi insan-robot etkileşimine yeni bir boyut getirmiştir. Artık robotların insanlarla yaşaması sıradan hale gelmeye başlayacaktır. İnsanlarla aynı mekânı paylaşmaları gerektiğinde, işlerini yaparken onlara zarar vermemelidirler [12].

#### 2.1. ROBOTİK

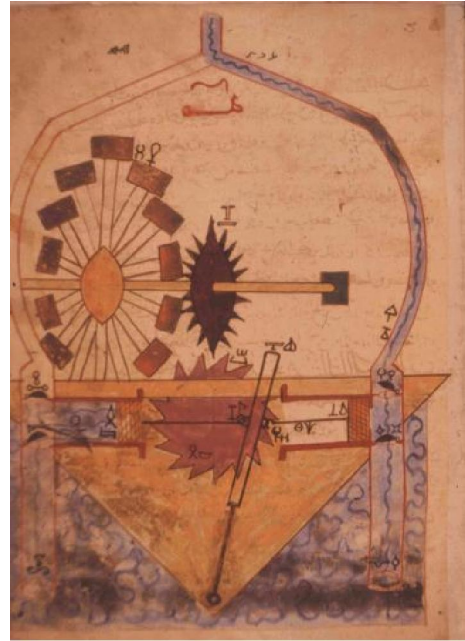
Robot, otonom veya önceden programlanmış görevleri yerine getirebilen elektro-mekanik bir cihazdır. Robot kelimesi, ilk olarak Karel Capek'in 1920 yılında yazdığı R.U.R. (Rossum's Universal Robots) adlı eserinde yer almış ve daha sonra tüm dünyada kullanılmaya başlanmıştır. Isaac Asimov, ünlü robot serisiyle teknolojik açıdan tutarlı bir robot kavramı yaratmış ve robotların amacının insana hizmet olduğunu, bir robotun kendi amaçlarını insanların amaçlarına hiçbir zaman tercih edemeyeceğini ortaya koyduğu 3 Robot Yasası'yla belirlemiştir. Bu robot yasalar, şu anda insanla robot arasındaki ahlaksal ve hukuksal ilişkinin temelini oluşturmaktadır. 3 Robot Yasası şu şekildedir; bir robot hiçbir şekilde insanoğluna zarar veremez veya pasif kalmak suretiyle zarar görmesine izin veremez, bir robot kendisine insanlar

tarafından verilen komutlara 1. kuralla çelişmediği sürece itaat etmek zorundadır ve bir robot 1. ve 2. kurallarla çelişmediği sürece kendi varlığını korumak zorundadır.

Robot olarak nitelendirebilecek ilk çalışma Ktesibius tarafından yapılmıştır. Ktesibius çağını aşan pek çok çalışmalar yapmış, sibernetik ve robotik biliminin kurucusu olmuştur. Ondan sonra bilinen en büyük sibernetik uzmanlarından biri El Cezeri'dir. El Cezeri Şekil 2.1 a ve b'de görüldüğü gibi çağının çok ilerisinde robotlar yapmıştır. Eski tarihlerde yaşamış olan meslektaşlarının icatlarını geliştirmiş ve kendine ait olan birçok şey yapmıştır. El Cezeri, otomatik makineler tarihinde çağın doruğuna erişmiş büyük mühendis İbni Razzaz Cesari adıyla anılmaktadır.



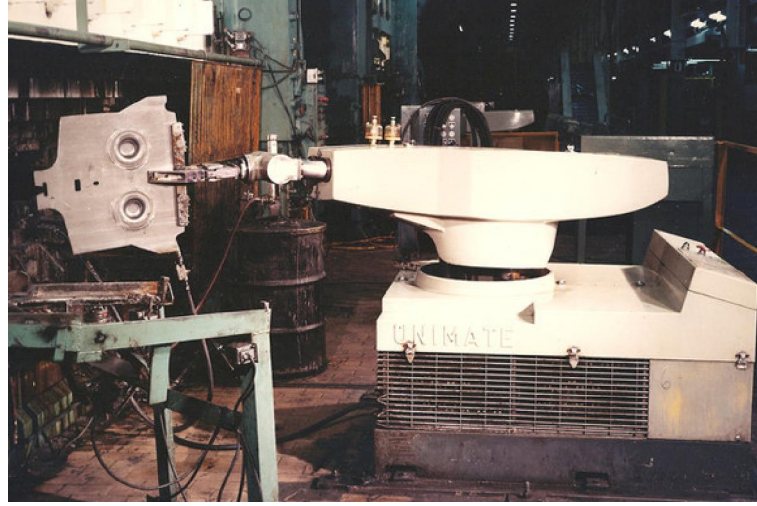
a) El-Cezeri 'nin filli saati



b) El-Cezeri'nin pistonlu su pompası

Şekil 2.1. El-Cezeri 'nin eserleri.

İlk elektromekanik robotlar, elektriğin yaygın bir şekilde kullanılmasıyla başlamıştır. Nikola TESLA, 1898 yılında radyo kontrollü teknesiyle ilk elektromekanik sistemi gerçekleştirmiştir. İlk endüstriyel robot ise 1954 yılında, George Devol tarafından Şekil 2.2'de görülen Unimate ismiyle endüstriyel bir kol olarak gerçekleştirilmiştir. İlk olarak General Motors'un Tenton'daki fabrikasında kullanılmaya başlayan Unimate sıcak metal parçaları ocaklardan çıkararak istifleme işini yapmaktadır.



Şekil 2.2. İlk endüstriyel robot Unimate.

### 2.1.1. Endüstride Robotik

Robotların karmaşıklığı arttıkça endüstrideki kullanım alanları da genişlemektedir. Endüstriyel robotiğin temel işlevi, tanımlanmış rutin görevlerin aynı şekilde ve sürekli tekrar edildiği üretim endüstrileri otomasyonudur. Otomotiv endüstrisi bu geniş ve karmaşık robotların görev aldığı en belirgin örnektir. Robotlar boyamada, sızdırmazlıkta, parçaların üretiminde ve birleştirilmesinde kullanılır (Şekil 2.3). Robotların bu görevlere uygun olmasının nedeni, sürecin kontrolü için çok az geribildirim yeterli olmasıdır. Endüstriyel robotlar çeşitli boyutlarda üretilebilirler ve bir insanın yapabileceğinden daha büyük işleri yapabilirler [13].



Şekil 2.3. Endüstride sıkça kullanılan robot kol.



Robot kolu, endüstriyel robotların en fonksiyonel parçasıdır. Son yıllarda, robot kollarının sabit istasyonlar şeklinde çalışmasının robot kolunun performansına ve kullanım alanlarına getirdiği kısıtlamaları fark eden araştırmacılar, bu sistemleri hareketli hale getirmek üzere çalışmalar yapmaktadır. Endüstriyel robotların gelişimi diyebileceğimiz bu sistemlere de mobil robotlar adı verilmektedir. Endüstriyel robot uygulamaları başlıca otomotiv, elektrik, elektronik ve mekanik olmak üzere endüstrinin hemen her alanında görülebilir [14].

### 2.1.2. Askeri Alanda Robotik

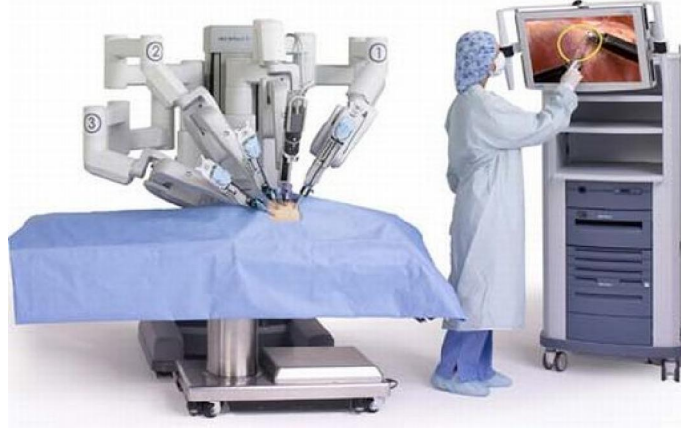
Robotlar yaygın olarak endüstride ve askeri alanlarda kullanılmaktadır. Askeri alanda kullanılan robotların başlıcaları mayın tarama ve insansız hava araçlarıdır. Askeri alanda kullanılan robotlar, insan kaybı yaşamadan operasyon bölgesine ulaşma ve bölgeden bilgi edinmeyi amaçlamaktadır. Şekil 2.4 de verilen Anka insansız hava aracı 30 Ocak 2015 tarihinde otomatik kalkış ve iniş modları test edildi. Anka'nın görevi uzaktan algılama ve keşiftir.



Şekil 2.4. TUSAŞ'ın yapmış olduğu insansız hava aracı Anka.

### 2.1.3. Tıp Alanında Robotik

Robotik cerrahinin dünyada kabul görmesi ve bu kadar hızlı ilerlemesinin altında yatan neden robotik cerrahi girişimlerinin yararlarının fazlalığıdır. En önemli yararı, ameliyat için kesinin az olmasıdır. Kesinin çok küçük olması hastalarda ağrı, kanama ve enfeksiyon oranlarının çok az olmasını sağlamaktadır.



Şekil 2.5. Da Vinci robotu.

Ayrıca, robot yardımlı laparoskopik cerrahi girişim sonrası hastaların rahat ve konfor düzeyinin iyi olması söz konusu olmakta, onların erken ayağa kalkıp, normal yaşamlarına kısa sürede dönmeleri gerçekleşmektedir. Kadınlar açısından beden imajında bütünlüğün bozulmaması, kozmetik açıdan üstünlük oluşturmaktadır [15]. Şekil 2.5 de tıp alanında küçük kesiklerle başarılı ameliyatlar yapan Da Vinci robotu verilmektedir.

## 2.2. ÇOKLU ROBOT SİSTEMLERİ

Çoklu robot sistemlerinin en önemli ortak özelliği sürü zekâsıdır. Sürü zekâsı kavramı ilk olarak 1989 yılında Gerardo Beni ve Jin Wang tarafından ortaya atılmıştır.

Sürü (toplu halde bulunma ve koordineli hareket etme) davranışını doğada bakteri ve kuşlardan balıklara kadar birçok canlıda gözlemlemek mümkündür. Araştırmalar, bu tür davranışın canlılara evrimsel avantajlar sağladığını ve bu sebepten dolayı evrimleştiğini göstermektedir. Örneğin, sürü halde dolaşan bireylerin yiyecek bulmakta ve yırtıcı hayvanlardan korunmakta daha etkili oldukları gözlenmiştir. Sistem kuramı açısından bakıldığında, sürü sistemlerin daha gürbüz, daha esnek, daha etkili ve ucuz olduğu söylenebilmektedir. Sürülerde bireyler ele alındığında birey zekâsının çok ilkel olduğu, fakat bu ilkel zekâlı bireylerden oluşan sürünün karmaşık işler yapabildiği görülmektedir.

Doğada yaşayan canlıların hareketlerinden esinlenerek geliştirilen biyolojik tabanlı algoritmaların, var olan problemlere çözüm üretmek için kullanılması sürü robotik sistem uygulamaları açısından son yıllarda oldukça önem kazanmaktadır. Sürü robotik sistemler konusu yeni olmasına rağmen hızla gelişen bir konudur. Bu tip sistemlerin faydalı yanları esnek, gürbüz, merkezi olmayan ve kendi kendine örgütlenme özellikleridir. Tek bir robot için fazla karmaşık olan bazı görevler sürü robotlarla kolayca başarılabilir. Bir kutunun itilmesi, tek bir robot için imkânsız bir görev olabilmekte ancak sürü robotlar eşgüdümlü çalışarak aynı kutuyu rahatlıkla itebilmektedir. Basit özelliklerle üretilen robotlar, karmaşık bir robota göre daha ucuz, daha sağlam, daha esnek, daha yüksek hata toleransına sahip ve daha güvenilir olmaktadır. Sürüyü oluşturan robotlardan bir veya birkaçı bozursa dahi sürünün geri kalanı görevi sürdürebilmektedir. Sürünün üyeleri arasındaki etkileşimin doğrudan değil de sürüyü çevreleyen ortam vasıtasıyla olması robotlar arasındaki iletişim yükünü de azaltabilmektedir. Sürü robotik sistemlerin sorunları olarak yerel bilgi kullanımından kaynaklanan yerel minimum problemi ve bu robotların çok basit olmaları sebebiyle özel görevler için nasıl programlanacakları sayılabilmektedir [16].

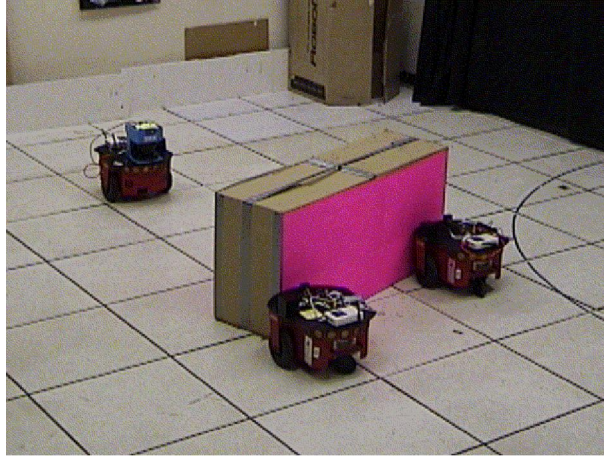
Yeni gelişen sürü robot çalışmaları teknolojik gelişme ve uygulamalar sayesinde literatürde geniş yer bulmaktadır. Çoklu robotlarda işbirliği kullanarak arama verimliliğinin artırılması, otonom robotlar için bir çoklu robot konuşlandırma tekniği, çoklu robot sistemlerinin ortak çalışması ve koordinasyonu için tümleşik planlama, görev arama ve yürütme mimarisi gibi çalışmalarla, çoklu robotlar için üzerinde uzmanlaşılması gereken konular geliştirilmektedir [17].



Şekil 2.6. Kilobots sürü robotları.

Çoklu robotlarda robotların bir birleriyle haberleşmesi ve ortak bir amaç için çalışması hayati önem arz etmektedir. Her ne kadar sürü robotlarda bir üye ile haberleşmenin kopması sürünün çalışmasını durdurmasa da olumsuz etkilemektedir. Bu durumu en aza indirmek ve kaliteli bir iletişim sağlamak, çoklu robotlarda üzerinde çalışılması gereken konuların başında gelmektedir. Şekil 2.6 'da verilen Kilobots sürü robotları büyük ölçekli özerk sürülerin toplu davranışlarını incelemek için tasarlanmaktadır.

Robotların bir birleriyle haberleşmesinden sonra ikinci öncelik olarak buldukları konumlardan haberdar olmaları ve bir birlerine göre olan konumlarını iyi hesaplayabilmeleridir. Konum bilgisinin doğru bir şekilde hesaplanabilmesi için robotların hareket mekanizmalarına takılan enkoder sistemleri, ivme metre gyro gibi sensörler aracılığıyla ve eğer dış ortamda çalışacaksa GPS (Global Positioning System) sistemleri ile konumlarını hesaplayabilirler. Şekil 2.7 de kutu itme görevini gerçekleştiren sürü robot uygulaması verilmiştir.



Şekil 2.7. Kutu itme görevi gerçekleştiren sürü robot uygulaması.

### 2.2.1. Çoklu Robot Sistemlerinde Haberleşme

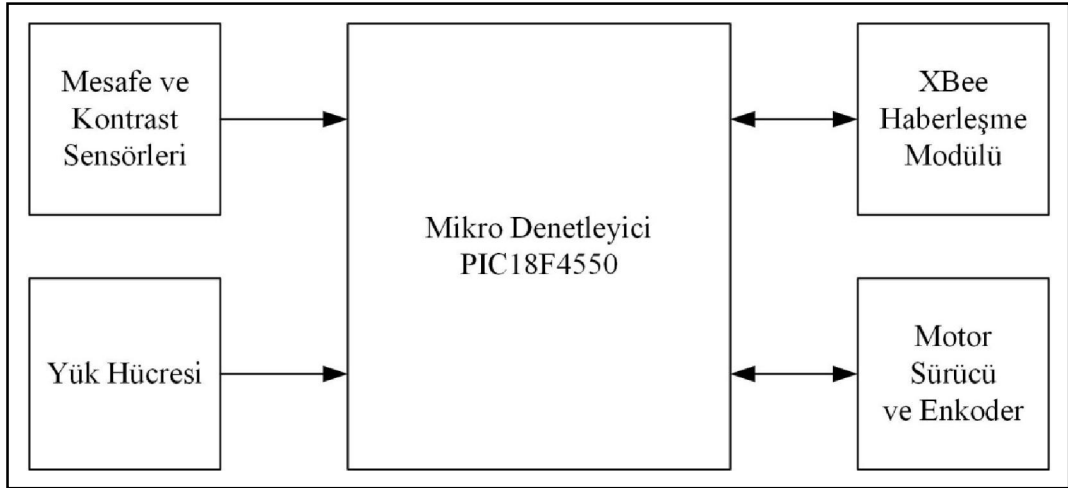
Çoklu robot sistemlerde haberleşme oldukça önemli bir konudur. Her bir üyenin çevreden almış olduğu verilerin diğer üyelere aktarabilmesi için, bu bilgilerin doğru bir şekilde tüm üyelere iletilmesi gerekmektedir. Haberleşmenin performansı, sistemin kararlı çalışmasında önemli bir etmendir.

Çok sayıda robotun birlikte işbirliği içerisinde çalıştığı bir sistemde merkezsiz bir denetimin ya da bütünsel bir bilginin (her erkinin diğer bütün erkinler hakkında bilgi sahibi olması) kullanılması, hem tasarlanması hem de gerçekleştirilmesi oldukça zor bir denetim yöntemidir. Düşük sayıda robot kullanıldığında, merkezsiz bir denetim uygulanabilmekte fakat robot sayısı artırıldığında, merkezsiz bir denetimin uygulanması oldukça zorlaşmaktadır. Bu sebeplerle, son yıllarda çoklu robot sistemlerinde merkezcil olmayan denetim yöntemleri tercih edilmeye başlanmıştır. Bu yöntemlerde, robotlar arası iletişim ve ağ yapısı önemli bir yere sahiptir. Başka bir deyişle, pratik, verimli çoklu robot uygulamaları yapılabilmesi için verimli denetim teknikleri dışında ölçeklenebilir, verimli ve gürbüz bir haberleşmenin yanı sıra gerekli ağ protokol ve teknolojilerinin de geliştirilmesi de önemli bir gereksinimdir [18].

### BÖLÜM 3

#### MOBİL ROBOTLARIN MEKANİK VE ELEKTRONİK TASARIMIN GERÇEKLEŞTİRİLMESİ

Eşyaların sürekli bir yerden bir yere taşınması söz konusu olduğunda, bu işi başarılı bir şekilde ve işçi sağlığını dikkate alarak gerçekleştirme ihtiyacı doğmaktadır. Robotlar verilen görevleri çok hassas yapabilme kabiliyetlerine sahiptirler. Bu nedenle eşya taşıma görevinde robotlar kullanılabilir. Bu tez çalışmasında yük taşıma görevini otonom mobil robotlar ile gerçekleştirilmektedir.

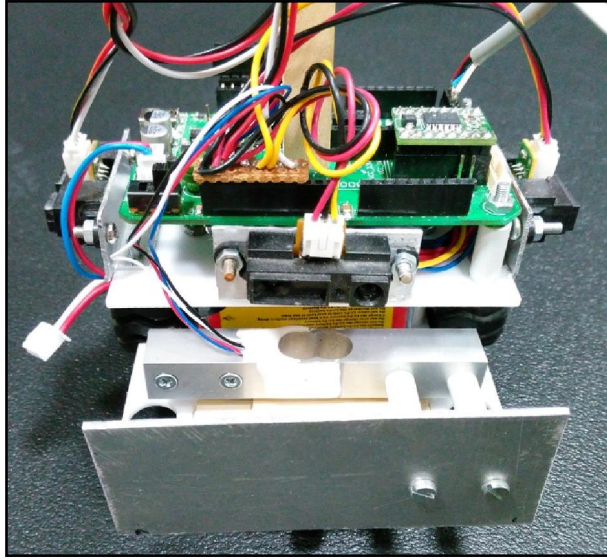


Şekil 3.1. Sistemin blok şeması.

Bu çalışmada 2 üyeden oluşan sürü robot uygulaması gerçekleştirilmiştir. Üyeler birbirleriyle XBee modüller aracılığıyla haberleşmektedirler. Önünde, sağında ve solunda olmak üzere toplam 3 tane cisim algılama sensörleri mevcuttur. Şekil 3.1 'de gerçekleştirilen robotların blok şeması verilmektedir. Çalışmada Microchip firmasının PIC18F4550 mikro denetleyicisi kullanılarak tasarlanan ANKAKIT V2.2 geliştirme kiti kullanılmaktadır. 6 adet kontrast algılayıcısı ile zemindeki çizgileri takip edebilecek bir sisteme sahiptir. Robotun itme noktasında, maksimum 5 kg lık

bir ağırlığı ölçebilecek yük hücresi bulunmaktadır. Bu yük hücresiyle itilen yük'e uygulanan kuvvetin ölçülmesi hedeflenmiştir. Robotların mekanik aksamı 2.5 mm alüminyum ve 3 mm pleksiglass malzemeden tasarlanmaktadır.

7.4 Volt giriş gerilimi olan robot motorlar çalışmıyorken 195 mA akım çekmektedir. Robot hareket halindeyken 360 mA akım çekmektedir. Robot 110x105x225 mm ölçülerinde ve 350 gram ağırlığındadır. 3 adet (sağında, solunda ve önünde olmak üzere) cisim algılama sensörü bulunmaktadır. Zemindeki çizgileri algılayabilmek için 6 adet CNY70 kontrast algılayıcısı kullanılmaktadır. Şekil 3.2 'de tasarlanan Öncü robotunun fotoğrafı verilmektedir.



Şekil 3.2. Tasarlanan Öncü robotu.

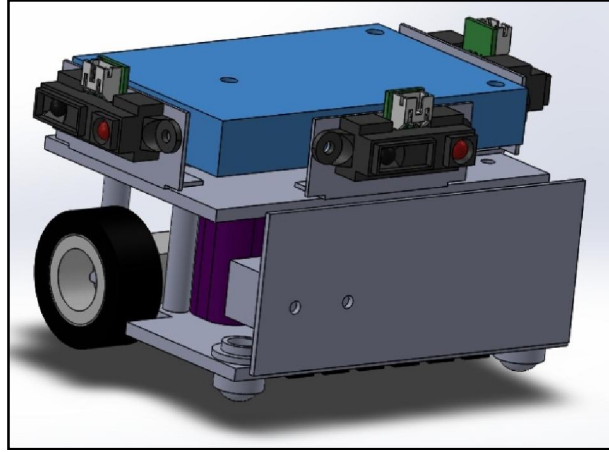
Robot tasarımı; mekanik donanım, elektronik tasarım ve çoklu robotların yazılımı olmak üzere 3 temel başlıkta toplanmaktadır. Bu bölümde mekanik donanım ve elektronik tasarım konuları açıklanacaktır.

### 3.1. MEKANİK DONANIM

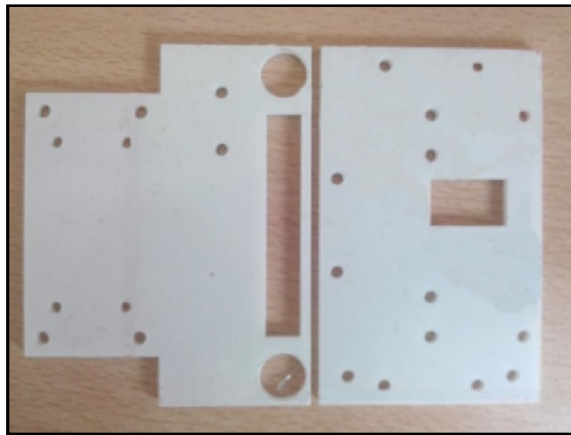
Robotun mekanik donanımı Solid Works programıyla 3 boyutlu olarak tasarlanmış olup pleksiglass malzemeden lazer kesim ile kestirilmiştir. Şekil 3.3'de tasarlanan çoklu robotun Solid Works çizimi ve Şekil 3.4 'de ise lazerde kestirilen pleksiglass

malzemeler verilmektedir. Motorları tutturmak için ve ikinci kattaki cisim sensörünü ve mikro denetleyici kartını tutan parçayı sabitlemek için delikler ayarlanmıştır. Robotun ön kısmında tam ortaya bir adet sarhoş tekerlek konularak sorun çözülmüştür. Alt şase üzerinde CNY70 kontrast sensör dizisinin konulması için bir dikdörtgen boşluk açılmıştır. Ayrıca kutuları itmek için hazırlanan sistemi tutacak L profil için de 2 adet delik ayarlanmıştır.

İkinci kattaki pleksiglass parçanın üzerinde 3 adet cisim sensörü için 6 adet delik, alt şaseyle sabitlenmesi için 4 delik ve mikro denetleyici kartını sabitlemek için 3 delik mevcuttur (Şekil 3.4). Yük hücresi alüminyumdan 13x80x13 mm boyutlarında seçilmiştir. Vidalar ile 37x90 mm lik alüminyum levhaya sabitlenerek cisimleri ya da yardım edeceği robotu itmesi amaçlanmıştır (Şekil 3.5).



Şekil 3.3. Tasarlanan sürü robot'un Solid Works çizimi.

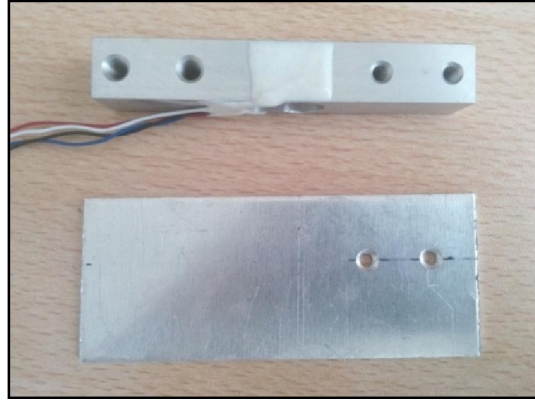


Şekil 3.4. Lazerde kesilen pleksiglass malzemeler.



Yük hücrelerini ana şaseye tutturmak için L profilden bir parça ayarlanmıştır ve yük hücrelerine iki vida ile sabitlenmiştir. Cisim sensörlerini montaj etmek için alüminyum L profilden faydalanılmış ve uygun ölçülerde kesilmiştir.

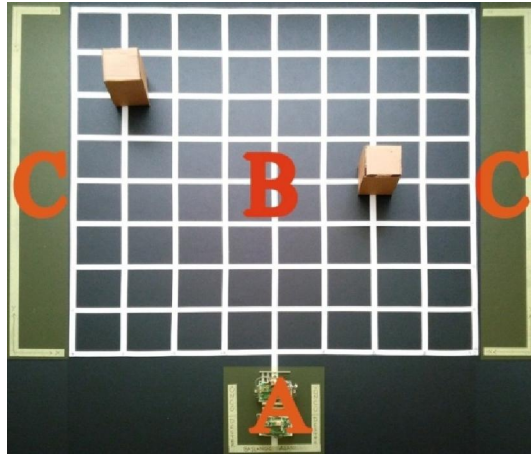
Yük hücreleri 80x13x13 mm ölçülerindedir. 30 gram ağırlığa sahip olan yük hücreleri maksimum 5 kg ağırlığı 1 gram hassasiyetle ölçebilmektedir. Yük hücreleri kazancı 100 olan bir enstrümantasyon yükselteci ile yükseltilerek kullanılmaktadır.



Şekil 3.5. Yük hücreleri ve cisim itme aparatı.

### 3.1.1. Deney Düzenegi

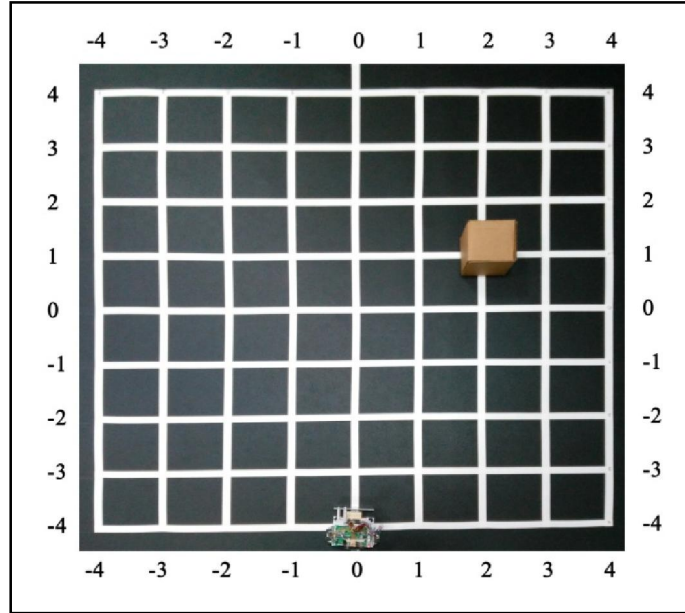
Şekil 3.6 da verilen deney alanı bir buçuk metre karelik siyah bir alan üzerine 18 mm kalınlığında beyaz elektrik bandı ile 150 mm aralıklarla 9 adet yatay ve dikey çizgiden oluşmaktadır.



Şekil 3.6. Deney alanı.

Yapılacak olan 2 adet robot “A” bölgesinden deney alanına (“B”) girdikten sonra yüklerin koordinatlarını bulacak ve yükleri “C” bölgesine taşıyacaktır.

Öncü robotu A noktasından bulunduğu istikamette dikey çizgiler bitene kadar ilerlemekte ve sağındaki ve solundaki yükleri algılayıp kaydetmektedir. Tarama işlemi bittikten sonra geri dönerek yüklerin bulunduğu koordinatlara gitmekte ve ağır olan hariç yükleri B bölgesine taşımaktadır. Ağır olan kutuyu yük hücresiyle algıladıktan sonra Destek robot’a bulunduğu koordinatı yollayarak yardım istemekte ve yardımın gelmesini beklemektedir. Yardımın geldiğini haberleşme yoluyla algıladıktan sonra Destek robotuyla birlikte kutuyu B bölgesine itmekteler. Tüm kutular temizlendikten sonra robotlar tekrar başlangıç noktasına gitmektedirler. Şekil 3.7 de deney alanının koordinatsal gösterimi verilmiştir.



Şekil 3.7. Deney düzeneğinin koordinatsal gösterimi.

### 3.1.2. Motorlar

Şekil 3.8 de verilen, 250:1 düşürme oranına sahip, 125 RPM (revolutions per minute), 6.45 kg/cm torka sahip iki DC motor kullanılmıştır. Motorun çalışma gerilimi 6 Volt, boşta çektiği akım 90 mA'dir. Motor'un kilitlenme akımı 2000mA'dir. Motorlarda dahili 90<sup>0</sup> faz farklı iki çıkışı olan enkoder yapısı

bulunmaktadır. Dişli kutusu açık olduğu için dış etmendeki olumsuz koşullardan etkilenebildiği için dişli kutusunu kapatacak şekilde tasarlanmış motor tutturucusu kullanılmıştır.



Şekil 3.8. Motor, motor tutucu ve sensör tutucular.

Bu motor'un tercih edilme sebebi enkoder sistemine sahip olması ve küçük boyutlarda olmasıdır. Motorların iç piyasadan temin edilebilmesi bir başka tercih sebebidir.

### 3.1.3. Tekerlekler

Robotlarda Pololu şirketinin üretmiş olduğu 42x19mm ölçülerinde Şekil 3.9 da görülen kauçuk tekerlek kullanılmaktadır. Tekerleğin yere geniş bir şekilde tutuyor olması kaymayı engellediği için enkoder ile adım sayma konusundaki hataların en aza indirgenmesi hedeflenmektedir. Tekerleğin bir başka özelliği ise D şaft denilen teknolojinin kullanılmasıdır. Çalışmada kullanılan motorun mili D şaft şeklinde olduğu için tekerleği motora vida ile sıkıştırma gibi bir zorluk ortadan kalkmıştır. Kolay bir şekilde motora takılmakta ve kullanım esnasında asla çıkmamaktadır.



Şekil 3.9. POLOLU 42x19 mm tekerlek.

## **3.2. ELEKTRONİK DEVRE TASARIM**

Robotun elektronik devre tasarımı daha önceden belirlenen robot kabiliyetlerine göre tasarlanmaktadır. Robot'un taşıyacağı yük'ün ağırlığı için yük hücresi yükselteci, çevresindeki yükleri algılayabilmesi için analog mesafe sensörü, yerdeki çizgileri algılayabilmek için kontrast sensörü, motorların PWM (Pulse Width Modulation) ile sürülebilmesi için motor sürücü, merkezle ve diğer sürü üyeleriyle iletişimde olabilmesi için kablosuz iletişim devresi ve çizgi takip etmeden ilerlediği durumlar için kullanılacak enkoder ile sürüş devresi eklenmektedir.

### **3.2.1. Mikro Denetleyici**

Microchip firmasının 8 bitlik işlemci mimarisine ait PIC18F4550 mikro denetleyicisinin özellikleri şu şekildedir;

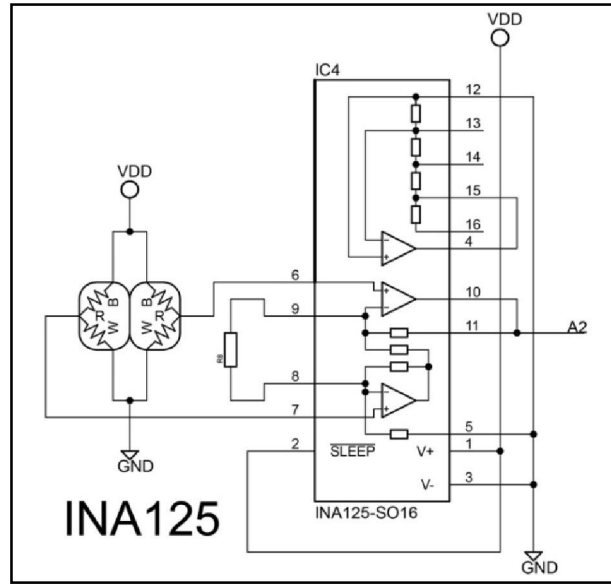
- 32 Kbyte flash hafıza
- 35 giriş/çıkış pini
- 2 Kbyte RAM
- 256 byte EEPROM
- Düşük hızda 1.5 Mb/s, yüksek hızda 12 Mb/s işlem hızı
- 8 MHz iç osilatörü ve 20MHz harici osilatörü
- 3 adet zamanlayıcı (Timer)
- 10 bit çözünürlüğe sahip 13 adet ADC (Analog Digital Converter)

PIC18F4550 mikro denetleyicisinin iki adet harici zamanlayıcı kesme girişi olduğu için ve 12 Mb/s işlem hızı olduğu için tercih edilmiştir. Harici kesme girişleri encoder bilgilerini okumak için kullanılmaktadır.

### **3.2.2. INA125 Enstrümantasyon Yükselteci ve Yük Hücresi**

Giriş empedansı 410 ohm olan yük hücresi 5-10 Voltta çalışmaktadır. Maksimum 5 kg lık bir kuvveti ölçebilen sensör alüminyumdan yapılmış ve 2 si sağında 2 si solunda olmak üzere 4 bağlantı noktasından oluşmaktadır. Şekil 3.10 da yük

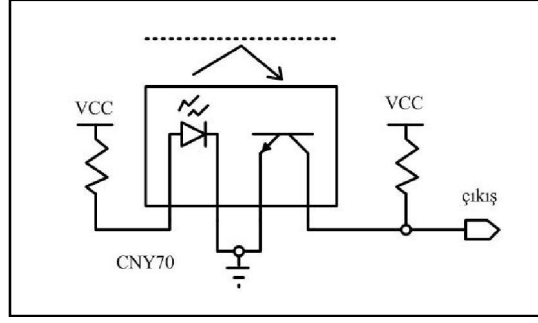
hücresinin elektronik yükselteç bağlantı şeması verilmiştir. Yük hücresi sabit noktası yere, ölçüm noktası havaya gelecek şekilde montaj edildiğinde ağırlık verisini vermektedir. Robota bağlanan yük hücresi yere göre dik konumda olduğu için ağırlık bilgisi vermeyecektir. Okunan analog değerler, itilmek istenen ve belli bir ağırlığa sahip olan yüklere uygulanan kuvvettir. Bu kuvveti ölçebilmek için yük hücresindeki gerilim değişimlerini yükseltmek gerekir. INA125 enstrümantasyon yükselteci fark yükseltecidir. Fark yükselteçleri yük hücresinin ölçüm noktaları arasındaki farkı yükseltmek için kullanılmaktadır.



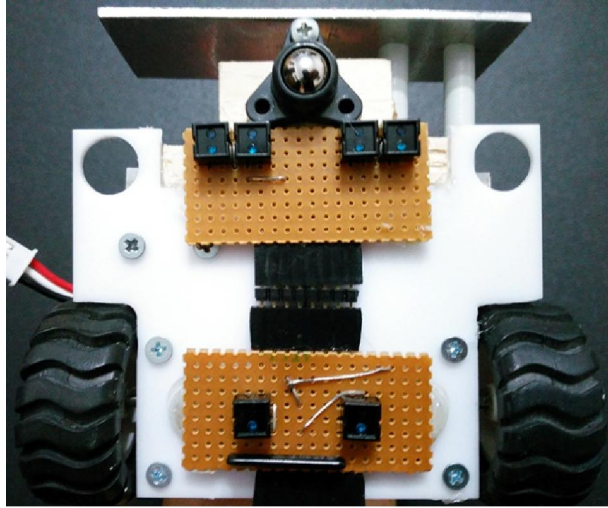
Şekil 3.10. INA125 yük hücresi yükselteci.

### 3.2.3. Kontrast Sensörü

Robotun pist üzerinde konumu hesaplamak ve pistin sınırlarını algılamak için 6 adet CNY70 kontrast sensörü kullanılmıştır. Şekil 3.12 de verilen kontrast algılayıcıları robotun çizgi üzerinde düz durabilmesi için 4'ü önde 2'si arkada olacak şekilde tasarlanmaktadır. Algılayıcının infrared diyot'u üzerinden 20 mA lik akım geçirildiğinde maksimum 0.7 mm uzaklıktaki beyaz renkli cisimden yansarak opto transistor'ü ilettime geçirebilmektedir. Siyah renkli cisimlerden yansıma çok az olduğu için transistör ilettime geçmemektedir. Şekil 3.11 de Kontrast algılayıcısının elektronik bağlantı şeması verilmektedir.



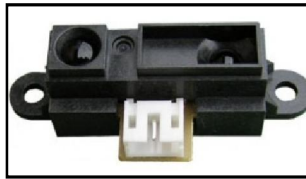
Şekil 3.11. CNY70 elektronik bağlantı şeması.



Şekil 3.12. Kontrast sensörlerinin yerleşimi.

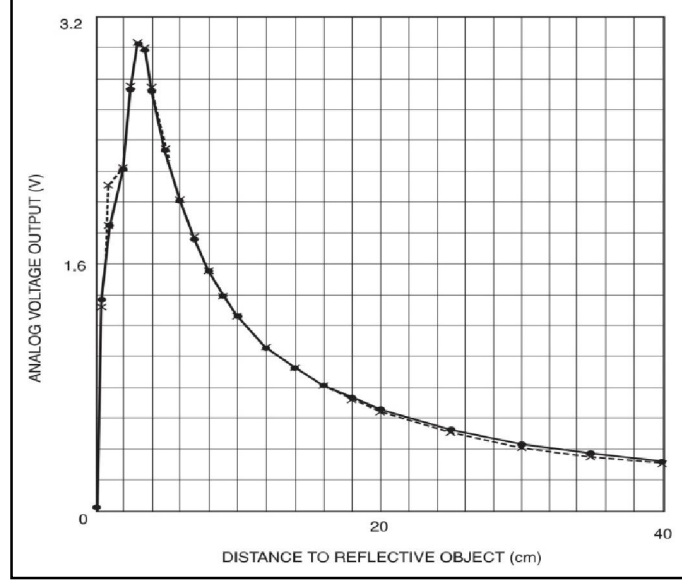
### 3.2.4. Mesafe Sensörü

Şekil 3.13 de verilen GP2D120X analog mesafe sensörünün çıkış karakteristiği Şekil 3.14 deki gibidir. Sensör kendisine 4 cm uzaklığa kadar bir gerilim yükselişi gösterdikten sonra maksimum çıkış gerilimine ulaşmakta ve cisim sensörden uzaklaştıkça gerilim düşmeye başlamaktadır. Bu düşüş lineer olmadığı için uzak mesafelerde küçük gerilim değişimleri yakın mesafelere göre daha çok uzaklığa tekabül etmektedir.



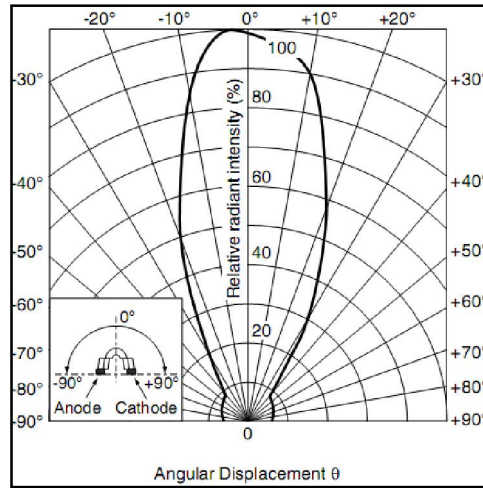
Şekil 3.13. GP2D120 4-30 cm mesafe sensörü.

Sensörün 2 cm'deki gerilim çıkışı değeri ile 5 cm'deki gerilim çıkış değeri aynı olduğu için, 4 cm' ye kadar olan mesafe sensörün kör noktası olarak tanımlanmaktadır. Sensör her açıldığında 44 ms'lik hazırlanma süresi vardır. Tepki süresi ise 39 ms'dir. Çalışırken ortalama 33 mA akım çekmektedir.



Şekil 3.14. GP2D120X sensörünün çıkış karakteristiği.

Robot üzerinde sağ, sol ve ön olmak üzere 3 adet cisim sensörü bulunmaktadır. Robot pist üzerinde hareket ederken aynı anda üç yöne bakarak cisimleri çok hızlı bir şekilde bulabilmesi için bu şekilde tasarlanmıştır.



Şekil 3.15. Sensör'ün optik yayılımı.

Yükleri algılamada noktasal ölçüm yapamayan sensörün optik ışımalarının genişliği nedeniyle okuma hataları oluşmaktadır. Sensörün optik ışması Şekil 3.15 de verilmektedir.

### 3.2.5. Batarya

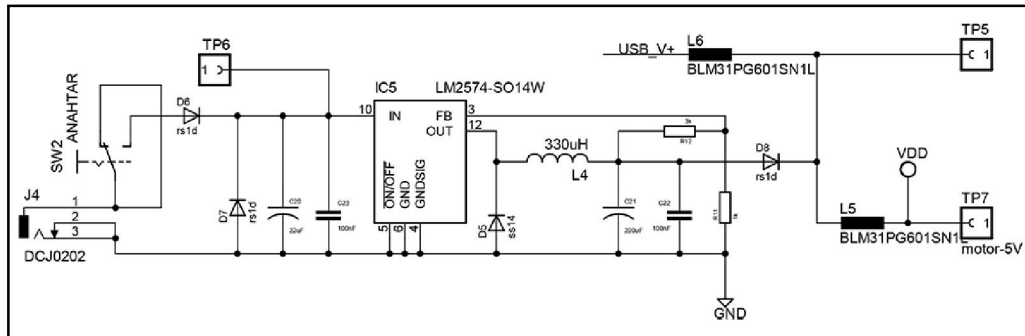
Robotun çalışması için gerekli olan enerjiyi çok yüksek akımlar sağlayabilen ve küçük boyutlara sahip olan lithium polymer batarya kullanılmıştır. Her bir hücresi 3.7V gerilime sahip olan Li-Po hücrelerinin seri olarak bağlanmasıyla oluşan 2 hücreli 7.4 Volt 900 mAh lik bir batarya tercih edilmiştir. Bu çalışmada kullanılan 2 hücreli li-po batarya Şekil 3.16 da verilmektedir.



Şekil 3.16. 7.4 V Li-Po batarya.

### 3.2.6. Regülatör Katı

LM2574 anahtarlama modlu (switch mode) regülatör kullanılmaktadır. Kullanılan regülatör devresi Şekil 3.17 de görülmektedir.



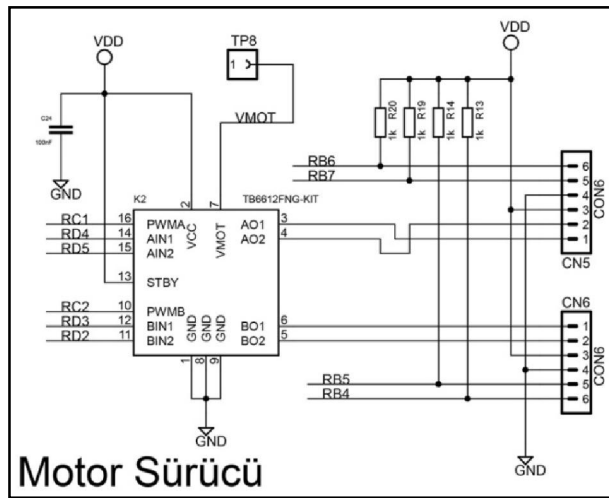
Şekil 3.17. Regülatör katı.



5 Volt çıkış için 7-40 Volt arasında bir giriş gerilime ihtiyaç duyan regülatör, çıkış akımı olarak 500 mA vermektedir. 52 kHz 'de anahtarlama yaparak çıkış gerilimini sağlayan regülatör yüksek anahtarlama kabiliyeti sayesinde düşük dalgalanmalara neden olmaktadır. Bu durum da mikro denetleyicinin sağlıklı bir şekilde çalışmasını sağlamaktadır.

### 3.2.7. Motor Sürücü

TB6612FNG motor sürücü 2 adet motor sürme kapasitesine sahiptir ve Şekil 3.18 de görülmektedir. DGM (Darbe Genişlik Modülasyonu) ve yön pinleri mevcut olan motor sürücü 5 ila 13 Volt aralığında motor beslemesine sahiptir.



Şekil 3.18. TB6612FNG motor sürücü kartı.

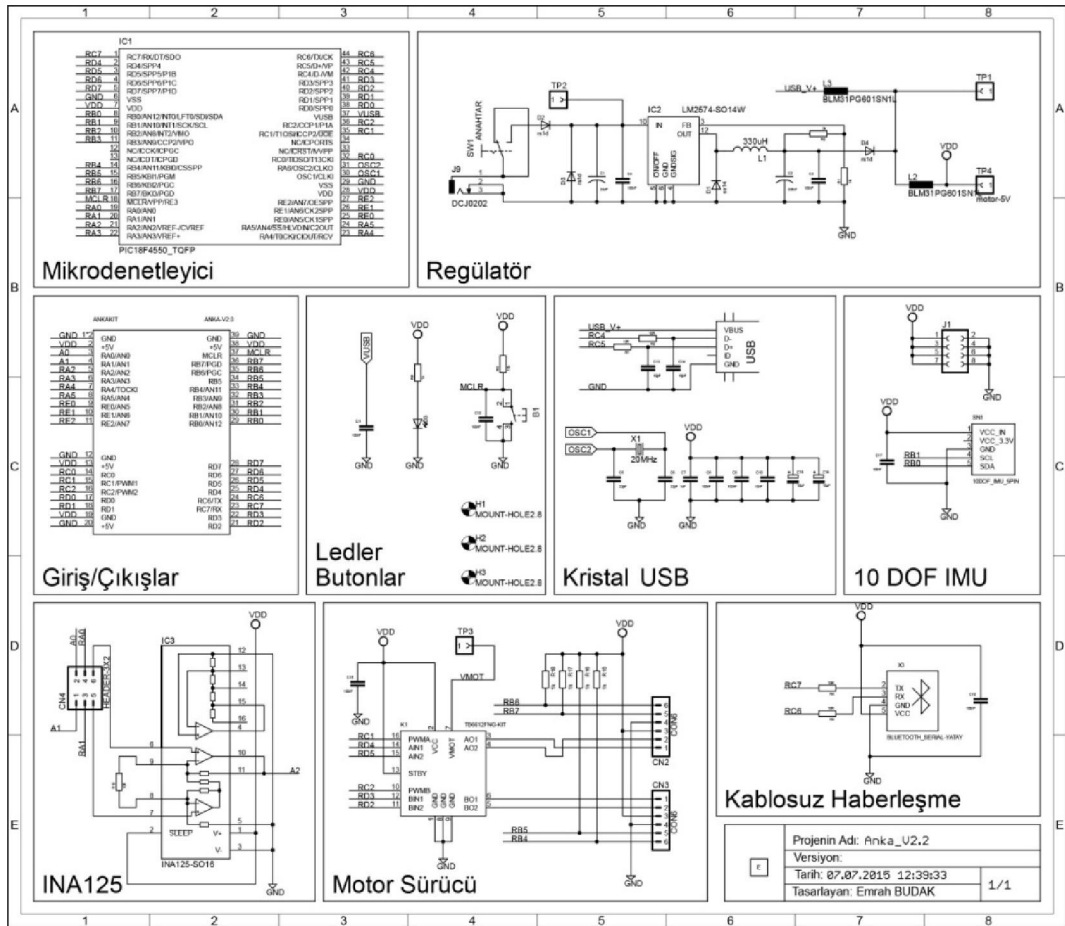
100 kHz'e kadar DGM sinyal girişini destekleyen motor sürücü 1 amperlik çıkış akımı sayesinde seçilmiş motorlar için uygundur. Motor sürücü içerisinde aşırı akım koruması olduğu için, verebildiği güçten fazlası çekildiğinde motor sürücüyü kapatarak kendisini korumaktadır.

### 3.2.8. Robot Denetim Kartı

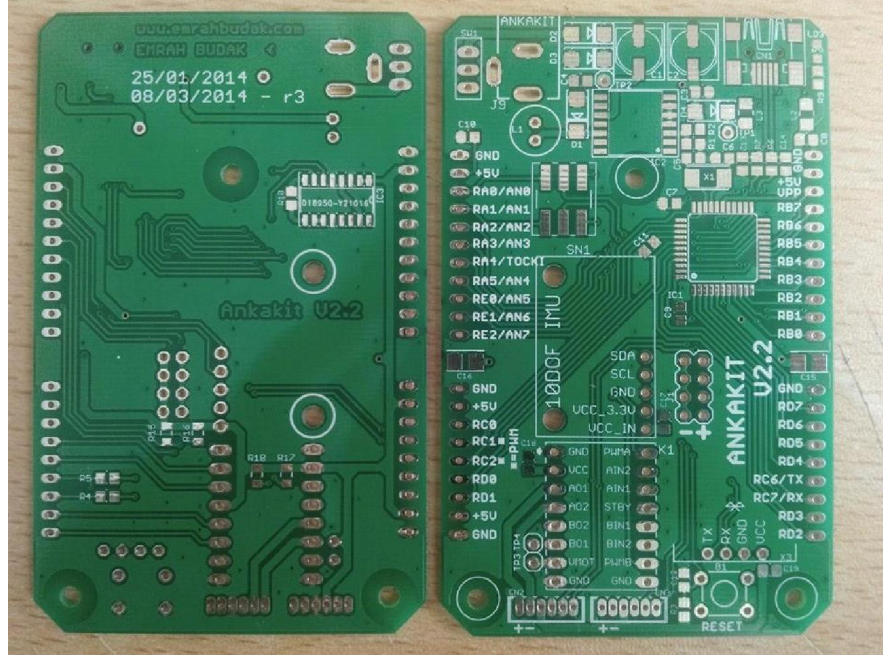
87x55 mm boyutlarında 1.6 mm kalınlıkta FR4 malzemesinden üretilen ANKAKIT V2.2 geliştirme kartı kullanılmaktadır. Kart ihtiyaç dahilindeki elemanlardan oluşturularak tasarlanmıştır. Şekil 3.19 da açık şeması, Şekil 3.20'de PCB'nin

fotoğrafi verilmektedir. Devre kartı üzerinde tasarlanan motor sürücü katı, seri haberleşme modülü girişleri, regülatör katı, analog okuma katı ve kontrast sensör giriş pinleri bulunmaktadır.

Motor beslemesini sabitlemek amacıyla 3Amper akım sağlayabilen 5 Voltluk bit regülatör haricen tasarlanmaktadır. XBee haberleşme modülü motor sürücünün oluşturduğu manyetik alandan etkilendiği için bir kablo yardımıyla motor sürücüdün uzaklaştırılmaktadır. Kart üzerinde besleme gerilimi ve regülatör çıkış gerilimlerinin ölçülebilmesi için pinler çıkartılmaktadır. Elektronik malzemeler yüzey montaj devre elemanı olarak seçilmektedir. Giriş/çıkış bağlantıları dışı sıralı pin ile verilmektedir.



Şekil 3.19. ANKAKIT V2.2 geliştirme kartı.



Şekil 3.20. ANKAKIT V2.2 PCB'si alttan ve üstten görünüş.

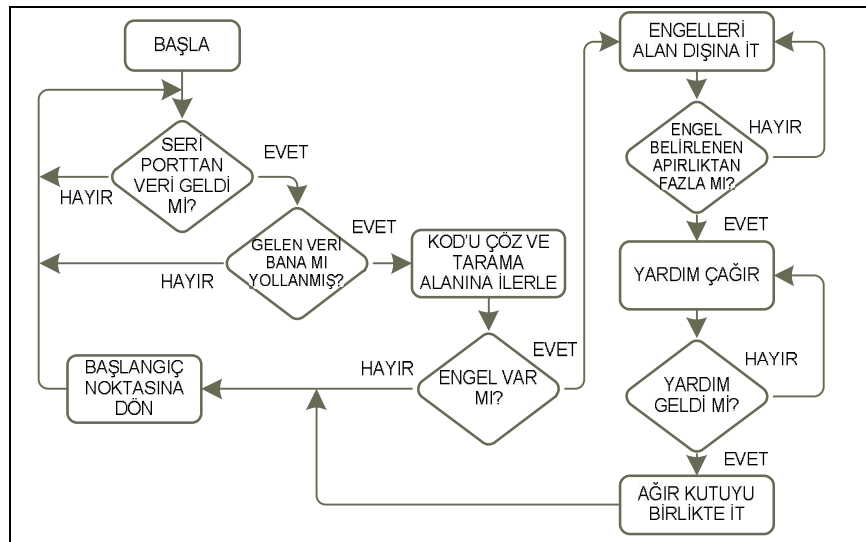
## BÖLÜM 4

### YAZILIM ALGORİTMALARI

Robotların yapacakları görevler göre ikiye ayrılmıştır. Çalışma alanındaki engellerin tamamını bulup hafif olanları alan dışına itecek ve ağır olanda ise yardım isteyecek olan robota Öncü, Öncü'nün yardım çağrısına cevap verip yardıma gidecek olan robota ise Destek ismi verilmiştir. Bu iki robotun yazılımları bir birlerinden farklı olarak kodlanmaktadır. Çizgi izleme, otomatik açılı dönüşler, çizgi sayma algoritmaları aynı olan bu iki robotun yapacakları görevler bakımından merkez algoritmaları farklıdır.

#### 4.1. ÖNCÜ ROBOTUNUN ALGORİTMASI

Öncü robotu bilgisayar yardımı ile almış olduğu bilgi ile alanı taramaya başlamaktadır. Alan taranırken kutuların yerleri belirlenmekte ve hafızada tutulmaktadır. Öncü robot için geliştirilen algoritma Şekil 4.1 'de verilmektedir.

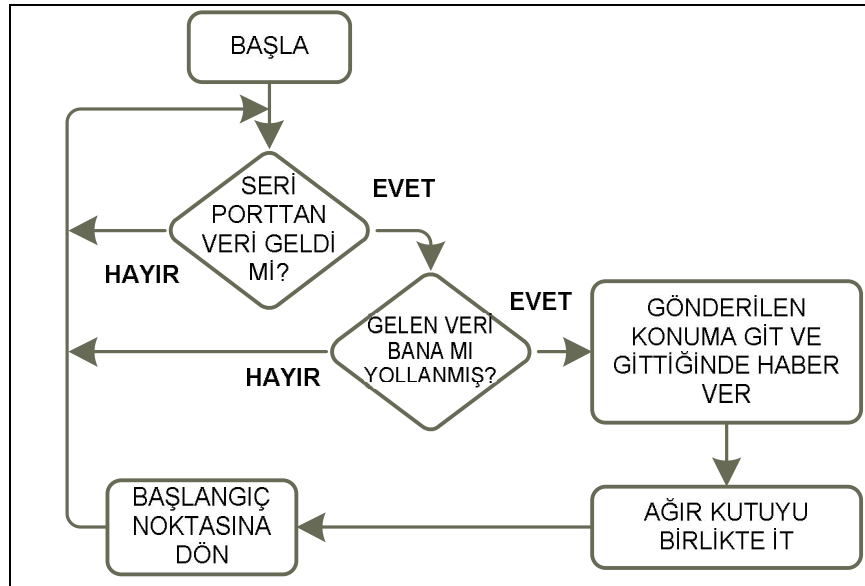


Şekil 4.1. Öncü robotunun algoritması.

Alan taraması bittikten sonra Öncü robotu geri dönerek kutuları alan dışına itmeye çalışmaktadır. Eğer kutunun ağırlığı itemeyeceği kadar ise, Destek robot çağırılmakta ve kutunun bulunduğu yere ulaşması beklenmektedir. Destek robot Öncü robot ile haberleştikten sonra birlikte kutuyu itmekte ve başlangıç noktasına geri dönmektedir.

#### 4.2. DESTEK ROBOTUNUN ALGORİTMASI

Destek robotu, Öncü robotu tarafından yardıma çağırılana kadar alanın dışında beklemektedir. Öncü robotu ağır olan kutuyu itmeye çalıştıktan sonra duracak ve Destek robotu çağıracaktır. Çağırırken göndermiş olduğu kodda Destek robotun hangi koordinata geleceği konusunda bir bilgede yollamaktadır. Destek robotu yardım çağırısını aldıktan sonra çağırıldığı koordinata gidip oraya ulaştığı bilgisini yolladıktan sonra Öncü robot ile birlikte kutuyu kutunun alan dışına itildiği bilgisini alana kadar itmektedir. Kutu alan dışına atıldıktan sonra sırayla başlangıç noktasına dönmektedir. Destek robotun algoritması ise Şekil 4.2 de verilmektedir.

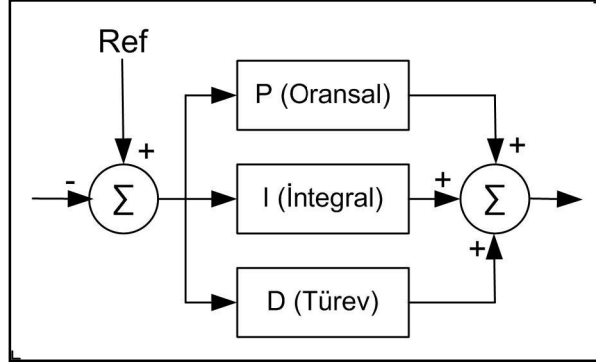


Şekil 4.2. Destek robotunun algoritması.

#### 4.3. PID DENETLEYİCİSİ

Robotlar çizgi izleme ve enkoder ile yol alma fonksiyonlarında PID denetleyicisini kullanmaktadır. Integral, oransal ve türevden oluşan PID algoritmasında, integral

geçmişte sistemde oluşan hatayı, oransal sistemde anlık oluşan hatayı ve türev ise gelecekte sistemde oluşacak hatayı düzeltmeyi hedeflemektedir. PID kontrolörler; otomotiv, uçuş kontrolü, motor sürücüler, manyetik ve optik hafızalar gibi birçok farklı alanda uygulanmaktadır [19]. Şekil 4.3 de temel PID denetleyicisinin yapısı verilmektedir.



Şekil 4.3. PID denetleyicisinin temel yapısı.

Yazılımsal olarak oluşturulan PID bloğu enkoder ile ilerleme ve kontrast sensörleriyle çizgi izleme fonksiyonlarında çağırılır. Enkoder ve çizgi izleme PID katsayıları her iki fonksiyon için ayrı ayrı tanımlanmıştır.

PID algoritmasının matematiksel ifadesi Eşitlik 4.1 de verilmektedir.  $K_p$  = Oransal kazanç katsayısını,  $K_i$  = İntegral kazanç katsayısı ve  $K_d$  = Türev kazanç katsayısı ifade etmektedir.  $e(t)$  PID denetimine girilen hata değeridir.

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{d}{dt} e(t) \quad (4.1)$$

#### 4.3.1. Oransal Terim

Oransal terim sistemden alınan hata değerinin  $K_p$  katsayısı ile çarpılmasıyla elde edilmektedir. Sistem sadece oransal denetim ile gerçekleştirildiğinde çıkışta referans değer üstünde ve altında seyreden bir çıkış izlenmektedir.

$$P_o = K_p e(t) \quad (4.2)$$

$$P_o = K_p * \text{hata} \quad (4.3)$$

### 4.3.2. İntegral Terimi

İntegral, eğrinin altında kalan alanı bulmak için kullanılan bir yöntemdir. Hata değeri referans değerimizden uzaklaştıkça integral değeri katlanarak artacak ve ters etkiyle sistemi referans değere çekmeye çalışacaktır. İntegral teriminin ifadesi eşitlik 4'de verilmiştir. İntegral teriminde bulunan  $K_i$  katsayısı, integralin sistem çıkışına yapacağı etkinin büyüklüğünü girmek için kullanılır. 4.4 deki eşitlikte integral teriminin mikro denetleyicide kullanılan kod karşılığı verilmiştir. dt PID denetiminin yenilenme süresidir. Bu süre enkoderla ilerleme ayarı için 40 ms, çizgi izleme ayarı için ise 20 ms dir.

$$I_o = K_i \int_0^t e(\tau) d\tau \quad (4.4)$$

$$I_o = I_o + (K_i * hata * dt) \quad (4.5)$$

### 4.3.3. Türev Terimi

Türev terimi, hatadaki değişimin açısını hesaplamaktadır. Yeni hatanın bir önceki çevrimdeki hataya göre pozitiflik/negatiflik ve dar/geniş açı olmasına göre farklı tepkiler vermektedir. Örneğin hata değeri referans değerini aştıktan sonra tekrar referansa dönüş yapacaktır, dönüş açısı genişse hızlı bir toparlanma, dar ise yavaş bir toparlanma gerçekleştirmektedir. Toparlanmanın durumuna göre sistemin çıkışına değer göndermektedir. Eşitlik 4.6 da türev teriminin matematiksel ifadesi verilmektedir. Eşitlik 4.7 'de verilen ehata bir önceki hata değeridir.

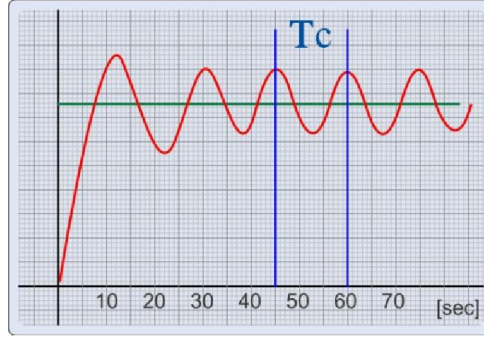
$$D_o = K_d \frac{d}{dt} e(t) \quad (4.6)$$

$$D_o = K_d * (hata - ehata) * dt \quad (4.7)$$

### 4.3.4. PID Katsayılarını Hesaplama Yöntemi

PID katsayılarının belirlenmesinde Ziegler Nichols'ün 2. metodu kullanılmıştır. Ziegler Nichols 'ün 2. Metoduna göre öncelikle oransal katsayı deneme yoluyla Şekil 4.4 de olduğu gibi frekansı sabit bir salınım yapacak şekilde bulunur ( $K_c$ ) ve

çıkıştaki salınım'ın periyodu ( $T_c$ ) ölçülür. Bu değerler elde edildiğinde Ziegler Nichols'un katsayıları ile çarpılarak PID katsayıları hesaplanır. Ziegler Nichols'un 2. Metodu Çizelge 4.1 de verilmiştir.



Şekil 4.4. Oransal denetleyici çıkışı.

Çizelge 4.1. Ziegler Nichols'un 2. metodu katsayıları.

Denetleyici Tipi	$K_p$	$K_i$	$K_d$
P	$0.5 * K_c$	0	0
PI	$0.4 * K_c$	$0.8 * T_c$	0
PID	$0.6 * K_c$	$0.5 * T_c$	$0.125 * T_c$

#### 4.4. PID DENETLEYİCİSİNİN UYGULANMASI

PID algoritması çizgi izleme fonksiyonunda ve enkoder ile yol alma fonksiyonunda kullanılmaktadır. Her iki fonksiyonda da PID katsayıları ve örnekleme süreleri bir birinden farklıdır. Çizgi izleme fonksiyonunda kontrast algılayıcılarından alınan bilgilerin robot düz olduğunda sıfır değeri gelecek şekilde PID denetim sistemine aktararak gerçekleştirilmektedir. Yol alma fonksiyonu enkoder ile gidilen adım sayılarak gidilen toplam mesafenin hesaplanması için kullanılmaktadır. Robotun düz ilerleyebilmesi için PID denetim sistemi kullanılmaktadır. Sağ ve sol enkoderden alınan bilgiler arasındaki fark PID denetleyicisine aktararak düz ilerlemesi sağlanmaktadır.



#### 4.4.1. Çizgi İzleme Fonksiyonunda PID Denetleyicisinin Uygulanması

Çizgi izleme algoritması için gerekli hata bilgisi 4 önde 2 arkada olmak üzere 6 adet kontrast algılayıcısından alınan bilgiler ile sağlanmaktadır. Fonksiyon robot çizgi üzerinde düz bir konumdayken hata değerini sıfır döndürecek şekilde yazılmaktadır. En sağ ve en solda bulunan sensörlerin aynı anda görmesi ile robota dik olan çizgilerin sayılması sağlanmaktadır. Aşağıda sensör algılama fonksiyonunun c dilindeki kod'u verilmektedir.

Kontrast algılayıcılarından robotun zemin üzerindeki konumuna göre değerler gelmektedir. Gelen değerler en soldaki kontrast algılayıcısı -8 katsayısı ile çarpılmaya başlayarak ortadaki sensörlerin -1 ve 1 katsayıları ile çarpılmasıyla devam etmektedir. En sağdaki sensör 8 katsayısı ile çarpılmaktadır. Bu katsayılar robotun çizgiden uzaklaştıkça daha hızlı tepki vermesi için doğrusal olmayarak seçilmektedir. Okunan her bir kontrast sensörü siyah renkte 0 beyaz renkte 1 gelecek şekilde yazılımsal olarak ayarlanmaktadır. Katsayılar ile çarpılan kontrast sensörleri beyazda olan algılayıcı sayısı ile bölünerek ara değerler elde edilmektedir. Bu yöntemle algılayıcılardan 11 farklı değer gelmektedir (-8,-5,-2,-1.5,-1,0,1,1.5,2,5,8). Kontrast algılayıcılarından gelen negatif değerler robotun sağa yöneldiğini, pozitif değerler sola yöneldiği anlamına gelmektedir. PID denetleyicisinden gelen hata değeri ile üretilen değer motorlara sabit bir değere ekleme yaparak gönderilmektedir. Bu sabit değerler Robotun çizgi izleme görevini gerçekleştirirken ki hızıdır. Motorlar arasındaki devir farkı nedeniyle bu değerler sol ve sağ motor için farklıdır. 10 KHz DGM frekansı için mikro denetleyici'de maksimum DGM değeri 500 dür. Robotun %60 hızla (75 RPM, 164 mm/s) gitmesi için DGM değeri 300 olarak belirlenmektedir. PID ile oluşan değer sol motor için sabit değerden çıkarılmakta, sağ motor için toplanmaktadır. Değerler maksimum DGM değerini geçmemesi için sınırlandırılmaktadır. PID bloğunda hata değeri kalıcı olduğu zaman integral değeri sürekli arttığı için sınırlandırılmaktadır. Böylelikle robot çizgiyi tekrar bulduğunda hızlı bir şekilde çizgi üzerinde istikrarlı olarak hareket edebilecektir.

```
#define CNY1 !input(pin_b0)
#define CNY2 !input(pin_b1)
#define CNY3 !input(pin_b2)
```

```

#define CNY4 !input(pin_b3)
#define CNY5 !input(pin_b4)
#define CNY6 !input(pin_b5)

#define Agirlik1 -8
#define Agirlik2 -2
#define Agirlik3 -1
#define Agirlik4 1
#define Agirlik5 2
#define Agirlik6 8
void CNY_Oku(float &Sensor, unsigned int8 &Goren_Sensor)
{
    unsigned int8 CNYs[7];
    signed int8 Sensor_Deger = 0.0;
    CNYs[0] = CNY6; [1] = CNY4; CNYs[2] = CNY5; CNYs[3] = CNY2;
    CNYs[4] = CNY3; CNYs[5] = CNY1;

    Goren_Sensor = CNYs[0] + CNYs[1] + CNYs[2] + CNYs[3] + CNYs[4] +
    CNYs[5];

    If(!((CNYs[0] && CNYs[2] && CNYs[3])|(CNYs[5] && CNYs[2] &&
    CNYs[3])))
    {
        if (Goren_Sensor!=0)
        {
            Sensor_Deger = CNYs[0] * (Agirlik1) + CNYs[1] * (Agirlik2) +CNYs[2]
            * (Agirlik3) + CNYs[3] * (Agirlik4) + CNYs[4] * (Agirlik5) + CNYs[5] *
            (Agirlik6);

            Sensor =(float)Sensor_Deger/Goren_Sensor;
        }
    }

    if((CNYs[0]==1 && CNYs[5]==1) && cizgi_say==0)cizgi_say=1;
    if((CNYs[0]==0 && CNYs[5]==0) && cizgi_say==1)
    {
        cizgi_say=0;
        Cizgi++;
    }
}

```

Hata değeri üretilirken sistemin hızlı bir şekilde toparlayabilmesi için en dıştaki sensörlerin ağırlığı daha yüksek seçilmektedir.

Çizgi izleme fonksiyonu için PID denetleyicisinin referans değeri sıfırdır. Referans değeri sabit olup dışarıdan değiştirilememektedir. pwm\_update fonksiyonu hesaplanan PWM değerlerinin motor sürücüsüne gönderilmesidir. Çizgi izleme fonksiyonunun c dilindeki kod'u aşağıda verilmektedir.

```

#define EKp  3.2 //encoder Kp
#define EKi  0.5 //encoder Ki
#define EKd  0.1 //encoder Kd

#define CKp  35 //çizgi izleyen Kp
#define CKi  0.1 //çizgi izleyen Ki
#define CKd  20 //çizgi izleyen Kd

void Cizgi_Izle(void)
{
    dt = 0.0135;
    CNY_Oku(CNYler,Goren_Sensor);
    PID = pid_hesapla(CNYler,1);
    SolPWM = 300 - PID; SagPWM = 318 + PID;
    if (SolPWM>499) SolPWM = 499;
    if (SolPWM<1) SolPWM = 1;
    if (SagPWM>499) SagPWM = 499;
    if (SagPWM<1) SagPWM = 1;
    PWM_update (SolPWM, SagPWM);
}

double Pid_hesapla(double GelenHata, unsigned int8 gonderici)
{
    if(gonderici==0) //encoder
    {
        Ref = encodersag * 0.001; //tekerleğin kayması nedeniyle
//referans alınan yolun binde biri kadar kaydırılır
        error = (Ref - GelenHata); //hata hesapla
        proportional = (ekp*error);
        integral += (eki * error * dt); //integral
        derivative = ekd * (error - errorlast); //türev
    }
    else
    {
        Ref = 0; error = (Ref - GelenHata); //hata hesapla
        proportional = (ckp*error);
        integral += (cki * error * dt); //integral
        derivative = ckd * (error - errorlast); //türev
    }
    PID = proportional + integral + derivative; //pid bulunuyor
    errorlast = error; //hata güncelleştiriliyor
    if(integral >200)integral = 200; if(integral <-200)integral = -200;

    return(PID);
}

```

#### 4.4.2. Enkoder ile Yol Alma Fonksiyonunda PID Denetleyicisinin Uygulanması

Enkoder ile yol alma fonksiyonu mikro denetleyicinin donanımsal kesme girişleri olan timer0 ve timer1'e bağlanan sağ ve sol enkoderlerden bilgi almaktadır. Alınan

bilgiler motorun ne kadar döndüğü bilgisidir. Motor 1 tur döndüğünde enkoderler ortalama 7226 adım saymaktadırlar. Tekerleğin çapı 42mm dir, bir turda  $2\pi r$  çevre hesabından 131 mm yol almaktadır. Robotun 1 mm ilerlemesi için  $7226/131 = 55.16$  adım gereklidir. Pid\_Enkoder fonksiyonuna mm cinsinden adım sayısı girilmektedir. Fonksiyon girilen değer kadar ilerledikten sonra durmaktadır.

Çizgi izleme fonksiyonundan farklı olarak, tekerlek sürtünmesindeki hatalar PID algoritmasına eklenemediği için robot ağırlık merkezinin etki ettiği yöne doğru her adımda 1/1000 oranında kaymaktadır. Bu kayma hatası Referans değerini her adımda 1/1000 oranında kaymanın zıttı yönde değiştirerek giderilmektedir. Kaymanın ne oranda olduğu robotun 100 mm düz ilerletilerek kaç mm ve hangi yöne kaydığı ölçülerek hesaplanmaktadır.

Robotlar hareket ettiklerinde mikro denetleyicinin harici kesmesine bağlı olan enkoder yapısından motorun tur bilgileri elde edilmektedir. Bu değerler sağ ve sol motor için ayrı ayrı elde edilmektedir. Sağ ve sol enkoder değerleri bir birinden çıkartılarak motorlar arasındaki fark bulunmaktadır. Enkoderler arası fark. PID denetleyicisine hata değeri olarak gönderilmektedir. PID denetleyicisi ile enkoder değerlerinin eşitlenmesi için motorların hızlarını değiştirmekte ve en az hata ile ilerlemesi sağlanmaktadır.

```
#define adim_sayisi 55.16

#INT_TIMER0
void TIMER0_isr(VOID)
{
    set_timer0 (65535);
    encodersol++;
}

#INT_TIMER1
void TIMER1_isr(VOID)
{
    set_timer1 (255);
    encodersag++;
}

void Pid_Enkoder(float step)
{
    dt = 0.045;
```

```

double hata=0.0;
signed int16 solen=0,sagen=0;
encodersag=0; encodersol=0;

while(encodersol<(int16)(step*adim_sayisi)||encodersag<(int16)(step*adim_sayisi))
{
    solen = encodersol; sagen = encodersag;
    hata = (double)(solen-sagen);
    Pid_hesapla(hata,0);
    SolPWM = 294 - PID;
    SagPWM = 319 + PID;
    if (SolPWM>499) SolPWM = 499;
    if (SolPWM<1) SolPWM = 1;
    if (SagPWM>499) SagPWM = 499;
    if (SagPWM<1) SagPWM = 1;
    PWM_update (SolPWM, SagPWM);
    delay_ms(30); //fonksiyon yenileme süresi
}
dur(); encodersag=0; encodersol=0;
}

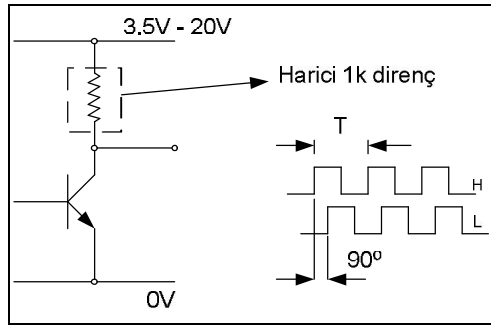
double Pid_hesapla(double GelenHata, unsigned int8 gonderici)
{
    if(gonderici==0) //encoder
    {
        Ref = encodersag * 0.001; //tekerleğin kayması nedeniyle
//referans alınan yolun binde biri kadar kaydırılır
        error = (Ref - GelenHata); //hata hesapla
        proportional = (ekp*error);
        integral += (eki * error * dt); //integral
        derivative = ekd * (error - errorlast); //türev
    }
    else
    {
        Ref = 0;
        error = (Ref - GelenHata); //hata hesapla
        proportional = (ckp*error);
        integral += (cki * error * dt); //integral
        derivative = ckd * (error - errorlast); //türev
    }
    PID = proportional + integral + derivative; //pid bulunuyor
    errorlast = error; //hata güncelleştiriliyor
    if(integral >200)integral = 200;
    if(integral <-200)integral = -200;

    return(PID);
}

```

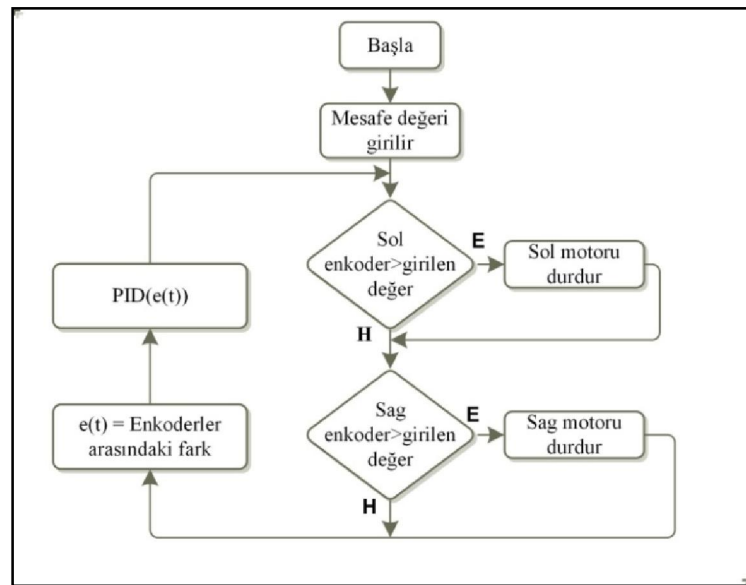
#### 4.5. YÖNELME ALGORİTMASI

Motorlarda bulunan enkoderler mikro denetleyicinin kesme (interrupt) pinlerine bağlanarak kaç tur attıklarının sayılmasıyla robotun ne kadar hareket ettiği bulunmaktadır.  $360^{\circ}$  lik bir dönüşte 16000 adım sayma hassasiyetine sahip olan enkoder ile  $1^{\circ}$  lik dönüş için 44.45 adım atılmaktadır. Enkoderlerdeki sensörler açık kollektör (open collector) şeklinde oldukları için Şekil 4.5 deki gibi pull-up direnci bağlamak gerekmektedir.



Şekil 4.5. Enkoder'in elektronik yapısı ve sinyal şekilleri.

Enkoder destekli ilerleme ve açısal dönüş görevini gerçekleştirmek için PID denetim sistemi kullanılmaktadır. Şekil 4.6 da Enkoder destekli ilerleme ve açısal dönüş algoritması verilmektedir.



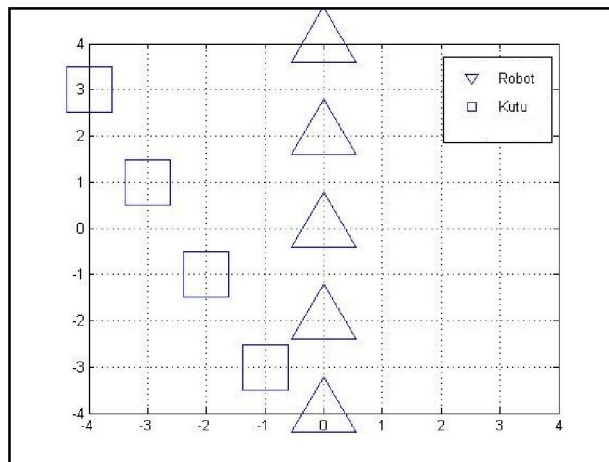
Şekil 4.6. Enkoder destekli ilerleme ve açısal dönüş algoritması.

Robotun gideceği mesafe santimetre cinsinden belirlendikten sonra mesafe bilgisi enkoder adım bilgisine çevrilir. 1 cm 55.16 adım'a tekabül etmektedir. Robotun her iki motoru da aynı adım sayısına ulaşması gereklidir. Motorların dönüş yönü ve yapım farklılıkları nedeniyle aynı hızda dönmedikleri için PID denetim sistemi ile adım sayısına ulaşma esansında eşit hızlarda gitmeleri sağlanmaktadır. Hangi motor istenilen mesafeye ulaştıysa o motor durur ve motorların ikisi de durduğunda yönünün başlangıç yönüyle aynı olması sağlanır.

#### 4.6. ENGEL ALGILAMA ALGORİTMASI

Robot hareket halindeyken sensörün verici LED'inin  $30^{\circ}$  lik optik yayılım açısı nedeniyle geniş bir alandan yansımalar sensör alıcısına gelmektedir. Şekil 3.15 de sensörün optik yayılımı verilmektedir. Bunun sonucunda robot daha yüke yaklaşmadığı halde çıkış da farklı gerilim değerleri okunmaktadır. Hataları en aza indirmek için mikro denetleyicinin ADC'si 10 bit çevirim yapacak şekilde ayarlanmaktadır.

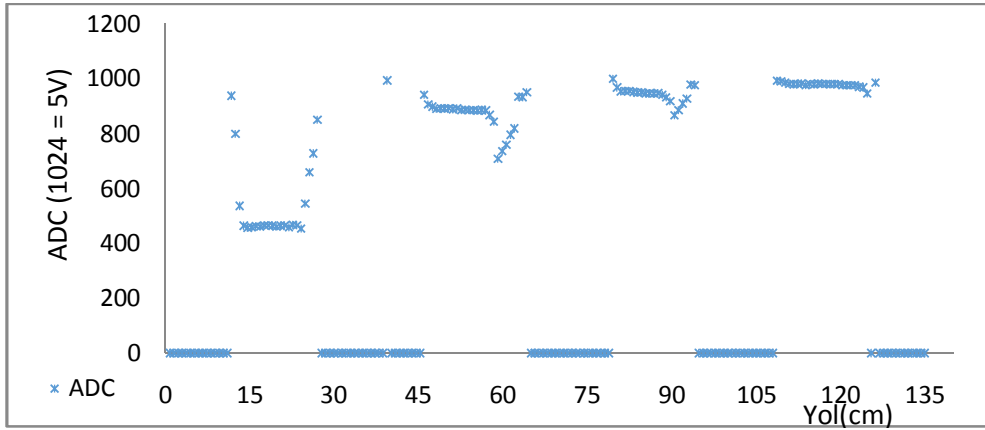
Matris kesişim noktalarına yerleştirilen yüklerin konumlarını hesaplamak için yüklerin robota olan uzaklıkları minimum ve maksimum değerleri alacak şekilde yerleştirilip ölçümler alınmaktadır.



Şekil 4.7. Deney düzeneği.

Bu ölçümler ışığında geliştirilen algoritmada, sensörün optik yayılımından kaynaklanan hatalar fark edilememiş ve yüklerin konumları birden fazla ya da aynı koordinatta arka arkaya iki yük varmışçasına sonuçlar alınmıştır. Hataların giderilmesi amacıyla sensör çıkış değerleri, yazılımsal alçak geçiren filtreden geçirilmiştir. Hatalar kısmen ortadan kalksa da %100 başarımla yükler tespit edilememiş ve çalışmada amaçlanan daha az güç harcaması hedefi engellenmiştir. Başarımı %100'e çıkarmak için sensör çıkışına öncelikle 20 adet verinin ortalaması alınarak yazılımsal alçak geçiren filtre gerçekleştirilmiştir.

Yükleri hatasız olarak algılayabilmek için Şekil 4.7 deki gibi bir deney düzeneği hazırlanmıştır. Bu deneyde robot y ekseninde -4 noktasından +4 noktasına çizgi izleyerek giderken farklı noktalara yerleştirilmiş olan yükler için sensör çıkış değerleri kaydedilmektedir.

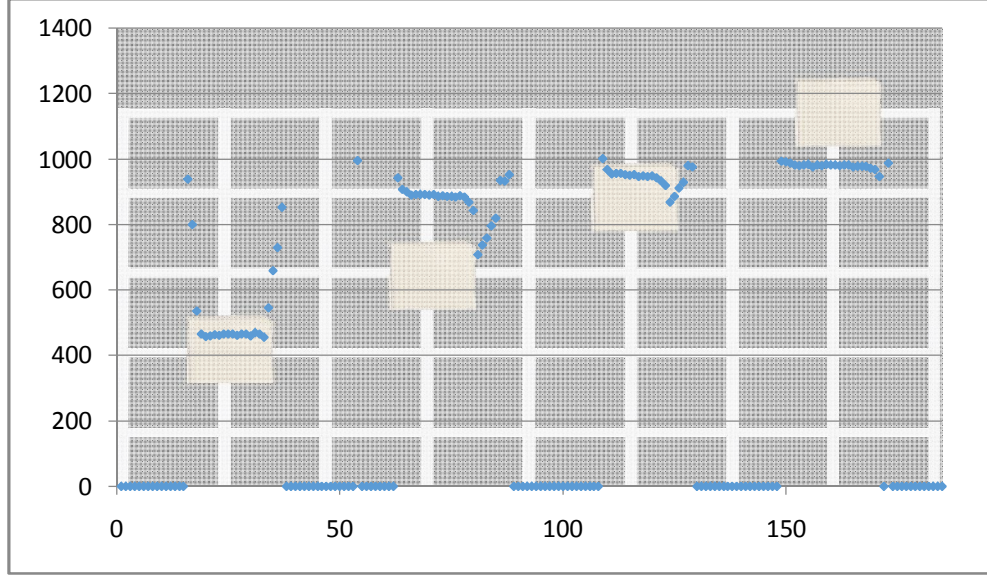


Şekil 4.8. Sensör çıkış değerleri grafiği.

Sensörün çıkış değerleri kaydedilerek elde edilen verilere ait grafik Şekil 4.8 deki verilmektedir. Veriler analizi yapılarak, robotun gidiş yönünde paralel oluşturan noktalar robotun sol tarafında farklı noktalara yerleştirilen yükler olarak tespit edilmektedir.

Grafik alanında yüklerin buldukları koordinatları gösterecek biçimde Şekil 4.9 da görülen alanlar oluşturulmuştur; a alanı 1. yük (-1,-3), b alanı 2. yük (-2,-1), c alanı 3. yük (-3,1) d alanı 4. yük (-4,3) tür.





Şekil 4.9. Sensör çıkış değerlerinin incelenmesi.

Şekil 4.9 de verilen optik sensörden gelen analog değerleri koordinat düzlemin ile anlamlandırılmak istenirse 1. aralık 1 birim uzaklığa, 2. aralık 2 birim uzaklığa, 3. aralık 3 birim uzaklığa, 4. aralık 4 birim uzaklığa denk gelmektedir. Bu aralıkların dar ya da geniş olması optik sensör çıkış karakteristiğinden kaynaklanmaktadır (bknz. Şekil 3.11).

Robotun -4 noktasından +4 noktasına ilerlemesi sırasında okuduğu değerleri birim aralıklara çevrildiğinde Çizelge 4.2 deki veriler elde edilmiştir. Çizelgede, robot 1. yükün yanından geçerken yükten yansıyan değerleri birim aralıklara göre değerlendirdiğinde 16 kez 1 birim uzaklıkta, 3 kez 2 birim uzaklıkta, 1 kez de 3 birim uzaklıkta olduğu sonuçları çıkmıştır. Aynı işlemler diğer yüklerde de gözlenmiştir.

Çizelge 4.2. Yüklerin ölçüm hataları.

	1 Birim (15 cm)	2 Birim (30 cm)	3 Birim (45 cm)	4 birim (60 cm)
1.Yük (-1,-3)	16	3	1	0
2.Yük (-2,-1)	0	17	4	1
3.Yük (-3,1)	0	3	15	4
4.Yük (-4,3)	0	0	3	18

Robot bir yükün yanından geçerken yükün koordinat düzlemindeki konumunun algılayabilmek için sensör verilerindeki konum bilgisinin doğruluğunu en çok hangi konumda olduğuna bakarak karar vermektedir. Gerçekleştirilen engel algılama

algoritması ile %100 başarıyla yüklerin yerini tespit etmektedir. Engel algılama algoritması, ölçüm süresince mesafe sensöründen gelen bilgilerin ağırlıklı olarak hangi mesafeden yansıdığını algılama fonksiyonudur. Robot çizgileri sayarak ilerlerken alandaki yükleri algılama işlemini gerçekleştirmektedir. Yük algılama işlemi, sensörden hiçbir cisim görmediği değerden farklı bir değer geldiği anda başlamaktadır. Hiçbir engel olmadığındaki değerden farklı bir değer geldiği anda en son güncellediği değer saklanır. Bu süreç içerisinde yük'ü geçene kadar sensörden gelen değerler koordinat düzlemindeki konumların uzaklıklarına göre belirlenmiş değerler aralığında bulunma haline göre bir sayaç tarafından sayılır. Yüklerin koordinat düzlemindeki konumlarına göre ölçümlerle alınan ADC değerleri aşağıda verilmiştir.

X Ekseninde;

- -1 veya +1 de olma durumu 450-600 arası
- -2 veya +2 de olma durumu 100-200 arası
- -3 veya +3 de olma durumu 50-90 arası
- -4 veya +4 de olma durumu 30-50 arası

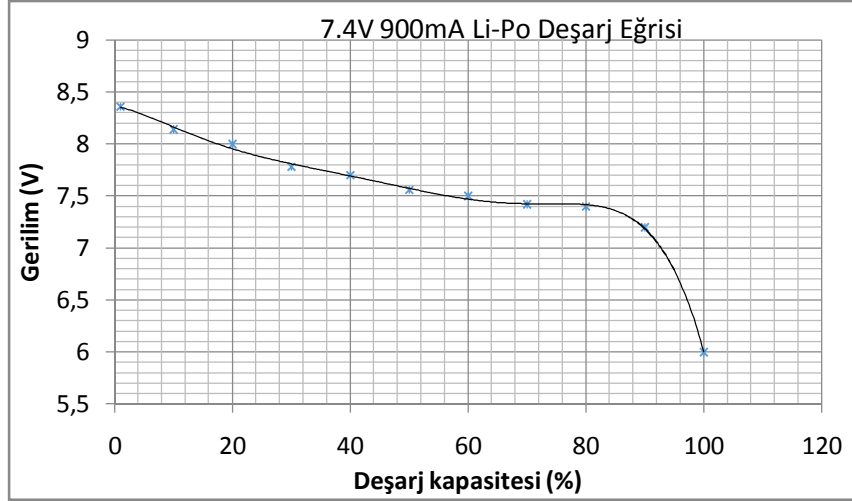
Sensörün hiçbir cisim algılamadığı değerden farklı bir değer geldiğinden itibaren tekrar hiçbir cisim görmediğindeki değere ulaşana kadar ki süreçte alınan ADC değerleri hangi değerler aralığındaysa o konumun sayacı arttırılır. Yük'ün yanından geçme süreci bittiğinde en çok hangi sayaç artmışsa yük'ün o konumda olduğu kararı verilmektedir.

#### **4.7. GÜÇ YÖNETİM SİSTEMİ**

Robotlar periyodik olarak besleme gerilimi ölçerler. Bu ölçümlerle kullanılan Li-Po pil'in deşarj eğrisindeki değerlerle karşılaştırılarak toplam gücünün ne kadarını harcadığı bilgisini elde etmektedir.

Robotlar merkezden ya da sürü robot üyelerinden gelen görevleri gerçekleştirmeden önce görev için gerekli güç harcamasını tahmin etmekte ve bataryasında kalan toplam güç ile karşılaştırarak görevi gerçekleştirip gerçekleştiremeyeceği bilgisini üretmektedir. Eğer görevi gerçekleştirebilecek kadar güç'e sahip değilse görevi

gerçekleştiremeyeceği bilgisini göndermektedir. Şekil 4.10 da kullanılan Li-Po bataryanın deşarj eğrisi verilmektedir.



Şekil 4.10. Kullanılan Li-Po bataryanın deşarj eğrisi.

Robotlar hareket halindeyken 300 mA ile 500 mA arasında akım çekmektedir. Robot kesintisiz çalıştığında 900 mA kapasiteye olan batarya 2 saat içerisinde tüketilecektir. Bir saniye de 164 mm ilerleyen robot kapasitenin %0.0139 'unu harcamaktadır. Örneğin robot 1200 mm lik bir mesafe sahip yük'e 7.3 saniyede yaklaşacaktır. 7.3 saniyelik görev gerçekleştirildiğinde kapasitenin %0.102 'sini harcayacaktır. Mevcut kapasite bataryanın deşarj eğrisindeki gerilim değerine göre çıkartılmaktadır. Görevi tamamladığında harcayacağı güç değerini bulurken, çalışacağı süre ve bir saniyede harcamış olduğu enerjinin çarpılmasıyla bulunmaktadır.

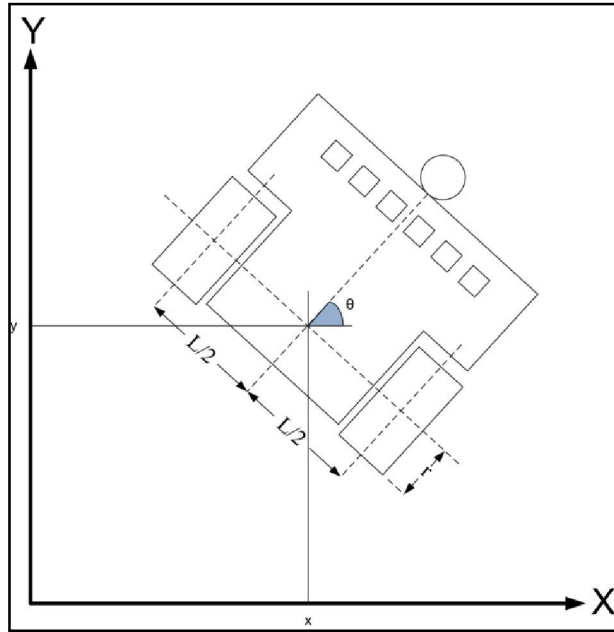
## BÖLÜM 5

### HAREKET FONKSİYONU VE HABERLEŞME PROTOKOLÜ

Bu bölümde robotun hareket fonksiyonlarında kullanılan yöntemler verilmektedir. Robot'un gideceği koordinata en kısa yolu tercih ederek gidebilmesi için eğri uydurma yöntemi kullanılmaktadır. Haberleşme protokolü olarak doğrusal topoloji seçilmiştir.

#### 5.1. EĞRİ UYDURMA

Eğri uydurma, iki ya da daha çok değişken arasındaki ilişkiyi ölçmek için kullanılan analiz yöntemidir. Eğer tek bir değişken kullanılarak analiz yapılıyorsa buna tek değişkenli regresyon, birden çok değişken kullanılıyorsa çok değişkenli regresyon analizi olarak isimlendirilmektedir. Şekil 5.1 'de robotun evrensel düzlemdeki konumlandırılması verilmiştir.



Şekil 5.1. Tasarlanan robot'un evrensel düzlemde konumlandırılması.

Birbirine bağılı olarak deęişen iki fiziksel büyüklük arasındaki matematiksel bağlantıyı, mümkün olduğunca gerçeęe uygun bir denklem olarak yazmak için en küçük kareler yöntemi kullanılmaktadır. Bir başka deyişle bu yöntem, ölçüm sonucu elde edilmiş veri noktalarına "mümkün olduğu kadar yakın" geçecek bir fonksiyon eğrisi bulmaya yaramaktadır. En küçük kareler yöntemi, doğrusal örnek ve doğrusal olmayan örnek şeklinde ikiye ayrılmaktadır.

Bu çalışmada 2 deęişkenli doğrusal örnek yöntemi kullanılmaktadır. Doğrusal olan örnekler  $Y = mx + n$  şeklinde ifade edilmektedir. Eşitlikteki  $m$  ve  $n$  deęerleri doğruyu çizmek için gerekli en doğru deęerler olarak tanımlanabilmektedir.  $m$  ve  $n$  deęerlerini bulabilmek için ön koşul, çizilecek olan doğrunun eşitliğe giren data deęerlerine olan uzaklığının karesinin en küçük olmasıdır.

Bu koşul eşitliğin  $m$ 'ye göre türevi (Eşitlik 5.2) ile  $n$ 'ye göre türevinin (Eşitlik 5.3) sıfır olmasıyla mümkündür. Eşitliğin  $m$ 'ye göre türevi ile  $n$ 'ye göre türevi yazıp bir dizi matematiksel işlemler yaparak eşitlik 4.8 elde edilmektedir. Eşitliğe Cramer's kuralı uygulanıp çözümlerse eşitlik 5.11 ve 5.12 elde edilmektedir. Eşitlik artık çözülebilir bir yapıya sahip olmuştur.

$$S = \sum_{i=0}^k mx_i + n - y_i \quad (5.1)$$

$$\frac{\partial S}{\partial m} = 0 \quad (5.2)$$

$$\frac{\partial S}{\partial n} = 0 \quad (5.3)$$

$$\frac{\partial S}{\partial m} = \sum_{i=1}^k 2(mx_i + n - y_i)x_i \quad (5.4)$$

$$\frac{\partial S}{\partial n} = \sum_{i=1}^k 2(mx_i + n - y_i) \quad (5.5)$$

$$m \sum_{i=1}^k x_i^2 + n \sum_{i=1}^k x_i - \sum_{i=1}^k x_i y_i = 0 \quad (5.6)$$

$$m \sum_{i=1}^k x_i + n \sum_{i=1}^k 1 - \sum_{i=1}^k y_i = 0 \quad (5.7)$$

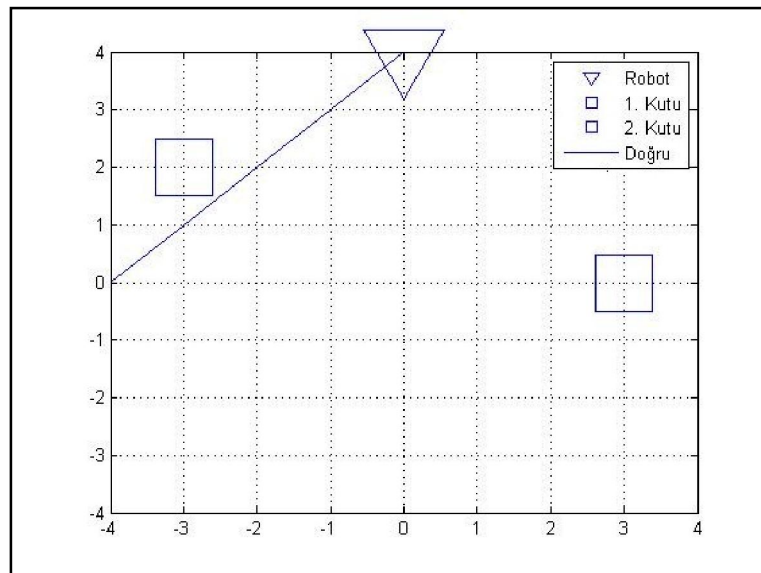
$$\begin{pmatrix} \sum_{i=1}^k x_i^2 & \sum_{i=1}^k x_i \\ \sum_{i=1}^k x_i & k \end{pmatrix} \begin{pmatrix} m \\ n \end{pmatrix} = \begin{pmatrix} \sum_{i=1}^k x_i y_i \\ \sum_{i=1}^k y_i \end{pmatrix} \quad (5.8)$$

$$m = \frac{\begin{vmatrix} \sum_{i=1}^k x_i y_i & \sum_{i=1}^k x_i \\ \sum_{i=1}^k y_i & k \end{vmatrix}}{\begin{vmatrix} \sum_{i=1}^k x_i^2 & \sum_{i=1}^k x_i \\ \sum_{i=1}^k x_i & k \end{vmatrix}} \quad (5.9)$$

$$n = \frac{\begin{vmatrix} \sum_{i=1}^k x_i^2 & \sum_{i=1}^k x_i y_i \\ \sum_{i=1}^k x_i & k \end{vmatrix}}{\begin{vmatrix} \sum_{i=1}^k x_i^2 & \sum_{i=1}^k y_i \\ \sum_{i=1}^k x_i & k \end{vmatrix}} \quad (5.10)$$

$$m = \frac{k \sum_{i=1}^k x_i y_i - \sum_{i=1}^k x_i \sum_{i=1}^k y_i}{k \sum_{i=1}^k x_i^2 - (\sum_{i=1}^k x_i)^2} \quad (5.11)$$

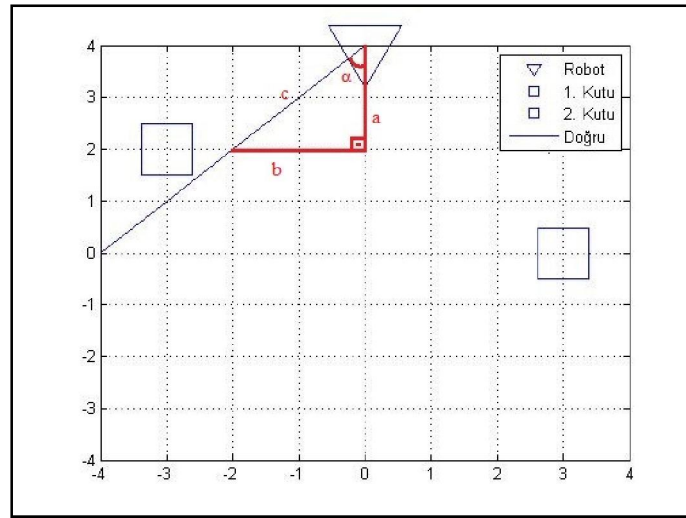
$$n = \frac{k \sum_{i=1}^k y_i \sum_{i=1}^k x_i^2 - \sum_{i=1}^k x_i \sum_{i=1}^k x_i y_i}{k \sum_{i=1}^k x_i^2 - (\sum_{i=1}^k x_i)^2} \quad (5.12)$$



Şekil 5.2. Eğri uydurma yöntemiyle robotun ile yük arasında çizilen doğru.

Çözümü yapılan matematiksel eşitliğe yük koordinatları girilerek robotla yük arasında çizilen doğru Şekil 5.2 'de verilmektedir. Robotun bu doğru üzerinde ilerleyebilmesi için gerekli açı bilgisi  $Y = mx + n$  formülündeki  $m$  ile bulunmaktadır. Açı bilgisi robot kuzey ve güney yönündeyken  $y$  düzlemiyle yaptığı açı bilgisi, doğu ve batı yönündeyse  $x$  düzlemiyle yatığı açı bilgisi için  $m$ 'nin ark tanjantı ile bulunur.

Robot gerekli açı bilgisini elde ettikten sonra enkoderlerden aldığı bilgi ile istenilen açı değeri kadar dönmekte ve yüke doğru ilerlemektedir (Şekil 5.3). Yükü itmek için yapmış olduğu dönüş tamamlandığında robot bir önceki konumuna göre yönü değişiyorsa yüke yaklaşırkenki döndüğü açıyı  $90^0$  ye tamamlayan derece kadar daha dönmekte ve yüke olan hareketini tamamlamış olmaktadır. Robot yüke yaklaşırkenki yapmış olduğu dönüş ve yüke yaklaşma görevinde yönü değişmiyorsa yüke yaklaşırkenki açının tersi kadar daha dönerek görevini tamamlamış olmaktadır.



Şekil 5.3. Robot'un yapacağı açı ve gideceği mesafenin hesaplanması.

Eğri uydurma yönteminin benzetimi MATLAB ortamında yapılmıştır. Benzetimde Robot üçgen ile yükler kare ile simgeleştirilmiştir. Eğri uydurma denkleminin sonucu  $x$  ekseninin tüm değerleri için yani  $-4$  ile  $+4$  arasında hesaplanıp grafikte çizilmiştir. 2 değişkenli doğrusal örnek analizi için yazılan MATLAB m file kodları aşağıda verilmektedir.

```
% Başlangıç konumu P(0,4) olan robotun
% Konumu F(-3,2) olan kutuya gitmesi için çizilen doğru
```

```

% Kutuyu itebilmesi için kutuya x ekseninde bir adım geriden
% yaklaşması gerekiyor
% Robotun pozisyonu haritadaki standart yönler göre seçilmiştir
Pos = 1; % robotun pozisyonu 0=kuzey, 1=güney, 2=doğu, 3=batı
x1 = 0;
y1 = 4;
X2 = -3;
y2 = 2;
if(Pos==0)
    plot(x1,y1,'^','MarkerSize',40) % Robot çizimi yaptırılıyor
elseif(Pos==1)
    plot(x1,y1,'v','MarkerSize',40) % Robot çizimi yaptırılıyor
elseif(Pos==2)
    plot(x1,y1,'<','MarkerSize',40) % Robot çizimi yaptırılıyor
elseif(Pos==3)
    plot(x1,y1,'>','MarkerSize',40) % Robot çizimi yaptırılıyor
end
hold on% çizimleri sakla
plot(X2,y2,'s','MarkerSize',40) % Kutu çizimi yaptırılıyor
plot(3,0,'s','MarkerSize',40) % Kutu çizimi yaptırılıyor
% Eğri uydurma denklemi
if(X2>0)
    x2 = X2 -1;
elseif(X2<0)
    x2 = X2 +1;
end
eXi = x1 + x2 ;
eYi = y1 + y2 ;
eX2i = (x1*x1)+(x2*x2) ;
eXiYi = (x1*y1)+(x2*y2) ;
m1 = ((2*eXiYi) - (eXi*eYi));
n1 = ((eYi*eX2i) - (eXi*eXiYi));
s = ((2*eX2i)-(eXi*eXi));
m = m1/s;
n = n1/s;
x = -4:4; % X ekseninin minimum ve maksimum değerleri
y = 1:9;
for i = 1:9
    y(i) = m*x(i) + n;% Eğri uydurma formülü
end
plot(x,y) % çizim yaptırılıyor
axis([-4 4 -4 4])
grid on

```

Robot yüke yaklaşma görevini tamamladıktan sonra bulunduğu konum bilgisine sahip olduğu için yükü alanın dışına taşımak için ne kadar mesafe kat edeceğini



bilmektedir. Bu bilgiyi kullanarak enkoder yardımıyla kutuyu dışarıya taşımakta ve tekrar matris alanına dönerek koordinat düzlemindeki bulunduğu yeri ve yönünü düzeltmektedir.

Kutuya yaklaşma, kutuyu dışarıya taşıma ve matris alanına dönüp yönünü ve konumu düzeltme işlemi bittikten sonra varsa ikinci yük için gerekli hesaplamaları yapmakta ve yüke doğru ilerlemektedir.

## 5.2. ROBOTUN KABLOSUZ İLETİŞİMİ

Wi-Fi ve Bluetooth gibi kablosuz haberleşme teknolojilerinin kimi uygulamalarda elverişsiz olmaları nedeniyle 1998 yılından itibaren Zigbee stili ağlar üzerinde çalışılmaya başlanmıştır. Amaç güvenilirliği yüksek, düşük maliyetli, enerji tasarruflu, görüntüleme ve yönetme amaçlı ağlar kurmaya elverişli ürünler ortaya çıkarmaktır. Zigbee teknolojisinin ihtiyacı olan standart IEEE tarafından 802.15.4 standardı Mayıs 2003'te tamamlanmıştır. Çizelge 5.1 de Zigbee'nin diğer RF çözümlerle olan karşılaştırılması verilmektedir.

Çizelge 5.1. Zigbee ve diğer RF çözümlerin karşılaştırılması.

Özellik	Zigbee	GPRS/GSM	wi-Fi	Bluetooth
Odaklanma Alanı	İzleme ve Kontrol	Geniş alan ses ve veri	Web, e-posta, görüntü	kablo yerine
Sistem Kaynağı	4-32 Kb	16 Mb+	1 Mb+	250 Kb+
Pil Ömrü(Gün)	100-1000+	1-7	0,5-5	1-7
Ağ boyutu (adet)	sınırsız $2^{64}$	1	32	7
Veri Genişliği (kb/s)	20-250	64-128+	11000+	720
Kapsama Alanı (m)	1-100+	1000+	1-100	1-10+
Başarı Alanları	Dayanıklılık, Maliyet, Güç tüketimi	Ulaşılabilirlik, kalite	Hız, esnelik	Maliyet, rahatlık

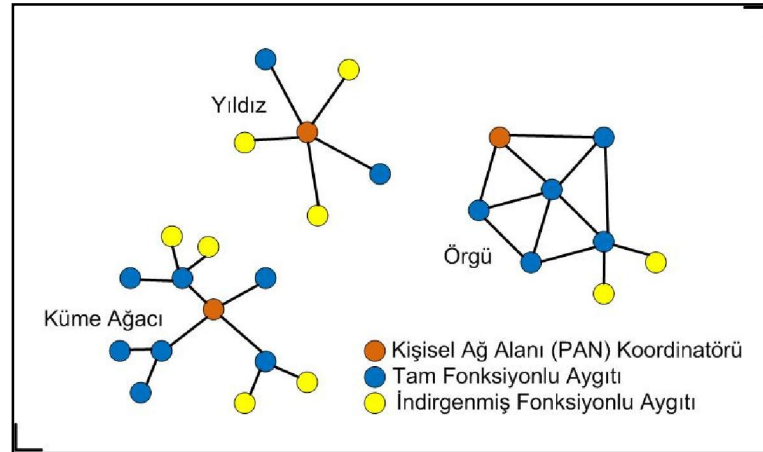
Zigbee sağladığı avantajlar sayesinde birçok sektörde sensör haberleşmesinde, kontrol ve izleme mekanizmalarında kullanılmaktadır. Zigbee'nin yaygın olarak kullanıldığı bazı sektörler ve bu sektörlerdeki uygulamaları aşağıda belirtilmiştir. Ticari Bina ve Ev Otomasyonu: HVAC (Isınma, Havalandırma ve Klima Sistemlerinde), ışıklandırma kontrolü, kapı – pencere – panjur – kilit kontrollerinde,

ev otomasyonunda müzik ve sinema sistemlerinin kontrolünde, bina içi giriş- çıkış kontrollerinde kullanılmaktadır.

Çizelge 5.1 de görüldüğü üzere Zigbee, pil ömrünün uzunluğu, istenildiği kadar ağ kurma avantajı, sistem kaynaklarını minimumda kullanması sayesinde izleme ve kontrol amaçlı uygulamalarda kullanılabilir en uygun teknolojidir. Dezavantajı ise Wi-Fi veya Bluetooth'da olduğu gibi büyük boyutlarda veri akışı sağlayamamasıdır. Bu da Zigbee'nin daha çok küçük boyutlarda veri akışının olduğu uygulamalarda kullanılması demektir. Ayrıca Zigbee lisans gerektirmeyen frekans bandını kullanır, kurulumu ucuz ve kolaydır. Esnek ve genişletilebilen ağ yapısına sahiptir.

Zigbee noktadan noktaya, yıldız, bir noktadan birden çok noktaya veya çok noktadan tek noktaya, ağaç ve örgü topolojileri olmak üzere birçok topolojiyi destekler.

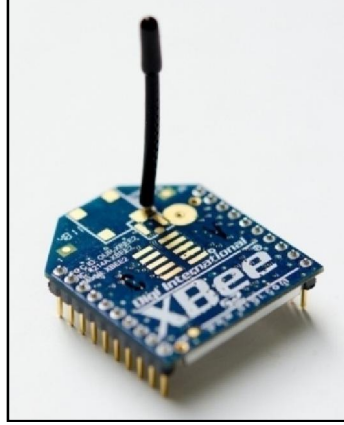
Yıldız topolojisinde merkezde koordinatör bulunur. Diğer bütün cihazlar noktadan noktaya olacak şekilde koordinatörle iletişim kurarlar. Mesajlaşma koordinatör üzerinden yapılır. Şekil 5.4 de merkezdeki cihaz koordinatör, diğer cihazlar son aygıttır.



Şekil 5.4. Zigbee topolojileri.

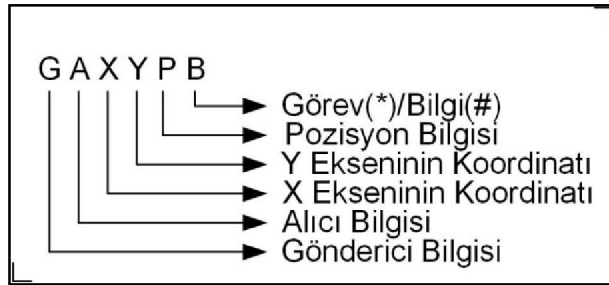
Hiyerarşik (küme ağacı) bir düzende cihazların birbirleriyle haberleşme biçimidir. Ağaç yapısının en üst noktasında koordinatör bulunur. Bir düşük seviyede yönlendiriciler ve en alt seviyede ise son veya yönlendirici aygıtlar bulunur. Örgü topolojisi, en çok kullanılan topoloji olup aynı zamanda projede de kullanılmıştır.

Her cihaz doğrudan diğer cihazlara bağlıdır [20]. XBEE Digi firmasının Zigbee alt yapısı ile oluşturduğu bir modüldür. Bu çalışmada kullanılan XBee modülü 1mW, 2.4GHz özelliklerine sahiptir. XBee Shield Ünitesi üzerinde PCB anten yer almaktadır. Yaklaşık olarak 100 metrelik alanda iletişim kurabilmektedir.



Şekil 5.5. Xbee modül.

Şekil 5.5.'de verilen XBee modülü, Digi firmasının “ZigBee and RF Modules” ürün kategorisinde bulunan 2.4 GHz bandında kablosuz haberleşme sağlayan, UART (Universal Asynchronous Receiver/Transmitter) arabirimi ile kullanılabilen bir modüldür. Temel amacı kabloları ortadan kaldırıp kolay bir şekilde haberleşmeyi sağlamaktır. Bunu yaparken birçok ağ topolojisine uygunluk göstermektedir. XBee modülü RF ağ oluşturarak, ağa bağlı olan tüm kullanıcılara bilgi aktarımı sağlayan bir platform olduğu için tercih edilmiştir.

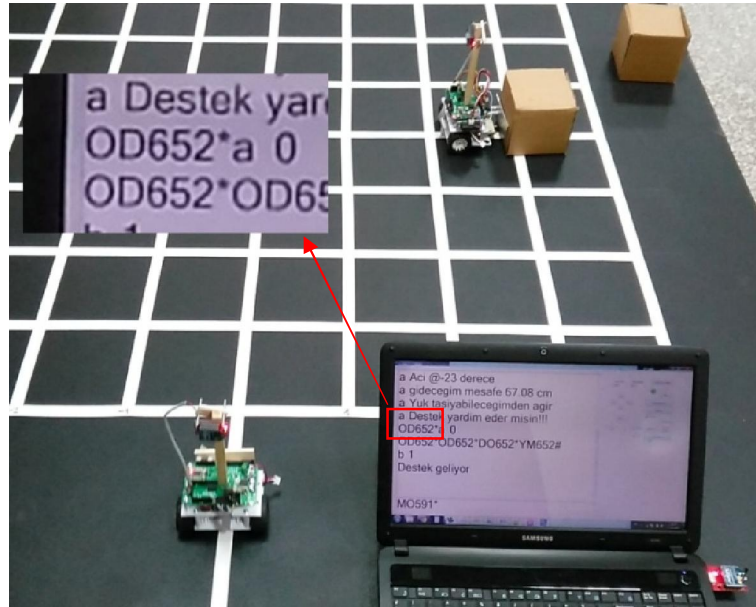


Şekil 5.6. Protokolün yapısı.

Robotlar arası iletişimde senkronizasyonu sağlamak için bir protokol hazırlanmıştır. Bu protokol'ün uzunluğu 6 byte dır. Tanımlanan protokolde, göndericinin ve alıcının

kim olduğuna dair bilgiler ve koordinat noktaları bulunmaktadır. Hazırlanan protokol Şekil 5.6 'da verilmektedir.

Protokol'ün ilk byte'ı göndericinin kim olduğu bilgisini taşımaktadır. Bu çalışmada Merkez bilgisayar için "M", Öncü robotu için "O" ve Destek robotu için "D" harfi kullanılmıştır. İkinci byte, alıcının kim olduğu bilgisini taşımaktadır. Protokolün ilk baytında olduğu gibi M,O ve D harfleri bu byte için de geçerlidir. Üçüncü ve dördüncü byte'lar X ve Y eksen bilgilerini taşımaktadır. Yapılan çalışmada koordinat düzlemi -4 ile +4 arasında olmasına rağmen, protokolün sadeliği için koordinat bilgileri 5 ile toplanarak gönderilmektedir. Bu şekilde protokol üzerinde negatif sayıların olması engellenmektedir. Alıcı robot gelen koordinat verilerinden 5 çıkartarak protokoldeki verileri koordinat düzlemine uyarlamaktadır. Beşinci byte, pozisyon bilgisini içermektedir. Pozisyon bilgisi +Y için 0, -Y için 1, -X için 2, +X için 3 olarak düzenlenmiştir. Altıncı byte, eğer "\*" ise robotun gönderilen koordinat ve pozisyon değerlerine gitmesi istenir, eğer "#" ise gönderilen koordinat bilgilerinden bağımsız olarak, robotun bulunduğu koordinat bilgilerinin istenmesidir. Şekil 5.7 'de Öncü robotunun Destek robotundan yardım isterken göndermiş olduğu protokol verisi verilmektedir.



Şekil 5.7. Protokol örneği.

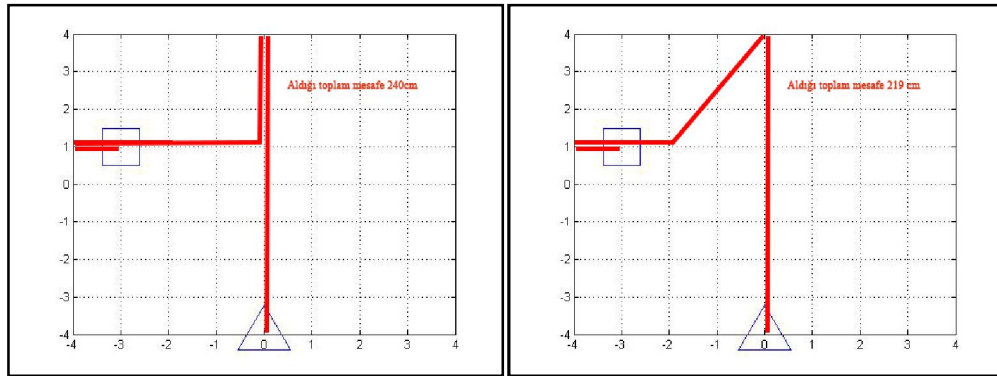
## BÖLÜM 6

### DENEYSEL ÇALIŞMALAR

Bu bölümde robotun pist üzerine yerleştirilen yüklerin konumuna göre vermiş olduğu tepkileri ve bu tepkilere bağlı olarak başarı yüzdeleri verilmektedir. Robotun yükleri taşıma görevini gerçekleştirirken eğri uydurma yöntemi ile çizgi takip etme yöntemi arasında bir karşılaştırma yapıldığında Çizelge 6.1 de verilen değerlerde daha az mesafe kat etmiştir. Eğri uydurma yöntemi ile alınan değerlerin ortalaması %11 dir.

Çizelge 6.1. Yük'ün Y eksenindeki değişimlerine göre aldığı mesafeler.

Konum	Çizgi izleme yöntemi	Eğri Uydurma Yöntemi	Yöntemler arası fark	Yöntemler arası fark (%)
-3,4	195	195	0	%0
-3,3	210	199	11	%19
-3,2	225	207	18	%12,5
-3,1	240	219	21	%11,4
-3,0	255	232	23	%11
-3,-1	270	246	24	%11,25
-3,-2	285	259	26	%10,9
-3,-3	300	274	26	%11,5
-3,-4	315	289	26	%12



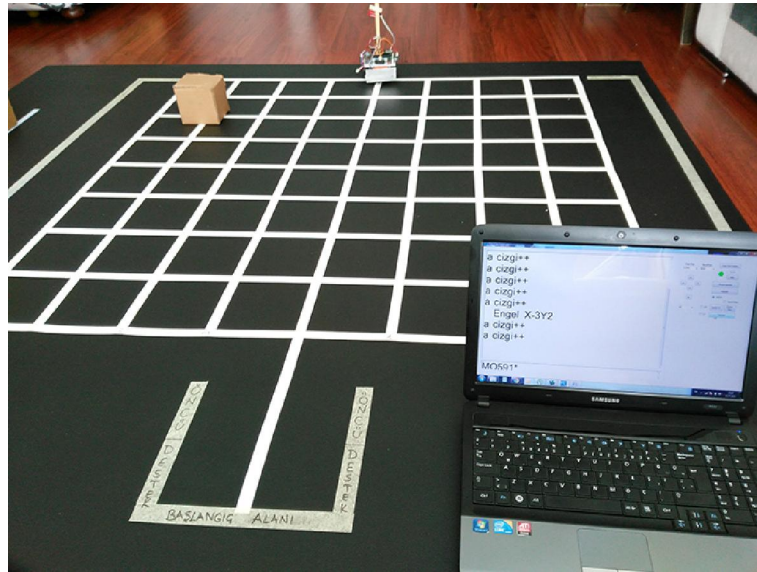
a) Çizgileri takip ederek aldığı yol      b) Eğri uydurma yöntemiyle aldığı yol

Şekil 6.1. Robotun çizgileri takip etmesi ve eğri uydurma yöntemini kullanması.

Şekil 6.1 de çizgi izleme yöntemi ile eğri uydurma yönteminin izlediği yollar karşılaştırmalı olarak verilmektedir.

### 6.1. YÜKÜN ROBOTA YAKIN KONUMDA OLMASI

Robot alana 0,-5 noktasından giriş yaparak 0,4 noktasına kadar çizgi takip ederek ilerlemektedir. Tarama görevi bittiğinde 180° dönerek kendisine en yakın yük'e yönelmektedir. Şekil 6.2 de -3,2 noktasına konulan yük için yapılan deneyin fotoğrafı verilmektedir. Robotun gerilim değeri ve görevi tamamlama yüzdeleri Çizelge 6.2 de verilmektedir.



Şekil 6.2. Yük'ün robot'a yakın koordinatta olması deneyi.

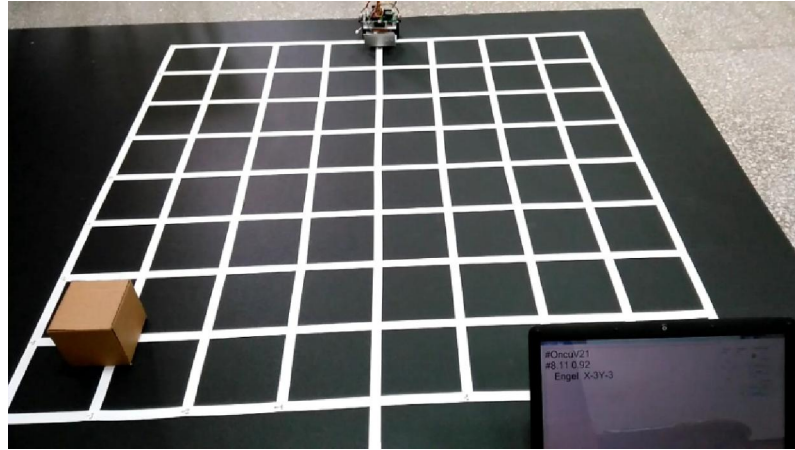
Çizelge 6.2. Yük'ün robota yakın koordinatta olma durumu.

Deneme	Gerilim (V)	Kalan Şarj Oranı	Algılanan Yük Koordinatı	Tarama Görevi	Yük'e Yaklaşma Görevi	Yük'ü Taşıma Görevi	Koordinat Güncelleme Görevi
1	7,8	%80	(-3,2)	+	+	+	+
2	7,79	%80	(-3,2)	+	+	+	+
3	7,75	%70	(-3,2)	+	+	+	+
4	7,7	%70	(-3,2)	+	+	+	+
5	7,67	%70	(-3,2)	+	+	+	+
6	7,7	%70	(-3,2)	+	-	-	+
7	7,7	%70	(-3,2)	+	+	+	+
8	7,66	%60	(-3,2)	+	+	+	+
9	7,7	%70	(-3,2)	+	+	+	+
10	7,66	%60	(-3,2)	+	+	+	+
Görev Tamamlama Yüzdeleri				%100	%90	%90	%100

Robot 6. deneme de tarama işlemi bittikten sonra 180 derecelik dönüşü yapmış ve çizgi üzerinde düz duramadığı için yük'e yaklaşma görevini gerçekleştirmemiştir. Yük'ün robota yakın olma durumunda başarı oranı tamamlaması gereken dört görevden en küçük değer ile belirlenir. Bu durumda yük'ün robota yakın olması durumundaki başarı yüzdesi %90 dır.

## 6.2. YÜKÜN ROBOTA UZAK KONUMDA OLMASI

Robot alana 0,-5 noktasından giriş yaparak 0,4 noktasına kadar çizgi takip ederek ilerlemektedir. Tarama görevi bittiğinde 180° dönerek kendisine en yakın yük'e yönelmektedir. Yapılan deneyde yük -3,-1 noktasına konulmuştur. Robotun gerilim değeri ve görevi tamamlama yüzdeleri Çizelge 6.2 de verilmektedir.



Şekil 6.3. Yük'ün robot'a uzak koordinatta olması deneyi.

Çizelge 6.3. Yük'ün robota uzak koordinatta olma durumu.

Deneme	Gerilim (V)	Kalan Şarj Oranı	Algılanan Yük Koordinatı	Tarama Görevi	Yük'e Yaklaşma Görevi	Yük'ü Taşıma Görevi	Koordinat Güncelleme Görevi
1	7,66	%60	(-3,-3)	+	+	+	+
2	7,66	%60	(-3,-3)	+	+	+	+
3	7,62	%60	(-3,-3)	-	-	-	-
4	7,60	%60	(-3,-3)	+	+	+	+
5	7,54	%50	(-3,-3)	+	+	+	+
6	7,54	%50	(-3,-3)	+	-	-	-
7	7,54	%50	(-3,-3)	+	-	-	-
8	7,54	%50	(-3,-3)	+	-	-	-
9	7,43	%40	(-3,-3)	+	+	+	+
10	7,43	%40	(-3,-3)	+	+	+	+
Görev Tamamlama Yüzdeleri				%90	%60	%60	%60

Robot 3. deneme de tarama işlemini gerçekleştirirken sonsuz döngüye girdiği için bu görevi tamamlayamamıştır. 6.,7. ve 8. denemelerde yük ile olan mesafesi uzak olduğu için küçük bir açı hatası büyük hatalara neden olmaktadır. Bu nedenle yük'e yaklaşamamış ve yük'ü dışarıya taşıyamamıştır. Yük'ün robota uzak olma durumunda başarı oranı tamamlaması gereken dört görevden en küçük değer ile belirlenir. Bu durumda yük'ün robota uzak olması durumundaki başarı yüzdesi %60 dır.

### 6.3. YÜKÜ YARDIMLAŞARAK TAŞIMA

Robot alana 0,-5 noktasından giriş yaparak 0,4 noktasına kadar çizgi takip ederek ilerlemektedir. Tarama görevi bittiğinde 180° dönerek kendisine en yakın yük'e yönelmektedir. Yapılan deneyde yük 3,0 noktasına konulmuştur.

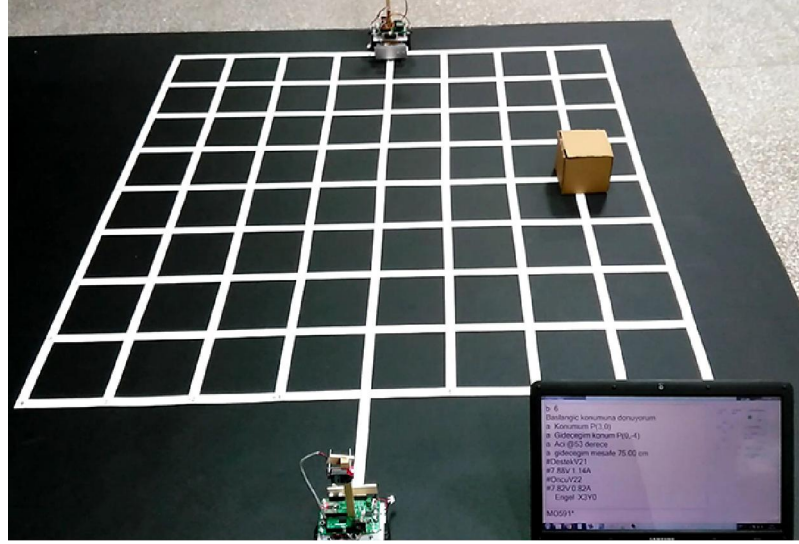
Çizelge 6.4. Öncü robotunun yardımlaşarak çalışma durumu.

Deneme	Gerilim (V)	Kalan Şarj Oranı	Algılanan Yük Koordinatı	Tarama Görevi	Yük'e Yaklaşma Görevi	Yük'ü Taşıma Görevi	Koordinat Güncelleme Görevi
1	8,25	%100	P(2,1)	-	+	+	+
2	8,23	%100	P(3,0)	+	+	+	+
3	8,26	%100	P(3,0)	+	-	-	-
4	8,17	%100	P(3,0)	+	+	+	+
5	8,18	%100	P(3,0)	+	+	+	+
6	8,19	%100	P(3,0)	+	+	+	+
7	8,17	%100	P(3,0)	+	+	+	+
8	8,10	%90	P(3,0)	+	+	+	+
9	8,08	%90	P(3,0)	+	+	+	+
10	8,10	%90	P(3,0)	+	+	+	+
Görev Tamamlama Yüzdeleri				%90	%90	%90	%90

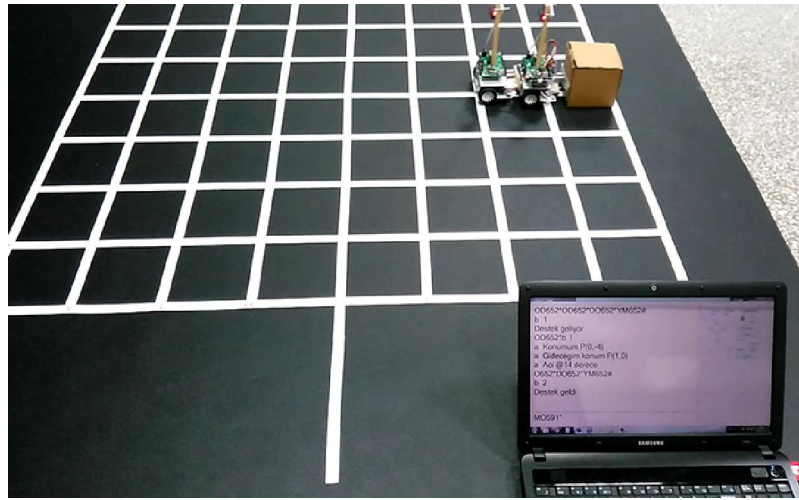
Çizelge 6.5. Destek robotunun yardımlaşarak çalışma durumu.

Deneme	Gerilim (V)	Kalan Şarj Oranı	Algılanan Yük Koordinatı	Yük'e Yaklaşma Görevi	Yük'ü Taşıma Görevi	Koordinat Güncelleme Görevi
1	8,29	%100	P(2,1)	-	-	-
2	8,29	%100	P(3,0)	+	+	+
3	8,27	%100	P(3,0)	-	-	-
4	8,27	%100	P(3,0)	-	-	-
5	8,24	%100	P(3,0)	+	+	+
6	8,25	%100	P(3,0)	+	+	+
7	8,20	%100	P(3,0)	+	+	+
8	8,23	%100	P(3,0)	+	+	+
9	8,23	%100	P(3,0)	+	+	+
10	8,17	%100	P(3,0)	+	+	+
Görev Tamamlama Yüzdeleri				%70	%70	%70





Şekil 6.4. Destek robotunun tarama işlemini bitirip yük'e yönelmesi.



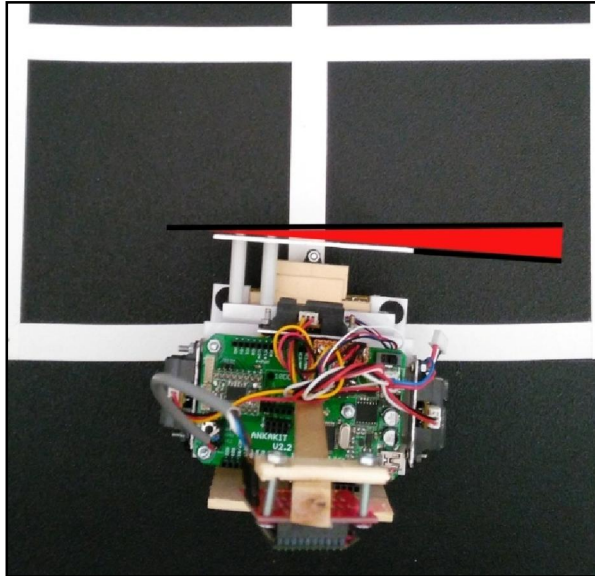
Şekil 6.5. Öncü ve Destek robotunun birlikte yük'ü itmesi.

Destek robotu Öncü robotunun yardım istediği koordinata 3 kez hatalı bir şekilde geldiği için bu deneyde başarı %70 dir. Deneyde Öncü robotu alanı taradıktan sonra (Şekil 6.4) yük'e yönelmekte ve yük'ü taşımaya çalışmaktadır. Robot üzerinde bulunan yük hücresiyle yük'ün taşıyabileceği ağırlıkta olup olmadığı kontrol edip taşıyamayacağı ağırlıktaysa Destek robotunu çağırılmaktadır. Şekil 6.5 de verildiği gibi Öncü robot yük robotun taşıyamayacağı ağırlıkta olduğu için Destek robotunu yardıma çağırılmıştır. Robotların gerilim değeri ve görevi tamamlama yüzdeleri Çizelge 6.3 ve Çizelge 6.4 de verilmiştir.

## 6.4. DENEY SONUÇLARI

### 6.5.1. Yüke Yaklaşamama Hatası

Yük'ü yaklaşma, yük'ün bulunduğu koordinatta yükle aynı açıda bulunmasıdır. Robot tarama işlemini bitirdikten sonra yüke yaklaşma görevini gerçekleştirmektedir. Bu görevi gerçekleştirmeden önce koordinat düzleminde doğru açıda bulunması gerekmektedir. Eğer tarama işlemi bittikten sonra koordinat düzleminde doğru açıda bulunamıyorsa %80 oranında bir sonraki görevi gerçekleştirememektedir. Yük'e yaklaşma hatasının sebebi tarama görevinden sonraki açı hatasıdır. Şekil 6.6 da tarama görevi bittikten sonraki hatalı konumu verilmektedir.

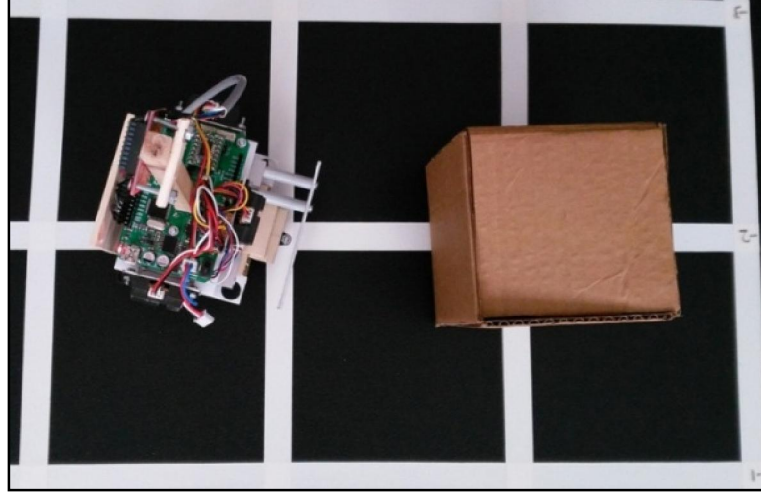


Şekil 6.6. Tarama görevinden sonraki hatalı açısal duruş.

Yük'e yaklaşma görevi tamamlanamadığında yük'ü taşıma görevi ve koordinat güncelleme görevini tamamlayamamaktadır. Şekil 6.7 de yük'e yaklaşma hatası verilmiştir. Robotun tarama görevinden sonra konumunun doğru olmamasını etkileyen etmenleri;

- Enkoderler motorlara bağlı olduğu için tekerlek kaymalarındaki hataların hesaplanamaması,

- Konum bilgisinin bir önceki konum bilgisinin doğruluğu kabul edilerek hesaplanmasından kaynaklanıyor olması
- Enkoder okumada kullanılan timer interrupt'larının aynı çözünürlükte olmamasıdır.



Şekil 6.7. Yük'e yaklaşma hatası.

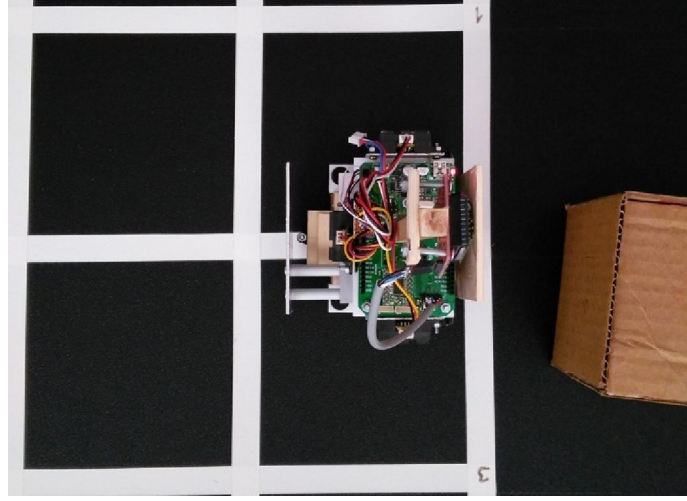
### 6.5.2. Yükü Taşıma Görevini Tamamlaması

Yük'ü taşıma görevi, yük'ün tamamını B alanına taşımaktır. Yük'ü taşıma görevinin tamamlanamamasının iki nedeni vardır. Birinci neden alanın başarılı bir şekilde taranamamasıdır, ikinci neden robotun Şekil 6.6 verildiği konumda olmasından dolayı yük'ü tam olarak taşıyamıyor olması, teğet geçmesi ya da tamamen farklı bir açıda bulunuyor olmasıdır.

### 6.5.3. Koordinat Güncelleme Görevi

Robot yük'ü alan dışına taşıdıktan sonra tekrar alana dönerek Y ekseninde kutunun bulunduğu eksen, X ekseninde maksimum eksen noktasının bir gerisine gelmesidir. Bu görev yük'e yaklaşma görevinde yük'e normalden daha uzakta olduğunda gerçekleşmektedir. Bunun nedeni yük'ün bulunduğu koordinatın bir eksiği olan koordinatta bulunması gerekirken, bir eksiği ile iki eksiği koordinatı arasında bulunuyor olmasıdır. Robotun o bölgede bulunması ise yanlış açı bilgisiyle yük'e yaklaşmasından kaynaklanmaktadır.

Koordinat gncelleme hatasında robot Őekil 6.8 de verildiđi gibi olması gereken koordinatın gerisinde bulunmaktadır. Bu hata ikinci yk'e giderken yanlış hesaplamalara neden olduđu için, hata her defasında daha fazlalařmaktadır. Bu durum robotun konumunda sapmalara neden olduđu için grevi bařarıyla tamamlayamamaktadır.



Őekil 6.8. Koordinat gncelleme hatası.

## BÖLÜM 7

### SONUÇ VE DEĞERLENDİRME

Bu çalışmada yüklerin konumları sabit tutulup koordinat düzlemindeki çizgileri izleyerek yükleri taşıma yöntemine göre eğri uydurma yöntemi %11 daha az mesafe kat ederek görevi tamamlanmaktadır. Robot'a eklenen yük hücresi sayesinde robot, taşımaya çalıştığı yüke dair bilgiler edinmektedir. Optik ışıklı engel sensörlerindeki ışık dağılımından kaynaklanan hataları gidermek için engel algılama algoritması geliştirilmektedir.

Robot boyutlarının küçüklüğü nedeniyle çizgileri takip etmelerinde karşılaşılan olumsuz etkiler sensörleri robotun önüne ve arkasına yerleştirilerek giderilmektedir. Çizgi üzerinde düz ve istenilen şekilde ilerlemesi PID algoritması ile sağlanmaktadır. Çizgi sayma algoritması ve robotun konumunu algılamadaki kullandığı fonksiyonlar hata yapma olasılığına neden olduğundan bir kamera yardımıyla gerçekleştirilmesi daha başarılı sonuçlar elde edilmesini sağlayacaktır.

Robotlar görevlerini gerçekleştirirken, besleme gerilim değeri ve kullanılan bataryanın deşarj eğrisi ile karşılaştırma yaparak güç yönetim sistemi gerçekleştirilmektedir. Bu sayede robotun taşıma görevini gerçekleştirebilecek enerjisinin olup olmadığı bilgisi elde edilmektedir. Robot şarj durumuna göre çalışma süresi ile yapacağı görevin süresi karşılaştırılarak, görevi mevcut enerjisi ile tamamlayıp tamamlayamayacağı kararını vermektedir.

Motorların aynı gerilimde vermiş oldukları farklı hız verileri nedeniyle robotun düz ilerlemesinde sorun yaşandığı için, daha kaliteli bir motor ve enkoder ile zeminde bulunan çizgiler kaldırılarak, bir kamera yardımıyla aynı sistem daha başarılı olarak gerçekleştirilebilir.

## KAYNAKLAR

1. Saeedi, S. , Paull, L. , Trentini, M. and Li, H. , “Occupancy grid map merging for multiple robot simultaneous localization and mapping”, *International Journal of Robotics & Automation*, Daeduk College, Korea, 149-157 (2015).
2. Saeedi, S. , Paull, L. , Trentini, M. and Li, H. , “Neural network-based multiple robot simultaneous localization and mapping”, *IEEE Transactions on Neural Networks*, 2376-2387 (2011).
3. Yasuda, G., “A distributed autonomous control architecture for synchronization and coordination of multiple robot systems”, *SICE Annual Conference*, Akita, Japan, 1864-1869 (2012).
4. Cheng C. , Zhuo, Y. and Kuo, G. “A multiple-robot system for home service”, *CACS International Automatic Control Conference*, Sun Moon Lake, Taiwan, 79-84 (2013).
5. Abdi, H. , Black, T. and Nahavandi, S. , “An adjustable force field for multiple robot mission and path planning”, *Systems, Man, and Cybernetics*, Anchorage, Alaska, 1950-1955 (2011).
6. Yetisenler, C. and Ozkurt, A. , “Multiple robot path planning for robot soccer” *Artificial Intelligence And Neural Networks*, İzmir, Turkey, 11-23 (2005).
7. Dearie, S. and Wedeward, K. “Path planning for a multiple robot system”, *System of Systems Engineering*, Albuquerque, New Mexico, 1-6 (2009).
8. Nagarajan, T. and Thondiyath, A. “An algorithm for cooperative task allocation in scalable, constrained multiple robot systems”, *Intelligent Service Robotics*, Berlin, Germany, 221-233 (2014).
9. Cabreira, T. , Aguiary, M. and Dimuro, G. “An extended evolutionary learning approach for multiple robot path planning in a multi-agent environment”, *IEEE*, Cancun, Mexico, 3363-3370 (2013).
10. Russel, S. and Norvig, P. “Artificial Intelligence a Modern Approach”, Second Edition, *Prentice Hall*, US, 97 (2003).
11. Goryca, J. and Hill, R. “Formal synthesis of supervisory control software for multiple robot systems”, *American Control Conference*, Washington DC, USA, 125-131 (2013).
12. Şabanoviç, A. ve Yannier, S. “Robotlar: Sosyal Etkileşimli Makineler”, *Sabancı*

- Üniversitesi*, İstanbul, Türkiye, 1 (2003).
13. Eşmen, S. “Endüstriyel Mobil Robotlar”, Lisans Tezi, *Karabük Üniversitesi*, Karabük, Türkiye, 13 (2015).
  14. Ersöz, H. “Endüstriyel Robotlar ve Uygulama Alanları”, Yüksek Lisans Tezi, *Gazi Üniversitesi*, Ankara, Türkiye, 15-16 (2007).
  15. Çelik, S. “Robot yardımlı laparoskopik cerrahide hemşirenin rolü”, *Yeni Tıp Dergisi*, Zonguldak, Türkiye, 83-85 (2011).
  16. Turduev, M. “Biyolojik tabanlı eniyileme yöntemleri kullanılarak çoklu robotlar ile bilinmeyen bir ortamın gerçek zamanlı kimyasal gaz yoğunluğu haritalanması”, Yüksek Lisans Tezi, *TOBB Ekonomi ve Teknoloji Üniversitesi*, Ankara, Türkiye, 2 (2010).
  17. Çayırpunar, Ö. “Çoklu Robot Sistemlerinde Robotlar Arası Haberleşme Ve İşbirliği Kullanarak Arama Verimliliğinin Arttırılması”, Yüksek Lisans Tezi, *TOBB Ekonomi ve Teknoloji Üniversitesi*, Ankara, Türkiye, 1 (2009).
  18. Bektaş, B. “Rfid Ve Xbee Tabanlı Depo Yönetim Sistemi Tasarımı ve Gerçekleştirilmesi”, Yüksek Lisans Tezi, *Marmara Üniversitesi*, İstanbul, Türkiye, 60 (2014).
  19. Bakça, F. “Kesir Dereceli PID-Bulanık Mantık Denetleyici ve Parametrelerinin Adaptif Metodla Ayarlanması”, Yüksek Lisans Tezi, *İnönü Üniversitesi*, Malatya, Türkiye, 6 (2014).
  20. Arslan, O. “Zigbee ile Bina İçin Güvenlik Otomasyon Sistemi”, Lisans Tezi, *İstanbul Üniversitesi*, İstanbul, Türkiye, 10-13 (2009).

## ÖZGEÇMİŞ

Emrah BUDAK 1987 yılında Sivas'da doğdu, ilk ve orta öğretimini Ankara'da tamamladı. 2005 yılında Dikmen Teknik ve Endüstri Meslek Lisesi, Elektronik Bölümü'nden mezun oldu. 2011 yılında Gazi Üniversitesi, Teknik Eğitim Fakültesi, Elektronik Öğretmenliği Bölümünü bitirdi. 2012 yılında Karabük Üniversitesi, Mühendislik Fakültesi, Elektrik-Elektronik Mühendisliği Bölümü'nde yüksek lisans öğrenimine başladı, halen aynı yerde öğrenim göremeye devam etmektedir. Altta, yazarın adresi bilgileri bulunur.

### **Adres Bilgileri**

Adres : Harbiye Mah. Nakış Sok. 7/5 Dikmen - ANKARA  
Tel : 553 324 4605  
e-posta : budakemrh@gmail.com