

**OTOMATİK KILAVUZLU ARAÇLARDA EŞ ZAMANLI KONUM
BELİRLEME VE YÖNLENDİRME**

YÜKSEK LİSANS TEZİ

Hasan SANCAK

**Danışman
Dr. Öğr. Üyesi Fatih BAYRAM**

MAKİNE MÜHENDİSLİĞİ ANABİLİM DALI

Eylül 2020

AFYON KOCATEPE ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

YÜKSEK LİSANS TEZİ

OTOMATİK KILAVUZLU ARAÇLARDA EŞ ZAMANLI KONUM
BELİRLEME VE YÖNLENDİRME

Hasan SANCAK

Danışman

Dr. Öğr. Üyesi Fatih BAYRAM

MAKİNE MÜHENDİSLİĞİ ANABİLİM DALI

Eylül 2020

TEZ ONAY SAYFASI

Hasan SANCAK tarafından hazırlanan “Otomatik kılavuzlu araçlarda eş zamanlı konum belirleme ve yönlendirme” adlı tez çalışması lisansüstü eğitim ve öğretim yönetmeliğinin ilgili maddeleri uyarınca 11/09/2020 tarihinde aşağıdaki jüri tarafından oy birliği ile Afyon Kocatepe Üniversitesi Fen Bilimleri Enstitüsü **Makine Mühendisliği Anabilim Dalı’nda YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Danışman : Dr. Öğr. Üyesi Fatih BAYRAM

Başkan : Prof. Dr. Osman TORUN
Afyon Kocatepe Üniversitesi, Teknoloji Fakültesi İmza

Üye : Dr. Öğr. Üyesi Fatih BAYRAM
Afyon Kocatepe Üniversitesi, Teknoloji Fakültesi İmza

Üye : Dr. Öğr. Üyesi Barış GÖKÇE
Necmettin Erbakan Üniversitesi, Müh. ve Mim. Fakültesi İmza

Afyon Kocatepe Üniversitesi
Fen Bilimleri Enstitüsü Yönetim Kurulu’nun
...../...../..... tarih ve
..... sayılı kararıyla onaylanmıştır.

.....
Prof. Dr. İbrahim EROL
Enstitü Müdürü

BİLİMSEL ETİK BİLDİRİM SAYFASI
Afyon Kocatepe Üniversitesi

Fen Bilimleri Enstitüsü, tez yazım kurallarına uygun olarak hazırladığım bu tez çalışmada;

- Tez içindeki bütün bilgi ve belgeleri akademik kurallar çerçevesinde elde ettiğimi,
- Görsel, işitsel ve yazılı tüm bilgi ve sonuçları bilimsel ahlak kurallarına uygun olarak sunduğumu,
- Başkalarının eserlerinden yararlanılması durumunda ilgili eserlere bilimsel normlara uygun olarak atıfta bulunduğumu,
- Atıfta bulunduğum eserlerin tümünü kaynak olarak gösterdiğimi,
- Kullanılan verilerde herhangi bir tahrifat yapmadığımı,
- Ve bu tezin herhangi bir bölümünü bu üniversite veya başka bir üniversitede başka bir tez çalışması olarak sunmadığımı

beyan ederim.

11/09/2020

Hasan SANCAK

ÖZET

Yüksek Lisans Tezi

OTOMATİK KILAVUZLU ARAÇLARDA EŞ ZAMANLI KONUM BELİRLEME VE YÖNLENDİRME

Hasan SANCAK

Afyon Kocatepe Üniversitesi

Fen Bilimleri Enstitüsü

Makine Mühendisliği Anabilim Dalı

Danışman: Dr. Öğr. Üyesi Fatih BAYRAM

Başta imalat sektörü olmak üzere otomotiv sektörü, perakendecilik sektörü, sağlık sektörü vb. sektörlerin tamamında ürünleri bir noktadan başka bir noktaya sevkiyatı söz konusu ise otomatik kılavuzlu araçlar kullanılmaktadır. Otomatik kılavuzlu araçlar öncelikle personel ve zaman tasarrufu sağlaması, güvenli taşıma yapabilmesi, düşük işletme maliyetlerine sahip olması ve işletmenin verimliliğini artırması gibi avantajlar sağlamaktadırlar. Birden fazla otomatik kılavuzlu araç sisteminin kullanıldığı ortamlarda araçların aynı ya da farklı hatlar üzerinde hareket ederken meydana gelebilecek olumsuz durumların önüne geçmek için araçların anlık olarak konumlarının belirlenmesi gerekmektedir.

Bu çalışmada, konum belirleme ve yönlendirme işlemi için prototip bir araç geliştirilmiştir. Araç üzerindeki farklı sensörler yardımıyla toplanan veriler bir algoritma ile değerlendirilmiş ve aracın konum bilgisi geliştirilen uygulama yazılımında gerçek zamanlı olarak takip edilmiş ve yönlendirme yapılmıştır. Geliştirilen prototip araç ve kontrol yazılımı ile otomatik kılavuzlu araçların çarpışmasını engellemek, trafik sıkışıklığını düzenlemek, aracı en yakın şarj istasyonuna yönlendirmek gibi ileride yapılabilecek çalışmalar için de bir altyapı oluşturulmuştur.

2020, xii + 98 sayfa

Anahtar Kelimeler: Otomatik Kılavuzlu Araç, Konum Belirleme, Yönlendirme, Rota

ABSTRACT
M.Sc Thesis

**REAL TIME POSITION LOCALIZATION AND GUIDANCE ON AUTOMATED
GUIDED VECHILES**

Hasan SANCAK

Afyon Kocatepe University

Graduate School of Natural and Applied Sciences

Department of Mechanical Engineering

Supervisor: Asst. Prof. Fatih BAYRAM

The automotive industry, the retail industry, the health sector, and so on. In all sectors, automatic guided vehicles are used if the products are shipped from one point to another. Automatic guided vehicles provide advantages such as saving personnel and time, being able to transport safely, having low operating costs and increasing the efficiency of the business.

In environments where more than one automatic guided vehicles system is used, it is necessary to determine the positions of the vehicles momentarily in order to prevent adverse situations that may occur while moving on the same or different lines.

In this study, a prototype tool has been developed for the positioning and orientation process. The data collected with the help of different sensors on the vehicle were evaluated with an algorithm and the location information of the vehicle was tracked and directed in real time in the developed application software. With the developed prototype vehicle and control software, an infrastructure has been created for future work such as preventing collisions with automatic guided vehicles, regulating traffic jams and directing the vehicle to the nearest charging station.

2020, xii + 98 pages

Keywords: Positioning and Routing, Automatic Guided Vehicle, Route Tracking, Route

TEŐEKKÜR

Bu arařtırmanın konusu, deneysel alıřmaların ynlendirilmesi, sonuların deęerlendirilmesi ve yazımı ařamasında yapmıř olduęu byk katkılarında dolay tez danıřmanım Sayın Dr. ęr. yesi Fatih BAYRAM 'a, arařtırma ve yazım sresince yardımlarını esirgemeyen Sayın Dr. ęr. yesi Gray SONUGR 'e, olumlu ynlendirmeleriyle deęerli vaktini ayıran Sayın Prof. Dr. İbrahim Hakkı CİĖERCİ 'ye, her konuda neri ve eleřtirileriyle yardımlarını grdęm hocalarıma ve emeęi geen tm arkadařlarıma teőekkr ederim.

Bu arařtırma boyunca manevi desteklerini esirgemeyen bařta eřim İlknur SANCAK olmak zere tm aileme teőekkr ederim.

Hasan SANCAK
AFYONKARAHİSAR 2020

İÇİNDEKİLER DİZİNİ

	Sayfa
ÖZET	i
ABSTRACT	ii
TEŞEKKÜR	iii
İÇİNDEKİLER DİZİNİ.....	iv
SİMGELER ve KISALTMALAR DİZİNİ	viii
ŞEKİLLER DİZİNİ	ix
ÇİZELGELER DİZİNİ.....	xii
1. GİRİŞ.....	1
2. LİTERATÜR BİLGİLERİ	3
2.1 Literatürde Yapılan Çalışmalar.....	3
2.2 Otomatik Kılavuzlu Araçların Tanımı.....	7
2.3 Otomatik Kılavuzlu Araçların Tarihsel Gelişimi.....	7
2.4 Otomatik Kılavuzlu Araçların Türleri	9
2.4.1 Lazer Güdümlü Otomatik Kılavuzlu Araçlar.....	9
2.4.2 Sabit Yol Otomatik Kılavuzlu Araç Sistemleri.....	10
2.4.3 Serbest Seri Otomatik Kılavuzlu Araç Sistemleri.....	10
2.4.4 Kamera Yönlendirmeli Otomatik Kılavuzlu Araçlar	11
2.4.5 Optik Güdümlü Otomatik Kılavuzlu Araçlar.....	11
2.4.6 Atalet Kılavuzlu Otomatik Kılavuzlu Araç Sistemleri	11
2.4.7 Diğer Otomatik Kılavuzlu Araç Sistemleri.....	12
2.5 Otomatik Kılavuzlu Araçların Faydaları	12
2.6 Konum Belirlemenin Önemi.....	13
2.7 Konum Belirleme Yöntemleri	14
2.7.1 Lazer Üçgenleme Navigasyon Sistemleri	14
2.7.2 Ataletsel Navigasyon Sistemleri	15

2.7.3 Manyetik Bant Navigasyon Sistemleri	16
2.7.4 Doğal Özellikli Navigasyon Sistemleri.....	16
2.7.5 Tel Navigasyon Sistemleri	17
2.7.6 Optik Navigasyon Sistemleri	17
3. MATERYAL ve METOT	18
3.1 Otomatik Kılavuzlu Araç.....	19
3.1.1 Araba Şasesi	19
3.1.2 Redüktörlü Enkoderli DC Motor	20
3.1.2.1 DC Motorların Çalışma Prensipleri.....	21
3.1.2.2 DC Motorlarının Kısımları ve Özellikleri	22
3.1.2.3 DC Motorlarda Yön ve Hız Kontrolü.....	24
3.1.2.4 DC Motor Çeşitleri.....	26
3.1.2.5 DC Motor Enkoder Sistemi	27
3.1.3 Kontrol Kartı	28
3.1.3.1 Kontrol Kart Yapısı	29
3.1.3.2 Kontrol Kartı Pin Açıklamaları ve Özellikleri	29
3.1.4 Kızılötesi Sensör	31
3.1.5 Bluetooth Modülü	33
3.1.6 Motor Sürücü Kartı	34
3.1.7 İvme ve Gyroscope Sensörü.....	35
3.1.8 Lityum Polimer Batarya	36
3.1.9 Tekerlekler	37
3.2 Aracın Takip Edeceği Yol	38
3.3 Bilgisayar	39
3.4 Bilgisayar Yazılımları.....	40
3.4.1 Visual C Sharp Programlama Dili.....	40

3.4.2 Arduino IDE.....	41
3.5 Sistemin Çalışma Prensipleri	41
3.5.1 Çizgi Takip Etme Sistemi	42
3.5.2 İvme Hesaplama Sistemi.....	43
3.5.3 Oransal Integral Türev (PID) Kontrolü.....	47
3.5.4 Mesafe ve Dönme Açısı Kontrolü.....	49
4. BULGULAR	52
4.1 Uygulama Adımları	52
4.2 PID Kontrol Uygulama Adımları	56
4.3 QTR-8RC Sensörlerinden Okunan Değerlere Göre Yön Tayini İşlemi	58
4.4 Yön Tayini Arduino Program Kodları.....	62
4.5 Araç Bir Tur Attığında Alınan Değerler.....	63
4.5.1 QTR-8RC Sensör Değerleri	64
4.5.1.1 QTR-8RC Sensör Position Değerleri	64
4.5.1.2 Sensör Anlık Değerleri ve Position Değerleri – Kalibrasyon	66
4.5.2 MPU 6050 Sensöründen Alınan Değerler.....	68
4.5.3 Sağ ve Sol Motor Enkoder Değerleri ve Dönme Açılımları	74
4.6 Uygulama Yazılımı Arayüz Ekranı	76
4.6.1 Bağlantı GroupBox Kontrol Elemanı.....	77
4.6.2 Çizgi Takibi Başlat Buton Elemanı	77
4.6.3 Araç Kontrol GroupBox Kontrol Elemanı.....	78
4.6.4 QTR-8RC GroupBox Kontrol Elemanı	78
4.6.5 Gelen Veriler GroupBox Kontrol Elemanı	79
4.6.6 Konum Bilgisi GroupBox Kontrol Elemanı	80
5. TARTIŞMA ve SONUÇ	81
6. KAYNAKLAR.....	83

ÖZGEÇMİŞ	89
EKLER.....	90

SİMGELER ve KISALTMALAR DİZİNİ

Simgeler

Mm	Milimetre
F	Kuvvet
r	Yarıçap
L	Uzunluk
M	Kütle
s	Saniye
Hz	Hertz
N	Newton
E	Elastisite Modülü
I	Atalet Momenti
V	Volt
mA	Miliamper
A	Amper
Mhz	MegaHertz

Kısaltmalar

AGV	Otomatik Kılavuzlu Araç
AS/RS	Otomatik Stoklama ve Geri Alma Sistemleri
CAD	Bilgisayar Destekli Tasarım
CPR	Devir Başına Döngü
ISO	Uluslararası Standartlar Örgütü
LED	Işık Yayan Diyot
LKA	Lazer Kılavuzlu Araçlar
PLC	Programlanabilir Mantıksal Denetleyici
PWM	Sinyal Genişlik Modülasyonu
S/R	Yükleme/Boşaltma
USB	Evrensel Seri Veriyolu

ŞEKİLLER DİZİNİ

	Sayfa
Şekil 2.1 İlk Otomatik kılavuzlu araç, 1953 Arthur Barret Jr (İnt. Kyn.8).	8
Şekil 2.2 Volvo'nun 1973 yılında geliştirdiği otomatik kılavuzlu araçlar (İnt. Kyn.9)...	8
Şekil 2.3 Lazer güdümlü sistem (İnt. Kyn.12).	10
Şekil 2.4 Manyetik bant sistemi ve Manyetik bant sensörü (İnt. Kyn.14).	10
Şekil 2.5 Kamera yönlendirmeli bir sistem (İnt. Kyn.16).	11
Şekil 2.6 Atalet kılavuzlu sistem (İnt. Kyn.11).	12
Şekil 2.7 Lazer üçgenlemeli sitemi (İnt. Kyn.17).	15
Şekil 2.8 Manyetik sensör algılama (a), Yalpa açısı hesaplama (b) (Yin vd. 2018).	15
Şekil 2.9 Manyetik bant ve sensör modeli (Wu vd. 2014).	16
Şekil 2.10 Doğal özellikli navigasyon sistemi için haritalar (İnt. Kyn.17).	16
Şekil 2.11 Tel navigasyon sistemi (İnt. Kyn.15).	17
Şekil 2.12 Optik navigasyonlu sistem (İnt. Kyn.13).	17
Şekil 3.1 Sisteminin çalışma şeması.	18
Şekil 3.2 Rapid Robot araba şasisi (İnt. Kyn. 44).	20
Şekil 3.3 Tork, devir ve güç ilişkisi (İnt. Kyn.34).	21
Şekil 3.4 DC motorun çalışma prensibi (İnt. Kay. 55).	22
Şekil 3.5 DC dış kesit görünümü (İnt. Kyn. 24).	22
Şekil 3.6 Endüvi (Rotor) kesiti (İnt. Kyn. 25).	23
Şekil 3.7 DC motor stator görünümleri (İnt. Kyn. 26).	23
Şekil 3.8 Fırça ve kolektör kesitleri (İnt. Kyn. 26).	24
Şekil 3.9 DC motor devir yönü değiştirme H köprü devresi (İnt. Kyn. 27).	25
Şekil 3.10 PWM görev döngüsü ortalama voltaj değerleri (İnt. Kyn. 58).	25
Şekil 3.11 12 Volt 1200 RPM Redüktörlü Enkoderli DC Motor (İnt. Kyn. 45).	26
Şekil 3.12 Optik sistem enkoder yapısı (İnt. Kyn. 61).	27
Şekil 3.13 Arduino Uno bağlantı noktaları diyagramı (İnt. Kyn. 29).	29
Şekil 3.14 Arduino Uno rev. 3 mikro denetleyici kontrol kartı (İnt. Kyn. 46).	31
Şekil 3.15 QTR-8RC Kızılötesi sensörü (İnt Kyn. 47).	32
Şekil 3.16 QTR-8RC – arduino uno bağlantı şekli.	32
Şekil 3.17 HC-06 modül ve arduino uno bağlantı şekli.	33
Şekil 3.18 ArduMoto L298P motor sürücü kartı ve arduino uno bağlantı şekli.	35
Şekil 3.19 MPU6050 sensörü ve arduino uno bağlantı şekli.	36
Şekil 3.20 500 mA Lityum polimer batarya (İnt. Kyn. 51).	37

Şekil 3.21 Bond silikon tekerlek ve motor bağlantı şekli (İnt Kyn. 39).	37
Şekil 3.22 Plastik sarhoş mini tekerlek (İnt. Kyn. 52).	38
Şekil 3.23 Otomatik kılavuzlu aracın takip edeceği yol güzergâhı.	39
Şekil 3.24 İvmeölçer hesabı için koordinat sistemi (İnt. Kyn.31).	43
Şekil 3.25 İvmeölçer hesabı için R vektörünün eksenlere yaptığı açı (İnt. Kyn.31).	44
Şekil 3.26 Koordinat sisteminde R vektörü için izdüşümleri (İnt. Kyn.31).	45
Şekil 3.27 Aracın X,Y,Z eksenleri için roll, pitch ve yaw hareketleri (İnt. Kyn.40).	47
Şekil 3.28 PID Kontrol blok şeması (İnt. Kyn.35).	48
Şekil 3.29 PID Kontrol, Orantısal Kazanç Kp grafiği (İnt. Kyn.36).	48
Şekil 3.30 PID Kontrol, İntegral Kazanç Ki grafiği (İnt. Kyn.36).	48
Şekil 3.31 PID Kontrol, Türev Kazanç Kd grafiği (İnt. Kyn.36).	49
Şekil 4.1 Araç şasesine motor ve tekerleklerin sabitlemesi.	52
Şekil 4.2 Şaseye sensör, tekerlek, bluetooth modülleri ve bataryanın sabitlemesi.	52
Şekil 4.3 Motor sürücü kartı ve motor bağlantılarının yapılması.	53
Şekil 4.4 Kablo bağlantılarının yapılması.	53
Şekil 4.5 PD parametreleri program kodu.	57
Şekil 4.6 PID yapısı ile hız farkı hesaplama program kodu.	57
Şekil 4.7 QTR-8RC kızılötesi sensör pozisyon değerleri – 1 grafiği.	60
Şekil 4.8 QTR-8RC kızılötesi sensör pozisyon değerleri – 2 grafiği.	61
Şekil 4.9 QTR-8RC kızılötesi sensör değeri okuma fonksiyonu.	62
Şekil 4.10 Hız farkları tespiti için kullanılan Line_Tracking fonksiyonu.	62
Şekil 4.11 Hız farkları için kullanılan M_CONTROL fonksiyonu.	63
Şekil 4.12 QTR-8RC kızılötesi sensör position değerleri ve hata miktarları.	64
Şekil 4.13 QTR-8RC kızılötesi sensör position değerleri ve motor farkları.	65
Şekil 4.14 QTR-8RC kızılötesi sensör kalibrasyon maksimum ve minimum değerleri.	66
Şekil 4.15 Kalibrasyon için kullanılan Manuel_Calibration fonksiyonu.	67
Şekil 4.16 Anlık sensör ve position değerleri için kullanılan Loop fonksiyonu.	67
Şekil 4.17 QTR-8RC kızılötesi anlık sensör ve position değerleri.	68
Şekil 4.18 MPU 6050 ivme sensörü YAW açı değerleri.	68
Şekil 4.19 YAW açı değeri hesaplaması arduino program kodu(İnt. Kyn. 56).	69
Şekil 4.20 YAW açı değerine göre konum tahmini -1.	70
Şekil 4.21 YAW açı değerine göre konum tahmini -2.	71
Şekil 4.22 YAW açı değerine göre konum tahmini -3.	71
Şekil 4.23 YAW açı değerine göre konum tahmini -4.	71

Şekil 4.24 YAW aç1 deęerine göre konum tahmini -5.....	72
Şekil 4.25 YAW aç1 deęerine göre konum tahmini -6.....	72
Şekil 4.26 YAW aç1 deęerine göre konum tahmini -7.....	72
Şekil 4.27 YAW aç1 deęerine göre konum tahmini -8.....	73
Şekil 4.28 YAW aç1 deęerine göre konum tahmini -9.....	73
Şekil 4.29 Aracın hareket yönü ve dönme noktaları.....	73
Şekil 4.30 Sağ ve Sol Enkoder Pulse Deęerleri ve Dönme Bölgeleri.....	74
Şekil 4.31 Dönme bölgelerinde sağ ve sol motor dönme aç1ları.....	75
Şekil 4.32 Uygulama Programı arayüz ekranı.....	76
Şekil 4.33 Baęlan butonuna tıklanması.....	77
Şekil 4.34 Çizgi Takibi Başlat butonuna tıklanması.....	78
Şekil 4.35 Kızılötesi sensör position deęerine göre label kontrolleri görünümü.....	79
Şekil 4.36 Gelen veriler groupbox kontrol elemanı.....	79
Şekil 4.37 Konum bilgisi groupbox kontrol elemanı.....	80

ÇİZELGELER DİZİNİ

	Sayfa
Çizelge 4.1 ArduMoto - DC Motor bağlantıları.....	54
Çizelge 4.2 Arduino-ArduMoto pin bağlantıları.....	54
Çizelge 4.3 MPU6050-Arduino pin bağlantıları.....	54
Çizelge 4.4 QTR-8RC-Arduino pin bağlantıları.....	55
Çizelge 4.5 HC-06 Bağlantıları.....	55
Çizelge 4.6 Enkoder Bağlantıları.....	56
Çizelge 4.7 QTR-8RC kızılötesi sensör pozisyon değerleri – 1.....	59
Çizelge 4.8 QTR-8RC kızılötesi sensör pozisyon değerleri – 2.....	61

1. GİRİŞ

Teknolojinin gelişmesine paralel olarak insanların hızla artan ihtiyaçları karşısında bu ihtiyaçlara cevap verebilmek çok çeşitli teknolojik aletler ve araçlar geliştirilmektedir. Günümüzde hayatın her alanına girmeye başlayan robotlar ve yapay zekâ ürünleri teknolojik gelişmişliğin bir göstergesi haline gelmiştir. Ülkeler her gün çeşitli özelliklerde robotlar geliştirmekte, firmalar teknolojiye bir adım öne geçmek için değişik fikirler uygulamaktadırlar. Otomatik kılavuzlu araçların tarihçesine bakıldığında da benzer bir durum ortaya çıkmaktadır.

1953 yılında inşa edilen ilk otomatik kılavuzlu araç sisteminden bu zamana kadar çok çeşitli tiplerde otomatik kılavuzlu araç sistemleri geliştirilmiştir. otomatik kılavuzlu araç sistemleri başlangıçta ilk olarak depolama sistemleri için geliştirilmiştir. Ancak sonraki yıllarda birçok sektörde farklı amaçlar için kullanılmak üzere otomatik kılavuzlu araç sistemleri geliştirilmiştir. Bu sistemlere örnek olarak ortama lazer tarayıcılar yerleştirilerek aracın rotasını takip etmesi için geliştirilmiş lazer güdümlü otomatik kılavuzlu araç sistemleri, yine rota takibi için aracın hareket ettiği yüzeye gömülü tel, manyetik bant ya da renkli boya vasıtasıyla hareket sağlanan sabit yol otomatik kılavuzlu araç sistemleri verilebilir.

Otomatik kılavuzlu araç sistemlerinde konum belirleme işlemini gerçekleştirmek için çeşitli yöntemler kullanılmaktadır. Lazer tarayıcıların kullanıldığı lazer üçgenleme sistemi, jiroskop kullanılan ataletsel konum belirleme sistemi, manyetik bantlı konum belirleme sistemi ve çalışma alanının referans görüntülerinin aracın hafızasına kaydedildiği doğal özellikli konum belirleme sistemi önemli konum belirleme sistemleridir.

Farklı yöntemler kullanan otomatik kılavuzlu araç sistemleri günümüzde tüm sektörler kullanılmaktadır. İmalat sektöründe hammadde, yarı mamül ve mamül sevkiyatları otomatik kılavuzlu araç sistemleri ile yapılmaktadır. Benzer şekilde otomotiv sektörü, e-ticaret, kimya endüstrisi, sağlık sektörü, perakendecilik sektörü, gıda sektörü ve benzeri gibi sektörlerin tamamında ürünleri bir noktadan başka bir noktaya sevkiyatı söz konusu ise otomatik kılavuzlu araç sistemleri tercih edilmektedir.

Otomatik kılavuzlu araç sistemleri öncelikle personel ve zaman tasarrufu sağlaması, sistemin üstünde bulunan çeşitli sensörler vasıtasıyla güvenli taşıma işlemi gerçekleştirilmesi, düşük işletme maliyetlerine sahip olması, uzaktan kontrollü serbest dolaşım ya da sabit kılavuz ile hareket ettirilebilmeleri ve işletmenin verimliliğini artırması gibi avantajlar sağlamaktadır.

Bu çalışmada özellikle üretim merkezleri, imalata yönelik firmalar, lojistik ve depolama sistemleri için gerekli olduğuna inanılan insansız bir aracın belirli bir rotayı takip ederken eş zamanlı olarak nerede olduğunu bulabilmek ve aracı yönlendirip istenilen hedef noktalara götürebilmek için bir araç tasarımı ve programlanması üzerine durulmaktadır. Aracın hareket halindeyken eş anlı olarak nerede olduğunu bulabilmek için çeşitli sensörlerden ve kodlayıcılardan faydalanılarak aracın konumu tespit edilmiştir. Bunun için bir zemine çizilmiş yolda hareket ederken eş zamanlı olarak aracın yaptığı hareket uygulama programında görüntülenmiştir. Çalışmada eş zamanlı konum belirleme yapılması hedeflendiği için aracın hareket ettiği güzergâhın da yine sensörlerden alınan veriler sayesinde rotası çıkarılmıştır.

Otomatik kılavuzlu araçların trafik sıkışıklığı, çarpışma gibi olası sorunlarının önüne geçmek için eş zamanlı olarak konumlarının belirlenmesi gerekmektedir. Tasarlanan prototip aracın rota üzerindeki konumunun belirlenebilmesi ile olası otomatik kılavuzlu araç sorunlarının çözümü için bir örnek oluşturulması hedeflenmektedir. Bu prototip araç ile otomatik kılavuzlu araç sistemlerinin eğitimi, sistemin uygulanabilmesi ve bunun yanında çalışma zamanında meydana gelebilecek sorunların en aza indirgenmesi amaçlanmaktadır.

2. LİTERATÜR BİLGİLERİ

2.1 Literatürde Yapılan Çalışmalar

Literatürde yapılan çalışmalara bakıldığında Otomatik kılavuzlu araçların sanayide çok farklı sektörlerde kullanım alanına sahip olduğu ve bu araçların eş zamanlı konumlarını belirlemenin, aracı yönlendirmenin en önemli problemleri olduğu görülmektedir.

Guldner vd. (1995) bir mobil robotun hazırlıksız bir çalışma alanında belirli bir hedef noktaya yönlendirmek için üç katmanlı bir yol kontrol sistemi üzerine çalışmışlardır. Robotun sabit bir noktaya göre konumunu bulmak için algoritma geliştirmişlerdir. Yol kontrolünde planlama, konum belirleme, çarpışmadan kaçınma şeklinde üç modül üzerinde durmuşlardır.

Um vd. (2009) otomatik kılavuzlu araçlı esnek üretim sistemi simülasyonunun tasarım ve analizi çalışmalarında esnek üretim sisteminin otomatik kılavuzlu araçlar vasıtasıyla performansını arttırmak için otomatik kılavuzlu aracın numarası, hızı ve dağıtım kuralı, parça türleri, zamanlama ve tampon boyutları dâhil olmak üzere birçok parametreyi göz önünde bulundurmuşlardır. Bu amaçla çok amaçlı doğrusal olmayan programlama ve evrim stratejisini kullanmışlardır.

Lu vd. (2017) akıllı fabrika ortamında radyo frekanslarıyla otomatik kılavuzlu araçların konumlarını belirleme üzerine yaptıkları çalışmada dairesel alandaki dairesel antenlerle manyetik alandaki otomatik kılavuzlu araçların konumlarının hassasiyeti ve denge üzerinde durmuşlardır. Çalışmalarında sürüş bölgesinde büyük çaplı antenler, park bölgelerinde küçük çaplı antenler kullanmışlardır.

Yang vd. (2018) Otomatik Konteyner Terminallerinde otomatik kılavuzlu araçların yönlendirilmesi için entegre bir programlama üzerine çalışmışlardır. Çalışmada Otomatik Konteyner Terminallerinde ritim vinçlerinin, otomatik kılavuzlu araçların ve bahçe vinçlerinin eş zamanlı programlanması kritik problem olmakla birlikte otomatik kılavuzlu araç yönlendirmesi ve ekipman koordinasyonu için Tıkanıklığı Önleme Kuralına Dayalı Bi-seviye Genetik Algoritmanın modelin çözümü için uygun olduğunu göstermişlerdir

Vis (2006) yaptığı çalışmada otomatik kılavuzlu araçların tasarım ve kontrolüne ilişkin araştırma yapmıştır. Bu çalışmada otomatik kılavuzlu araç sistemlerinin imalat, dağıtım, aktarma ve taşıma sistemlerinde tasarım ve kontrol konularıyla ilgili literatür ele alınmıştır. Büyük otomatik kılavuzlu araç sistemleri için uzun hesaplama sürelerinin, sistemdeki kilitlenme ve gecikmelerin üstesinden gelmek için yeni analitik ve simülasyon modellerinin geliştirilmesi gerekmesi kanısına varılmıştır. Dağıtım, aktarma ve nakliye sistemlerinde otomatik kılavuzlu araç sistemlerinin tasarımı ve kontrolünde daha spesifik araştırma perspektifleri belirlemişlerdir.

Le-Anh vd. (2006) Otomatik kılavuzlu araç sistemlerinin tasarımı ve kontrolü hakkında bir inceleme sunmuşlardır. Çalışmada otomatik kılavuzlu araç sistemlerinin kılavuz yol tasarımı, araç ihtiyaçlarının belirlenmesi, konumlandırma, enerji yönetimi, rota planlama, kilitlenme problemleri gibi birçok kritik problemi ele alınmışlardır.

Galasso vd. (2019) yaptıkları çalışmada dört tekerlekli otomatik kılavuzlu araçların dışsal ve içsel otomatik kalibrasyonu için yeni bir yöntem önermişlerdir. Yöntem Programlanabilir Mantık Denetleyicileri üzerinde uygulanmış ve endüstriyel otomatik kılavuzlu araçlar üzerinde test edilmiştir. Bu yöntemle kalibre edilen otomatik kılavuzlu araçların elde ettiği pozisyon doğruluğu elle kalibre edilenlerden daha yüksek olmuştur.

Ge vd. (2011) eş zamanlı olarak bilinmeyen ortamlardaki yol planlaması sorununu ele almak için Eşzamanlı Yol Planlama ve Topolojik Haritalama (SP2ATM) adlı yeni bir yaklaşım sunmuşlardır. Deney sonuçlarına göre SP2ATM'in karmaşık ve dinamik ortamlarda etkili ve küresel olarak yakınsak olduğunu göstermişlerdir.

Mirkhani vd. (2013) çalışmalarında mobil robotların temel hedeflerinin özerk hareket etmek olduğunu belirtmişler ve bir robotun kendi doğru pozisyonunu ve yönünü bulma yeteneği geliştirmesi için tarama eşleştirme yöntemiyle Harmony algoritmasına dayanan yeni bir yöntem önermişlerdir. Tarama sonuçlarına göre bu algoritmanın Genetik Algoritmaya göre daha kesin sonuçlar ürettiği görülmüştür.

Shah vd. (1997) dinamik araç yolu belirlemesi ve çatışmasız yönlendirme gibi temel özellikleri içeren entegre bir otomatik kılavuzlu araç Sistemi kontrol modelinin geliştirilmesi üzerine çalışmışlardır. Otomatik kılavuzlu araç modelinin nesne yönelimli bir uygulaması önerilmiştir. Bu uygulama metodolojisi hızlı bir değişim ve gelişim yeteneği sağlamıştır.

Kirsch vd. (2011) yaptıkları çalışmada otomatik kılavuzlu araçların ana sorunları (Küresel lokalizasyon, konum takibi, yol planlama ve iletişim) hakkında bilgiler vermiş, küresel yerleştirme tekniğinin doğru konumlandırmayı garanti ettiğini ve sensör füzyonunun yerleştirme uygulamaları için kullanılabileceğini göstermişlerdir.

Yan vd. (2018) Çoklu otomatik kılavuzlu araç sisteminin tasarım, işletim ve bakımını optimize etmek için yeni metodoloji önermişlerdir. Çoklu otomatik kılavuzlu araç sistemindeki arızalı otomatik kılavuzlu araçların geri dönüşüm sürecinin, bakım sahasının yerinin değişmesine bağlı olarak, renkli Petri ağları (CPN'ler) yaklaşımıyla analiz edilmesini uygulamışlardır. Elde edilen simülasyon sonuçları, bakım alanlarının ve bakım stratejilerinin konumunun, birçok otomatik kılavuzlu araç sisteminin performansı üzerinde etkisi görülmüştür.

Awasthi vd. (2009) imalat sistemlerinde kullanılan otomatik kılavuzlu araçların parça seçimi ve rotalama için bir modelleme ve simülasyon geliştirmişlerdir. Sistemin performansını iyileştirmek için parça seçimi ve yönlendirmeye uyulması gereken en iyi kural sırasını belirlemek için deneylerin tasarımı yapılmıştır. Modelin bir uygulaması, varsayımsal otomatikleştirilmiş bir kılavuzlu araç sistemi düzeni kullanılarak gösterilmiştir.

Azenha ve Carvalho (2003) otomatik kılavuzlu araçların konumlandırma sorununu ele almışlardır. Çalışma ultrasonik sensörlere veya elektromanyetik dalga verici / alıcı anten kümelerine dayanmaktadır. Otomatik kılavuzlu araç hareketinin kontrol sisteminin verimliliğini doğrulaması için simülasyon sonuçları ve devam etmekte olan çalışmaların uygulama sorunları sunulmuştur.

Roszkowska (2003) çalışma, çift yönlü kılavuz yol ağı olan kapalı otomatik kılavuzlu araç sistemlerini, tek yönlü araçları, çarpışmaları önlemek için bölge kontrolünü ve dinamik rota planlamasını içerir. Bağlantılı ağa sahip sistemler için olay oluşumunu korumak için optimal bir politika ve ağ yapısının genel durumu için asgari bir politika önermiştir.

Gualda vd (2019) bu çalışmalarında, geniş bir iç mekân alanını kapsayan birden fazla Ultrasonik Yerel Konumlandırma Sisteminin (UYKS) bir Eşzamanlı Kalibrasyon ve Navigasyon (EKN) algoritmasını önermektedirler. Algoritma, Mobil Robot (MR) ile ilgili genel ve yerel yörüngelerini tahmin eden paralel olarak çalışan birden çok filtreye dayanmaktadır. Yöntem, çalışması şartıyla filtre türünden bağımsızdır. Birçok filtreye performanslarını karşılaştırmak maksadıyla test edilmiştir.

Cheong ve Lee (2018) çalışmalarında görüntü algılama ve konumlandırmaya dayalı konsept bir otomatik kılavuzlu araç tasarımı yapmışlardır. Otomatik sürüş konusu ele alınmış otomatik kılavuzlu araçlar için sanayi devriminde imalat fabrikalarında otomatik sürüş kavramının çözümüne yönelik öneride bulunmuşlardır.

Jaiganesh vd. (2014) çalışmalarında kullandıkları araç, özel malzeme taşıma görevini yerine getireceği ve ayrıca özel uygulamalar için kullanacağı özelleştirilmiş bir otomatik kılavuzlu araçtır. Otomatik kılavuzlu aracın başlama ve durma işlemini doğrudan kontrol eden araç hareketini tespit etmek için otomatik kılavuzlu aracın yolunda yakınlık sensörleri kullanılmıştır. Akış yoluna göre, manyetik bant yöntemi bu araca en iyi sonucu almak için en uygun yöntem olarak belirlenmiş akış yolu tüm yönleri dikkate alarak tasarlanmıştır. Daha iyi performans elde etmek için araçta endüktif güç aktarma yöntemleri uygulanmıştır.

Heger ve Voss (2018) çalışmalarında, bir sipariş üzerine üretimde karmaşık zamanlama probleminin küçük bir örneğine en uygun çözümü bulmak için karışık bir tam sayılı doğrusal programlama (DP) formülasyonu sunmuşlardır. Farklı işlere sahip engelleyici bir reentrant işyeri ortamında otomatik kılavuzlu araçlar için en uygun programı

oluşturmaya çalışmışlardır. Sonuç olarak hiçbir teknik uygun bir program oluşturma ihtiyacını karşılamamakla birlikte olası çözümler sunulmuştur.

2.2 Otomatik Kılavuzlu Araçların Tanımı

Otomatik kılavuzlu araçlar, operatöre veya sürücüye ihtiyaç duymadan bir ya da daha fazla bilgisayar ya da kontrol kartıyla kontrol edilebilen tekerlek bazlı yük taşıyıcı mekanik sistemler olarak adlandırılabilir. Otomatik kılavuzlu araçlar yerdeki işaretleri, telleri izleyerek ya da radyo frekansları, lazerleri takip ederek tesis içinde mevcut bir yükü bir noktadan başka bir noktaya taşıyabilen sistemlerdir (İnt.Kyn.6).

"Otomatik güdümlü araç" terimi, sürücüsüz çalışabilen tüm taşıma sistemlerini kapsayan genel bir terimdir. Otomatik kılavuzlu araçlar için en basit tanım; bir tesisteki materyalleri verimli bir şekilde taşımak için kullanılan sürücüsüz bir araç olarak ifade edilebilir (İnt. Kyn.6).

Otomatik kılavuzlu araçların hareketleri bir yazılım ve sensör tabanlı rehberlik sistemlerinin bir araya gelmesiyle yönlendirilir. Kesin olarak kontrol edilen hızlanma ve yavaşlama ile öngörülebilir bir yolda hareket ettiklerinden ve otomatik engel algılama tamponlarını içerdiklerinden, otomatik kılavuzlu araçlar yüklerin güvenli bir şekilde hareket etmesini ve kaynak noktadan hedef noktaya ulaşmasını sağlar (İnt.Kyn.4).

2.3 Otomatik Kılavuzlu Araçların Tarihsel Gelişimi

İlk otomatik kılavuzlu araç sistemi 1953 yılında inşa edilmiştir. Arthur Barret Jr. Bakkal deposundaki bir treyleri, zemindeki bir kabloyu takip eden çekme traktörü vasıtasıyla depodaki ürünlerin taşınmasını otomatik olarak yapmıştır (Şekil 2.1) (İnt.Kyn.3).

Kısa süre sonra, Illinois, Northbrook ve Barrett Electronics, bir depo veya fabrika zeminine gömülü teli takip eden basitçe çekici kamyon olan ilkel bir otomatik kılavuzlu araç sistemini pazarlamaya başladı (İnt. Kyn.4).



Şekil 2.1 İlk Otomatik kılavuzlu araç, 1953 Arthur Barret Jr (İnt. Kyn.8).

Otomatik kılavuzlu araç sistemleri için sonraki büyük gelişme 1973'te İsveç'in Kalmar kentindeki Volvo otomobil montaj fabrikasında çalışan mühendislerin geleneksel konveyör merkezli montaj hattına alternatif olarak senkronize olmayan bir montaj ekipmanı geliştirmesiyle başladı. Sonunda, depoya 280 bilgisayar kontrollü otomatik kılavuzlu araç sistemi kuruldu. Volvo daha sonra birim yük otomatik kılavuzlu araçları diğer otomobil şirketlerine pazarladı (İnt. Kyn.3).

İlk model otomatik kılavuzlu araçlar teller ve bantlar kullanılarak yönlendirildi, ancak bugün çoğunlukla lazerler kullanılarak yönlendirilmektedir. Lazer Kılavuzlu Araçlar (LKA) olarak bilinen bu tip araçlar, ürünlerin depo içerisinde verimli bir şekilde taşınmasını sağlamak için diğer robotlarla bağlantılı şekilde yönlendirilmektedir (Şekil 2.2) (İnt. Kyn.3).



Şekil 2.2 Volvo'nun 1973 yılında geliştirdiği otomatik kılavuzlu araçlar (İnt. Kyn.9).

1970'lerin ortalarında bir birim yük aracının piyasaya sürülmesiyle bu birim yük otomatik kılavuzlu araçları, çeşitli fonksiyonlara hizmet verebilmeleri nedeniyle malzeme taşıma pazarında yaygın kabul görmüştür. Bugün, birçok üretici tarafından üretilen operasyonel yük taşıma araçlarını kullanan yüzlerce sistem oluşturulmuştur (İnt. Kyn.1).

1980'lerde endüstri liderleri bu sistemlere “otomatik kılavuzlu araç” adını verdiler. O zamandan önce bu makineler “sürücüsüz araçlar” olarak biliniyordu. Ayrıca, 80'li yıllarda lazerler ve manyetik bant gibi telsiz kılavuzluk mekanizmaları tanıtıldı. Lazer güdümlü araçlar 1987'de tanıtıldı. 1989 bilgisayar kontrollü otomatik kılavuzlu araçlar üretilmeye başlanmıştır (İnt. Kyn.2).

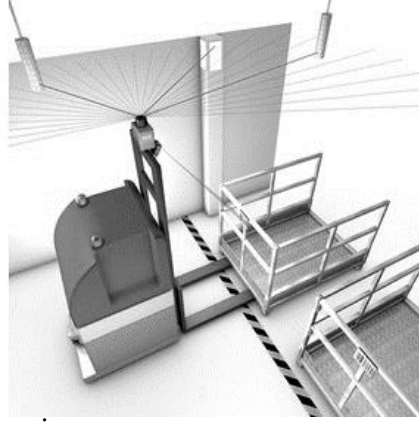
1992'de hem kablolu hem de kablosuz otomatik kılavuzlu araçları yönlendirebilecek ilk sistemler tasarlanmıştır. 2003 yılında, izlediği yolu değiştirebilen otomatik kılavuzlu araçlar geliştirilmiştir. Bugün, otomatik kılavuzlu araçlar her zamankinden daha özerk ve yükleri her zamankinden daha güvenli ve kolay bir şekilde hedefe aktarır pozisyona ulaşmışlardır (İnt. Kyn.10).

2.4 Otomatik Kılavuzlu Araçların Türleri

Bir Otomatik kılavuzlu araç, yapılacak işe bağlı olarak değişik karmaşıklığa sahip önceden programlanmış bir kılavuz sistemi tarafından yönlendirilir. Otomatik kılavuzlu araçların çoğu sabit yol ve serbest menzilli sistemler olarak tasarlanır ancak lazerler tarafından yönlendirilen sistemlerde mevcuttur. Uygulamada kamera güdümlü otomatik kılavuzlu araçlar, optik güdümlü otomatik kılavuzlu araçlar, atalet güdümlü otomatik kılavuzlu araçlar, lazer güdümlü otomatik kılavuzlu araçlar gibi birçok türü mevcuttur.

2.4.1 Lazer Güdümlü Otomatik Kılavuzlu Araçlar

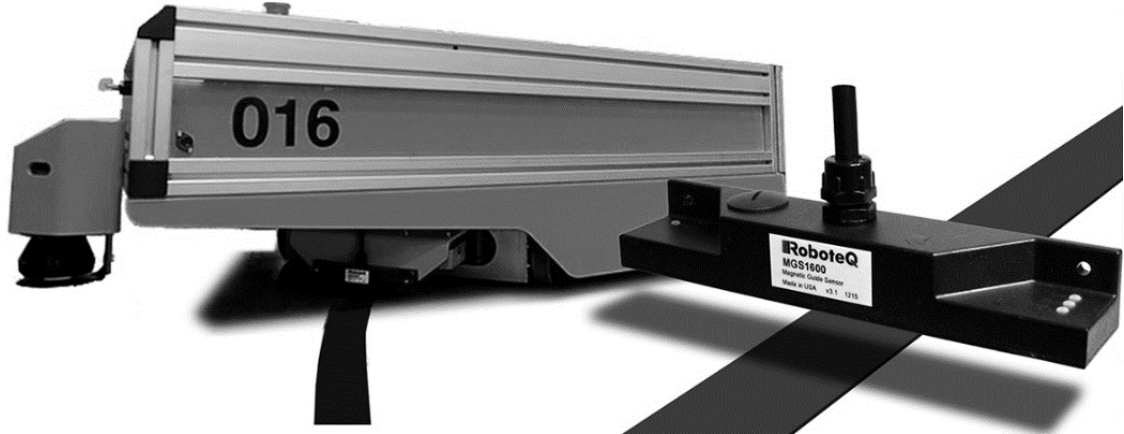
Sistemin düzgün çalışması ve rotasını takip edebilmesi için ortama lazer tarayıcılar yerleştirilmiştir. Tarayıcılar hedefe bir lazer yayar, sonra onu geri yansıtır. Yansımanın geri dönmesi için geçen süreye bağlı olarak, otomatik kılavuzlu araç hedef mesafeyi belirleyebilir. Hedefle olan mesafesine göre kendisini yönlendirir (Şekil 2.3) (İnt. Kyn.12).



Şekil 2.3 Lazer güdümlü sistem (İnt. Kyn.12).

2.4.2 Sabit Yol Otomatik Kılavuzlu Araç Sistemleri

Kılavuz olarak gömülü tel, manyetik bant veya renkli boya kullanan otomatik kılavuzlu araç sistemleridir. En basit otomatik kılavuzlu araç sistemleri bunlardır. Sistem kablonun ya da bantın bulunduğu yolu takip etmesini sağlayan sensörler ve frekanslarla donatılmıştır. Bu basit sistemler, bazı uygulamaların gerektirebileceği esnekliğe izin vermemesine rağmen, oldukça doğru ve güvenilir sistemlerdir (Şekil 2.4) (İnt. Kyn.14).



Şekil 2.4 Manyetik bant sistemi ve Manyetik bant sensörü (İnt. Kyn.14).

2.4.3 Serbest Seri Otomatik Kılavuzlu Araç Sistemleri

Serbest seri sistemler daha esnek ve uyarlanabilir bir seçenektir. Serbest çalışan bir sistemde herhangi bir yol takibi için iz kullanılmaz, otomatik kılavuzlu aracı, yerleşik mikroişlemci ve kontrol sistemiyle iletişim kuran bilgisayar tabanlı bir program kullanarak yönlendirir. Bu programlama kontrollü ve ayarlanabilir harekete dönüşerek sistemin hareketi ve yönlendirilmesi sağlanır (İnt. Kyn.20).

2.4.4 Kamera Yönlendirmeli Otomatik Kılavuzlu Araçlar

Kamera yönlendirmeli otomatik kılavuzlu araçlar buldukları ortamda hareket edebilmek için yerleşik kameraları kullanır. Bu otomatik kılavuzlu araçlar, kalabalık bir alanda ya da küçük ortamlarda hassas navigasyona ihtiyaç duyulduğunda çok faydalıdır (Şekil 2.5) (İnt. Kyn.16).



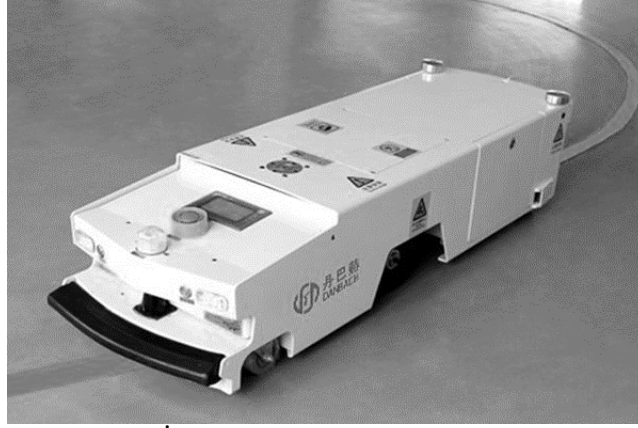
Şekil 2.5 Kamera yönlendirmeli bir sistem (İnt. Kyn.16).

2.4.5 Optik Güdümlü Otomatik Kılavuzlu Araçlar

Optik güdümlü otomatik kılavuzlu araçlar, operatörlerin zeminlerine yapışan ışığa duyarlı bant ile kılavuzluk alırlar. Konumlarını bilmelerine yardımcı olmak için, optik güdümlü otomatik kılavuzlu araçlar, kilometre sayacı kullanarak durma noktaları arasındaki mesafeyi ölçebilirler (İnt. Kyn.20).

2.4.6 Atalet Kılavuzlu Otomatik Kılavuzlu Araç Sistemleri

Atalet kılavuzlu otomatik kılavuzlu araçlar, yönlerini anlamak için zemine monte edilmiş mıknatıslara güvenir. Konumlarını ve nerelerde hareket etmeleri gerektiğini mıknatıslar vasıtasıyla belirlerler. Sistemde mıknatısları algılayan bir sensör, ünitenin yönünü ölçmek için bir jiroskop ve aracın başlangıca göre konumunu hesaplamak için bir tekerlek sayacı bulunur (Şekil 2.6) (İnt. Kyn.11).



Şekil 2.6 Atalet kılavuzlu sistem (İnt. Kyn.11).

2.4.7 Diğer Otomatik Kılavuzlu Araç Sistemleri

Otomatik kılavuzlu arabalar, uçak parçaları gibi zor hareket eden yükleri taşımak için kullanılır. Çekme araçları, ağır palet yüklerini taşıyabilen ancak küçük parçaları bir araya getirmede yardımcı olamayan basit otomatik kılavuzlu araçlardır. Payanda otomatik kılavuzlu araçları payandalar olarak bilinen iki yatay ayakla inşa edilmiştir. (İnt. Kyn.20).

Bunların haricinde çatallı otomatik kılavuzlu araç sistemleri, forkliftler gibi, çok çeşitli yükleri kaldırabilir ve farklı seviyelere taşıyabilir. Büyük şasilere sahip otomatik kılavuzlu araç sistemleri olarak da bilinen birim yük otomatik kılavuzlu araçlar, tekerlekli yataklar ve kaldırma indirme mekanizmaları gibi bileşenlerin yardımıyla ağır yükleri taşırlar (İnt. Kyn.20).

2.5 Otomatik Kılavuzlu Araçların Faydaları

Otomatik kılavuzlu araçların başta imalat sektörü olmak üzere çeşitli sektörlerde kullanılmasının birçok nedeni vardır. Otomatik kılavuzlu araç sistemlerinin sağladığı faydalar aşağıda maddeler halinde listelenmiştir.

- Hesap verebilirlik – Ürün takibi için bir ürün otomatik kılavuzlu araca yerleştirildiğinde, “kaybedilir” veya yanlış yerleştirilmiş ürün en aza indirilecek şekilde izlenilebilir.
- Otomatik hat dengeleme – Farklı zamanlarda birden fazla işlem yapılan bir üretim ortamında, otomatik kılavuzlu araçlar hat dengelemede yardımcı olabilir.

- Maliyet kontrolü – Bir otomatik kılavuzlu araç sisteminin maliyetleri önceden tahmin edilebilirken, işgücü maliyetleri ekonomik koşullara bağlı olarak hızlı bir şekilde değişebilir bu durumda personel ve zamandan tasarruf sağlanır.
- Tesis bakımı – Engelden kaçınma, çarpışma önleme yetenekleri sayesinde tesis içindeki hasarı önler.
- Esneklik – Bir otomatik kılavuzlu aracın izlediği yol, üretim ve kullanım ihtiyaçları arttıkça değiştirilebilir.
- Daha az kısıtlama – Serbest dolaşımda bulunan otomatik kılavuzlu araçlar, konveyörler tarafından oluşturulan erişim sorunlarını ortadan kaldırır ve daha dar koridorlara izin vererek geleneksel forkliftlere göre daha az alan gerektirir.
- Düşük işletme maliyetleri – Şarj ve akü kullanımı bir otomatik kılavuzlu araç sistemiyle otomatik olabilir ve kontrollü hızlanma / yavaşlama bileşenlerin yıpranmasını en aza indirir.
- Ürün hasarında azalma – Otomatik kılavuzlu araçlar taşıdıkları ürünün cinsine fiziki yapısına göre ürünleri nazikçe ele alarak hurda ve atıkları azaltır.
- Tekrarlı bilirlilik – Otomatik kılavuzlu araçlar tekrarlanabilir hareket görevlerini öngörülebilir ve güvenilir bir şekilde gerçekleştirir.
- Güvenlik – Otomatik kılavuzlu araçlar her zaman rehber yollarını takip eder ve bir engelle karşılaşırlarsa, çevredeki personelin güvenliğini artırarak durur.
- Çizelgeleme – Güvenilirlikleri ve zamanında teslimatları nedeniyle otomatik kılavuzlu araçlar çizelgeleme yeteneklerini ve işlemlerin verimliliğini artırır.
- Ölçeklenebilirlik – Kapasiteyi ve kapasiteyi arttırmak için daha fazla otomatik kılavuzlu araç sisteme eklenebilir.
- Verimlilik – Otomatik kılavuzlu araç sistemlerinin bulunduğu tesislerde verimlilik yükselir. (İnt. Kyn.4).

2.6 Konum Belirlemenin Önemi

Otomatik kılavuzlu araçlarda en önemli konulardan bir tanesi aracının bulunduğu ortamdaki konumunun belirlenmesidir. Aracın gerekli faaliyeti yürütebilmesi için hem başlangıç noktasına göre konumu hem de bulunduğu ortam için konumu bilgisinin mutlaka hesaplanması gerekir. Belirlenen konuma göre aracın bir sonraki faaliyeti için

yönlendirilmesi gerekir. Gerek konum belirleme gerekse konumu hesaplanan aracın yönlendirilmesi için birçok yöntem yazılım ve algoritma geliştirilmiştir.

Birden fazla otomatik kılavuzlu araç sisteminin kullanıldığı ortamlarda araçların aynı ya da farklı hatlar üzerinde hareket ederken meydana gelebilecek olumsuz durumların önüne geçmek için araçların anlık olarak konumlarının belirlenmesi gerekmektedir. Trafik sıkışıklığını düzenlemek, otomatik kılavuzlu araçların çarpışmasını engellemek, aracı en yakın batarya ünitesine yönlendirmek gibi durumlar tamamen konum belirleme ile gerçekleştirilmektedir.

Konum belirleme işlemi yapılırken aracın hareketini sağlayan kılavuz çeşitleri önem kazanmaktadır. Bu kılavuz çeşitlerine göre farklı konum belirleme yöntemleri kullanılmaktadır.

2.7 Konum Belirleme Yöntemleri

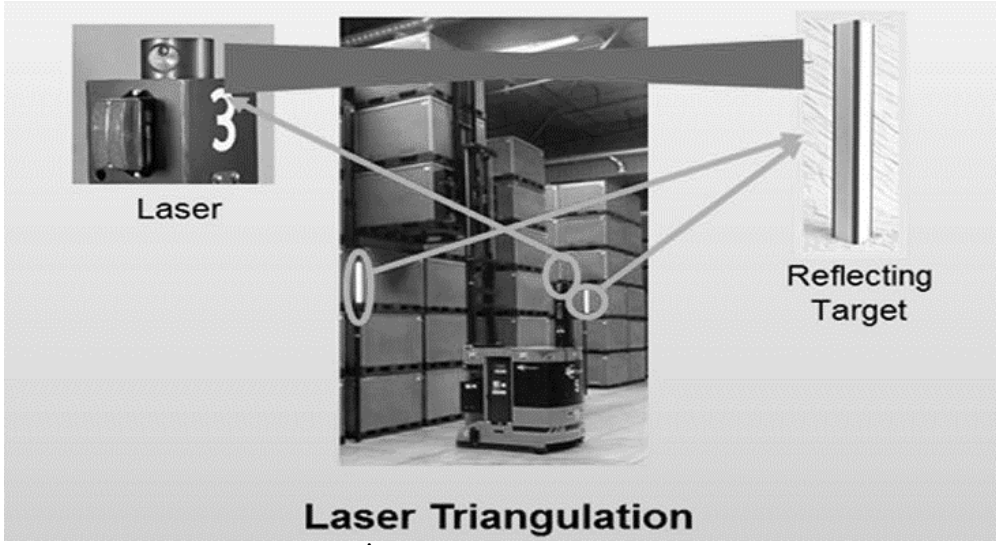
Otomatik kılavuzlu araçlar için kullanılan tipik konum belirleme yöntemleri ya da başka bir ifadeyle navigasyon türlerinden günümüzde sık kullanılan çeşitleri aşağıda listelenmiştir (İnt. Kyn.21).

- Lazer üçgenleme
- Atalet
- Manyetik bant
- Manyetik ızgara
- Doğal özellik
- Tel
- Optik

2.7.1 Lazer Üçgenleme Navigasyon Sistemleri

En popüler otomatik kılavuzlu araç navigasyon yöntemidir. Öncelikle işletmenin genelinde yansıtıcı hedefler bilinen konumlara yerleştirilir. Aracın üzerine bir lazer tarayıcı monte edilmektedir. Konumun belirlenmesi için yansıtıcı hedeflere araçtaki lazer tarayıcı tarafında sinyaller gönderilir. Araç kontrol algoritmaları, üçgenleme yoluyla tam

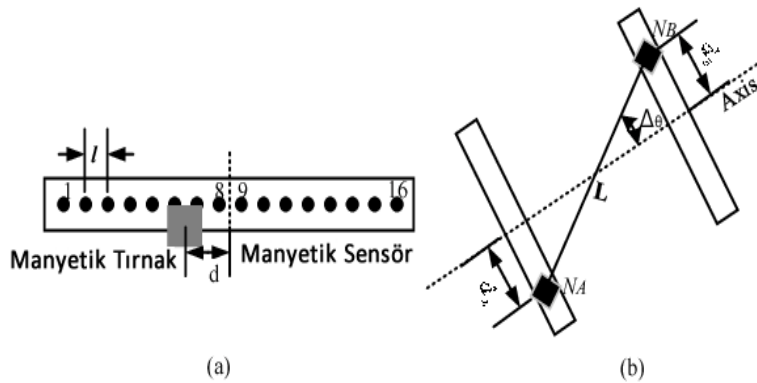
araç konumunu hesaplar (Şekil 2.7).



Şekil 2.7 Lazer üçgenlemeli sitemi (İnt. Kyn.17).

2.7.2 Ataletsel Navigasyon Sistemleri

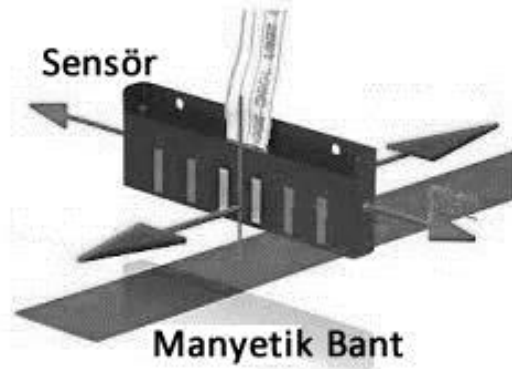
Referans noktaları (genellikle mıknatıslar) zemine belirli bir x, y koordinatlarında sistem haritasında yerleştirilmiş vaziyettedir. Referans noktaları, araç referans noktasından geçerken araçtaki bir sensör tarafından algılanır. Araçtaki jiroskop, aracın yönünü ölçmek ya da korumak amacıyla kullanılmaktadır. Araçtaki bir tekerlek kodlayıcı, kat edilen mesafeyi hesaplar. Araç konumu belirlemek için üç cihazdan da gelen geri bildirimler kullanılarak hesaplama yapmaktadır (Şekil 2.8).



Şekil 2.8 Manyetik sensör algılama (a), Yalpa açısı hesaplama (b) (Yin vd. 2018).

2.7.3 Manyetik Bant Navigasyon Sistemleri

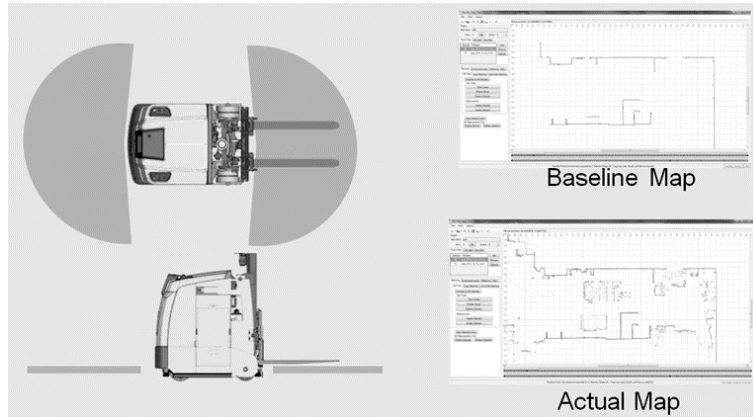
Kablo kılavuzuna benzer şekilde, kılavuz yolu tanımlamak için zemine manyetik bant yapıştırılmaktadır. Aracın alt tarafındaki bir sensör manyetik bandı algılar ve bandı takip eder. Yol ya da rota değişiklikleri için bant çıkarılıp yeniden takılarak güzergâh değişiklikleri tanıtılmaktadır (Şekil 2.9).



Şekil 2.9 Manyetik bant ve sensör modeli (Wu vd. 2014).

2.7.4 Doğal Özellikli Navigasyon Sistemleri

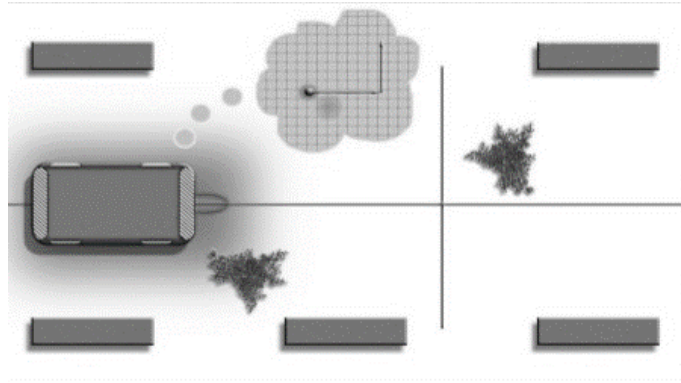
Çalışma alanının referans görüntüleri aracın hafızasına kaydedilip saklanmaktadır. Çalışma alanında benzersiz şekilde tanımlanabilir, doğal olarak ortaya çıkan özellikler tanımlanmıştır. Aracın gerçek konumu, bu doğal özelliklerle karşılaştırıldığında göreceli konumuna göre hesaplama işlemi yapılmaktadır. Kurulum sırasında özellikleri kaydetmek ve navigasyon sırasında özellikleri algılamak için bir kamera veya lazer kullanılabilir (Şekil 2.10).



Şekil 2.10 Doğal özellikli navigasyon sistemi için haritalar (İnt. Kyn.17).

2.7.5 Tel Navigasyon Sistemleri

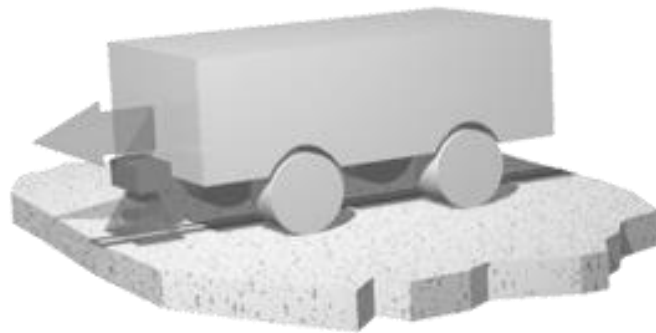
Araç zemine gömülü kesintisiz bir tel kullanarak hareketini gerçekleştirmektedir. Aracın üzerinde bulunan antenler telden gelen sinyalleri algılar. Mesafeyi hesaplamak için tekerleklerde kodlayıcı kullanılmaktadır. Genellikle sistem iyileştirmelerde, sistem değişikliklerinde ve genişletmelerde kullanılır (Şekil 2.11).



Şekil 2.11 Tel navigasyon sistemi (İnt. Kyn.15).

2.7.6 Optik Navigasyon Sistemleri

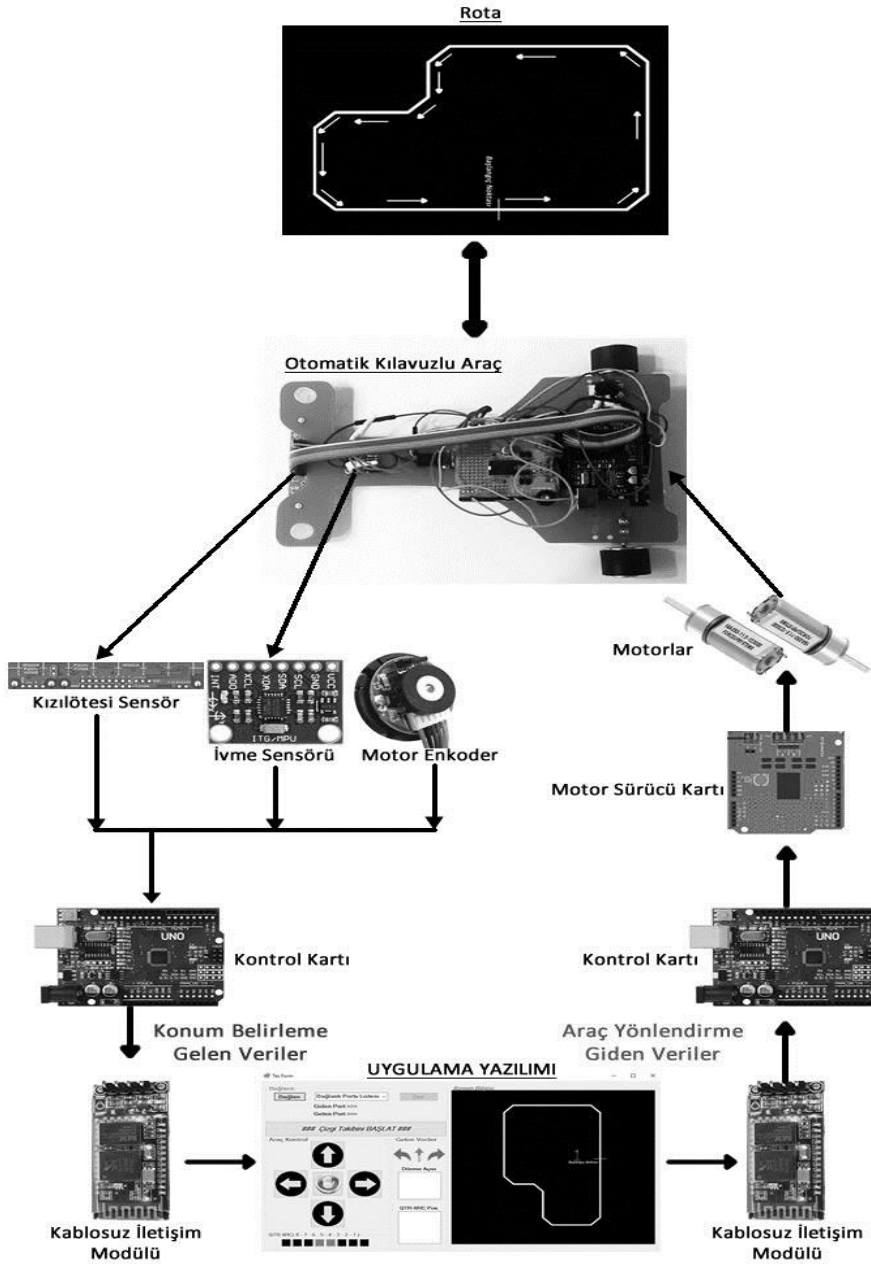
Araçın hareket edeceği zemin kimyasal boyama ya da şerit bant ile sabitlenmiş vaziyettedir. Araçın hareket yolunu tespit etmesini sağlayan yerleşik bir sensörü bulunur. Bazı sistemler, şeridi aydınlatmak için bir ultraviyole (UV) ışık kaynağı kullanmaktadır. Genellikle tesislerde veya depolarda kullanılmaz, çünkü zemin hattının temizlenmesi veya yeniden uygulanması gerekmektedir (Şekil 2.12).



Şekil 2.12 Optik navigasyonlu sistem (İnt. Kyn.13).

3. MATERYAL ve METOT

Otomatik kılavuzlu araçlarda eş zamanlı konum belirleme ve yönlendirme çalışmasında otomatik kılavuzlu araç, aracın takip edeceği yol, bilgisayar, bilgisayar yazılımları ve kontrol kartları olmak üzere beş ana kısım bulunmaktadır. Her kısım kendi içinde irdelenmiş kullanılan parçalara ilişkin alternatifler de değerlendirilip optimum konum belirleme ve aracın yönlendirilebilmesi dikkate alınmıştır (Şekil 3.1).



Şekil 3.1 Sistemin çalışma şeması.

3.1 Otomatik Kılavuzlu Araç

Bu tez çalışmasında prototip olarak geliştirilen araç için aşağıda listelenen materyaller kullanılmıştır.

1. 1 Adet Araba Şasesi
2. 2 Adet Redüktörlü Enkoderli DC Motor
3. 2 Adet Kontrol Kartı
4. 2 Adet Kızılötesi Sensör
5. 2 Adet Bluetooth Modülü
6. 1 Adet Motor Sürücü Kartı
7. 1 Adet İvme ve Gyroscope Sensör
8. 1 Adet Lityum Polimer Batarya
9. 4 Adet Silikon ve Sarhoş Tekerlek
10. 2 Adet Şaseye Motor Tutturucu Aparat
11. Çeşitli Vida ve Somunlar
12. Dişi-Dişi 20cm Jumper Kablolar
13. Erkek-Dişi 20cm Jumper Kablolar
14. Erkek-Erkek 20cm Jumper Kablolar

Kullanılan önemli bazı parçaların özellikleri alt maddeler halinde belirtilmiştir.

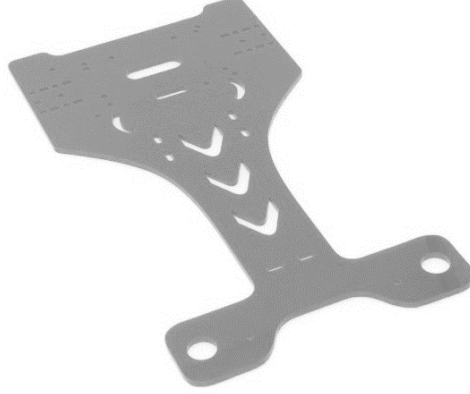
3.1.1 Araba Şasesi

Otomatik kılavuzlu araç için şase olarak rapid şase kullanılmıştır. Bu şase özellikle çizgi takibi yapan araçlarda aracın stabil bir şekilde çizgiyi takip etmesi için tercih edilmektedir. Hafif ve dayanıklı olması nedeniyle pleksi glass tercih edilmiştir. Şasenin özellikleri aşağıda verilmiştir. Şekil 3.2’de araba şasesi gösterilmiştir.

Özellikleri;

- Lazerli kesme yöntemiyle üretilmiş bir şasidir.
- Arka kısmına 2 adet motor monte edilebilir.

- Arka kısmına 2 adet teker kullanılacak gövdedir.
- Ön kısmına 2 adet sarhoş teker monte edilebilir.
- Malzeme cinsi: Pleksi Glass malzemeden üretilmiştir.
- Şasi uzunluk: 250 mm.
- Şasi en: 135 mm.
- Şasi arka taraf en: 100 mm.



Şekil 3.2 Rapid Robot araba şasisi (İnt. Kyn. 44).

3.1.2 Redüktörlü Enkoderli DC Motor

Otomatik kılavuzlu aracın hareketini sağlamak için iki adet 12V 1200 RPM redüktörlü enkoderli Pololu marka DC motor kullanılmıştır. Robotik uygulamalarda en çok kullanılan motor tipi DC motorlardır. DC motorlar düşük maliyetli, boyutları ufak ve etkilidir. Birçok farklı yapıda seçeneğe sahip olması da DC motorların tercih sebebidir. DC motorlar robotik uygulamalarda ya da herhangi bir sistemde dişli kutulu veya dişli kutusuz şeklinde aynı sistemde bulunabilirler. Doğru akım motorların uygulamalarda tercih edilme nedenlerine ilişkin nitelikleri aşağıda listelenmiştir.

Yön; Doğru akım motorlarında dönüş yönünü uygulanan akım yönü belirler. Ters akım verildiğinde motor ters tarafa döner.

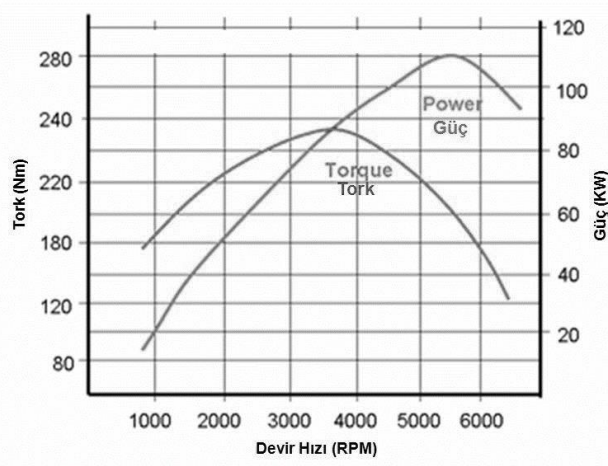
Hız; dakikadaki tam tur sayısı ile ifade edilir. RPM olarak isimlendirilir. Bu hız uygulanan gerilime, yük miktarına bağlıdır.

Gerilim; mini doğru akım motorları minimum 1,5 Volt maksimum 48 Volt arasında farklı

gerilim değerlerine sahiptirler.

Akım; doğru akım motorlarının devreden çektiği akım üzerine düşen yüke göre değişmektedir. Yük miktarı ile devreden çekilen akım şiddeti doğru orantılıdır.

Güç; doğru akım motorlarına uygulanan potansiyel enerji farkı ve devreden çektiği akım şiddeti çarpılırsa motorun gücü ortaya çıkar. Bazı durumlarda güç ifadesi kullanılırken bazı durumlarda tork ifadesi daha önemlidir. Tork, dönüş momentine karşılık gelen ifadedir. Bu değer ne kadar yüksek ise motorda o kadar güçlüdür (Şekil 3.3) (İnt. Kyn.33).



Şekil 3.3 Tork, devir ve güç ilişkisi (İnt. Kyn.34).

Robotik uygulamalarda güç yerine tork değeri daha önemlidir. Yapılacak işe uygun tork değerine sahip motor seçimi yapılır.

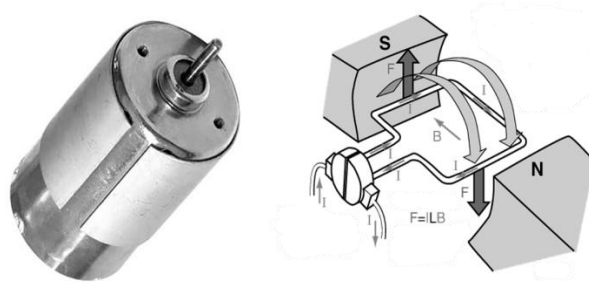
3.1.2.1 DC Motorların Çalışma Prensibi

DC motor sabit ve hareketli bölümlerden oluşur. Hareketsiz bölüme gövde, hareketli bölüme endüvi denir. İletkenler, maddeleri mıknatıs gibi çekmeye çalışır. Benzer kutupa sahip manyetik alanlar itilir; diğer manyetik alana sahip olanları çekmeye çalışır. N'den S'e meydana gelen kuvvete manyetik akı denir.

İletken cisme doğru akım verildiğinde bu cisim, sabit manyetik alan meydana getirir. Bu alan N-S kutuplarını oluşturur. Bu alan içine giren metaller mıknatısın niteliklerine sahip olur. İletken metalleri çeker. Aynı kutupa sahip alanlar birbirini iter. Farklı kutupa sahip

olanlar birbirini çekerler. Oluşan N-S kutuplarında arasında güçlü manyetik akım meydana gelmektedir.

Gövdede oluşan N-S kutupları, endüvide(makinaların dönen bölümleri) oluşan itme veya çekme ile dönüş hareketi meydana gelir. Endüvi ve gövdenin durmalarını engellemek için, endüvide birden çok sargı oluşturulur. Sargı, endüvideki kolektörde yer alır. Kolektördeki gerilim, fırçayla iletilir. Kömür fırçası hareketsiz olduğundan dönmeye devam eden endüvinin voltajı sargılar ile değiştirilmektedir (Şekil 3.4) (İnt. Kyn. 54).

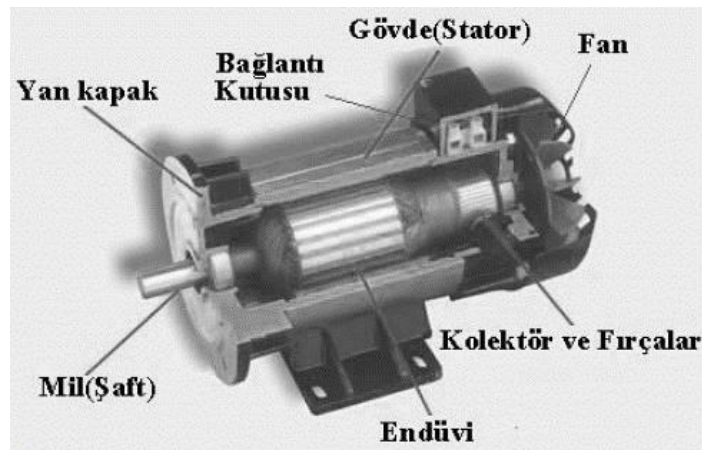


Şekil 3.4 DC motorun çalışma prensibi (İnt. Kay. 55).

Şekil 3.4'te F uygulanan kuvveti, B manyetik alanı, L iletkenin uzunluğunu, I iletkenin geçen akım şiddetini ifade etmektedir.

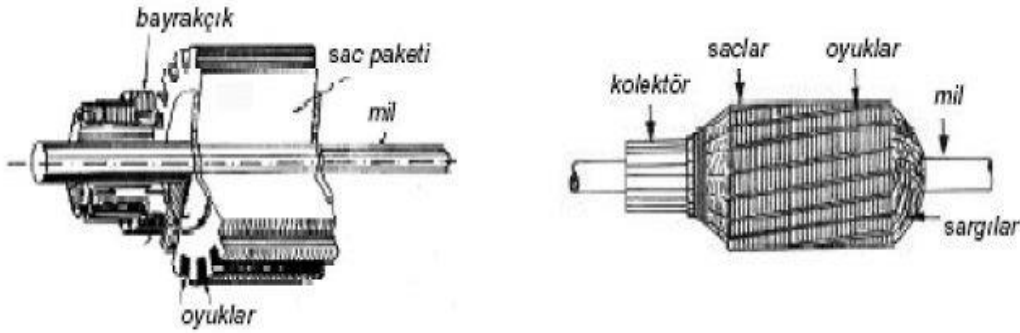
3.1.2.2 DC Motorlarının Kısımları ve Özellikleri

Şekil 3.5'te DC motorların kısımları dış kesit görünümü ile gösterilmiştir.



Şekil 3.5 DC dış kesit görünümü (İnt. Kyn. 24).

Endüvi(Rotor); Motorların hareketi bölümüdür. 0,30,7 mm kalınlıđa sahip elik sacdan oluřmuř silindir řeklindeki gvdenin stne aılan ve kanallara yerleřtirilen sargılar ile oluřturulmuřtur. Rotor sargısının ularına, bakır dilimlerden elde edilmiř ve stne fıranın temasını yaptığı blme bađlı bulunmaktadır (řekil 3.6) (İnt. Kyn. 24).



řekil 3.6 Endvi (Rotor) kesiti (İnt. Kyn. 25).

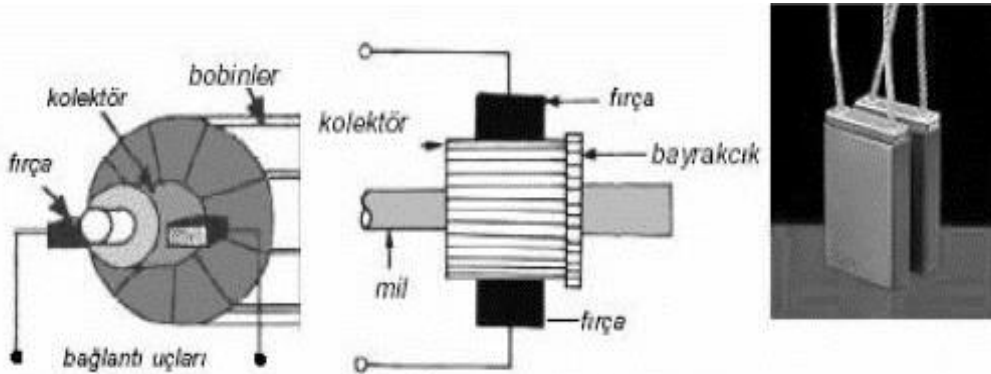
Gvde(Stator);Dođru akım veya alternatif akım makinelerinde N ve S kutupların oluřması amacıyla yapılan sargıların yerleřtiđi blme endktr adı verilir. Mini motorların endktr dođal mıknatıs ile yapılıyorken, yksek gce sahip makinelerin endktr bobinler vasıtasıyla oluřmaktadır (řekil 3.7) (İnt. Kyn. 24).



řekil 3.7 DC motor stator grnmleri (İnt. Kyn. 26).

Fıra ve Kolektrler; Dođru akım ve alternatif akımla alıřabilen kolektre sahip motorların kolektrn stne temas eden paraya fıra ya da kmr adı verilir. Fıra, makinedeki akımın, gerilimin miktarına bađlı olarak deđiřik niteliklerde retimi yapılır. Fıranın kolektr stne uygun řekilde oturması iinde baskılı yay kullanılmaktadır. Fıra ařıyan bir para olduđu iin belli zaman sonra zelliđini yitirir. DC motorlarda

rotora verilen gerilim kolektör vasıtasıyla iletimi sağlanır. Kolektörün dilim parçaları, haddeden geçmiş dayanıklı bakır baskılanarak elde edilir. Bakır dilimlerin aralarına 0,5 - 1,5 mm kalınlığa sahip mikadan veya mikanite benzer yalıtkan yerleştirilir. Kolektörler, DC motorların kritik ve en çok arızalanan parçalarıdır. Bu yüzden kolektörün dilim parçaları, hassasiyetle üretilir ve dilim parçaları arasındaki potansiyel enerji farkının 15 voltu geçemeyecek miktarda ayarlanması gerekmektedir (Şekil 3.8) (İnt. Kyn. 26).



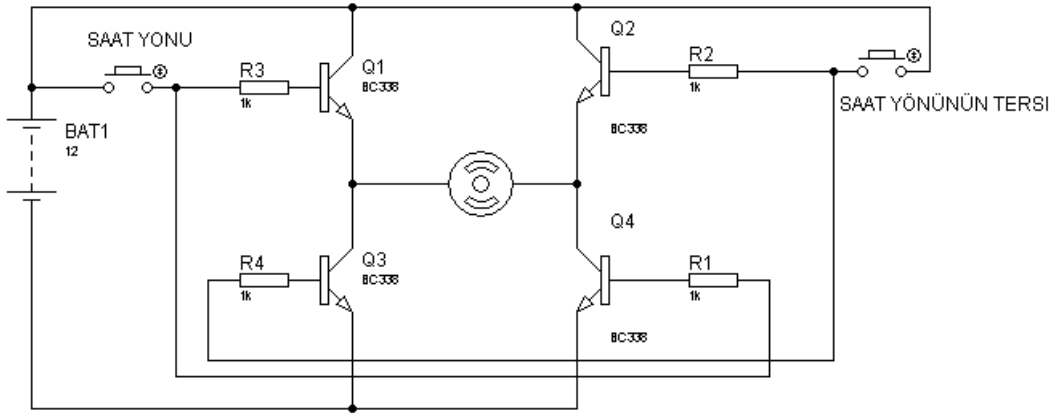
Şekil 3.8 Fırça ve kolektör kesitleri (İnt. Kyn. 26).

Motor Yan Kapakları; Stator ve kapakların motoru dış etkilere karşı korumak için alüminyum, demir ya da demir alaşımından üretilir. Rotorun stator içinde merkezi olarak yataklanması görevini ise kapaklar yapmaktadır (İnt. Kyn. 26).

3.1.2.3 DC Motorlarda Yön ve Hız Kontrolü

DC motorlarda motorun devir yönünü ve devir hızının değiştirilmesi için birçok fazla seçenek bulunmaktadır. Bu işlemler için çeşitli devreler ya da kontrol kartları mevcuttur.

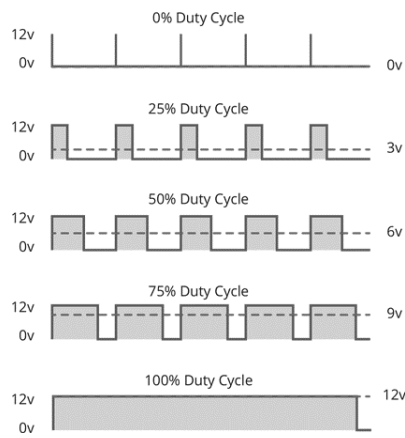
DC motorlarda yön kontrolünü gerçekleştirmek için H köprüsü adı verilen bir devre kullanılmaktadır. Aşağıdaki şekil 3.9’da H köprüsü devresi verilmiştir. Bu devrede saat yönü anahtarı aktif hale getirildiğinde Q1 ve Q4 transistörleri devreye girerek motorun içinden geçen elektrik akımı motorun saat yönünde hareket etmesini sağlamaktadır. Benzer şekilde saat yönünün tersi anahtarı kapatıldığında Q2 ve Q3 transistörleri devreye girerek elektrik akımının ters yönde gitmesini ve motorun ters yönde hareket etmesini sağlamaktadır. Her iki anahtarın aynı anda kapalı olması kısa devre meydana getirmektedir (Şekil 3.9) (İnt. Kyn. 57).



Şekil 3.9 DC motor devir yönü değiştirme H köprü devresi (İnt. Kyn. 27).

DC motorlarda hız kontrolü için kısaca PWM olarak adlandırılan sinyal genişlik modülasyonu kullanılmaktadır. Bu yöntemde görev döngüsü(Duty Cycle) kavramı kullanılmaktadır. PWM tekniğinin temeli kare dalgalar üretme prensibine dayanmaktadır.

Bir dizi ON-OFF kare dalgalar gönderilerek motorun giriş voltajının ortalama değeri ayarlanmaktadır. Ortalama voltaj değeri darbe genişliği olarak ta adlandırılan görev döngüsü ile doğru orantılıdır. Darbe genişlikleri ne kadar yüksek olursa motora uygulanan ortalama voltaj değeri o kadar yüksek olacağı için motorun hızı yükseltilmiş olmaktadır. Benzer şekilde görev döngüsü(Duty Cycle) yani darbe genişlikleri ne kadar düşük olursa da motora uygulanan ortalama voltaj değeri o kadar düşük olacağı için motorun hızı düşürülmüş olmaktadır (Şekil 3.10) (İnt. Kyn 58).



Şekil 3.10 PWM görev döngüsü ortalama voltaj değerleri (İnt. Kyn. 58).

3.1.2.4 DC Motor Çeşitleri

Fırçasız Doğru Akım Motorları; Motorda güç transferi, fırçalar ve kolektörlere alternatif olarak elektronik anahtar bu işi yerine getirir. Bu nedenle ark önlenir. Motorlar, yüksek hıza ulaşabilmektedirler. Sık tercih edilen gerilim seçeneği 24 voltur.

Şönt Motorlar; Uyarım sargısını endüvinin sargısına paralel şekilde bağlı olduğu DC motordur. Şönt motorunun devri yüke göre değişmeyen bir yapıya sahiptir. Motorun devrinin miktarı, kaynağının voltajı veya endüvinin akım miktarı ile kontrol edilmektedir.

Seri Motorlar; Uyarım sargısının endüvinin sargısına birbirlerine seri şekilde bağlandığı DC motorudur. Motora, yüklenildiğinde devrinin miktarı hızlı şekilde düşmektedir. Bunun sebebi, yükün akımıyla uyarımın akımının aynı olmasıdır.

Kompunt Motorlar; Seri ve şönt motorunun birleşiminden elde edilmiş bir DC motor çeşididir. Kompunt motorlar, başlangıç seviyesinde “n” gibi devir değerindedirler. Nedeni ise motorlarda var olan şönt sargısı, az miktarda akımı çekmektedir. Bu bir ϕ manyetik akı oluşturmaktadır(İnt. Kyn. 59).



Şekil 3.11 12 Volt 1200 RPM Redüktörlü Enkoderli DC Motor (İnt. Kyn. 45).

Tez çalışmasında kullanılan motorun görüntüsü Şekil 3.11’de, motorun özellikleri aşağıda özellikleri;

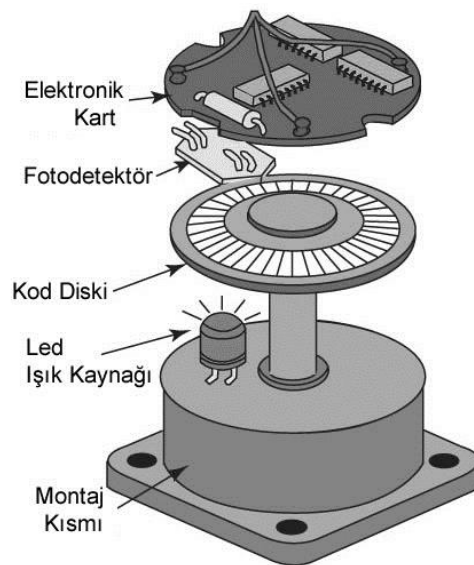
- Çalışma Gerilimi: 12 Volt.
- Dakikadaki Devir Sayısı: 1200 Rotations Per Minute (Devir/Dakika).

- Boştaiken Çekilen Akım Miktarı: 100 mili Amper.
- Durma Anında Çekilen Akım Miktarı: 1100 mili Amper.
- Durma Torku Miktarı: 0.8 kg/cm
- Motor Çap Uzunluđu: 25 mm
- Mil Çapı: 4 mm D Şaft Çıkış Noktası Orta
- Mil Uzunluk Miktarı: 13 mm
- Temel Uzunluk: 63 mm
- Motor Ađırlıđı: 95 gr

3.1.2.5 DC Motor Enkoder Sistemi

Enkoder, DC motorlarda kullanılabilen motor miline bađlı olarak geri bildirim sađlayan elektromekanik bir sensördür. Endüstrinin birçok alanında enkoderler kullanılmaktadır. Konum, Hız, Mesafe gibi geri bildirim gerektiren uygulamalarda enkoderler tercih edilmektedir.

Enkoderler sinyal oluşturmak için mekanik, manyetik, dirençli ya da optik olmak üzere farklı teknolojilerde üretilirler. En yaygın kullanılan optik sistemidir. Optik algılama sisteminde enkoder ışık kesintisine bađlı olarak geri bildirim sinyali üretir. Şekil 3.12’de optik teknolojiyi kullanan bir enkoder sistemi gösterilmiştir.



Şekil 3.12 Optik sistem enkoder yapısı (İnt. Kyn. 61).

Motor milinin hareketi ile LED ışık kaynağından gelen ışık huzmesinin opak çizgilerle desenli kod diskinden belirli aralıklarla geçerek ışığın karşı tarafa geçmesi ya da engellenmesi ile fotodedektörde bir darbe sinyali oluşturur. Işık=Açık ya da Işık=Kapalı şeklinde üretilen bu sinyaller kontrol kartına gönderilir ve motorun hızı, yönü ve mesafesi kontrol edilebilir.

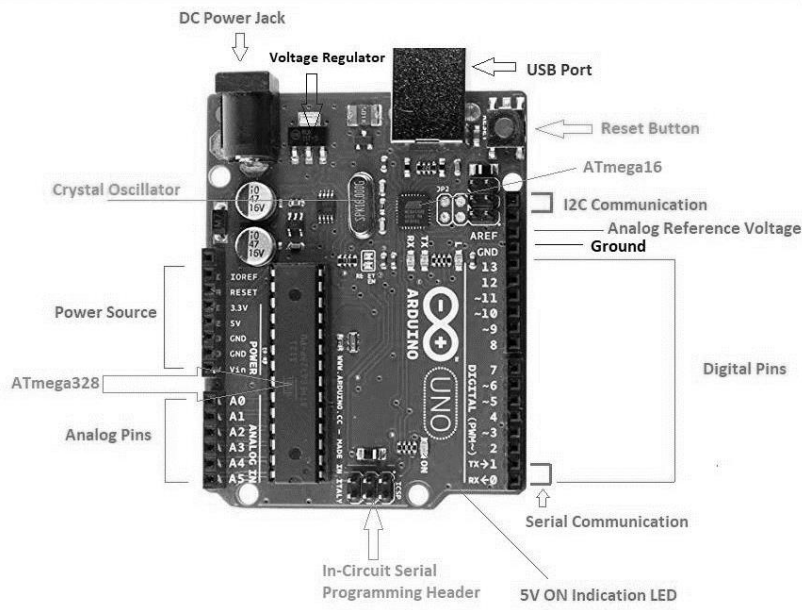
3.1.3 Kontrol Kartı

Aracın kızılötesi sensörler, ivme sensörü, motor sürücü kartı ve diğer bileşenler ile iletişimini sağlamak için iki adet arduino uno kontrol kartı kullanılmıştır. Bu kontrol kartından birincisi aracın yönlendirme işlemini gerçekleştirmek amacıyla, ikinci kontrol kartı ise kızılötesi sensör ve ivme sensöründen gelen verileri uygulama yazılımına aktarmak amacıyla kullanılmaktadır.

Arduino.cc tarafından geliştirilen ve Atmega328 mikro denetleyici ile devre elemanlarını yönetebilen bir kontrol kartıdır. Arduino 2003 yılındaki D. Cuartielless ve M. Benzi tarafından gerçek ortamda cihazların kontrolünü yapabilmek amacıyla öğrenciler ve profesyonellere yönelik olarak düşük maliyetli projeler gerçekleştirmeleri amacıyla geliştirilmiştir.

Arduino Uno son versiyonu Universal Serial Bus ara yüzü, altı adet analog pini, diğer devrelerin bağlanması amacıyla on dört adet giriş çıkış sayısal pinle birlikte gelir. On dört giriş çıkış pininin altısı PWM çıkışı için bulunur. Seri iletişim için Tx ve Rx pinleri kullanılabilir.

Karta enerji sağlamak için adaptör kullanılabilir. Bu kartlar Arduino ürünlerindeki kartlara çok benzeyen yapıya sahiptir, ancak bu kartlarda seri sürücü yongası yoktur. Uygulamanın özelliği karmaşık hale geldiğinde, depolama miktarını yükseltmek amacıyla Mikro SD kart eklenebilmektedir (Şekil 3.13) (İnt. Kyn. 29).



Şekil 3.13 Arduino Uno bağlantı noktaları diyagramı (İnt. Kyn. 29).

3.1.3.1 Kontrol Kart Yapısı

Arduino Uno bilgisayar iletişimi için USB'yi kullanabilir. 16MHz hızı sahip Atmega 328 mikro denetleyicisi kullanılmaktadır. Açık kaynak kodlu bir platformdur. Kontrol kartı voltaj regülatörüne sahiptir ve fabrika ayarlarına dönmek için Reset tuşu mevcuttur. Kartta on dört adet dijital ve altı analog pin yer almaktadır. Harici bağlantı yapılabilmeyle beraber bağlanabilen harici cihazlara esneklik ve kullanım kolaylığı sağlamaktadır. Cihazları karta bağlamak için zor bir ara yüz gerekmemektedir. Harici cihazı, kart üzerinde pinlere takılması yeterlidir.

A0, A1, A2, A3, A4, A5 analog pinlerdir ve 10 bit çözünürlük söz konusudur. Bu pinler +0V ve +5V ile çalışmaktadır. Komutların program kodları türünde saklanabilmesi maksadıyla 13 KB ekstra bellek kullanılmaktadır (İnt. Kyn. 29).

3.1.3.2 Kontrol Kartı Pin Açıklamaları ve Özellikleri

+5V'da çalışan kartta var olan birkaç adet sayısal ve analog pin bulunur. Bunlar 20 mA ile 40 mA arasında farklı akım değerlerine sahiptir.

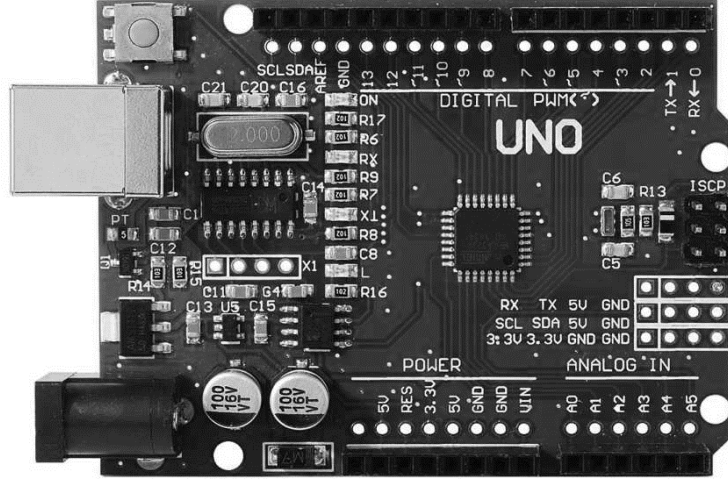
- LED, Arduino Uno, pin 13 üzerinden bağlanan dahili LED ile birlikte gelir. Pine

- yüksek bir değer verilirse led yanıp söner.
- Vin, Arduino Kartına sağlanan giriş voltajıdır. Bir USB portu ile verilen 5 V'tan farklıdır. Bu pin voltaj beslemesi için kullanılır. Güç girişi üzerinden bir voltaj sağlanırsa, bu pinle erişilebilir.
 - 5V, Bu pin voltaj regülasyonu sağlama yeteneğine sahiptir. Çıkış ayarlı voltaj sağlamak için 5V pin kullanılır. Kart, USB, kartın Vin pini veya DC güç girişi gibi üç yolla çalıştırılır.
 - Vin, Power Jack 7V - 20V arası bir voltajı desteklerken, USB 5V civarında voltajı destekler. Tahtanın 5V'da çalıştırılması önerilir. Eğer 5V veya 3.3V pinlerden bir voltaj verilirse, voltaj limit sınırını aşarsa karta zarar verebilecek voltaj regülasyonunun atlanmasıyla sonuçlandığını not etmek önemlidir.
 - GND, Bunlar topraklama pinleri. Kart üzerinde, ihtiyaçlara göre kullanılacak birden fazla topraklama pini bulunmaktadır.
 - Reset, Bu pin, pano üzerinde çalışan programı sıfırlayan pindir. Kart üzerindeki fiziksel sıfırlama yerine IDE, kartı programlama ile sıfırlama özelliğine sahiptir.
 - IOREF, Bu pin, panoya voltaj referansı sağlamak için çok kullanışlıdır. Bu pin üzerindeki gerilimi okumak için bir kalkan kullanılır ve ardından uygun güç kaynağını seçer.
 - PWM 3,5,6,9,10,11 numaralı pinler tarafından sağlanır. Bu pinler 8-bit PWM çıkışı sağlayacak şekilde yapılandırılmıştır.
 - SPI, Seri Çevre Birim Ara yüzü olarak bilinir. Kütüphanesinin yardımıyla SPI iletişim sağlanır.
 - Aref, Analog Referans olarak adlandırılır. Bu pin, analog girişlere referans voltajı sağlamak için kullanılır.
 - TWI, İki telli ara yüz denir. TWI iletişimine Wire Library aracılığıyla erişilir. Bu amaçla A4 ve A5 pinleri kullanılır.
 - Seri iletişim, Seri iletişim, Pin 0 (Rx) ve Pin 1 (Tx) olarak adlandırılan iki pin vasıtasıyla gerçekleştirilir.
 - Veri iletmek için Tx pini kullanılırken veri almak için Rx pini kullanılır.
 - Dış Kesmeler, Pin 2 ve 3 harici kesmeler sağlamak için kullanılır(İnt. Kyn. 29).

Arduino Uno Özellikleri ve Görünümü (Şekil 3.14);

- Mikro Denetleyi Çip Modeli: ATmega328.

- Çalışma Gerilimi: 5 Volt.
- Nominal Giriş Gerilimi: 7-12 Volt.
- En Yüksek Giriş Gerilimi: 6-20 Volt.
- Dijital Giriş/Çıkış Pinleri Sayısı: 14 (6'sı PWM çıkışı).
- PWM Dijital Giriş/Çıkış Pinleri Sayısı: 6.
- Analog Giriş Pinleri Sayısı: 6.
- Her Giriş/Çıkış İçin Akım Miktarı: 40 mili Amper.
- 3.3 V Çıkış İçin Akım Miktarı: 50 mili Amper.
- Flash Bellek Miktarı: 32 KB (ATmega328).
- SRAM Miktarı: 2 KB (ATmega328).
- EEPROM Miktarı: 1 KB (ATmega328).
- Saat Frekansı: 16 MHz.
- Arduino Uno Kart Uzunluğu: 68.6 mm.
- Arduino Uno Kart Geniřliđi: 53.4 mm.
- Arduino Uno Kart Ađırlıđı: 25 g.



Şekil 3.14 Arduino Uno rev. 3 mikro denetleyici kontrol kartı (İnt. Kyn. 46).

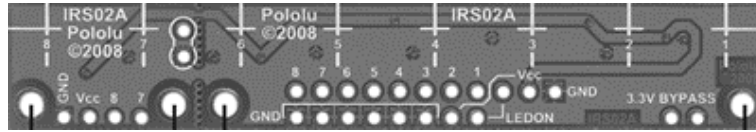
3.1.4 Kızılötesi Sensör

Konum belirleme işlemini gerçekleştirmek ve aracın çizgi üstünde stabil bir şekilde hareket etmesini sağlamak için iki adet QTR-8RC kızılötesi sensör kullanılmıştır. Bu sensörlerden birincisi aracın çizgiyi takip edebilmesi için ikinci ise kızılötesi sensörün position değerini kullanarak konum belirleme işleminin yapılabilmesi için

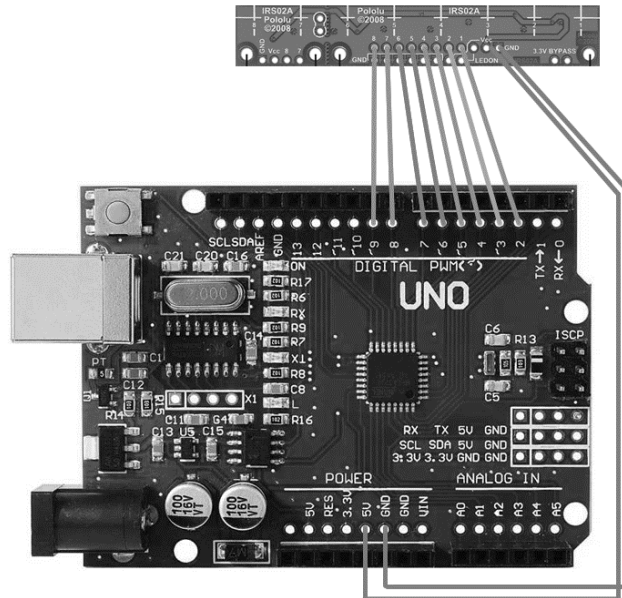
kullanılmaktadır. Kızılötesi sensörün özellikleri ve görünümü aşağıda verilmiştir (Şekil 3.15).

Özellikleri;

- Dijital yapıdaki bir sensördür.
- Sensör Uzunluğu: 75 mm.
- Sensör Genişliği: 13 mm.
- Sensör Kalınlığı: 3 mm.
- Çalışma Gerilimi Değer Aralığı: 3,3-5,0 Volt.
- Koşma Akımı: 100 mA.
- Çıkış Pini: 8 Dijital Giriş/Çıkış uyumlu pin.
- Minimum Algı Mesafesi: 3 mm.
- Maksimum Algı Mesafesi: 9,5 mm.
- Sensör Ağırlığı: 3,1 gr.



Şekil 3.15 QTR-8RC Kızılötesi sensörü (İnt Kyn. 47).



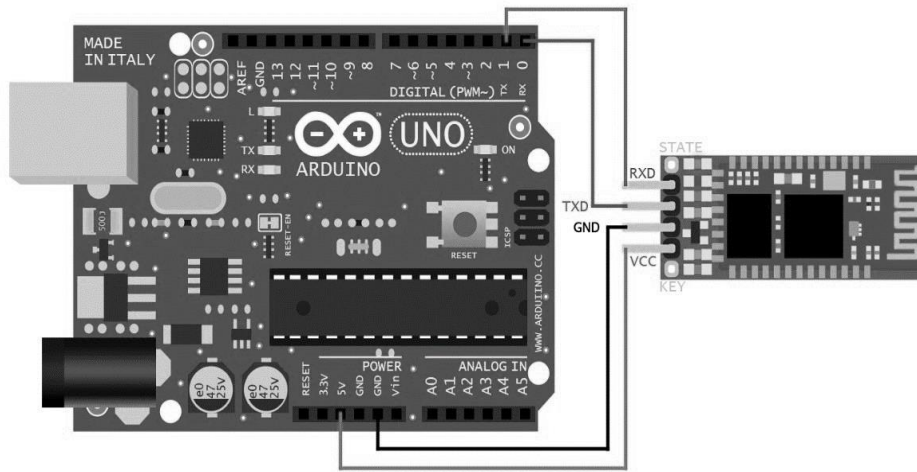
Şekil 3.16 QTR-8RC – arduino uno bağlantı şekli.

3.1.5 Bluetooth Modülü

Bu tez çalışmasında iki adet HC-06 bluetooth modülü kullanılmıştır. Bu modüller ile uygulama yazılımı ve otomatik kılavuzla araç arasındaki veri iletişimi gerçekleştirmektedir. Birinci bluetooth modülü aracın ileri, geri, sağa, sola yönlendirilmesi ve gerektiğinde durdurmak amacıyla araca komut göndermek için kullanılmaktadır. İkinci modül ise ivme sensöründen gelen yaw açısı değeri ve kızılötesi sensöründen gelen position değerini uygulama yazılımına göndererek konum belirleme maksadıyla kullanılmaktadır. Bluetooth modülünün özellikleri ve arduino bağlantı şekli aşağıda verilmiştir (Şekil 3.17).

Özellikleri;

- USB adaptörü ile çalışır.
- Varsayılan Baud Rate: 9600 bps.
- Dâhili Alıcı/Verici.
- İletişim Frekansı: 2.4 GHz.
- Hassaslık Değeri: $\leq -80\text{dBm}$.
- Gönderi Gücü: $\leq +4\text{dBm}$.
- Bluetooth Versiyon: V2.0+EDR.
- Çalışma Gerilim Miktarı: 3.3 Volt.
- Sinyal Kapsama Alanı: 30ft.
- Boyutları: 1.55 cm x 3.98 cm.



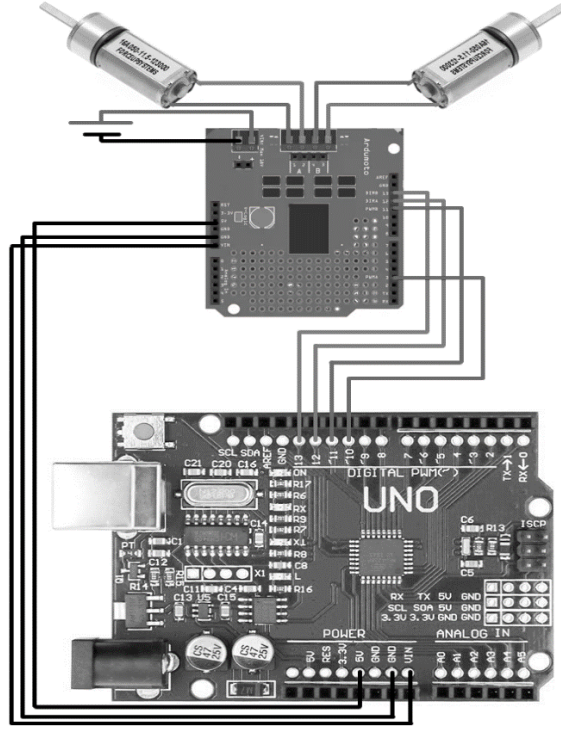
Şekil 3.17 HC-06 modül ve arduino uno bağlantı şekli.

3.1.6 Motor Sürücü Kartı

Redüktörlü DC motorların hareketini sağlamak için ArduMoto motor sürücü kartı kullanılmıştır. Bu kart iki adet motoru bağımsız olarak hız ve yön açısından kontrolünü gerçekleştirmektedir. Motor sürücü kartın özellikleri ve bağlantı şekli aşağıda verilmiştir (Şekil 3.18).

Özellikleri;

- L298P entegresi kullanılmıştır.
- Bağımsız iki motor için hız ve yön kontrolü.
- 2A çift kanallı motor sürücü.
- Voltaj ayarlayıcı, aşırı ısı ve yüksek akım koruması mevcuttur.
- Pin Uçları:
 - o OUT 1/2: Sağ motor çıkışlar pinleri.
 - o DIR A: Sağ motor yön pini.
 - o PWM A: Sağ motor hız pini.
 - o OUT 3/4: Sol motor çıkış pinleri
 - o DIR B: Sol motor yön pini.
 - o PWM B: Sol motor hız pini.
 - o VIN: En Yüksek +18 Volt
 - o GND: Toprak bağlantısı.
- VS: Besleme gerilimi (4.8V - 46V).
- VSS: Kart gerilimi (6V - 12V).
- Boyutları: 57mm x 43mm.



Şekil 3.18 ArduMoto L298P motor sürücü kartı ve arduino uno bağlantı şekli.

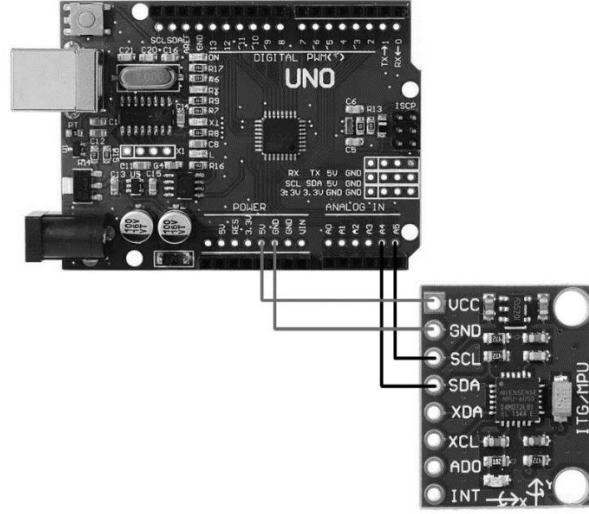
3.1.7 İvme ve Gyroscope Sensörü

Otomatik kılavuzlu aracın eş zamanlı olarak konumu belirlemek için MPU6050 ivme gyroscope sensörü kullanılmıştır. Bu sensör altı eksenli bir IMU sensör olmakla birlikte üç yönlü gyroscope ve üç yönlü açılı ivme ölçme işlemini gerçekleştirebilmektedir.

Araç düz bir zemin üzerinde hareke edeceği için roll, pitch, yaw açısı değerlerinden sadece yaw açısı değerinin hesaplanması yeterli olacaktır. MPU6050 sensörünün özellikleri ve bağlantı şekli aşağıda verilmiştir (Şekil 3.19).

Özellikleri;

- MPU6050 Entegresi.
- 3 yönlü jiroskop, 3 yönlü açılı ivme ölçerli 6 eksenli bir IMU sensör kartıdır.
- Çalışma Gerilimi Aralığı: 3.0 – 5.0 Volt.
- Gyro Ölçüm Aralığı: $\pm 250, 500, 1000, 2000$ °/s.
- Açılı İvme Ölçer Ölçüm Aralığı: $\pm 2, 4, 8, 16$ g.
- İletişim Ara yüzü: I2C.
- Boyutlar: 20,3 mm X 15,6 mm.



Şekil 3.19 MPU6050 sensörü ve arduino uno bağlantı şekli.

3.1.8 Lityum Polimer Batarya

Çalışmada kullanılan aracın enerjisini sağlamak için 11.1 Volt 500 mAh lityum polimer batarya kullanılmıştır. Lityum polimer bataryalar şarj edilebilir olması, değişik güç miktarlarına göre farklı hücre sayılarına sahip olması ve farklı kapasite seçenekleri nedeniyle tercih edilmektedirler. Araçta kullanılan lityum polimer bataryanın özellikleri ve görünümü aşağıda verilmiştir (Şekil 3.20).

Özellikler;

- Şarj edilebilir lityum polimer batarya.
- Batarya Hücre Sayısı: 3S.
- Depolanan Çalışma Gerilimi: 11.1 Volt.
- Elektrik Akımı Kapasite Miktarı: 500 miliAmper.
- Batarya Ağırlığı: 54 gr.
- Batarya Boyutları: 47 x 24 x 24 mm



Şekil 3.20 500 mA Lityum polimer batarya (İnt. Kyn. 51).

3.1.9 Tekerlekler

Aracın hareket edebilmesi ön ve arka tarafında ikişer adet olmak üzere toplam dört adet tekerlek kullanılmıştır.

Silikon Arka Tekerlek; aracın yüzey üzerinde hareket edebilmesi için arka tarafında iki adet tekerlek kullanılmıştır. Bu tekerlekler motorların miline sabitlenmiş vaziyettedir. Motorların ileri geri hareketleri ile tekerlekler yüzeye tutunarak aracın hareket etmesini sağlamaktadır. Yüzeye tutunma işleminin daha sağlam olabilmesi için silikon yapıdaki tekerlekler tercih edilmiştir. Tekerleklerin fiziksel özellikleri ve görünümü aşağıda verilmiştir (Şekil 3.21).

Özellikleri;

- Orta kısım deliğinin çapı 3 mm.
- Tekerlek Dışının Çapı: 30 mm.
- Tekerlek Genişliği: 23 mm.
- Tekerlek sertlik değeri: Shore 10:A.
- Tekerlek Ağırlığı: 23 gr.



Şekil 3.21 Bond silikon tekerlek ve motor bağlantı şekli (İnt Kyn. 39).

Sarhoş Ön Tekerlek; aracın ön tarafında iki adet plastik yapıda sarhoş tekerlek kullanılmıştır. Bu tekerlekler aracın sağa, sola, ileri, geri yönde 360° hareket edebilmesini sağlamaktadır. Sarhoş tekerlekler otomatik kılavuzlu araçlarda aracın özellikle dönme hareketi yapma esnasında hareketini kısıtlamaması için tercih edilmektedir. Sarhoş tekerleklerin fiziksel özellikleri ve görünümü aşağıda verilmiştir (Şekil 3.22).

Özellikleri;

- Plastik rulman.
- Şasi delik çapı: 12 mm.



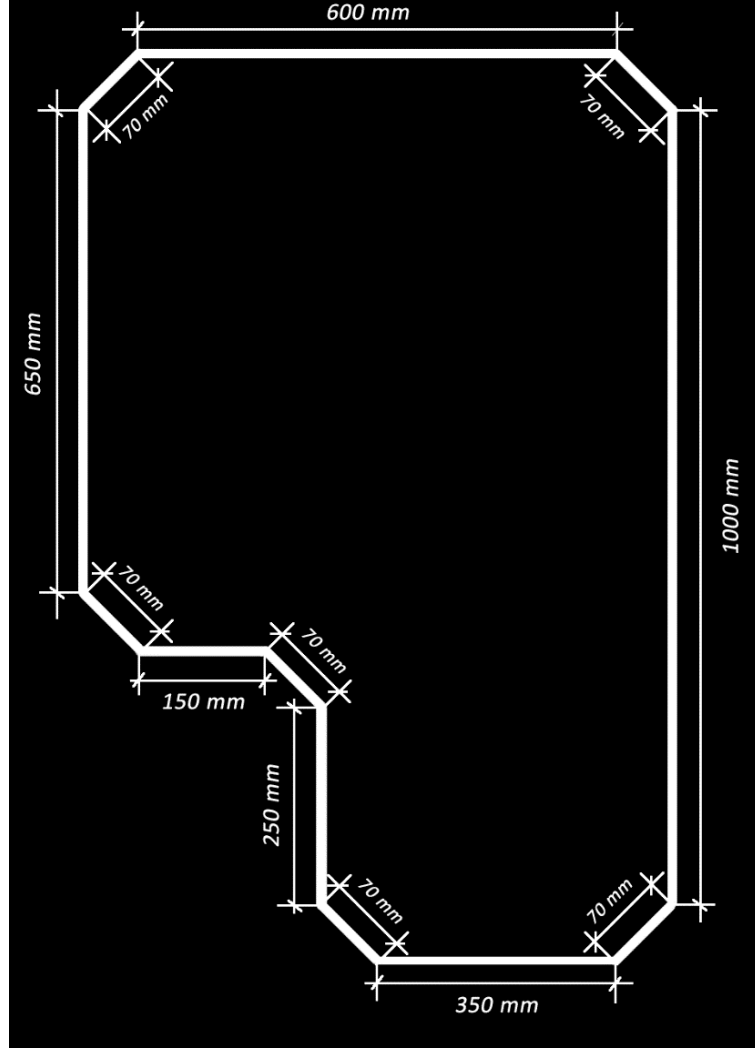
Şekil 3.22 Plastik sarhoş mini tekerlek (İnt. Kyn. 52).

3.2 Aracın Takip Edeceği Yol

Otomatik kılavuzlu araç çizgi üzerinde belirli güzergâhta hareket edecektir. Bu güzergâhta siyah zemin üzerindeki beyaz çizgiyi stabil şekilde takip ederek çalışacaktır. Şekil 3.23'te otomatik kılavuzlu aracın takip edeceği yol güzergâhı gösterilmiştir. Hazırlanan zemin 3 mm kalınlığındaki 70 x 100 cm'lik iki adet dekota levhanın yan yana birleştirilmesi ile elde edilmiştir.

Dekota levha özellikleri;

- Dekota levha hafif pvc malzemeden üretilmiştir.
- Yüzeyi düzgün, yapısı sert ve zemini siyah renktedir.
- Algılayıcı sensörün hata payını azaltmak için mat bir yapıya sahiptir.
- Dayanıklı olduğu ısı aralıkları: -40 ile +70 derece arasındadır.



Şekil 3.23 Otomatik kılavuzlu aracın takip edeceği yol güzergâhı.

3.3 Bilgisayar

Tezin gerçekleştirilebilmesi için gerekli yazılımlar ve bu yazılımlar ile kodlanan uygulama programlarının kullanılabilmesi için HP EliteBook 820 G2 model bilgisayar kullanılmıştır.

Hp EliteBook 820 G2 özellikleri;

- İşletim sistemi: Windows 10 Pro 64 bit
- Mikro işlemci: Intel® Core™ i5-5300U 2300 Mhz.
- Dahili ekran kartı: Intel® HD Grafik Kartı 5500.
- Toplam bellek: 8 GB 1600 MHz DDR3L SDRAM.
- Dahili sürücü:500 GB HDD.
- Ekran: 12,5" LED ekran (1366x768).

- Bağlantı noktaları.
 - 2 Adet USB 3.0, 1 Adet USB 3.0 şarj.
 - 1 Adet Display port 1.2.
 - 1 Adet VGA port.
 - 1 Adet birleşik kulaklık/mikrofon.
 - 1 Adet RJ-45.
 - 1 Adet yerleştirme konektörü.
- Genişletme yuvaları:1 çok biçimli dijital ortam okuyucu.
- Ağ arabirimleri.
 - Intel I218-LM GbE (10/100/1000 NIC).
 - Birleşik Intel® Çift Bantlı Kablosuz-AC 7265 802.11a/b/g/n/ac WiFi.

3.4 Bilgisayar Yazılımları

3.4.1 Visual C Sharp Programlama Dili

Otomatik kılavuzlu araçlarda eş zamanlı konum belirleme ve yönlendirme tezinin uygulama programını kodlamak için Microsoft yazılım şirketi tarafından geliştirilmiş C Sharp programlama dili kullanılmıştır.

Visual C Sharp programlama dili ilk olarak 2000 yılında ortaya çıkmıştır. 2002 yılında ise Visual Studio.Net tümleşik program geliştirme paketi içinde kullanılmaya başlanmıştır.

Visual C Sharp programlama dili genel özellikleri;

- Kolay, modern, genel programlama amacına uygun bir dildir.
- Sistem programlama dilidir.
- Gömülü sistem uygulama yazılımları için tasarlanmıştır.
- Nesne yönelimli bir programlama dili olarak tasarlanmıştır.
- Fonksiyonel ve esnek bir programlama dilidir.
- Güçlü veri tipleri ve veri yapılarına sahiptir.
- .Net çatısı altındaki sınıfları kullanabilir.

3.4.2 Arduino IDE

Arduino bir Giriş/Çıkış kartı ve işleme/kablolama programlama dilinin bir uygulama yazılımını içeren tümleşik geliştirme ortamından oluşan programlama platformudur. Arduino yalnız başına çalışan etkileşimli nesnelere geliştirmek için kullanılmakla birlikte bilgisayarda çalışan diğer yazılımlara da bağlanabilir.

Arduino kartları kapasiteleri ve bağlantı pin sayılarına göre çok farklı modeller sunmaktadır. Bu çeşitliliğin nedeni ise Arduino IDE'nin açık ve özgür bir yazılım platformu olmasından kaynaklanmaktadır.

Uno, Leonardo gibi başlangıç seviyesi kartlar haricinde, Mega, Zero gibi gelişmiş özelliklere sahip kartları mevcuttur. Nesnelere interneti için geliştirilmiş MKR ürün ailesi kartlarına ve giyilebilir teknolojiler için geliştirilmiş Gemma kartına ve LilyPad ürün ailesine sahiptir.

Arduino IDE, kod yazım editörü ve derleme işlemi için görev yapan, bununla birlikte derleme sonucu hatasız olan kaynak programın karta yüklenmesini sağlayan bir geliştirme ortamıdır. Platformdan bağımsız olarak çalışabilen Java dili ile yazılmış bir tümleşik geliştirme ortamıdır.

3.5 Sistemin Çalışma Prensipleri

Bu çalışmadaki temel amaç belirlenmiş bir güzergâh üzerinde hareket eden otomatik kılavuzlu aracın eş zamanlı olarak takip ettiği yolun rotasını oluşturabilmektir. Çalışmanın sonuca ulaşabilmesi için anlık olarak araç üzerindeki sensörlerden alınan çizgi pozisyonu, dönüş yönü, dönüş açısı, kat edilen mesafe gibi bilgiler birlikte değerlendirilip bir rota bilgisi elde edilmiştir. Elde edilen rota bilgisinin uygulama programında yer alan konum bilgisi alanında rota üzerinde gösterilmiştir. Sistemin temel çalışma prensibi robot çizgi sensörü vasıtasıyla çizgi üzerinde hareket halindeyken kızılötesi yansıma sensöründen gelen verilerle aracın takip ettiği yol ve motorların enkoderlerinden gelen bilgilerle mesafe ve yön bilgileri tespit edilmiştir. Verilerin uygulama programına transferi için aracın üzerindeki bir başka parça olan bluetooth

modülü kullanılmıştır. Bilgisayar ortamına gelen veriler işlenip aracın takip ettiği yolun haritası çıkarılacaktır.

3.5.1 Çizgi Takip Etme Sistemi

Bu çalışmada çizgi takibi için QTR-8RC kızılötesi sensörü kullanılmıştır. 11 girişli QTR-8RC modülünde 1, 2, 3, 4, 5, 6, 7, 8 numaralı pinler sensörlerden alınan verileri okumak için, VCC pini +5 V besleme ve GND pini topraklama için kullanılmıştır. IR pini kullanılmamıştır. Sistemin beyaz bant çekili pistte hareket ettirilmesi gerçekleştirilmiştir.

Çizgi takip etme sisteminde kullanılan QTR-8RC kızılötesi modülünün her bir sensörü beyaz rengi algıladığında 0 ya da zeminin parlaklığına beyaz renginin tonuna bağlı olarak 0'a yakın bir değer alır. Benzer şekilde siyah rengi algıladığında ise 1000 ya da zeminin matlığına ve siyah rengini tonuna bağlı olarak 1000'e yakın bir değer alır. Sensörlerden gelen değerler QTR-8RC kütüphanesinin readLine fonksiyonu ile anlık olarak okunur ve sensörlerin çizgi üzerindeki pozisyonu formül 3.1'e göre hesaplanmaktadır.

$$Position = \frac{0*Value0+1000*Value1+2000*Value2+\dots+7000*Value7}{Value0+Value1+Value2+\dots+Value7} \quad (3.1)$$

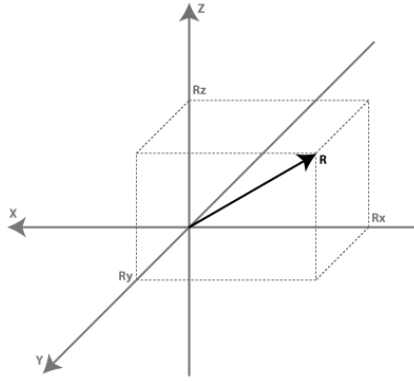
Hesaplanan bu pozisyon değerinin tüm sensörler aktif iken olması gerek optimum değeri 3500'dür. Eğer hesaplan değer 3500 ise aracın 4. ve 5. Sensörlerden beyaz çizgiyi takip ettiği anlamına gelmektedir. Bu 4 ve 5 numaralı sensörler QTR-8RC modülünün en ortasında yer alan sensörlerdir. Bu durumda beyaz çizginin düzgün şekilde takip edildiği anlaşılmaktadır. Modül beyaz çizgiyi tam ortadan takip etmek yerine 1,2 ya da 3 numaralı sensörler üzerinden takip ederse readLine fonksiyonu tarafından hesaplanacak pozisyon değeri 3500'ün üstüne çıkmaktadır. Bu durumda aracın sola doğru ilerlediği görülmektedir ve sağ motorun hızı düşürülüp sol motorun hızı yükseltilerek aracın sağa doğru hareket ettirilmesi gerekmektedir.

Modül beyaz çizgiyi tam ortadan takip etmek yerine 6,7 ya da 8 numaralı sensörler üzerinden takip ederse readLine fonksiyonu tarafından hesaplanacak pozisyon değeri 3500'ün altına inmektedir. Bu durumda aracın sağa doğru ilerlediği görülmektedir ve sağ motorun hızı yükseltilip sol motorun hızı düşürülerek aracın sola doğru hareket ettirilmesi

gerekmektedir. Sensörlerden gelen değerlerden anlık olarak hesaplanan pozisyon değerine göre aracı beyaz çizgiyi tam ortadan takip edebilmesi için sağ ve sol motora gerekli PWM değerleri aktarılmalıdır.

3.5.2 İvme Hesaplama Sistemi

İvmeölçer cihazı üzerine düşen yer çekimi ya da ani hız ya da ani duraksama durumlarındaki ivme miktarını ölçer. Sensörün verdiği değeri m/s² ya da yerçekimi (g-Force) cinsinden belirlenebilir. Uygulamalarda genel olarak yer çekimi cinsinden belirlenmektedir. Uzay ya da rastgele çekim alanı kapsamı içinde değilse sensör üstüne 1 g'lık yer çekimi kuvveti etki eder. Buda aşağı yukarı 9.8 m/s²'dir ve dünyada bulunan yere göre değişiklik gösterir (Şekil 3.24) (İnt.Kyn.31).



Şekil 3.24 İvmeölçer hesabı için koordinat sistemi (İnt. Kyn.31).

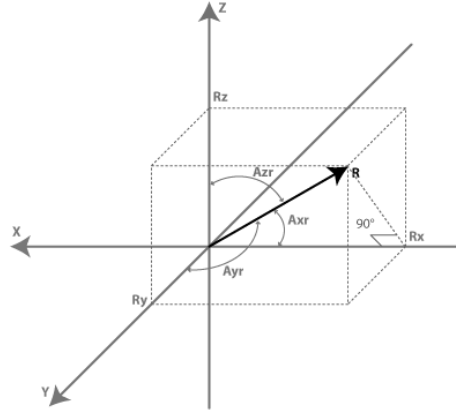
R vektörü ivmeölçerin üstündeki kuvvet vektörü olsun. İlgili kuvvet yer çekimi ya da sensör hareketinin sonucunda yer çekimi kuvveti ile eylemsizlik kuvvetinin bileşkesi olur. R vektörü 3 bileşene sahiptir ve $R=[R_x,R_y,R_z]$.

R vektörü hesabı:

$$R^2 = R_x^2 + R_y^2 + R_z^2 \quad (3.2)$$

R_x , R_y ve R_z bileşenleri bilindiği takdirde Şekil3.40' ta görüleceği gibi trigonometrik fonksiyon yardımıyla R vektörü X ve Y eksenleri ile oluşturduğu açıyı hesaplayarak

sensörün yeryüzüne göre konumu bulunabilir. R vektörü bileşeni de ivmeölçer g kuvveti türünden vermektedir (Şekil 3.25) (İnt. Kyn.31).



Şekil 3.25 İvmeölçer hesabı için R vektörünün eksenlere yaptığı açı (İnt. Kyn.31).

Yukarıda şekil ile gösterilmiş R vektörü X ve Y eksenleri ile yapmış olduğu açı hesaplanacak olursa. Şekildeki görünen Axr açısına ait cosinüs Rx/R değerini verir. Rx ve R değerlerinden arccosinüs ile Axr açısı elde edilmektedir. Aynı durum Ayr açısının da hesaplanmasında kullanılmaktadır. R vektörü de yukarıda ilk eşitlikten hesaplanıp aşağıda arccos fonksiyonuna değerler yerleştirildiğinde istenilen açılar bulunmaktadır (İnt. Kyn.31).

R vektörünün X eksenine göre açısı:

$$\cos(Axr) = Rx/R, Axr = \arccos(Rx/R) \quad (3.3)$$

R vektörünün Y eksenine göre açısı:

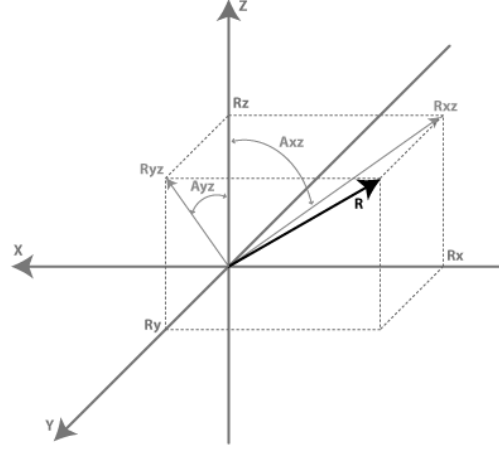
$$\cos(Ayr) = Ry/R, Ayr = \arccos(Ry/R) \quad (3.4)$$

R vektörünün Z eksenine göre açısı:

$$\cos(Azr) = Rz/R, Azr = \arccos(Rz/R) \quad (3.5)$$

Jiroskoplar basit şekilde bir tekerin kendi eksenini etrafında hızlı bir şekilde dönmesi sonucunda oluşmaktadır. Tekerin etrafında yer alan çemberin dik açı ile bağlanmış

diğer bir çember ve bu çemberlere dik açı ile sabitlenmiş başka bir çember jiroskobun modelini oluşturmaktadır (Şekil 3.26) (İnt. Kyn.31).



Şekil 3.26 Koordinat sisteminde R vektörü için izdüşümleri (İnt. Kyn.31).

İki eksen (X,Y) gyro kullandığını varsayarsak. Burada R vektörü için X-Z uzayında oluşan izdüşümünü Rxz, Y-Z uzayında izdüşümünü ise Ryz vektörü ile belirlenmektedir. Bu vektörler pisagor teoremiyle;

Rxz vektörü hesabı:

$$R_{xz}^2 = R_x^2 + R_z^2 \quad (3.6)$$

Ryz vektörü hesabı:

$$R_{yz}^2 = R_y^2 + R_z^2 \quad (3.7)$$

Vektörlerin Z eksenini ile yaptığı açı değerleri Axz ve Ayz' dir. Sistem Y ekseninin çevresinde döndürüldüğünde Axy açısı, X ekseninin çevresinde döndürüldüğünde ise Ayz açı değeri değişir. Jiroskop dönme hızı ölçmektedir. Dönme hızı zaman ile çarpılırsa dönme açısı bulunur. t0 anındaki açı Axz0 olduğu, t1 anındaki açının ise Axz1 olduğu varsayılacak olursa.

Dönüş açısı hesabı:

$$(Axz1 - Axz0) = RateAxz * (t1 - t0) \quad (3.8)$$

Sensörden alınan değerleri dönüş hızına çevirmek içinse aşağıdaki formüller kullanılmaktadır.

Dönüş hızı hesapları:

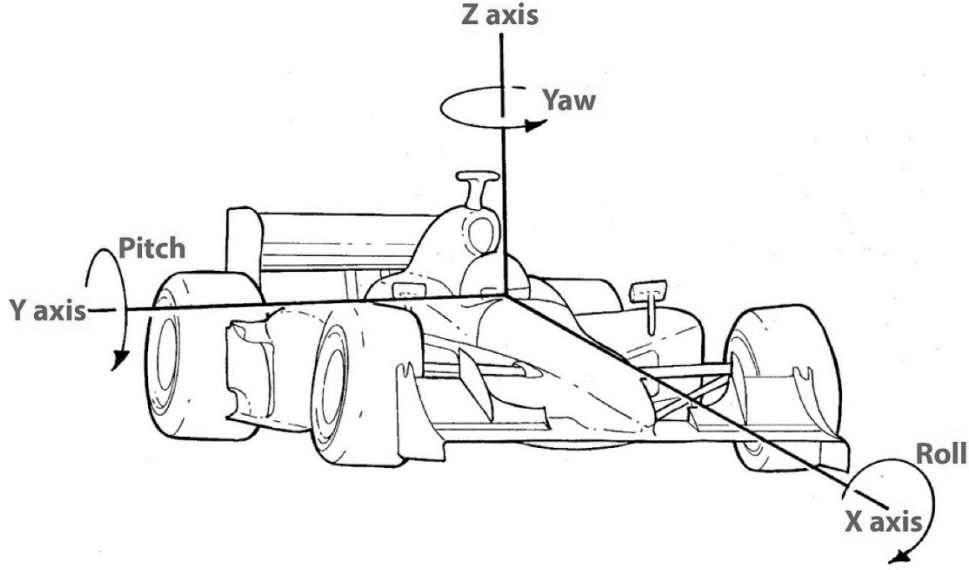
$$RateAxz = (AdcGyroXZ * \frac{Vref}{1023} - VzeroRate) / Sensitivity \quad (3.9)$$

$$RateAyz = (AdcGyroYZ * \frac{Vref}{1023} - VzeroRate) / Sensitivity \quad (3.10)$$

Çalışmada kullanılan MPU6050 sensörün SDA pini Uno'da Analog 4 pinine, sensörün SCL pini Uno'da Analog 5 pinine bağlanmıştır.

Otomatik kılavuzlu araç düz bir zeminde hareket edeceği için bu tezde sadece aracın sağa ve sola dönüş hareketleri sırasında yapacağı dönüş açısı gerekmektedir. MPU6050 sensörü ile roll, pitch ve yaw açı değerleri hesaplanabilmektedir. Roll pitch ve yaw hareketleri Şekil 3.27'de gösterilmiştir.

- Aracın önünden arkasına doğru olan(X eksen) eksen etrafında yapılan dönme hareketine roll denir.
- Aracın sağ yanından sol yanına doğru olan(Y Eksen) eksen etrafında yapılan dönme hareketine pitch denir.
- Aracın dikey eksen etrafında yapılan dönme hareketine yaw denir.



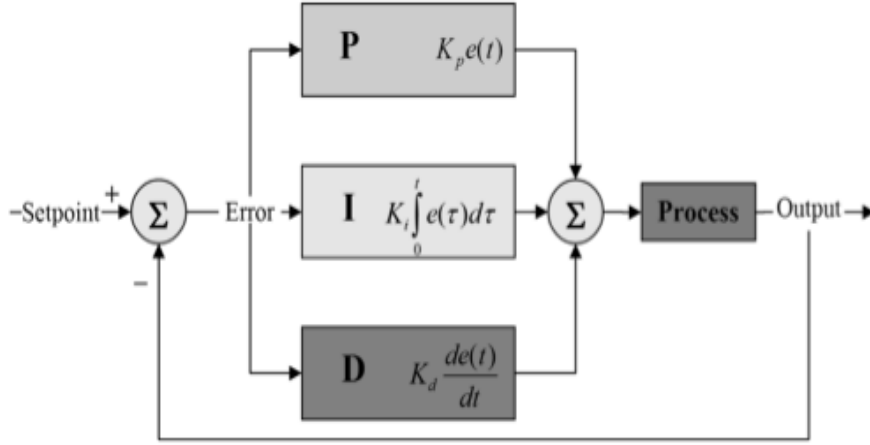
Şekil 3.27 Aracın X,Y,Z eksenleri için roll, pitch ve yaw hareketleri (İnt. Kyn.40).

Yaw değeri Z ekseninde yapılan dönme hareketini ifade etmektedir. Bu nedenle bu tezde otomatik kılavuzla araç düz zemin üzerinde hareket edeceği için dönme hareketleri sırasındaki yaw açısı değeri aracın sağa ya da sola dönme işlemini ne kadar yaptığının tespit edebilmesi amacıyla kullanılmaktadır.

3.5.3 Oransal Integral Türev (PID) Kontrolü

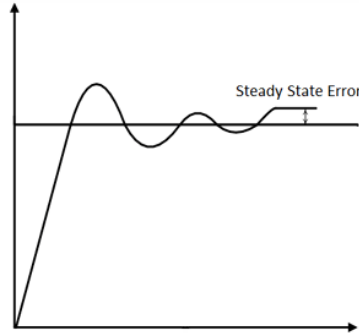
Bu tez çalışmasında çizgiyi takip eden sağ ve sol motorlar için aracın yalpalama yapmasını önlemek amacıyla PID kontrolü kullanılmaktadır. Normal şartlarda araca bu kontrol uygulanmadığı takdirde düşük hız ile ileri yönde hareket ederken çizginin takip edilmesinde herhangi bir sorun bulunmamaktadır. Ancak hız miktarı arttırılınca araç düz çizgiyi istikrarlı olarak takip etmekle beraber dönüş noktalarında yalpalama ve hatta çizgi dışına çıktığı görülmektedir. Bu durumu engellemek için PID kontrolü kullanılmaktadır.

PID (oransal-integral türev) denetleyici, genel bir kapalı döngü denetleyicidir. Endüstride kullanılan kontrolörlerin %80'inden fazlası PID kontrolörlerdir. Bir PID kontrolörünün üç terimi vardır: orantılı, integral ve türev terimi. Bu üç ayrı terim, üç sonucu ortaya çıkarmaktadır. Üç çıkışın toplamı, motor sürücülerinin olması gereken hızlarını ayarlamak için kullanılmaktadır. Şekil 3.28'de PID Kontrol blok şeması gösterilmiştir.



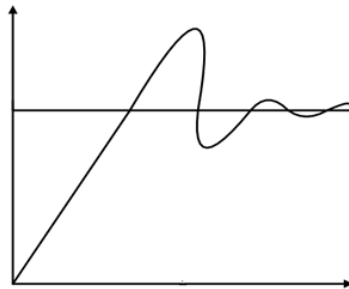
Şekil 3.28 PID Kontrol blok şeması (İnt. Kyn.35).

Orantılı Terim(P), Kazanç olarak da bilinen oransal terimi ifade eder, mevcut hata değerini orantısız kazanç K_p ile çarpılarak bulunmuştur. Oransız kazanç, çıkıştaki değişimin hızını tanımlamaktadır (Şekil 3.29).



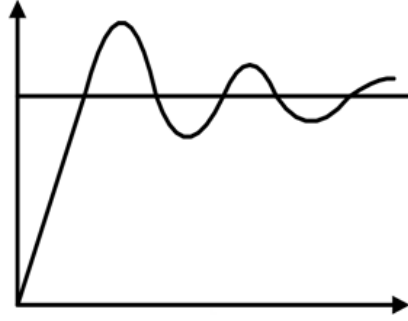
Şekil 3.29 PID Kontrol, Orantısız Kazanç K_p grafiği (İnt. Kyn.36).

İntegral Terim, integral terimi zaman içindeki oluşan hata değerlerinin toplamıdır. Bu sistemde zaman içinde düzeltilmesi gereken birikmiş hataları vermektedir. Hataların toplamı integral kazanç K_i ile çarpılarak bulunmuştur (Şekil 3.30).



Şekil 3.30 PID Kontrol, İntegral Kazanç K_i grafiği (İnt. Kyn.36).

Türev Terim, hatalardaki deęişim oranını belirler ve türev kazanç deęeri K_d ile çarpılarak bulunmaktadır (Şekil 3.31).



Şekil 3.31 PID Kontrol, Türev Kazanç K_d grafięi (İnt. Kyn.36).

PID deęerlerini hesaplamamanın çeşitli yöntemleri vardır.

- Deneme Yanılma Yöntemi
- Cohen-Coon Yöntemi
- Yuwana- Seborg Yöntemi
- Ziegler-Nichols Yöntemi

Deneme yanılma yönteminde yeterli miktarda deney yapılarak olması gereken K_p , K_d , K_i deęerleri bulunmaya çalışılmaktadır. Sistemin en stabil çalıştığı deęerlerin bulunması yapılan deney sayısına baęlıdır. Otomatik kılavuzlu araçlarda genellikle integral kazanç K_i deęeri sıfır kabul edilmektedir. Oransal deęer K_p ve türev kazanç deęeri K_d deęerleri üzerinde yeterli miktarda deneyler gerçekleştirilmektedir. Bu tezde K_p ve K_d deęerleri, benzer projelerde bulunan deęerler baz alınarak yapılan deneyler sonucunda deneme yanılma yöntemiyle bulunmuştur. K_i deęeri sıfır kabul edilmektedir.

3.5.4 Mesafe ve Dönme Açısı Kontrolü

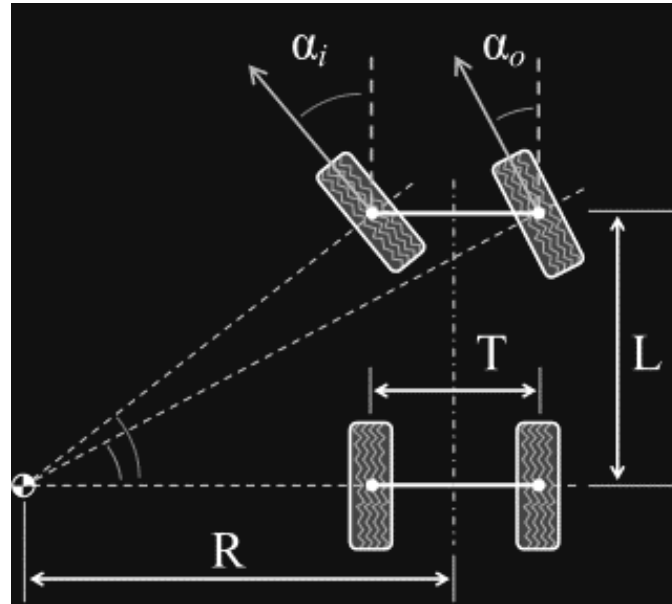
Prototip araçta iki adet enkoderli DC motor kullanılmıştır. Enkoder motordaki mekanik dönme hareketini dijital ya da analog sinyale çevirebilen bir kodlayıcı cihazdır. Konum algılama, hız ve yön takibi sistemlerinde kullanılmaktadır. Oluşturulan prototip araçta kullanılan motorların enkoder çözünürlükleri 48 CPR deęerindedir. Motorların 360° derecelik bir tam turu atması ile motora baęlı enkoder 48 pulse deęerini göndermektedir.

Aracın rota üzerinde hareket etmesini sağlayan arka tekerleklerin dış çapı ise 30 mm'dir. Tekerleklerin rota üzerindeki hareketleri sonucunda kat edilen mesafe, dairenin çevresi ve enkoderlerden gelen pulse değeri ile çarpılarak bulunabilmektedir.

$$Mesafe = Gelen Pulse Değeri * (2 * \pi * r) \quad (3.11)$$

Yukarıdaki 3.11'de belirtilen formülde Gelen Pulse Değeri motorların enkoderlerinden gelen pulse değerini r ise arka tekerleklerin yarıçapını ifade etmektedir.

Tekerleklerin dönme açılarının hesabı Ackerman Geometrisi sistemine göre hesaplanmaktadır.



Şekil 3.32 Ackerman Geometrisi sistemine göre dönme açıları (İnt. Kyn.60).

Şekil 3.32'de Ackerman Geometrisine göre dönme açıları hesabı verilmiştir. Bu şekilde T her tekerleğin merkez çizgisi ile arasındaki mesafeyi, L aracın iki aks arasındaki mesafesini, R aracın merkez çizgisi ile dönme merkezi arasındaki yarıçapı, α_i iç tekerleğin dönme açısını, α_o ise dış tekerleğin dönme açısını göstermektedir.

Bu durumda iç tekerleğin dönme açısı:

$$\alpha_i = \tan^{-1} * \left(\frac{L}{R - \frac{T}{2}} \right) \quad (3.12)$$

Dış tekerleğin dönme açısı:

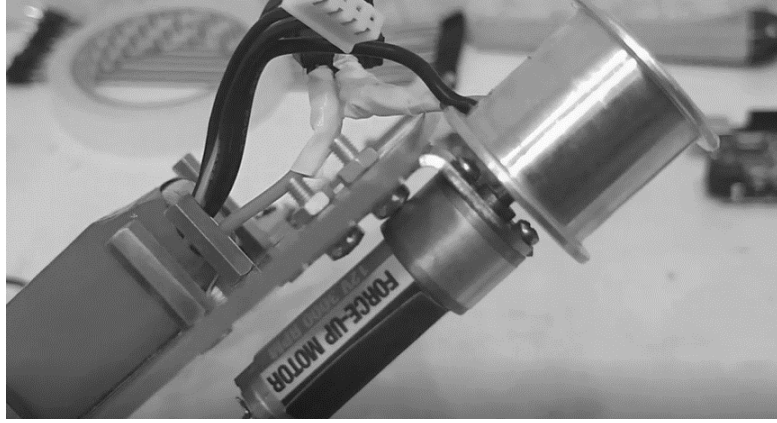
$$\alpha_o = \tan^{-1} * \left(\frac{L}{R + \frac{T}{2}} \right) \quad (3.13)$$

Formül 3.12 ile iç tekerlek dönme açısı, 3.13 ile dış tekerlek dönme açısı hesaplanmaktadır.

4. BULGULAR

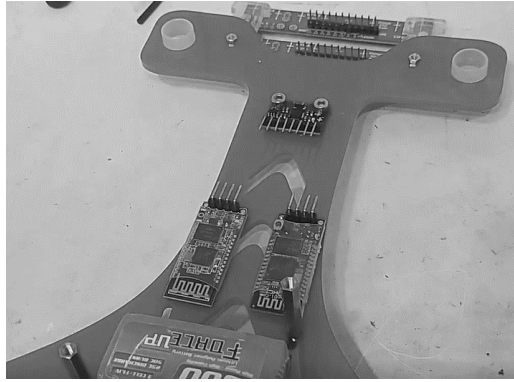
4.1 Uygulama Adımları

Uygulamanın ilk adımında araç şasesi üzerine motor sabitleme aparatları takılarak motorlar bu aparatlar vasıtasıyla şaseye sabitlenmiştir. Daha sonra motorların mil uçlarına arka tekerlekler sabitlenmiştir (Şekil 4.1).



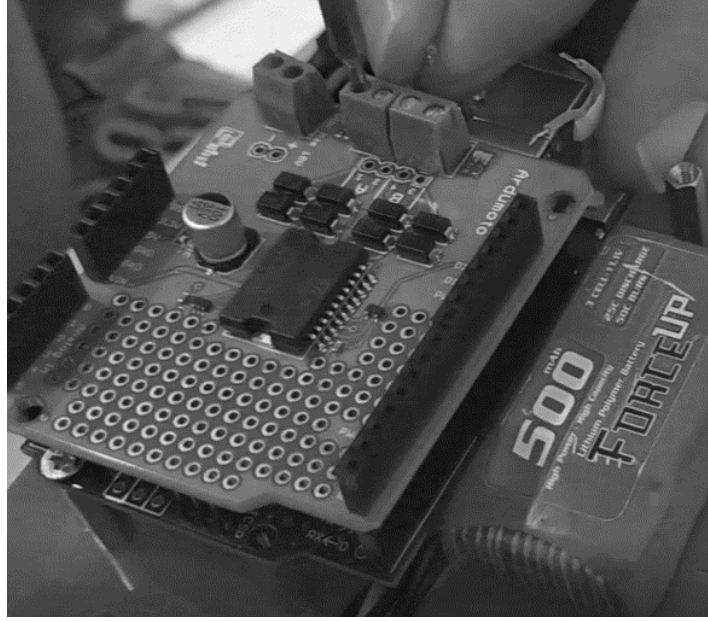
Şekil 4.1 Araç şasesine motor ve tekerleklerin sabitlenmesi.

Bir sonraki adımda kızılötesi sensörler, ivme sensörü, ön tekerlekler ve bluetooth modülleri ve lityum polimer bataryaya araç şasesine sabitlenmiştir (Şekil 4.2).



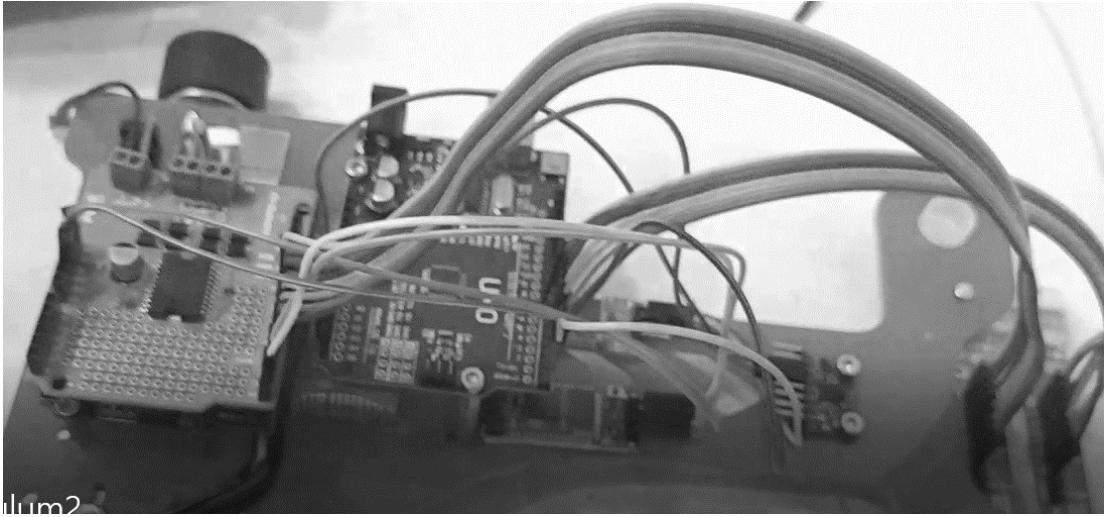
Şekil 4.2 Şaseye sensör, tekerlek, bluetooth modülleri ve bataryanın sabitlenmesi.

Bir sonraki adımda genişleme modülü şeklindeki motor sürücü kartının sabitlenebilmesi için kontrol kartı şaseye sabitlenmiş ve motor sürücü kartı kontrol kartının üzerine geçirilmiştir. Daha sonra motor sürücü kartı ile motor bağlantıları gerçekleştirilmiştir (Şekil 4.3).



Şekil 4.3 Motor sürücü kartı ve motor bağlantılarının yapılması.

Bir sonraki adımda verileri almak maksadıyla kullanılan kontrol kartı şaseye sabitlenerek kablo bağlantıları yapılmıştır (Şekil 4.4).



Şekil 4.4 Kablo bağlantılarının yapılması.

Motorları sürmek için ArduMoto kontrol kartındaki Out1, Out2, Out3, Out4 çıkışları ile aracın sağ ve solunda bulunan 12 Volt 3000 RPM DC motorların bağlantısı gerçekleştirilmektedir (Çizelge 4.1).

Çizelge 4.1 ArduMoto - DC Motor bağlantıları.

Çıkış Pinleri	Açıklama
Out 1	Sağ DC motor
Out 2	Sağ DC motor
Out 3	Sol DC motor
Out 4	Sol DC motor

Arduino’da ArduMoto kontrol kartı DC motorların hız ve yön kontrollerini gerçekleştirebilmek amacıyla kontrol kartındaki PWMA, PWMB, DIRA, DIRB pinlerinin bağlantıları yapılmıştır. Motor yönlerini ayarlamak için dijital çıkış pinlerine HIGH ve LOW değerleri gönderilmektedir her iki pine de HIGH ya da her iki pine de LOW değeri gönderilirse gerilim olmayacağı için motorlar çalışmayacaktır. Buna ilişkin kodlama yaparken belirtilmesi gereken değerler Çizelge 4.2’de gösterilmiştir.

Çizelge 4.2 Arduino-ArduMoto pin bağlantıları.

ArduMoto Çıkış Pinleri	Açıklama	Arduino Pinleri	PWM-Yön (HIGH/LOW)
PWMA	Sağ DC motor	Dijital 3	PWM
PWMB	Sol DC motor	Dijital 11	PWM
DIRA	Sağ DC motor	Dijital 12	HIGH /LOW
DIRB	Sol DC motor	Dijital 13	HIGH /LOW

MPU6050 sensör bağlantıları; bu sensör analog değerler ürettiği için Uno’da analog pin girişleri ya da kendi çıkış isimleri ile adlandırılan pinlerin kullanılması gerekir. Buna ilişkin bağlantı pinleri Çizelge 4.3’te gösterilmiştir.

Çizelge 4.3 MPU6050-Arduino pin bağlantıları.

MPU6050 Çıkış Pinleri	Arduino Pinleri
MPU6050 SDA pini	Analog A4
MPU6050 SCL pini	Analog A5
MPU6050 VCC	Power 5V
MPU6050 GND	Power GND

11 pinli QTR-8RC çizgi izleme sensörünün uygulamada 10 pini kullanılmıştır. Bu pinlerden VCC pini Uno'da +5 V çıkışına, GND pini Uno'da GND pinine bağlanmıştır çizgi izleme sensör pinlerinin ve diğer pinlerin bağlantıları Çizelge 4.4' te gösterilmiştir.

Çizelge 4.4 QTR-8RC-Arduino pin bağlantıları.

QTR-8RC Çıkış Pinleri	Arduino Bağlantı Pinleri
1 numaralı pin	Dijital 2
2 numaralı pin	Dijital 4
3 numaralı pin	Dijital 5
4 numaralı pin	Dijital 6
5 numaralı pin	Dijital 7
6 numaralı pin	Dijital 8
7 numaralı pin	Dijital 9
8 numaralı pin	Dijital 10
VCC pini	Power 5V
GND pini	Power GND

Bluetooth HC-06 bağlantıları, VCC pini Uno'da +5 V çıkışına, GND pini Uno'da GND pinine RX pini Uno'da 1 numaralı TX Dijital çıkışa, TX pini ise Uno'da 0 numaralı RX Dijital çıkışa bağlanmıştır. HC-06 veri alışveriş bağlantıları Çizelge 4.5'te gösterilmiştir.

Çizelge 4.5 HC-06 Bağlantıları.

HC-06 Çıkış Pinleri	Arduino Pinleri
RX pini	Dijital 1 TX
TX pini	Dijital 0 RX
VCC pini	Power 5V
GND pini	Power GND

Prototip araçta, sağ motor enkoderinin Data A pini Uno'da Dijital 2 pinine , Data B pini

Uno'da Dijital 4 pinine, sol motor enkoderinin Data A pini Uno'da Dijital 3 pinine, Data B pini Uno'da Dijital 5 pinine bağlanmıştır. Enkoderlerin VCC pinleri Uno'da +5 V çıkışına, GND pinleri Uno'da GND çıkışlarına bağlanmıştır. Enkoder pinlerinin bağlantıları Çizelge 4.6'da gösterilmiştir.

Çizelge 4.6 Enkoder Bağlantıları.

Enkoder	Arduino
Çıkış Pinleri	Pinleri
Sağ Motor Pin A	Digital 2
Sağ Motor Pin B	Digital 4
Sol Motor Pin A	Digital 3
Sol Motor Pin B	Digital 5
Enkoder VCC Pinleri	Power 5V
Enkoder GND Pinleri	Power GND

Çalışmada iki adet Arduino Uno kontrollü araçta bir adet MPU6050 ivme ve gyro sensörü ve iki adet QTR-8RC kızılötesi sensörü olmak üzere toplam üç adet sensör kullanılmıştır.

Sensör pinlerinin bağlantıları yukarıdaki çizelgelerde belirtilmiştir. Tüm sensörlerden gelen bilgileri okuyup uygulama programına aktarmak için iki adet HC-06 bluetooth modülü kullanılmıştır. Gelen veri paketleri ayrıştırılıp QTR-8RC sensöründen gelen position değeri, dönme noktalarında MPU6050'den gelen yaw açısı değerleri geliştirilen algoritma ile yorumlanıp uygulama programında rota çizdirilmiştir.

4.2 PID Kontrol Uygulama Adımları

Tez çalışmasında aracın beyaz çizgi üzerinde sağa sola yalpalama yapmadan düzgün şekilde hareket edebilmesi için oransal sabit(K_p) ve türevsel sabit(K_d) değerleri kullanılmıştır. İntegral sabiti(K_i) değerlendirmeye alınmamıştır. Deneme yanılma yöntemiyle K_p ve K_d değerleri tespit edilmiştir.

Deneme yanılma yöntemiyle ilk olarak K_p değeri 0.70, K_d değeri ise 2.0 olarak kabul edilmiştir. Aracın bu değerle hareketi sonucunda aşırı yalpalama yaptığı görüldükten

sonra deęerler Kp deęeri iin 0.10 ve Kd deęeri iin 0.20 miktarlarında adım adım azaltılmıřtır. 12. adım sonunda Kp deęeri 0.10, Kd deęeri ise 0.80 deęerlerinde aracın herhangi bir yalpalama yapmadan stabil olarak beyaz izgiyi takip ettięi grlmřtr. Bu deęerler ile birlikte hata, son hata ve hız farkları deęiřkenleri de programda kodlanmıřtır (řekil 4.5).

```
int FAULT, COR_SPEED;

int LAST_FAULT = 0;
float Kp = 0.10;
float Kd = 0.80;
```

řekil 4.5 PD parametreleri program kodu.

Kp ve Kd deęerleri kullanılarak hesaplanan pozisyon deęeri iin PID yapısıyla motorlara uygulanması gereken hız farklarının da hesaplanabilmesi iin programda gerekli kodlama yapılmaktadır (řekil 4.6).

```
COR_SPEED = Kp * FAULT + Kd*(FAULT - LAST_FAULT);
LAST_FAULT = FAULT;
```

řekil 4.6 PID yapısı ile hız farkı hesaplama program kodu.

PID kontroll araç kullanılması nedeniyle aracın beyaz izgiyi dzgn ve istikrarlı takip edebilmesi amacıyla ařaęıdaki adımların takip edilmesi gerekir.

- Yalnızca 1 nolu sensr beyaz izgiyi algıladıęında, araç sola doęru beyaz izgiden ayrılmak zeredir. Aracın saęa dnmesi gerekmektedir
- 1 ve 2 nolu sensrler beyaz izgiyi algıladıęında, araç sola doęru hareket etmektedir. Aracın saęa dnmesi gerekmektedir.
- 2 ve 3 nolu sensrler beyaz izgiyi algıladıęında, araç sola doęru hareket etmektedir. Aracın saęa dnmesi gerekmektedir.
- 3 ve 4 nolu sensrler beyaz izgiyi algıladıęında, araç sola doęru hareket etmektedir. Aracın saęa dnmesi gerekmektedir.
- 4 ve 5 nolu sensrler beyaz izgiyi algıladıęında, araç doęru hareket etmektedir.
- 5 ve 6 nolu sensrler beyaz izgiyi algıladıęında, araç saęa doęru hareket etmektedir. Aracın sola dnmesi gerekmektedir.

- 6 ve 7 nolu sensörler beyaz çizgiyi algıladığında, araç sağa doğru hareket etmektedir. Aracın sola dönmesi gerekmektedir.
- 7 ve 8 nolu sensörler beyaz çizgiyi algıladığında, araç sağa doğru hareket etmektedir. Aracın sola dönmesi gerekmektedir.
- Yalnızca 8 nolu sensör beyaz çizgiyi algıladığında, araç sağa doğru beyaz çizgiden ayrılmak üzeredir. Aracın sola dönmesi gerekmektedir.

4.3 QTR-8RC Sensörlerinden Okunan Değerlere Göre Yön Tayini İşlemi

QTR-8RC dijital sensöründe, sensörlerden her biri 0 ile 1000 arasında bir değer döndürür. Sensör siyahı çizgiyi gördüğünde zeminin parlaklığına, ortamdaki ışık miktarına ve yansıma gibi etkenlere bağlı olarak 1000' e yakın bir değer döndürür. Eğer sensör beyaz rengi görüyor ise yine aynı etkenlere bağlı olarak 0'a yakın bir değer döndürmektedir. Yani sensörün standart değerlerinde 1000 siyah zemini, 0 ise beyaz zemini ifade etmektedir.

Yukarıda bahsedilen çevresel etkenler nedeniyle sensör çoğu zaman tam olarak 0 ya da 1000 değerini döndürmemektedir ancak zemine göre bu değerlere yakın değerleri döndürdüğü görülmektedir. Örneğin beyaz zeminde 250, siyah zeminde 800 gibi. Sensörün bu alt ve üst sınırları tespit edebilmesi için çizgi takibine başlamadan önce sensörün ilgili fonksiyonları ile kalibrasyonunun yapılması gerekmektedir. Bu sayede elde edilen minimum değer 0'a (örneğin 250), maksimum değer ise 1000'e (örneğin 800) eşitlenir ve yön tayininde ya da başka bir ifadeyle sensörlerden hangisinin siyah çizginin üstünde olduğu hangisinin beyaz çizginin üstünde olduğu ağırlıklı ortalama yöntemiyle bulunur. Ağırlıklı ortalama bulmak için 4.1'de gösterilen ana formül kullanılmaktadır.


$$Position = \frac{0*Value0+1000*Value1+2000*Value2+\dots+7000*Value7}{Value0+Value1+Value2+\dots+Value7} \quad (4.1)$$

Bu hesaplama yönteminde kullanılan sensörler 0 indis değerinden başlayarak 1000, 2000, vs. şeklinde devam eder uygulamada 8 renk algılayıcıyı sensörün tamamı kullanıldığı için son sensörün indisi 7000'e karşılık gelir value terimi ise o sensörün okuduğu değeri göstermektedir.

Uygulamada 8 sensör kullanılmaktadır bu maksatla beyaz çizginin düzgün takip edilebilmesi için 4. ve 5. sensörlerin çizginin üstünde olması gerekmektedir. Bu durumda ağırlıklı ortalama hesabına göre arzu edilen pozisyon değerinin 3500 olması gerekir. Bu nedenle sensörlerden ağırlıklı ortalama yöntemiyle hesaplanan position değeri 3500 ise, hedef değer de 3500 olduğu için hata değeri 0 olacaktır ve motor farkı ortaya çıkmayacaktır.

Eğer sensörlerden gelen ortalama değer 3500 değerinden daha düşük ise aracın sağa doğru ilerlediği görülmektedir ve buna uygun olarak motor farkları sağ motora ilave edilmektedir. Benzer şekilde hesaplanan position değeri 3500 değerinden daha yüksek ise bu defa aracın sola doğru hareket yaptığı görülmektedir ve ilgili motor farkları sol motora ilave edilerek çizgiyi yalpalama yapmadan izlemesi sağlanmaktadır.

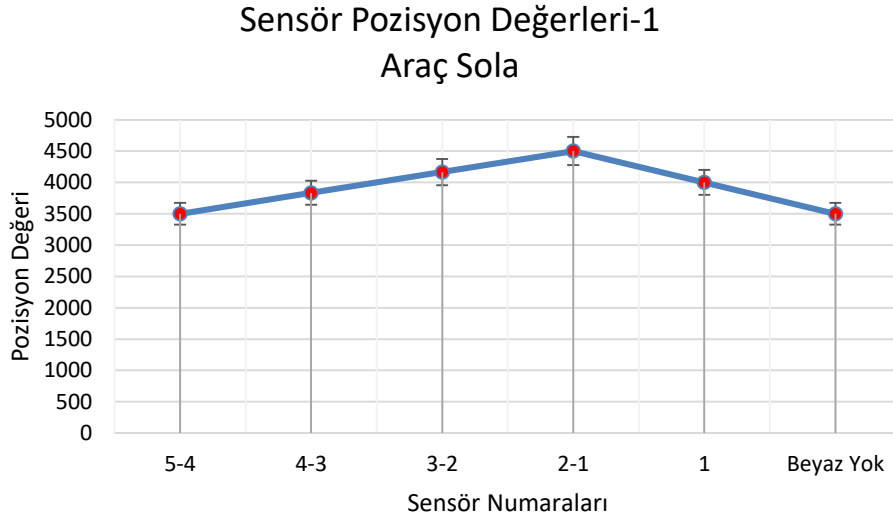
Çizelge 4.7 QTR-8RC kızılötesi sensör pozisyon değerleri – 1.

Sensör No	Value Çarpan	Beyaz Çizgiyi Gören Sensör Numaraları					Beyaz Yok
		5-4	4-3	3-2	2-1	1	
1	0x	1000	1000	1000	0	0	1000
2	1000x	1000	1000	0	0	1000	1000
3	2000x	1000	0	0	1000	1000	1000
4	3000x	0	0	1000	1000	1000	1000
5	4000x	0	1000	1000	1000	1000	1000
6	5000x	1000	1000	1000	1000	1000	1000
7	6000x	1000	1000	1000	1000	1000	1000
8	7000x	1000	1000	1000	1000	1000	1000
Pozisyon Değerleri		3500	3833	4166	4500	4000	3500

Yukarıdaki Çizelge 4.6’da görüleceği gibi QTR-8RC’nin kalibrasyonu yapıldıktan sonra beyaz çizgi için 0 değeri (value), siyah çizgi için 1000 değeri (value) kullanılarak ağırlıklı ortalama sonuçları hesaplanmaktadır.

Bu hesaplama sonucunda aracın çizgiyi tam ortadan takip etmesi için 4 ve 5 numaralı sensörlerin beyaz çizgiyi görmesi gerekir ve bu durumda pozisyon değeri 3500 olarak hesaplanır. Eğer 3 ve 4 numaralı sensörler beyaz çizgiyi görüyorsa pozisyon değeri 3833,

2ve 3 numaralı sensörler beyaz çizgiyi görüyorsa pozisyon değeri 4166 olacaktır. 1ve 2 numaralı sensörler beyaz çizgiyi görürse pozisyon değeri 4500, sadece 1 numaralı sensör beyaz çizgiyi görüyorsa pozisyon değeri 4000 olarak hesaplanmaktadır. Bu durumda araç sola doğru hareket etmektedir. Sola doğru hareket etmeye devam ederse en son 1 numaralı sensörden beyaz çizgiyi görerek ayrılır ve tüm sensörler siyah görmeye başlayacaktır. Aracın tekrar optimum çizgi takibine yani 4 ve 5 numaralı sensörlerin beyaz gördüğü duruma geri dönebilmesi için aracın sağa döndürülmesi gerekmektedir. Bunun içinde elde edilen hata değerlerinin sol motorun taban hızına eklenerek sol motorun hızlanması sağ motorun yavaşlaması sağlanır ve araç sağa doğru hareket ettirilir. Şekil 4.7’de araç sola giderken hesaplanan pozisyon değerleri grafiği yer almaktadır.



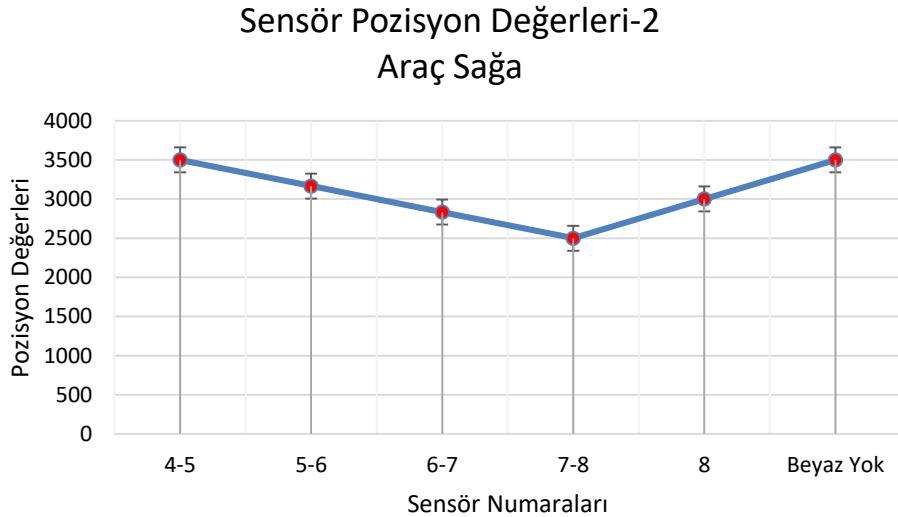
Şekil 4.7 QTR-8RC kızılötesi sensör pozisyon değerleri – 1 grafiği.

Çizelge 4.7’de görüleceği gibi aracın çizgiyi tam ortadan takip etmesi için 4 ve 5 numaralı sensörlerin beyaz çizgiyi görmesi gerekir ve bu durumda pozisyon değeri 3500 olarak hesaplanır. Eğer 5 ve 6 numaralı sensörler beyaz çizgiyi görüyorsa pozisyon değeri 3166, 6 ve 7 numaralı sensörler beyaz çizgiyi görüyorsa pozisyon değeri 2833 olacaktır. 7 ve 8 numaralı sensörler beyaz çizgiyi görürse pozisyon değeri 2500, sadece 8 numaralı sensör beyaz çizgiyi görüyorsa pozisyon değeri 3000 olarak hesaplanır. Bu durumda araç sağa doğru hareket etmektedir. Araç sağa doğru hareket etmeye devam ederse en son 8 numaralı sensörden beyaz çizgiyi görerek beyaz çizgiden ayrılıp tüm sensörler siyah görmeye başlayacaktır.

Aracın tekrar optimum çizgi takibine yani 4 ve 5 numaralı sensörlerin beyaz gördüğü duruma geri dönebilmesi için aracın sola döndürülmesi gerekmektedir bunun içinde elde edilen hata değerlerinin sağ motorun taban hızına eklenerek sağ motorun hızlanması, sol motorun yavaşlaması sağlanır ve araç sola doğru hareket ettirilir. Şekil 4.8’de araç sağa giderken hesaplanan pozisyon değerleri grafiği yer almaktadır.

Çizelge 4.8 QTR-8RC kızılötesi sensör pozisyon değerleri – 2.

Beyaz Çizgiyi gören Sensör Numaraları							
Sensör No	Value Çarpan	4-5	5-6	6-7	7-8	8	Beyaz Yok
1	0x	1000	1000	1000	1000	1000	1000
2	1000x	1000	1000	1000	1000	1000	1000
3	2000x	1000	1000	1000	1000	1000	1000
4	3000x	0	1000	1000	1000	1000	1000
5	4000x	0	0	1000	1000	1000	1000
6	5000x	1000	0	0	1000	1000	1000
7	6000x	1000	1000	0	0	1000	1000
8	7000x	1000	1000	1000	0	0	1000
Pozisyon Değerleri	⇒	3500	3166	2833	2500	3000	3500



Şekil 4.8 QTR-8RC kızılötesi sensör pozisyon değerleri – 2 grafiği.

4.4 Yön Tayini Arduino Program Kodları

Yön tayini işlemini gerçekleştirebilmek için öncelikle QTR-8RC sensörünün program kütüphanesi içinde yer alan readLine fonksiyonu ile sensör değerlerinin okutulup ağırlıklı ortalamaya göre pozisyon değerinin hesaplanması gerekir. Hesaplanan pozisyon değeri 3500 optimum pozisyon değerinden çıkarılarak hata değeri tespit edilir (Şekil 4.9).

```
void Sensor_Read() {  
    unsigned int position = qtrrc.readLine(sensorValues,1,GROUND_COLOR);  
    FAULT = position-3500;  
}
```

Şekil 4.9 QTR-8RC kızılötesi sensör değeri okuma fonksiyonu.

Hata değeri tespit edildikten sonra, elde edilen hata değeri pozitif bir değer ise hesaplanan pozisyon değerinin 3500' den yüksek bir değer olduğu anlamına gelir. Bu durumda araç sola doğru gitmektedir ve sağa hareket ettirilmesi gerekmektedir. Eğer hata değeri negatif bir değer ise bu durumda hesaplanan pozisyon değerinin 3500' ün altında bir değer olduğu anlamına gelir yani araç sağa gitmektedir ve sola hareket ettirilmesi gerekmektedir.

Bu hata değerine PID kontrol parametreleri uygulanarak motorlara eklenmesi ya da çıkarılması gereken hız farkları bulunur ve bu hız farkları motorlara parametre olarak gönderilmektedir (Şekil 4.10).

```
void Line_Tracking() {  
    COR_SPEED = Kp * FAULT + Kd*(FAULT - LAST_FAULT);  
    LAST_FAULT = FAULT;  
  
    RMPWM = RMHIZ + COR_SPEED ;  
    LMPWM = LMHIZ - COR_SPEED ;  
  
    RMPWM = constrain(RMPWM, -250, 200);  
    LMPWM = constrain(LMPWM, -250, 200);  
  
    M_CONTROL(RMPWM, LMPWM);  
}
```

Şekil 4.10 Hız farkları tespiti için kullanılan Line_Tracking fonksiyonu.

Line_Tracking fonksiyonu ile hesaplanan yeni hız değerleri Şekil 4.11’de gösterilen M_Control fonksiyonuna gönderilerek sağ ve sol motora uygulanarak aracın çizgiyi ortadan takip etmeye devam etmesi sağlanmaktadır.

```
void M_CONTROL(int RMPWM, int LMPWM){
  if(RMPWM<=0) {
    RMPWM=abs(RMPWM);
    digitalWrite(RMYON, LOW);
    analogWrite(RMPWMPIN, RMPWM);
  }
  else {
    digitalWrite(RMYON, HIGH);
    analogWrite(RMPWMPIN, RMPWM);
  }

  if(LMPWM<=0) {
    LMPWM=abs(LMPWM);
    digitalWrite(LMYON, HIGH);
    analogWrite(LMPWMPIN, LMPWM);
  }
  else {
    digitalWrite(LMYON, LOW);
    analogWrite(LMPWMPIN, LMPWM);
  }
}
```

Şekil 4.11 Hız farkları için kullanılan M_CONTROL fonksiyonu.

4.5 Araç Bir Tur Attığında Alınan Değerler

Şekil 3.20’ deki aracın izleyeceği rotada aracın bir tam tur attığında gerekli değerleri elde etmek için Serial.println fonksiyonu kullanılmaktadır.

Bu fonksiyon ile bir tam turda elde edilen değerlerin sayısı, malzemenin cinsine ve aracın enerjisini sağlayan lityum polimer pilin o an sağladığı enerji miktarına göre değişmektedir. Örneğin araç bir tam tur attığında sağ motor PWM değeri için yaklaşık 1440 değer elde edilmişken, sol motor PWM değeri için yaklaşık 1550 değer, QTR-8RC sensör position değeri için de yaklaşık 1980 değer elde edilmektedir.

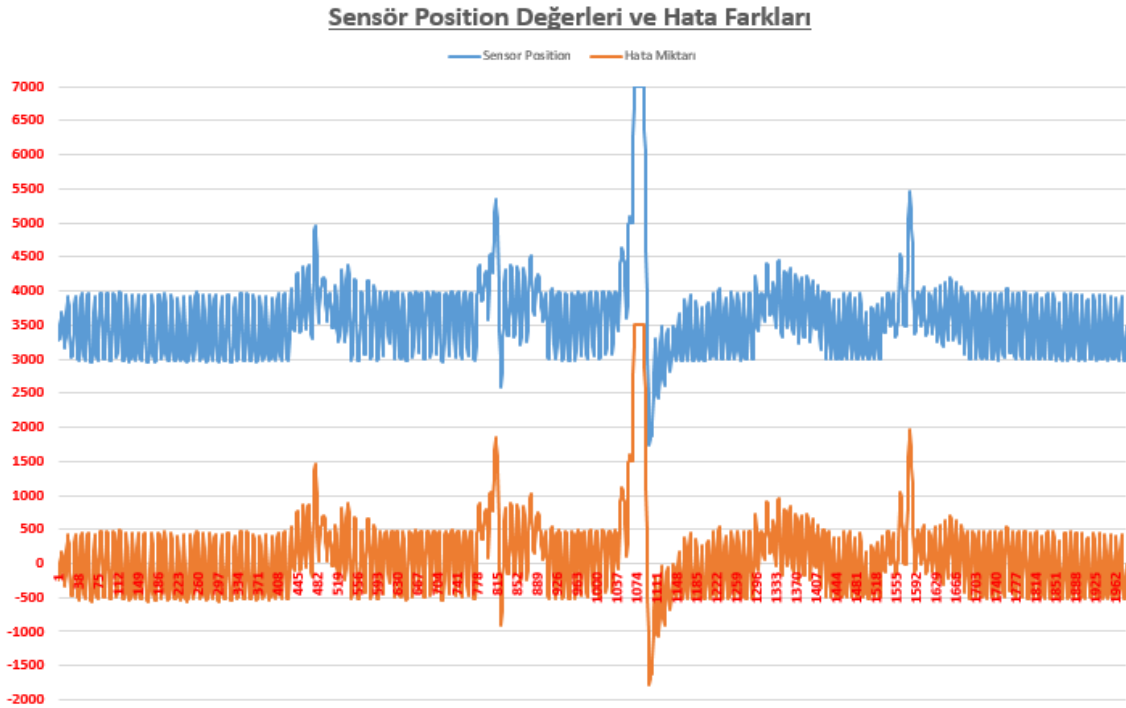
Bu farklı değer sayılarının nedeni her farklı malzeme değerlerini elde etmek için araç arka

arkaya çalıştırılmakta ve her çalıştırma da lityum polimer pilin sağladığı enerji miktarı bir öncekine göre azalmaktadır. Bir diğer neden ise malzemenin hassasiyetinden kaynaklanmaktadır.

4.5.1 QTR-8RC Sensör Değerleri

4.5.1.1 QTR-8RC Sensör Position Değerleri

Şekil 4.12’de aracın bir tam turunda elde edilen sensör position değerleri ve bu değerler için elde edilen hata miktarları gösterilmektedir. Bir tam turda sensörden yaklaşık 1980 değer elde edilmiştir. Dikey eksen sensör değeri ve hata miktarını yatay eksen ise değer sayısını ifade etmektedir. Sensör position değerlerini hesaplamak için şekil 4.7’de gösterilen Sensor_Read fonksiyonundaki qtrrc.readLine fonksiyonu kullanılmaktadır. Elde edilen bu position değeri en uygun position değeri olan 3500 değerinden çıkarılarak hata miktarı (FAULT) bulunmaktadır.



Şekil 4.12 QTR-8RC kızılötesi sensör position değerleri ve hata miktarları.

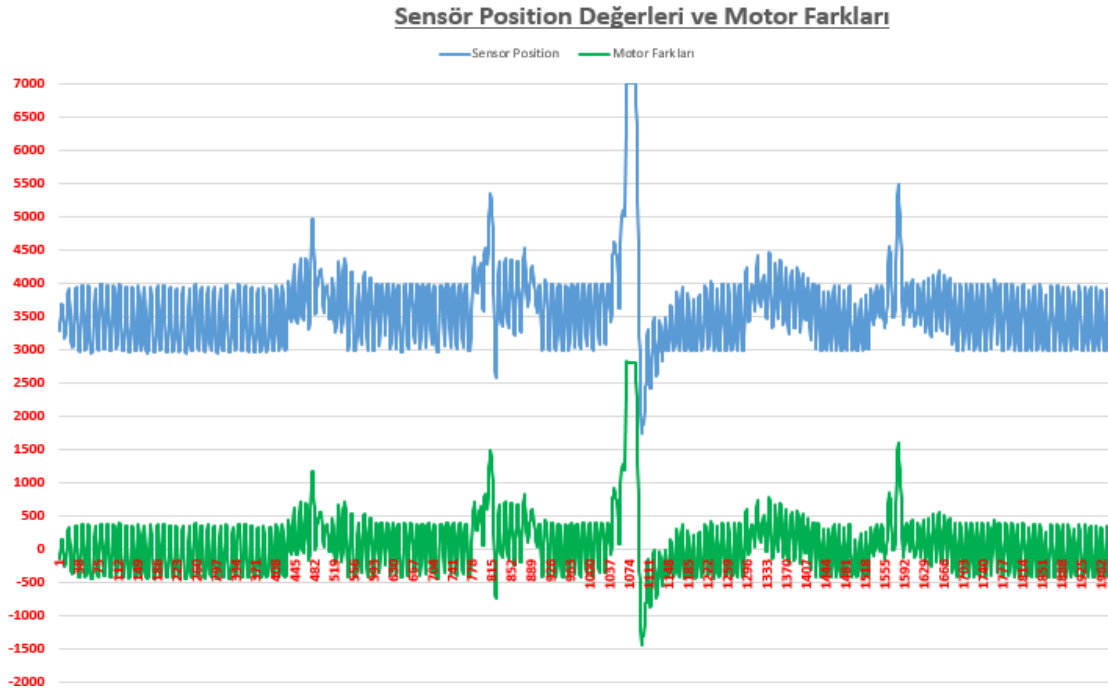
Grafik incelendiğinde position değerlerinin genel olarak 3500 değerine yakın değerlerde seyrettiği gözlenmektedir. Hata miktarlarında +500 ile -500 arasında değer aldığı

gözlenmektedir. Bu bölgelerde aracın düz çizgi üzerinde beyaz çizgiyi takip ederek hareket ettiği anlaşılmaktadır.

Dalgalanmaların olduğu bölgelerde ise ani ve yüksek miktarlarda position değeri değişimi olduğu gözlenmektedir. Benzer durum hata miktarları içinde geçerlidir. Bu bölgelerde aracın dönme hareketi gerçekleştirdiği anlaşılmaktadır. En yüksek dalgalanmanın olduğu bölge ise izlenen rotada küçük düz çizgi ve iki dönme noktasının olduğu bölge değerleri olduğu anlaşılmaktadır.

Şekil 4.13'te position değerleri ile hata miktarlarından elde edilen motor farkları grafiği yer almaktadır. Motor farklarını hesaplamak için şekil 4.8' deki Line_Tracking fonksiyonunda yer alan CORE_SPEED değişkeni kullanılmaktadır.

Belirlenen PID katsayıları ile position değerinden elde edilen hata miktarı (FAULT) kullanılarak motor farkları (CORE_SPEED) hesaplanır. PID katsayılarında D' nin karşılığı olan Türev farkları için bir alt satırda yer alan komut ile son hata (LAST_FAULT) değeri hata miktarına (FAULT) eşitlenmektedir.

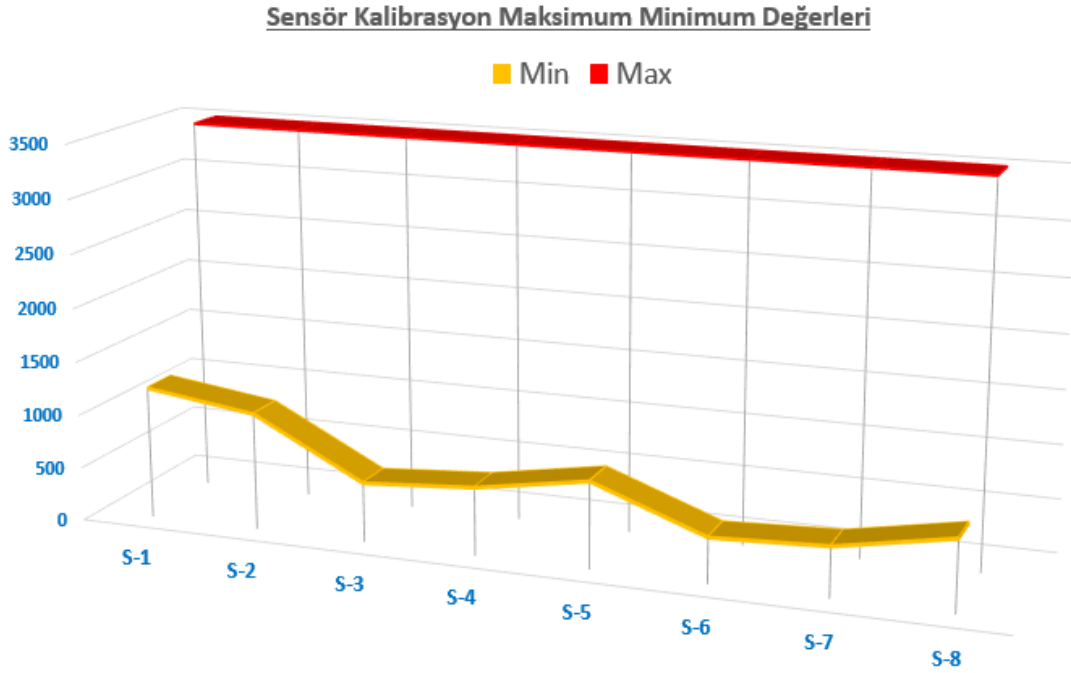


Şekil 4.13 QTR-8RC kızılötesi sensör position değerleri ve motor farkları.

4.5.1.2 Sensör Anlık Değerleri ve Position Değerleri – Kalibrasyon

QTR-8RC kızılötesi sensörünün doğru değerler üretebilmesi için birkaç saniye manuel ya da otomatik kalibrasyon işlemine tabi tutulması gerekmektedir. Bu işlem için araç hareket etmez durumda iken siyah zemin, beyaz çizgi ve siyah zeminde araç şasesine bağlı olan sensörün sağa ve sola yalpalama hareketinin kalibrasyon süresince sürekli olarak yapılması gerekmektedir. Kalibrasyon işleminden sonra sensörlerin maksimum ve minimum kalibrasyon değerleri elde edilmektedir.

Şekil 4.14'te kalibrasyon işleminden sonra sensörlere ait maksimum ve minimum kalibrasyon değerleri verilmiştir.



Şekil 4.14 QTR-8RC kızılötesi sensör kalibrasyon maksimum ve minimum değerleri.

Bu tezde kalibrasyon işlemi için otomatik kalibrasyondan daha düzgün sonuçlar elde edilmesini sağlayan manuel kalibrasyon yöntemi kullanılmıştır. Şekil 4.15'te kalibrasyon için kullanılan Manuel_Calibration fonksiyonu gösterilmiştir.

```

// -----KALİBRASYON-----
void Manuel_Calibration() {
  digitalWrite(13, HIGH);

  for (i = 0; i < 400; i++)
  {
    qtrrc.calibrate();
    delay(20);
  }

  digitalWrite(13, LOW);

  delay(1000);
}
// -----KALİBRASYON-----

```

Şekil 4.15 Kalibrasyon için kullanılan Manuel_Calibration fonksiyonu.

Sensörlerin doğru değer üretip üretmediği görmek amacıyla manuel kalibrasyon işlemi gerçekleştirildikten sonra serial.println fonksiyonu ile birkaç defa sensörün sağa sola yalpalama hareketi yapılmış ve anlık olarak sensör değerleri ile bu sensör değerlerinden elde edilen position değerleri tespit edilmiştir. Bu işlem için Şekil 4.16’da gösterilen Loop fonksiyonu kullanılmıştır.

```

void loop()
{
  unsigned int position = qtrrc.readLine(sensorValues);

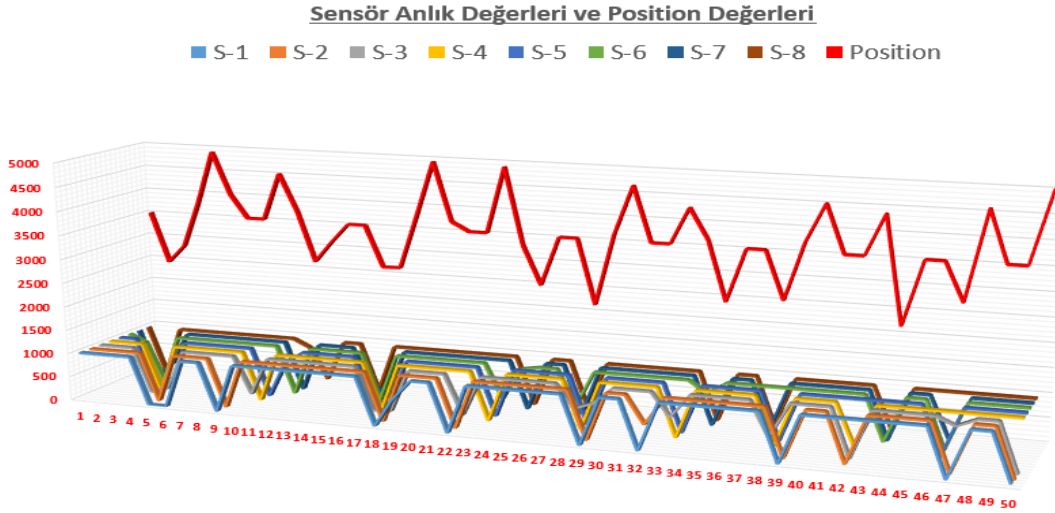
  unsigned char i;
  for (i = 0; i < 8; i++)
  {
    Serial.print(sensorValues[i]);
    Serial.print(' ');
  }
  Serial.print(" ");
  Serial.println(position);

  delay(250);
}

```

Şekil 4.16 Anlık sensör ve position değerleri için kullanılan Loop fonksiyonu.

Siyah zemin, beyaz çizgi ve siyah zemin üzerinde ileri ya da geri hareketi olmaksızın sensörün sağa ve sola doğru yalpalama hareketleri sonucunda elde edilen değer grafiği Şekil 4.17’de gösterilmektedir. Sensör anlık değerlerinden 1000 değeri siyah zemini, 0 değeri ise beyaz çizgiyi göstermektedir.

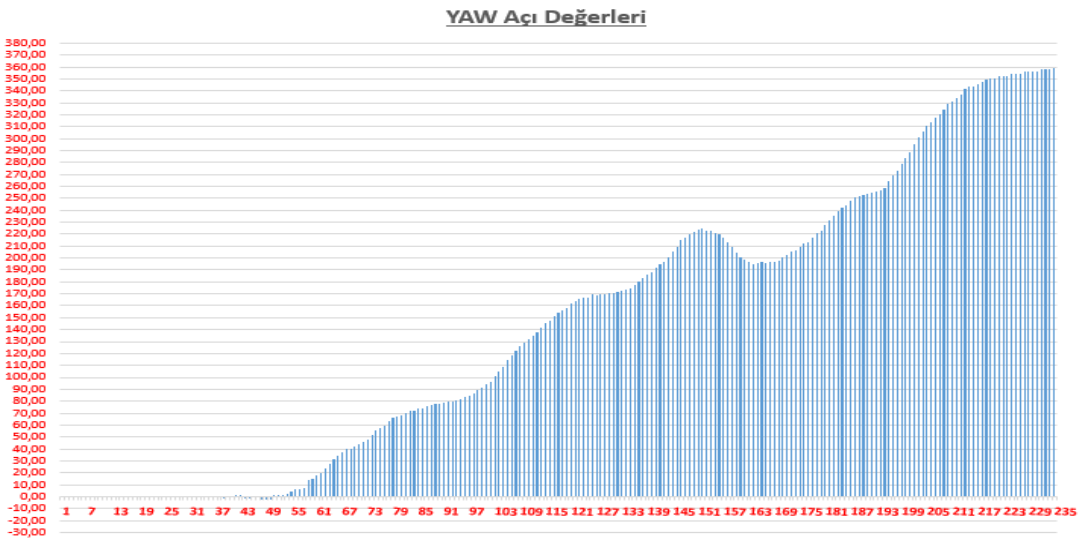


Şekil 4.17 QTR-8RC kızılötesi anlık sensör ve position değerleri.

Grafikteki dalgalanma hareketleri sensörlerin düzgün değerler okuduğunu göstermektedir. Anlık sensör değerlerinden elde edilen position değerleri ise grafikte üst kısımda gösterilmektedir.

4.5.2 MPU 6050 Sensöründen Alınan Değerler

Aracın bir tam tur hareketi sırasında MPU 6050 sensöründen yaw açısı değerleri alınarak rotada araç hareket halindeyken meydana gelen açı değişimleri gözlemlenmiştir. Şekil 4.18’de yaw açısı değerleri gösterilmiştir.



Şekil 4.18 MPU 6050 ivme sensörü YAW açısı değerleri.

MPU 6050 sensörü ile ivme değerleri ve gyro değerleri hesaplanabilmektedir. 3 boyutlu eksen üzerindeki açılar; X eksenindeki hareket değişimleri sonucunda oluşan ROLL açısını, Y eksenindeki hareket değişimleri sonucunda oluşan PITCH açısını ve Z eksenindeki hareket değişimleri sonucunda oluşan YAW açısını ifade etmektedir. Bu tezde araç düz bir zemin üzerinde hareket edeceği için aracın sadece Z eksenine göre sağa ya da sola dönme açısı yani YAW açısı önemli olmaktadır. Araç X eksenine göre ya da Y eksenine göre herhangi bir hareket gerçekleştirmeyeceği için ROLL ve PITCH açılarının hesaplanması gerekmemektedir. YAW açı değerinin hesaplanabilmesi için Şekil 4.19’da gösterilen Arduino program kodu kullanılmıştır.

```
/* MPU6050 Triple Axis Gyroscope & Accelerometer.
   Pitch & Roll & Yaw Gyroscope Example.
   GIT: https://github.com/jarzebski/Arduino-MPU6050
   (c) 2014 by Korneliusz Jarzebski */
#include <Wire.h>
#include <MPU6050.h>

MPU6050 mpu;
unsigned long timer = 0;
float timeStep = 0.1;
float yaw = 0;

void setup() {
  Serial.begin(9600);
  mpu.begin(MPU6050_SCALE_2000DPS, MPU6050_RANGE_2G);
  mpu.calibrateGyro();
}

void loop() {
  timer = millis();
  Vector norm = mpu.readNormalizeGyro();
  yaw = yaw + norm.ZAxis * timeStep;
  Serial.print(" Yaw = ");
  Serial.println(yaw);
  delay((timeStep*1000) - (millis() - timer));
}
```

Şekil 4.19 YAW açı değeri hesaplaması arduino program kodu(İnt. Kyn. 56).

Şekil 4.18’deki grafikte dikey eksen YAW açı değerlerini, yatay eksen ise elde edilen değer sayılarını ifade etmektedir. Araç beyaz çizgi üzerine düzgün olarak yerleştirildikten sonra MPU sensörünün kalibrasyonu için bir süre hareketsiz bekletildikten sonra sensör

aracın hareket yönünü 0 derece olarak kabul etmektedir.

Araç hareketine başladığı andan tam tur yapılıncaya kadar 235 adet YAW açısı değeri elde edilmiştir. Aracın hareketi sırasında araç sürekli düz hareket ya da sola dönüş hareketi gerçekleştirdiği için tamamen pozitif açısı değerlerinin döndürüldüğü görülmektedir. Eğer araç sağa doğru bir dönme hareketi yapmış olsaydı o zaman sensörün negatif açısı değerleri de döndürmesi gerekmektedir.

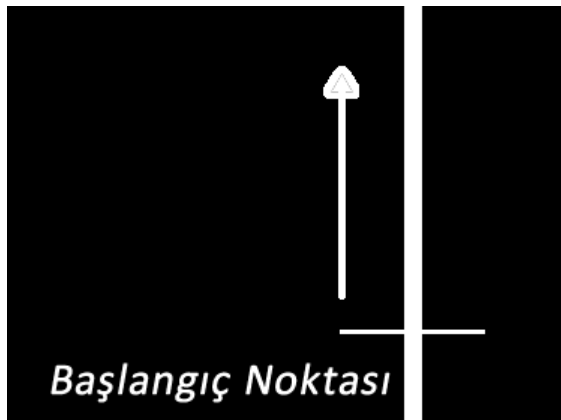
Grafik incelendiğinde açısı değerleri belirli bölgelerde birbirine yakın değeri döndürdüğü noktalarda aracın düz hareket ettiği gözlenmektedir. Açısı değerinde ani yükselmelerin olduğu bölgelerde ise araç sola doğru dönme hareketini yaptığı gözlenmektedir.

Araç düz çizgi üzerinde ileriye doğru ilk hareketini yapmaya başladığı anda döndürülen açısı değeri 0 derecedir. Tur tamamlanıp yine aynı noktaya geldiğinde ise açısı değerinin yaklaşık 359 derece olduğu gözlenmektedir.

Elde edilen açısı değerleri kullanılarak birbirine yakın açısı değerlerinin düz çizgi, ani açısı değeri yükselmesi durumunda sola dönme hareketinin yapıldığı anlaşılmaktadır.

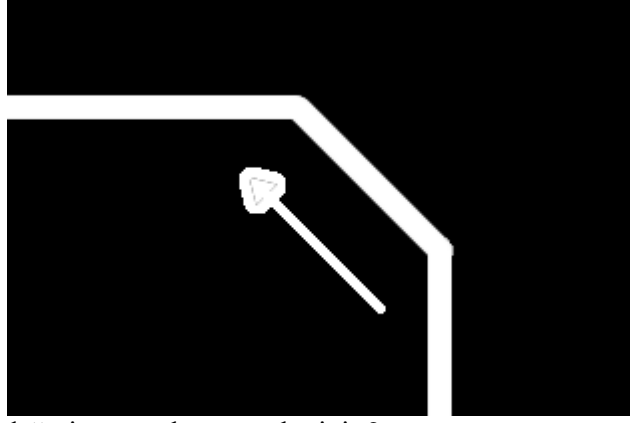
YAW açısı değerleri araç konum bilgisi elde etmek için kullanılmıştır. Buna göre;

- Araç başlangıç noktasında ilerleme yönüne doğru YAW açısı değeri 0 derece olarak kalibre edilmiştir(Şekil 4.20).



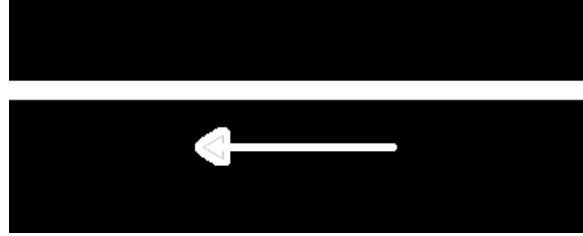
Şekil 4.20 YAW açısı değerine göre konum tahmini -1.

- İlk dönüş noktasına geldiğinde YAW açı değeri başlangıç noktasına göre 25-35 derece arası değerler almaktadır(Şekil 4.21).



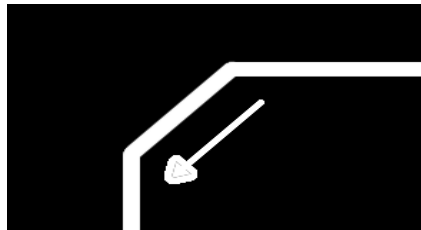
Şekil 4.21 YAW açı değerine göre konum tahmini -2.

- Dönüş yaptıktan sonra gelen yatay düz çizgide yine başlangıç noktasındaki konumuna göre YAW açı değeri olarak 80-90 arası değerler gelmektedir (Şekil 4.22)



Şekil 4.22 YAW açı değerine göre konum tahmini -3.

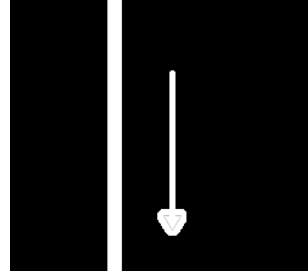
- Araç rotada yer alan sonraki dönüş noktasına geldiğinde sensör YAW açı değeri olarak 125-135 derece arasında değerler döndürmektedir (Şekil 4.23).



Şekil 4.23 YAW açı değerine göre konum tahmini -4.

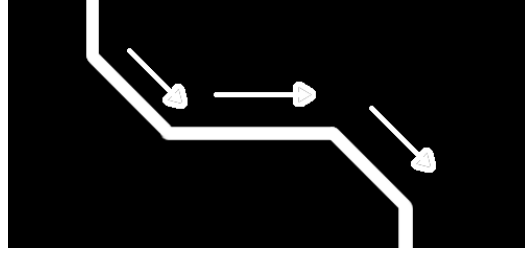
- Araç dönme hareketinden sonra dikey düz çizgi üzerine geldiğinde başlangıçtaki

konumunun tam tersi yönünde hareket olduğu görülmektedir. Bu durumda sensörden 165-180 derece arasında değerler gelmektedir (Şekil 4.24).



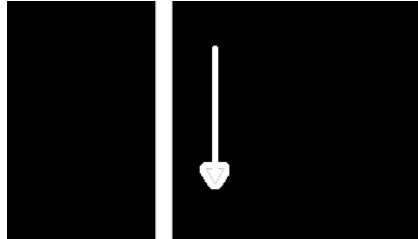
Şekil 4.24 YAW açısı değerine göre konum tahmini -5.

- Araç tekrar dönme noktası kısa düz hareket ve yine dönme noktası bölümüne geldiğinde önce 200-210 derece arası açı değerleri, düz çizgide 210-230 arası açı değerleri tekrar dönme noktasına geldiğinde 220-210 derece arası açı değerleri aldığı gözlenmektedir (Şekil 4.25).



Şekil 4.25 YAW açısı değerine göre konum tahmini -6.

- Araç hareketin devamında gelen dikey çizgideyken MPU 6050 sensörü YAW açısı değeri olarak 200-190 derece arası değerler döndürmektedir (Şekil 4.26).



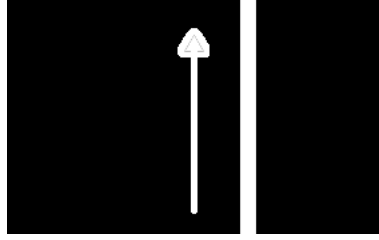
Şekil 4.26 YAW açısı değerine göre konum tahmini -7.

- Aracın hareketi sırasındaki diğer dönme, düz hareket, dönme bölgesinde sensör tarafından döndürülen açı değerleri sırasıyla 190-210 derece arası, 210-230 derece arası, 230-260 derece arası değerler gelmektedir (Şekil 4.27).



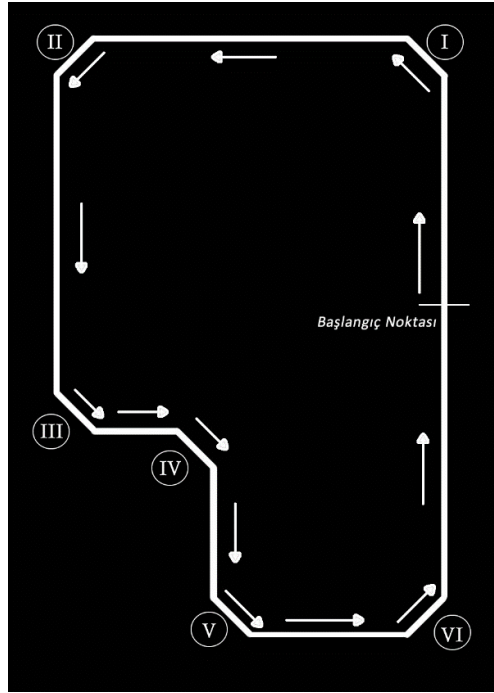
Şekil 4.27 YAW açısına göre konum tahmini -8.

- Araç tekrar başlangıç kısmındaki düz çizgi üzerine geldiğinde sensör YAW açısı olarak 330-360 derece arası değerler döndürdüğü gözlenmektedir (Şekil 4.28).



Şekil 4.28 YAW açısına göre konum tahmini -9.

Aracın izlediği rota, hareket yönü ve dönme noktaları aşağıdaki şekil 4.29'da gösterilmektedir.

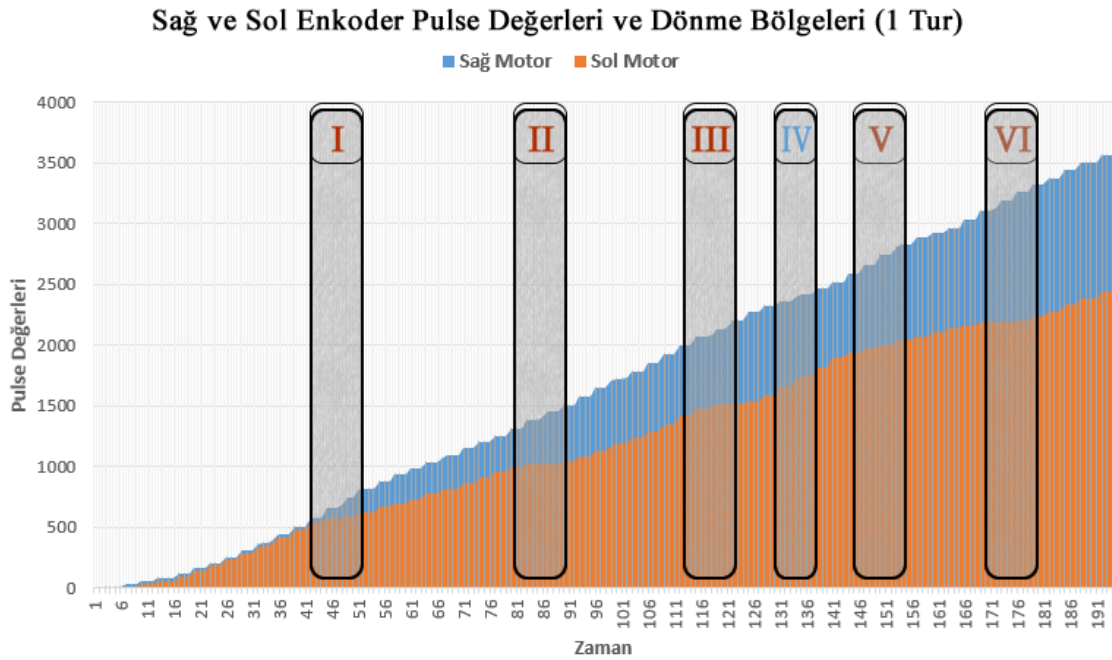


Şekil 4.29 Aracın hareket yönü ve dönme noktaları.

4.5.3 Sağ ve Sol Motor Enkoder Değerleri ve Dönme Açılı

Prototip araçta kullanılan Enkoderli DC motorlarda, çıkış milinin tam dönüş başına 211.2 sayım gerçekleştiren entegre 48 CPR dört evreli kodlayıcı bulunmaktadır. Uygulama yazılımında iki evreli kodlama kullanıldığı için çıkış milinin bir tam turunda 105.6 sayım elde edilmektedir.

Motorların hareketi sırasında enkoderlerden elde edilen sayım sayısı yani pulse değerleri kullanılarak aracın çizgi üzerinde kat ettiği mesafe hesaplanabilmektedir. Şekil 4.30'da aracın çizgi üzerindeki bir tam turunda sağ ve sol motordan elde edilen enkoder pulse değerleri ve dönme bölgeleri gösterilmiştir. Dönme bölgeleri Şekil 4.29'da gösterilen bölgelerdir.

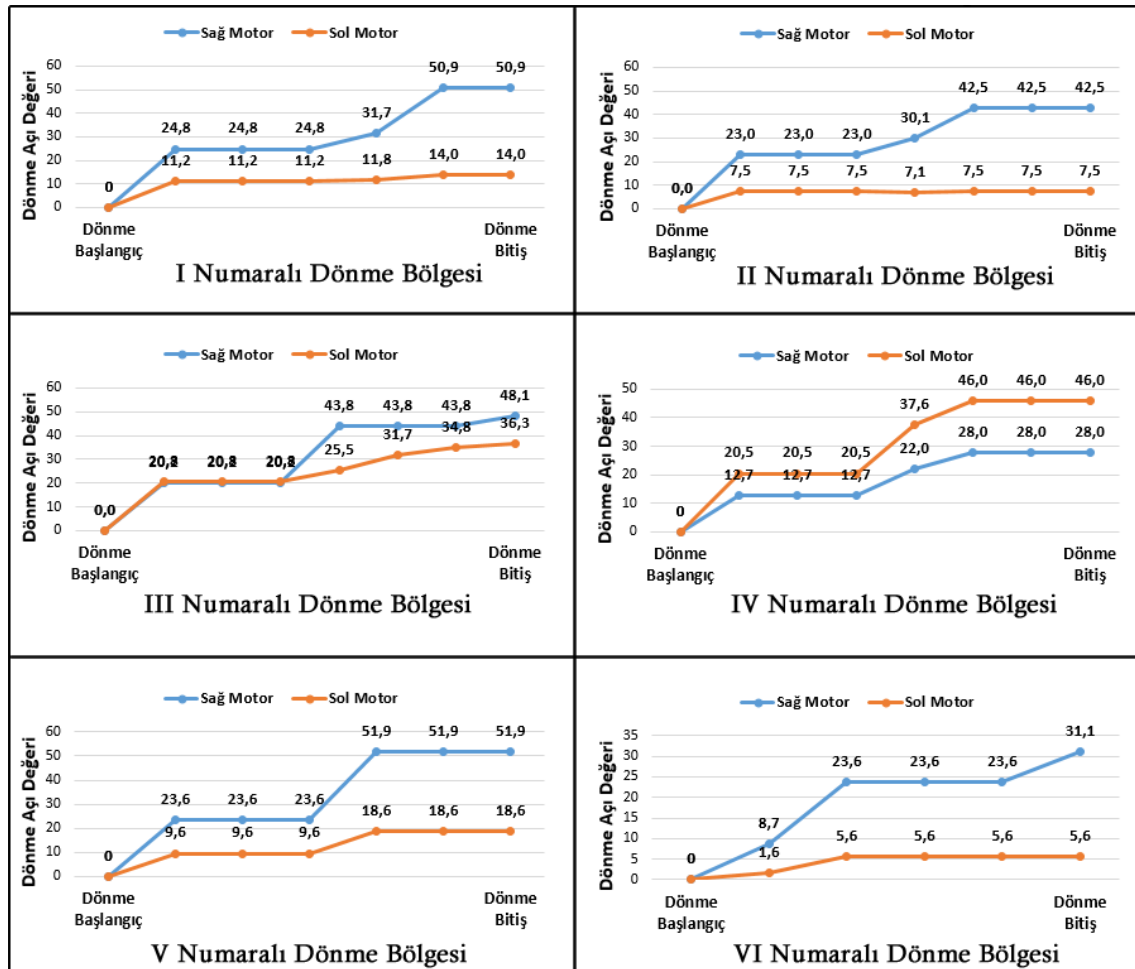


Şekil 4.30 Sağ ve Sol Enkoder Pulse Değerleri ve Dönme Bölgeleri.

Prototip aracın hareket edeceği rota üzerinde toplam 6 adet dönme bölgesi bulunmaktadır. Rotadan da görüleceği gibi III, IV ve V numaralı bölgeler birbirine yakın olan bölgelerdir. Araç düz çizgi üzerinde hareket halindeyken sağ ve sol motor enkoder değerlerinin artış miktarları aynı seviyededir.

I, II, III, V ve VI numaralı bölgelerde araç sola dönme hareketini yapmaktadır. Bu durumda dönme merkezine göre sol tekerlek(iç tekerlek) daha yavaş hareket ve daha düşük bir dönme açısına sahip olurken sağ tekerlek(dış tekerlek) daha hızlı hareket ve daha yüksek dönme açısına sahip olmaktadır.

Dönme bölgelerinden sadece IV numaralı bölgede araç sağa dönme hareketi yapmaktadır. Bu durumda yine dönme merkezine göre sağ tekerlek(iç tekerlek) daha yavaş hareket ve daha düşük bir dönme açısına sahip iken sol tekerlek(dış tekerlek) daha hızlı hareket ve daha yüksek dönme açısına sahip olduğu görülmektedir.



Şekil 4.31 Dönme bölgelerinde sağ ve sol motor dönme açıları.

Sağ ve Sol motor enkoderlerinden gelen pulse değerlerinin dönme noktalarının başlangıcında ve bitişinde elde edilen değerlere göre artış miktarlarının dönme hareketi

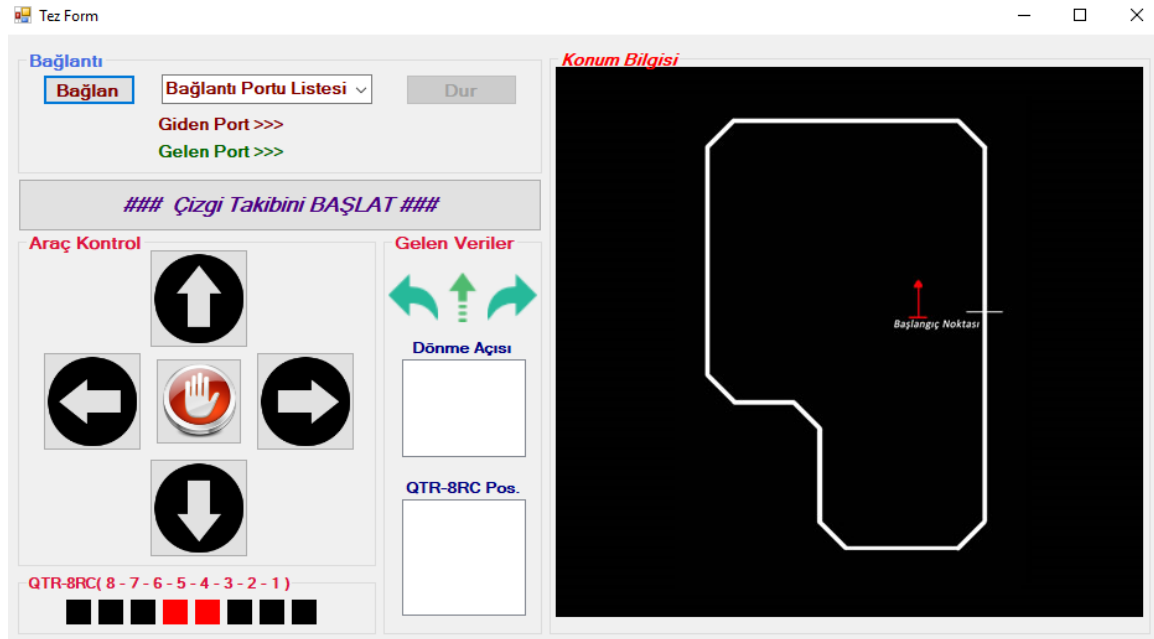
nedeniyle farklı olduğu görülmüştür. Bu durumda dönme bölgelerinde elde edilen sağ ve sol motor dönme açıları Şekil 4.31’de gösterilmiştir.

Sadece IV numaralı bölgede araç sağa dönme hareketi yaptığı için sol motor dönme açısı sağ motor dönme açısından yüksek, diğer tüm bölgelerde araç sola dönme hareketi yaptığı için sağ motor dönme açıları sol motor dönme açılarından daha yüksektir.

4.6 Uygulama Yazılımı Arayüz Ekranı

Bilgisayar ile prototip aracın kontrolünün sağlandığı uygulama yazılımının arayüzü 6 ana bölümden oluşmaktadır (Şekil 4.32).

- Bağlantı GroupBox kontrol elemanı,
- Çizgi Takibi Başlat buton elemanı
- Araç Kontrol GroupBox kontrol elemanı
- QTR-8RC GroupBox kontrol elemanı
- Gelen Veriler GroupBox kontrol elemanı
- Konum Bilgisi GroupBox kontrol elemanı



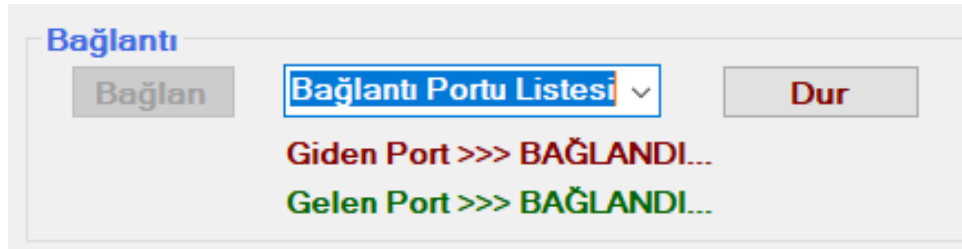
Şekil 4.32 Uygulama Programı arayüz ekranı.

4.6.1 Bağlantı GroupBox Kontrol Elemanı

Arayüzün bu bölümünde 2 adet buton kontrolü 2 adet label kontrolü ve 1 adet combobox kontrolü kullanılmaktadır. Combobox kontrolünde bağlantı portlarının listesi yer almaktadır. Aynı anda birden fazla porta bağlanması gerektiği için bu kontrolde bir seçim yapılmayacaktır. Hangi portlara bağlanması gerektiği kod bölümünde belirtilmektedir.

Bağlan butonuna tıklandığında programın kod kısmında belirtilen portlara bağlanıp label kontrollerinde “Giden Port >>> BAĞLANDI...” ve “Gelen Port >>> BAĞLANDI...” ifadeleri gösterilmektedir. Bağlantı işlemi gerçekleştiğinde DUR butonu da aktif hale getirilmektedir.

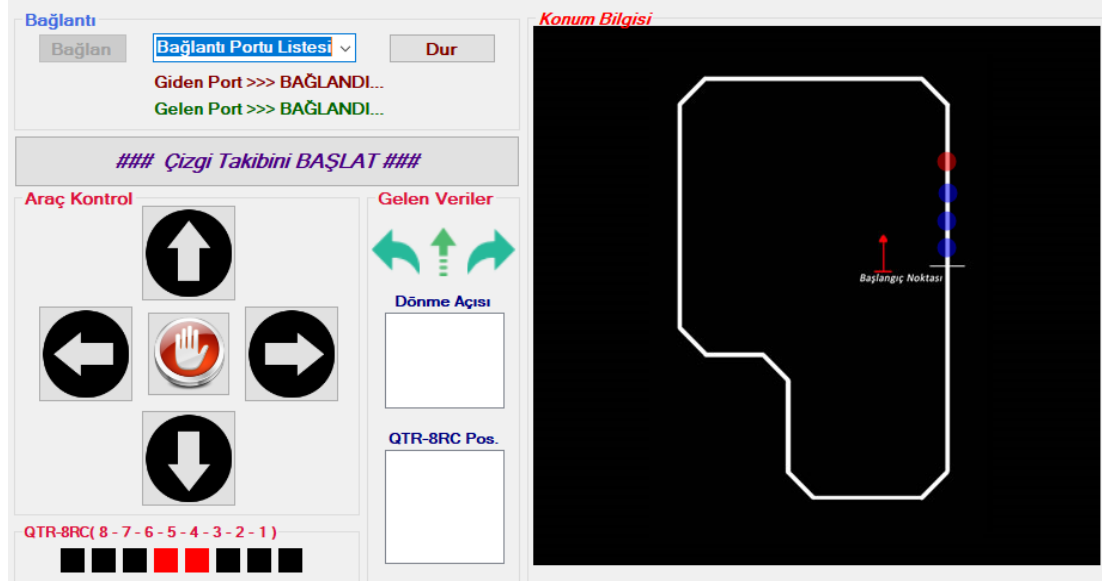
Dur butonuna tıklandığında tüm bağlantılar kesilmekte ve label kontrollerinde “Giden Port >>> BAĞLI DEĞİL !!!” ve “Gelen Port >>> BAĞLI DEĞİL !!!” ifadeleri belirlemektedir (Şekil 4.33).



Şekil 4.33 Bağlan butonuna tıklanması.

4.6.2 Çizgi Takibi Başlat Buton Elemanı

Çizgi takibi başlat butonuna tıklandığında araç başlangıç noktasında beyaz çizgiyi takip etmeye başlamakta ve aynı zamanda rotadaki konumu arayüzde konum bilgisi kısmında kırmızı noktalarla gösterilmektedir (Şekil 4.34).



Şekil 4.34 Çizgi Takibi Başlat butonuna tıklanması.

4.6.3 Araç Kontrol GroupBox Kontrol Elemanı

Bu groupbox kontrol elemanında 5 adet buton yer almaktadır. Bu butonlar aracı sağa, sola, ileri ve geri yönlendirmek için kullanılmaktadır. Ortada yer alan buton ise aracı durdurmak için kullanılmaktadır. Aracın kontrol işlemi yönlendirme butonlarına basıldığı sürece gerçekleşmektedir. Butona basma eylemi bittiğinde yönlendirme işlemi de sona ermektedir. Durdurma butonu ise tek tıklama yapıldığında aracın durmasını sağlamaktadır.

4.6.4 QTR-8RC GroupBox Kontrol Elemanı

QTR-8RC groupbox kontrolünde kızılötesi sensör modülünün 8 adet sensörüne karşılık 8 adet arka plan renkleri değiştirilmiş label kontrolleri kullanılmaktadır. Araç çizgi takibi yaparken hangi sensörler beyaz çizgiyi görmekte ise o numaralı label kontrollerinin arka plan renkleri kırmızı yapılmaktadır. Diğer label kontrollerinin arka plan renkleri siyah renkte kalmaktadır.

Bu renk ayarlamasını gerçekleştirmek için çizgi izleme sensörünün o anki position değeri kullanılmaktadır. Position değeri anlık olarak bağlantılı porttan uygulama programına gönderilerek ilgili label kontrollerinin arka plan renkleri kırmızı yapılmaktadır. Position

değeri olarak sensörden 4166 değeri gönderildiğinde label kontrollerinin görünümüleri Şekil 4.35' te gösterilmiştir.

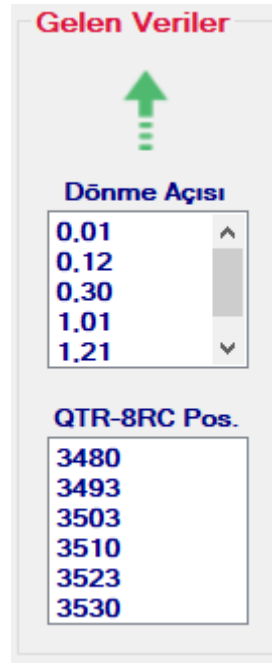


Şekil 4.35 Kızılötesi sensör position değerine göre label kontrolleri görünümü.

4.6.5 Gelen Veriler GroupBox Kontrol Elemanı

Bu kontrol elamanında MPU 6050 sensöründen gelecek YAW açısı değerine göre sağa sola ve ileri ok işaretleri, yine aynı sensörden gelen YAW açısı değerinin gösterilmesi amacıyla Dönme Açısı listbox kontrol elemanı ve çizgi izleme sensöründen gelen position değerinin gösterileceği listbox kontrol elemanı bulunmaktadır.

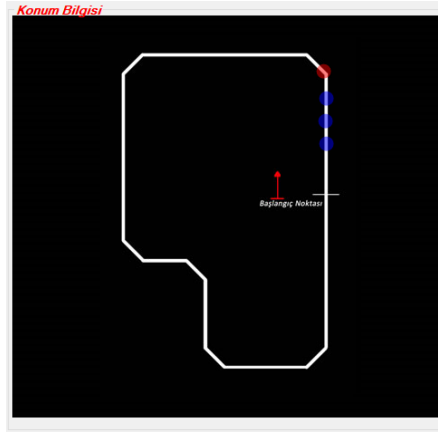
Değerler anlık olarak gelmeye devam ettiği sürece gelen sensör değerleri bu listbox kontrollerinde listelenmektedir (Şekil4.36).



Şekil 4.36 Gelen veriler groupbox kontrol elemanı.

4.6.6 Konum Bilgisi GroupBox Kontrol Elemanı

Konum bilgisi alanında Çizgi Takibi Başlat butonuna basıldığında araç çizgi takibine başlayarak anlık olarak aracın rota üzerindeki konum bilgisi görüntülenmektedir. Bu işlemi gerçekleştirmek için konum bilgisi groupbox kontrolünün location değeri MPU 6050 sensöründen gelen YAW açı değeri ve QTR-8RC sensöründen gelen position değerleri ile değiştirilmektedir. Bu sayede aracın rota üzerindeki anlık konumu kırmızı nokta şeklinde gösterilmektedir. Mavi noktalar bir önceki konum noktalarını ifade etmektedir (Şekil 4.37).



Şekil 4.37 Konum bilgisi groupbox kontrol elemanı.

5. TARTIŞMA ve SONUÇ

Otomatik kılavuzlu araçlar gelişen teknoloji ile beraber her tür sektörde kendine daha fazla yer edinmeye devam etmektedir. Dördüncü sanayi devrimi olarak da adlandırılan Endüstri 4.0 nesnelerin interneti, hizmetlerin interneti ve siber fiziksel sistemler kavramlarından oluşmaktadır. Siber fiziksel sistemler kavramı ile artık fabrika ortamlarında, imalat sanayisinde, lojistik sektöründe ve daha birçok sektörde her şeyin neredeyse tamamen insan gücünden bağımsız olarak kendini yönetebilen makinalar tarafından yapılması hedeflenmektedir. Bu ortamda da otomatik kılavuzlu araçlar önemi daha da arttırmaktadır.

Otonom sistemlerin günlük hayatın her alanında karşımıza çıkmaya başladığı bu dönemde insanların hayatını kolaylaştıran teknolojiler hem bireyler hem de kurumlar tarafından çok çabuk kabul görmektedir. Yeni teknolojilerin kabul edilebilir ve uygulanabilir olmasındaki en önemli ölçüt maliyet faktörüdür. Bireysel ya da kurumsal olarak yeni bir teknolojinin kullanımına karar verme sürecindeki en dikkat edilen konu fayda maliyet ilişkisinden geçmektedir.

Endüstri 4.0 kavramının temelinde akıllı fabrikalar, otonom sistemler ve kendi karar verebilen sistemler söz konusudur. Örnek olarak bir fabrika düşülürse bu fabrikada yapılan işe göre kullanılan insan gücü oldukça az olmaktadır. Çünkü birçok işlem robotlar tarafından ya da otonom sistemler tarafında yerine getirilmektedir. Bu akıllı sistemler sayesinde insan faktörü azaldığı için ortaya çok daha kaliteli ürünler çıkmaktadır. Bir işletmenin en önemli gideri insan gücü maliyetidir, Endüstri 4.0 altyapısına sahip bir fabrikada insan gücüne ihtiyaç az olacağı için ortaya çıkan ürünün maliyeti de minimum olmaktadır. Birim maliyet azaldığı için ürünler çok ucuza mal edilebilmektedir. İşletmeler için temel olarak varılmak istenen nokta bu olmaktadır. Çok daha kaliteli ürünleri çok daha ucuza mal etmek hedeflenmektedir.

Yapılan çalışmada temel hedef bir aracın belirli kılavuzla işaretlenmiş yolu takip ederken izlediği rotanın ortaya çıkarılmasıdır. Burada en çok üzerinde durulması gereken konu ise araç düz yolda ya da çizgide hareket ederken herhangi bir problem söz konusu olmamakla birlikte sadece araçtaki çok ufak titreşimlerden kaynaklı çok hassas olan ivme sensörünün

gönderdiği açı değerlerinde değişiklikler meydana gelmektedir. Özellikle bir dönme noktasına gelindiğinde sensörünün döndürdüğü açı değerlerinde anlık olarak farklılıklar ortaya çıkmaktadır. Ancak enkoderlerden gelen pulse değerleri düzgün şekilde gelmeye devam ettiği için eş zamanlı olarak konum belirlenebilmektedir.

İvme sensöründen gelen açı sorununun çözülebilmesi için ivme sensörünün mümkün olduğunca dış ortamdan kaynaklı hassasiyet kaybının engellenmesi gerekmektedir. Bunun için endüstride kullanılan daha hassas veriler ölçebilen sensörlerin tercih edilmesi gerekmektedir. Seri iletişimle gelen veriler arasındaki farklılıkları kontrol ederek belirli bir miktardan büyük farka sahip değerleri yok saymanın ya da dikkate almamanın sistemi biraz daha optimize edebileceği düşünülmektedir. Bununla birlikte sensörden gelen değerlerin optimizasyonu için başta Kalman Filtresi olmak üzere daha farklı filtreler ivme sensöründen gelen değerlere uygulanabilir.

Gerçekleştirilen bu prototipin devamında otomatik kılavuzlu bir aracın eş zamanlı konumunu belirlemek yerine serbest dolaşan bir aracın eş zamanlı konumunun belirlenmesi çalışmasına ait bir prototipin gerçekleştirilebilir olduğu öngörülmektedir. Ayrıca gerçekleştirilen prototip araç ve kontrol yazılımı ile otomatik kılavuzlu araçların çarpışmasını engellemek, trafik sıkışıklığını düzenlemek, aracı en yakın şarj istasyonuna yönlendirmek gibi ileride yapılabilecek çalışmalar için de bir altyapı oluşturulmuştur.

Otomatik kılavuzlu bir aracın yönlendirilmesi ve eş zamanlı olarak konunun belirlenmesi ile özellikle depolama ve yük taşıma işlemlerinde insan gücü maliyetini azaltıp fabrika ya da imalat sektöründe daha güvenli ortamlar oluşturacaktır. Çalışma sadece iş hayatına değil otomatik kılavuzlu araçların eğitimi konusunda da eğitim sektörüne de katkı sağlayacaktır. Benzer çalışmalar yapmak isteyen öğrenci ve akademisyenler için yeni fikirler vererek kaynak oluşturacaktır.

6. KAYNAKLAR

- Awasthi A, Chauhan S S, Goyal S K, 2009, A simulation model for parts selection and routing in manufacturing systems, Proceedings of the 13th IFAC Symposium on Information Control Problems in Manufacturing, 3-5 June, Moscow Russia, 618-623.
- Azenha A, Carvalho A, 2003, Instrumentation And Localisation In Quasi-Structured Environments For Agv Positioning, IFAC Intelligent Components and Instruments for Control Applications, 9-11 July, Aveiro Portugal, 65-72.
- Cheong H, Lee H, 2018, Concept Design of AGV (Automated Guided Vehicle) Based on Image Detection and Positioning, Procedia Computer Science, 139, 104–107.
- Çınar S, 2005, Gps İle Araç Takip Ve Yönlendirme Sistemi, Hacettepe Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, 114s, Ankara.
- Doughangi H, 2017, Kapalı Alanda Konum Belirleme Sistemi, İstanbul Ticaret Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, 52s, İstanbul.
- Galasso F, Rizzini D L, Oleari F, Caselli S, 2019, Ecient calibration of four wheel, Robotics and Computer Integrated Manufacturing, 57, 116-128.
- Ge S, Zhang Q, Abraham A T, Rebsamen B, 2011, Simultaneous Path Planning and Topological Mapping (SP2ATM) for environment exploration and goal oriented navigation, Robotics and Autonomous Systems, 59, 228–242.
- Gualda D, Ureña J, García J C, García E, Alcalá J, 2019, Simultaneous calibration and navigation (SCAN) of multiple ultrasonic local positioning systems, Information Fusion, 45, 53–65.
- Guldner J, Utkin V I, Bauer R, 1995, A three-layered hierarchical path control system for mobile robots: Algorithms and experiments, Robotics and Autonomous Systems, 14, 133-147.
- Heger J, Voss T, 2018, Optimal scheduling of AGVs in a reentrant blocking job-shop, Procedia CIRP, 67, 41–45.
- Jaiganesh V, Kumar D J, Girijadevi J, 2014, Automated Guided Vehicle with Robotic Logistics System, Procedia Engineering, 97, 2011–2021.
- Kirsch C, Röhrig C, 2011, Global Localization and Position Tracking of an Automated Guided Vehicle, Proceedings of the 18th World Congress The International

- Federation of Automatic Control, 28 August - 2 September, Milano Italy, 14036-14041.
- Koca H, 2007, Robot Manipulator Denetimi, Gazi Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, 126s, Ankara.
- Le-Anh T, De Koster M B M, 2006, A review of design and control of automated guided vehicle systems, *European Journal of Operational Research*, 171, 1–23.
- Lu S, Xu C, Zhong R Y, Wang L, 2017, A RFID-enabled positioning system in automated guided vehicle for smart factories, *Journal of Manufacturing Systems*, 44, 179–190.
- Mercan E, 2010, Otomatik Yönlendirmeli Araç Tasarımı Ve Gerçek Zamanlı Kontrolü, Atatürk Üniversitesi Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, 55s, Erzurum.
- Mirkhani M, Forsati R, Shahri A M, Moayedikia A, 2013, A novel efficient algorithm for mobile robot localization, *Robotics and Autonomous Systems*, 61, 920–931.
- Özdemir N B, 2015, Hibrit Yöntemler İle Kapalı Mekân Konum Belirleme, Selçuk Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, 56s, Konya.
- Özışık O, 2010, Robot Takımı İle Eş Zamanlı Konum Belirleme Ve Haritalama, Yıldız Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, 138s, İstanbul.
- Özyalçın İ, 2006, Kartezyen Robot Tasarımı, Mustafa Kemal Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, 97s, Antakya.
- Roszkowska E, 2003, Supervisory Control For Closed Agv Systems With Dynamic Vehicle Routing, *IFAC Robot Control*, 1-3 September, Wroclaw Poland, 413-418.
- Shah M, Lin L, Nagi R, 1997, A production order-driven AGV control model with object-oriented implementation, *Computer Integrated Manufacturing Systems*, 10-1, 35-48.
- Tuncer B, 2009, Konum Belirleme Algoritma Uygulamaları, İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, 48s, İstanbul.
- Um I, Cheon H, Lee H, 2009, The simulation design and analysis of a Flexible Manufacturing System with Automated Guided Vehicle System, *Journal of Manufacturing Systems*, 28, 115-122.
- Vis I F A, (2006). Survey of research in the design and control of automated guided vehicle systems, *European Journal of Operational Research*, 170, 677–709.

- Wu X, Lou P, Shen K, Peng G, Tang D, 2014, Precise Transshipment Control Of An Automated Magnetic-Guided Vehicle Using Optics Positioning, International Journal On Smart Sensing And Intelligent Systems, Vol. 7, No. 1, Nanjing China, 48-71.
- Yan R, Dunnett S J, Jackson L M, 2018, Novel methodology for optimising the design, operation and maintenance of a multi-AGV system, Reliability Engineering and System Safety, 178, 130–139.
- Yang Y, Zhong M, Dessouky Y, Postolache O, 2018, An integrated scheduling method for AGV routing in automated container terminals, Computers & Industrial Engineering, 126, 482–493.
- Yıldırım G, 2012, Kablosuz Sensör Ağı İle Konum Belirleme Ve İzleme, Fırat Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, 103s, Elazığ.
- Yılmaz O, 2010, Kablosuz Teknolojilerle Kapalı Alanda Konum Belirleme, Yıldız Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, 60s, İstanbul.
- Yılmaz S, 2006, Bir Robot Kolu Mekanizmasında Adım Motorları Vasıtasıyla Verilen Koordinatlara Hareketin Gerçekleştirilmesi, Sakarya Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, 142s, Sakarya.

İnternet Kaynakları

- 1- <https://www.coursehero.com/file/17017894/AGVS/>, 10.05.2019
- 2- <https://www.encyclopedia.com/entrepreneurs/encyclopedias-almanacs-transcripts-and-maps/automated-guided-vehicle-agv>, 10.05.2019
- 3- <http://www.reconditionedforklifts.com/blog/forklift-history-2/history-automated-guided-vehicles>, 12.05.2019
- 4- <http://www.mhi.org/fundamentals/automatic-guided-vehicles>, 20.05.2019
- 5- <https://prezi.com/6s6eypmwtidd/the-history-of-the-agv/>, 12.05.2019
- 6- <https://www.handling.com/company/blog/what-is-an-agv-system/>, 23.05.2019
- 7- <https://www.conveyco.com/future-agvs-new-technology-keep-eye/>, 13.05.2019
- 8- https://link.springer.com/chapter/10.1007/978-3-662-44814-4_1, 10.05.2019
- 9- https://www.kollmorgen.com/en-us/blogs/_blog-in-motion/articles/samuel-alexandersson/driverless-vehicles-a-reality-at-volvo-already-in-1973/, 12.05.2019

- 10- <https://www.automaticguidedvehicles.com/>, 24.05.2019
- 11- <https://www.dbhrobot.com/AGV-system/Inertial-navigation-AGV.html>, 16.05.2019
- 12- <https://www.sick.com/ag/en/automated-guided-vehicle-laser-navigation-with-reflectors/c/p170640>, 23.05.2019
- 13- <https://www.goetting-agv.com/components/optical/introduction>, 14.05.2019
- 14- <https://www.roboteq.com/index.php/roboteq-products-and-services/magnetic-guide-sensors>, 28.05.2019
- 15- <https://www.slideshare.net/alamkhan111/aotomated-guided-vehicle>, 23.05.2019
- 16- <https://danetechnologies.com/agv-robotics/>, 23.05.2019
- 17- <https://www.jbtc.com/en/north-america/automated-systems/expertise/navigation-innovations?lr=1>, 23.05.2019
- 18- <http://www.axter-agv.com/axter-agv-solutions/agv-navigation/>, 24.05.2019
- 19- <http://www.modoya.com/ozgun-urunlerimiz/agv-oya-nedir/>, 12.05.2019
- 20- https://www.globalspec.com/learnmore/material_handling_packaging_equipment/material_handling_equipment/automatic_guided_vehicles_agv, 24.05.2019
- 21- <https://www.transbotics.com/learning-center/guidance-navigation>, 24.04.2019
- 22- <https://www.electroschematics.com/12275/motor-speed-sensor-module-circuit/>, 12.04.2019
- 23- <https://www.jungheinrich.com.tr/ueruenler/otomatik-enduestriyel-araclar/otomatik-yoenlendirmeli-araclar-sueruecuesuez/>, 15.04.2019
- 24- <http://www.elektrikrehberiniz.com/elektrik-motorlari/dc-motor-nedir-454/>, 25.05.2019
- 25- <https://isccompanies.com/parts-distribution/motors/dc-motors/>, 12.05.2019
- 26- <http://www.elektrikbilgisi.com/YaziOku.asp?id=104#.XO8XDdIzbZ6>, 28.05.2019
- 27- <http://temelelektronik.info/devre.aspx?devre=263>, 26.05.2019
- 28- <https://elektronikhobi.net/dc-motor-hiz-kontrolu/>, 23.05.2019
- 29- <https://www.theengineeringprojects.com/2018/06/introduction-to-arduino-uno.html>, 27.05.2019
- 30- <https://buraktahtacioglu.blogspot.com/2016/02/arduino-ile-gyroscope-ve-accelerometer.html>, 29.05.2019
- 31- <http://www.barissamanci.net/Makale/26/accelerometer-gyroscope-imu-nedir/>, 23.05.2019

- 32- <https://www.datasheetspdf.com/mobile-pdf/50345/MotorolaInc/LM393.html>,
29.05.2019
- 33- <http://physicistmfa.blogspot.com/2013/11/14-motor-kontrolleri.html>, 30.05.2019
- 34- <https://canerkiziloglu.blogspot.com/2017/11/tork-guc-devir-iliskisi.html>,
30.05.2019
- 35- <https://.machinedesign.com%2Fmotion-control%2Fget-attuned-servo-motor-driver-tuning-0&psig=AOvVaw2IejZXJRvBaQf9oqILc54&ust=1569583096798812>,
06.06.2019
- 36- <https://circuitdigest.com/article/what-is-pid-controller-working-structure-applications>, 08.06.2019
- 37- <https://www.pololu.com/docs/0j19/all>, 10.07.2019
- 38- <https://www.elprocus.com/mpu6050-pin-diagram-circuit-and-applications/>,
10.06.2020
- 39- <https://www.robotzade.com/Bond-Silikon-Tekerlek-Seti-Siyah,PR-1495.html>,
01.06.2020
- 40- <https://www.racecar-engineering.com/tech-explained/racecar-vehicle-dynamics-explained/attachment/racecar-vehicle-dynamics-roll-pitch-yaw/>, 12.05.2020
- 41- <https://howthingsfly.si.edu/flight-dynamics/roll-pitch-and-yaw>, 10.05.2020
- 42- https://acikders.ankara.edu.tr/pluginfile.php/70147/mod_resource/content/0/GDM4_04_11.pdf, 20.04.2020
- 43- https://www.generationrobots.com/media/capteurs-de-reflexion-infrarouge-QTR-8RC/QTR_arduino_library.pdf, 01.04.2020
- 44- <https://www.robotzade.com/Rapid-Cizgi-Izleyen-Robot-Govdesi,PR-1400.html>,
23.05.2020
- 45- <https://www.f1depo.com/12V-3000-Rpm-DC-Motor-Force-Up-Ozel-Uretim-Cizgi-Izleyen-Motoru,PR-1572.html>, 12.05.2020
- 46- <https://market.samm.com/arduino-uno-r3-smd-ch340-chipset-klon-usb-kablo-dahil>,
10.05.2020
- 47- <https://roboticx.ps/product/pololu-qtr-8rc-reflectance-sensor-array/>, 05.04.2020
- 48- <https://www.robotistan.com/qtr-8rc-kizilotesi-sensor>, 06.04.2020
- 49- <http://www.roboweb.net/qtr-8rc-reflektans-sensoru-dizisi-rw-pl-961.html>,
06.04.2020

- 50- <https://learn.sparkfun.com/tutorials/ardumoto-kit-hookup-guide/resources--going-further>, 02.04.2020
- 51- <https://www.robotzade.com/111V-Lipo-Pil-500mAh,PR-1480.html>, 26.05.2020
- 52- <https://www.f1depo.com/Sarhos-Tekerlek-2-Adet,PR-1230.html>, 23.05.2020
- 53- <https://info4idea.com/c-mssql-timeout-suresini-arttirma-veya-sinirsiz-yapma/>, 01.06.2020
- 54- <https://www.elektrikrehberiniz.com/elektrik-motorlari/dc-motor-calisma-prensibi-472/>, 05.05.2020
- 55- <http://www.robotpark.com.tr/Dc-Motor-Nedir>, 12.01.2020
- 56- <http://www.jarzebski.pl/arduino/czujniki-i-sensory/3-osiowy-zyroskop-i-akcelerometr-mpu6050.html>, 04.05.2020
- 57- <https://www.elektrikport.com/teknik-kutuphane/h-koprusu-nedir/17187#ad-image-0>, 08.07.2020
- 58- <https://hayaletveyap.com/arduino-ile-dc-motor-kontrolu/>, 08.07.2020
- 59- <https://www.elektrikrehberiniz.com/category/elektrik-motorlari/>, 08.07.2020
- 60- <http://datagenetics.com/blog/december12016/index.html>, 08.07.2020
- 61- <https://www.anaheimautomation.com/manuals/forms/encoder-guide.php>, 08.07.2020

ÖZGEÇMİŞ

Adı Soyadı : Hasan SANCAK
Doğum Yeri ve Tarihi : AFYONKARAHİSAR 02/05/1977
Yabancı Dili : İngilizce
İletişim (Telefon/e-posta) : 0551 934 90 10 / hsancak@aku.edu.tr

Eğitim Durumu (Kurum ve Yıl)

Lise : Afyon Endüstri Meslek Lisesi, Elektrik Bölümü, (1990-1993)
Lisans : Çanakkale 18 Mart Üniversitesi, Mühendislik Mimarlık Fakültesi, Bilgisayar Mühendisliği Bölümü, (1995-1999)
Yüksek Lisans : Afyon Kocatepe Üniversitesi, Fen Bilimleri Enstitüsü Makine Mühendisliği ABD, (2011-2020)

Çalıştığı Kurum/Kurumlar ve Yıl

: Diyalog Bilgi İşlem İzmir (2000 – 2002)
: Afyon Kocatepe Üniversitesi Bolvadin Meslek Yüksekokulu (2002– Devam Ediyor)

EKLER

EK 1. Arduino -1 Program Kodları

```
/* Otomatik Klavuzlu Araçlarda Eş Zamanlı Konum Belirleme ve Yönlendir
 * Danışman : Dr. Öğr. Üyesi Fatih BAYRAM
 * Hasan SANCAK
 */
```

```
// Arduino -1 Motor Kontrol Programı
```

```
#include <SoftwareSerial.h>
#include <QTRSensors.h>
```

```
// -----HC-06 TANIMLAMA-----
SoftwareSerial bt(0, 1); // RX-1 TX-0
// -----HC-06 TANIMLAMA-----
```

```
// -----MOTOR TANIMLARI-----
#define RMHIZ 80
#define LMHIZ 80
#define LMYON 13
#define LMPWMPIN 11
#define RMYON 12
#define RMPWMPIN 10
// -----MOTOR TANIMLARI-----
```

```
// -----DEĞİŞKEN TANIMLARI-----
int i;
char InData;
int FAULT, COR_SPEED;
int LAST_FAULT = 0;
float Kp = 0.10;
float Kd = 0.80;
```

```
int RMPWM = 0;
int LMPWM = 0;
int GROUND_COLOR=1;
// -----DEĞİŞKEN TANIMLARI-----
```

```
// -----SENSÖR TANIMLARI-----
QTRSensorsRC qtrrc((unsigned char[]) { 2, 3, 4, 5, 6, 7, 8, 9} ,8, 3500,
QTR_NO_EMITTER_PIN);
unsigned int sensorValues[8];
// -----SENSÖR TANIMLARI-----
```

```
// -----KALİBRASYON-----
```

EK 1. (Devam) Arduino -1 Program Kodları

```
void Manuel_Calibration(){
  digitalWrite(13, HIGH);

  for (i = 0; i < 400; i++)
  {
    qtrrc.calibrate();
    delay(20);
  }

  digitalWrite(13, LOW);

  delay(1000);
}
// -----KALİBRASYON-----

// -----POZİSYON HESABI-----
void Sensor_Read(){
  unsigned int position = qtrrc.readLine(sensorValues,1,GROUND_COLOR);
  FAULT = position-3500;
}
// -----POZİSYON HESABI-----

// -----MOTOR FARKLARI-----
void Line_Tracking(){

  Sensor_Read();

  COR_SPEED = Kp * FAULT + Kd*(FAULT - LAST_FAULT);
  LAST_FAULT = FAULT;

  RMPWM = RMHIZ + COR_SPEED ;
  LMPWM = LMHIZ - COR_SPEED ;

  RMPWM = constrain(RMPWM, -120, 120);
  LMPWM = constrain(LMPWM, -120, 120);

  M_CONTROL(RMPWM,LMPWM);
}
// -----MOTOR FARKLARI-----

// -----MOTOR KONTROL-----
void M_CONTROL(int RMPWM, int LMPWM){
  if(RMPWM<=0) {
    RMPWM=abs(RMPWM);
    digitalWrite(RMYON, LOW);
    analogWrite(RMPWMPIN, RMPWM);
  }
}
```

EK 1. (Devam) Arduino -1 Program Kodları

```
}
else {
    digitalWrite(RMYON, HIGH);
    analogWrite(RMPWMPIN, RMPWM);
}

if(LMPWM<=0) {
    LMPWM=abs(LMPWM);
    digitalWrite(LMYON, HIGH);
    analogWrite(LMPWMPIN, LMPWM);
}
else {
    digitalWrite(LMYON, LOW);
    analogWrite(LMPWMPIN, LMPWM);
}
}
// -----MOTOR KONTROL-----

// -----ARAÇ KONTROL-----
void M_Tracking(){
    Line_Tracking();
}

void STOP_M(){
    M_CONTROL(0,0);
}

void TURN_RIGHT(){
    M_CONTROL(-80,80);
}

void TURN_LEFT(){
    M_CONTROL(80,-80);
}

void M_Backward(){
    M_CONTROL(-50,-50);
}

void M_Forward(){
    M_CONTROL(50,50);
}
// -----ARAÇ KONTROL-----

// -----SETUP-----
void setup(){
```

EK 1. (Devam) Arduino -1 Program Kodları

```
Serial.begin(9600);
bt.begin(9600);

pinMode(RMYON, OUTPUT);
pinMode(RMPWMPIN, OUTPUT);
pinMode(LMYON, OUTPUT);
pinMode(LMPWMPIN, OUTPUT);

Manuel_Calibration();
}
// -----SETUP-----

// -----LOOP-----
void loop(){
  if(bt.available(>0){
    InData=bt.read();
    switch(InData){
      case 'N': M_Tracking(); break;
      case 'F': M_Forward(); break;
      case 'L': TURN_LEFT(); break;
      case 'R': TURN_RIGHT(); break;
      case 'B': M_Backward(); break;
      case 'S': STOP_M(); break;
      case ' ': STOP_M(); break;
    }
  }
}
// -----LOOP-----
```

EK 2. Arduino -2 Program Kodları

```
#include <Encoder.h>
```

```
/* Otomatik Klavuzlu Araçlarda Eş Zamanlı Konum Belirleme ve Yönlendir
```

```
* Danışman : Dr. Öğr. Üyesi Fatih BAYRAM
```

```
* Hasan SANCAK
```

```
*/
```

```
// Arduino -2 Veri İletişim Programı
```

```
#include <SoftwareSerial.h>
```

```
#include <QTRSensors.h>
```

```
/* MPU6050 Triple Axis Gyroscope & Accelerometer. Pitch & Roll & Yaw Gyroscope  
Example.
```

```
* GIT: https://github.com/jarzebski/Arduino-MPU6050
```

```
* (c) 2014 by Korneliusz Jarzebski
```

```
*/
```

```
#include <Wire.h>
```

```
#include <MPU6050.h>
```

```
// -----HC-06 TANIMLAMA-----
```

```
SoftwareSerial bt(6,7); // RX-1 TX-0
```

```
// -----HC-06 TANIMLAMA-----
```

```
// -----DEĞİŞKEN TANIMLARI-----
```

```
int bs=10;
```

```
int GROUND_COLOR=1;
```

```
String data;
```

EK 2. (Devam) Arduino -2 Program Kodları

```
const byte RMPinA = 2;
const byte RMPinB = 4;
const byte LMPinA = 3;
const byte LMPinB = 5;
byte RMPinALast;
byte LMPinALast;
int RMPulse;
boolean RMDirection;
int LMPulse;
boolean LMDirection;
// -----DEĞİŞKEN TANIMLARI-----

// -----SENSÖR TANIMLARI-----
QTRSensorsRC qtrrc((unsigned char[]) { 8, 9, 10, 11, 12, 13 } ,8, 2500,
QTR_NO_EMITTER_PIN);
unsigned int sensorValues[8];
// -----SENSÖR TANIMLARI-----

-----MPU6050 TANIMLARI-----
MPU6050 mpu;
unsigned long timer = 0;
float timeStep = 0.01;
float yaw = 0;
-----MPU6050 TANIMLARI-----

// -----SETUP-----
void setup(){
  Serial.begin(9600);
  bt.begin(9600);
  EncoderInit();
  delay(1000);
```

EK 2. (Devam) Arduino -2 Program Kodları

```
mpu.begin(MPU6050_SCALE_2000DPS, MPU6050_RANGE_2G);
mpu.calibrateGyro();

digitalWrite(13, LOW);
delay(1000);
digitalWrite(13, HIGH);

for (int i = 0; i < 200; i++)
{
  qtrrc.calibrate();
  delay(20);
}

digitalWrite(13, LOW);
delay(1000);
}
// -----SETUP-----

// -----LOOP-----
void loop(){
  data = RMPulse;
  data += "&";
  data += LMPulse;
  data += "&";

  timer = millis();
  unsigned int position = qtrrc.readLine(sensorValues,1,GROUND_COLOR);
  Vector norm = mpu.readNormalizeGyro();
  yaw = yaw + norm.ZAxis * timeStep;
  data += position;
```


EK 2. (Devam) Arduino -2 Program Kodları

```
data += "&";
data += yaw;
bt.println(data);
delay((timeStep*1000) - (millis() - timer));
}
// -----LOOP-----

void EncoderInit()
{
  RMDirection = true;
  pinMode(RMpinB,INPUT);
  attachInterrupt(0, RMSpeed, CHANGE);
  LMDirection = true;
  pinMode(LMpinB,INPUT);
  attachInterrupt(1, LMSpeed, CHANGE);
}

void RMSpeed()
{
  int Lstate = digitalRead(RMpinA);
  if((RMpinALast == LOW) && Lstate==HIGH)
  {
    int val = digitalRead(RMpinB);
    if(val == LOW && RMDirection)
    {
      RMDirection = false;
    }
    else if(val == HIGH && !RMDirection)
    {
      RMDirection = true;
    }
  }
}
```

EK 2. (Devam) Arduino -2 Program Kodları

```
}  
RMPinALast = Lstate;  
  
if(RMDirection) RMPulse++;  
else RMPulse--;  
}  
  
void LMSpeed()  
{  
  int Lstate = digitalRead(LMpinA);  
  if((LMpinALast == LOW) && Lstate==HIGH)  
  {  
    int val = digitalRead(LMpinB);  
    if(val == LOW && LMDirection)  
    {  
      LMDirection = false;  
    }  
    else if(val == HIGH && !LMDirection)  
    {  
      LMDirection = true;  
    }  
  }  
  LMpinALast = Lstate;  
  
  if(!LMDirection) LMPulse++;  
  else LMPulse--;  
}
```