

T.C.
KIRKLARELİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
MAKİNE MÜHENDİSLİĞİ ANABİLİM DALI

ÇİZGİ İZLEYEN ROBOT İLE YÜK TAŞIMA SİSTEMİ TASARIMI



YÜKSEK LİSANS TEZİ

Aybüke ŞAHİN

MAYIS - 2024

T.C.
KIRKLARELİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
MAKİNE MÜHENDİSLİĞİ ANABİLİM DALI

ÇİZGİ İZLEYEN ROBOT İLE YÜK TAŞIMA SİSTEMİ TASARIMI

YÜKSEK LİSANS TEZİ

Aybüke ŞAHİN

Tez Danışmanı: Prof. Dr. Erol TÜRKEŞ

Mayıs – 2024

ÇİZGİ İZLEYEN ROBOT İLE YÜK TAŞIMA SİSTEMİ TASARIMI adlı tez çalışması adlı tez çalışması **Aybüke ŞAHİN** tarafından hazırlanmış olup aşağıdaki jüri tarafından **OY BİRLİĞİ** ile Kırklareli Üniversitesi Fen Bilimleri Enstitüsü Makine Mühendisliği Anabilim Dalı'nda Yüksek Lisans Tezi olarak kabul edilmiştir.

Tez Danışmanı:

Prof. Dr. Erol TÜRKEŞ
Kırklareli Üniversitesi

.....

Jüri Üyeleri:

Doç. Dr. Ümit HÜNER
Kırklareli Üniversitesi

.....

Doç. Dr. Gürkan İRSEL
Trakya Üniversitesi

.....

Tez Savunma Tarihi: 29/05/2024

.....
Doç. Dr. H. Hale KARAYER
Fen Bilimleri Enstitüsü Müdürü

ETİK BEYAN

Kırkırelı Üniversitesi Fen Bilimleri Enstitüsü Tez ve Proje Yazım Kurallarına uygun olarak hazırladığım bu tez çalışmasında; tez içinde sunduğum bilgileri, verileri ve dokümanları, değişik sonuç verebilecek şekilde araştırma araç gereçleri kullanmadan, işlem veya kayıt sonuçlarını değiştirmeden akademik ve etik kurallar çerçevesinde elde ettiğimi, bilimsel etik ve ahlak kurallarına uygun olarak sunduğumu, tez çalışmasında yararlandığım eserlerin tümüne uygun atıfta bulunarak kaynak gösterdiğimi, bu tezde sunduğum çalışmanın özgün olduğunu bildirir, aksi bir durumda aleyhime doğabilecek tüm hak kayıplarını kabullendiğimi beyan ederim.

Aybüke ŞAHİN
29/05/2024

ÖZET

ÇİZGİ İZLEYEN ROBOT İLE YÜK TAŞIMA SİSTEMİ TASARIMI

Aybüke ŞAHİN

Yüksek Lisans Tezi

Kırklareli Üniversitesi

Fen Bilimleri Enstitüsü

Danışman: Prof. Dr. Erol TÜRKES

Mayıs 2024, 66 sayfa

Çizgi izleyen robotlar, modern teknolojinin ilerlemesiyle endüstriyel sektörde yaygın bir şekilde kullanılmaya başlanmıştır. Özellikle büyük ölçekli sanayi tesislerinde lojistik ve otomasyon alanlarında, niteliksiz iş gücüne olan ihtiyacı azaltmak amacıyla bu tür robotlar giderek daha fazla tercih edilmektedir. Bu tez çalışmasında, literatürde yer alan çizgi izleyen robotlara dair projeler detaylı bir şekilde incelenmiş ve endüstriyel kullanıma uygun, yük dolum ve boşaltma istasyonlarını algılayabilen bir robotun tasarımı üzerine yoğunlaşmıştır. Bu tez çalışmasının amacı, endüstriyel ortamlarda çizgi izleme teknolojisinin potansiyel çözümler sunabilmesinin yanı sıra, yazılımsal ihtiyaçları karşılayarak yeni sistemlere entegre edilebilirliğini artırmaktır. Bu projenin çekirdeğinde, çizgileri algılayıp takip edebilen bir araç ve bu aracın üzerine monte edilmiş, kutuları belirli duraklardan alıp bırakabilen bir robot kol bulunur. Mobil robotta kol tasarımı LDR ile kontrol edilmiştir. Tasarım, Arduino mikrokontrolörü kullanarak mekatronik ve otomasyon prensiplerinin bir araya getirilmesiyle gerçekleşir. Birden fazla çeşidi olan Arduino kart içerisinden Arduino Nano kart kullanılmıştır ve bu sayede devre kartı boyutunun küçük olması sağlanmıştır.

Anahtar Kelimeler: Çizgi Takip Robot, Endüstriyel Çizgi Takip, Arduino, Robot Kontrolü.

ABSTRACT

LOAD CARRYING SYSTEM DESIGN WITH LINE FOLLOWING ROBOT

Aybüke ŞAHİN

MSc Thesis

Kirklareli University

Graduate School of Natural and Applied Sciences

Supervisor: Prof. Dr. Erol TÜRKEŞ

May 2024, 66 pages

Line-following robots have begun to be widely produced with the advancement of modern technology. Especially in the fields of logistics and automation in large-scale industrial facilities, such robots are increasingly preferred in order to minimize the price of unskilled labor. In this thesis study, the projects for line following robots in the literature have been examined in detail and the focus has been on the design of a robot suitable for industrial use that can detect cargo filling and unloading stations. The purpose of this thesis is to increase the potential solutions of industrial line tracking technology as well as its integration into new systems by providing software solutions. In this application, there is a vehicle that can detect and follow lines and a robotic arm mounted on this vehicle that can pick up and release from the operating selective stops. The arm design in the mobile robot is controlled by LDR. The design combines mechatronics and automation principles using the Arduino microcontroller. Among the Arduino cards that have more than one option, the Arduino Nano card was used, thus minimizing the circuit board replacement.

Anahtar Kelimeler: Line Follow Robot, Industrial Line Following, Arduino, Robot Control

TEŞEKKÜR

Sabrı, tecrübesi ve ilgisiyle tez çalışmamı sürdürmem için desteğini, bilgisini ve yol göstericiliğini hiçbir zaman esirgemeyen danışmanım Sayın Prof. Dr. Erol TÜRKEŞ'e,

Bana olan sonsuz inancını, güvenini ve sevgisini her zaman hissettiğim hayat arkadaşım Ahmet AKSOY'a,

Sevgisiyle, sabrıyla bana her zaman güç veren, beni bugünlere getiren ve en iyi şekilde yetiştiren anneme, babama ve her zaman moral desteğim olan kardeşime sonsuz içtenlikle

Teşekkür ederim.

İÇİNDEKİLER

ÖZET	IV
ABSTRACT	V
TEŞEKKÜR.....	VI
İÇİNDEKİLER.....	VII
ŞEKİLLERİN LİSTESİ.....	IX
RESİMLERİN LİSTESİ.....	X
SİMGELER VE KISALTMALAR	XI
1. GİRİŞ.....	1
2. ROBOT KAVRAMI	3
2.1. Robotların Çalışma Prensipleri.....	4
2.2. Robotların Sınıflandırılması	5
3. ENDÜSTRİYEL MOBİL ROBOTLARIN GELİŞİMİ	9
4. ENDÜSTRİYEL MOBİL ROBOTUN TASARIMI.....	15
4.1. Servo Motor.....	15
4.2. Kızılötesi Sensör.....	16
4.3. Sarhoş Tekerlek	19
4.4. Sürücü Kartı.....	20
4.5. Bağlantılar ve Ultrasonik Sensör.....	20
4.6. DC Motor.....	23
4.7. Arduino Nano	26
4.8. LDR ve Servo Sürücü Kartı	28
4.9. PID Algoritması.....	30
5. MATERYAL VE YÖNTEM	33
6. BULGULAR VE TARTIŞMA	37
6.1. Sensör Entegrasyonu ve Algılama Kapasitesi	37
6.2. Motor Kontrolü, Navigasyon ve İşlevselliği	37
6.3. Hata Yönetimi, Güvenlik Protokolleri ve Kullanıcı Arayüzü	38
6.4. Güç Yönetimi, Verimlilik, Çevresel Adaptasyon ve Ağ-İletişim Kapasitesi.....	39
6.5. Algoritmik Zorluklar ve İşlem Kapasitesi	40
6.6. Çoklu Öğrenme Yönetimi ve Öğrenme Kapasitesi	40

7. SONUÇ VE ÖNERİLER	43
KAYNAKLAR	47
EKLER	53
EK A. Projede Yazılan Kod	53
EK B. Robot Özellik Tabloları	62
ÖZGEÇMİŞ	66



ŞEKİLLERİN LİSTESİ

Şekil 4.1. DC servo motor	15
Şekil 4.2. Kablo bağlantıları gösterimi.....	16
Şekil 4.3. Pololu QTR-8RC çizgi algılayıcının boyutları	17
Şekil 4.4. Pin çıkışları	18
Şekil 4.5. Hat sensör şeması.....	18
Şekil 4.6. Sarhoş tekerlek parçaları	19
Şekil 4.7. L298N motor sürücü kartı ve blog diyagramı	20
Şekil 4.8. Sensörün çalışma prensibi.....	22
Şekil 4.9. Plastik engele karşı sensör verileri.....	23
Şekil 4.10. DC motor	24
Şekil 4.11. Arduino nano pin şeması	27
Şekil 4.12. Arduino nano pin şeması.....	28
Şekil 4.13. LDR'nin iç yapısı ve çalışma prensibi	28
Şekil 4.14. LDR'nin çalışma örneği.....	29
Şekil 4.15. PCA9685 servo sürücü kartı	30

RESİMLERİN LİSTESİ

Resim 5.1. Projede Tasarlanan Endüstriyel Mobil Robot.....	35
Resim 5.2. Projede Tasarlanan Endüstriyel Mobil Robot.....	36



SİMGELER VE KISALTMALAR

Bu çalışmada kullanılmış simgeler ve kısaltmalar, açıklamaları ile birlikte aşağıda sunulmuştur.

Simgeler

Açıklamalar

Ms	Milisanıye
mA	Miliamper
Kp	Orantısal Kazanç
Ki	İntegral Kazanç
Kd	Türev Kazanç
V	Volt
Ω	Direnç birimi
μs	Mikrosaniye
tk	Lojik 1 süresi
V	Volt
mAh	Miliamper saat
gr	Gram

Kısaltmalar

Açıklamalar

AC	Alternatif Akım
ADC	Analogdan Dijitale Dönüştürücü
DC	Doğru Akım
DAC	Dijitalden Analığa Dönüştürücü
GND	Toprak Bağlantısı
GPIO	Genel Amaçlı Giriş/Çıkış
IR	Kızılötesi
IDE	Tümleşik Geliştirme Ortamı
ISO	Uluslararası Standartlar Organizasyonu
LED	Işık Yayan Diyot

LDR	Işığa Bağımlı Direnç
OUT	Çıkış
PWM	Darbeli Genişlik Modülasyonu
PID	Oransal-İntegral-Türev
RPM	Dakikada devir sayısı
RF	Radyo Frekansı
UART	Evrensel Asenkron Alıcı/Verici
VCC	Ortak Kolektör Voltajı
QTR	Şarj Transfer Direnci



1. GİRİŞ

Endüstriyel otomasyonun gelişimi, insan müdahalesine gerek kalmadan başlangıçtan sona kadar işlemleri yönetebilen robot otomasyon sistemlerinin ortaya çıkışını zorunlu kılmıştır. Bu sistemler, algılama elemanlarının yardımıyla çevreden aldıkları bilgilerle ve kendi veri tabanlarındaki bilgileri kullanarak kararlar alabilir ve bu kararlar doğrultusunda işlemleri gerçekleştirebilir özelliklere sahiptirler [1]. Robotların endüstride yaygın olarak kullanılmasının nedenleri arasında işlem hızı, güvenilirlik ve insan sağlığı için riskli ortamlarda çalışabilme yeteneği gibi faktörler bulunmaktadır [2]. Özellikle seri üretimin kaçınılmaz hale gelmesiyle, robotlar firmalar için stratejik bir öneme sahip olmuştur [3]. Robotların geliştirilmesi ve kullanılmasında çevresel değişikliklere adaptasyon, sensörlerden gelen bilgilerin işlenmesi, özelleştirilmiş kontrol sistemlerinin geliştirilmesi ve PID algoritmalarının doğru parametrelerle ayarlanması gibi zorluklar bulunmaktadır [2]. Robot otomasyon sistemlerinin gelişiminde, Arduino gibi platformların düşük maliyet, geniş kütüphane desteği ve kullanıcı dostu arayüz sunması, bu tür projelerin daha da yaygınlaşmasına olanak tanımaktadır [1]. Bu sayede, endüstriyel çizgi takip eden robotlar ve bu robotların çeşitli uygulama alanlarındaki potansiyellerini keşfetmek ve bu alanda yenilikçi çözümler sunmak mümkün hale gelmiştir.

Çizgi izleme teknolojisi, fabrikalardaki parça taşıma işlemlerinde ve otomotiv sektöründeki araçlarda "şerit takip sistemleri" gibi alanlarda güncel ve etkili uygulamalara sahiptir [4]. Bu sistemler, özellikle lojistik ve otomasyon süreçlerinde, niteliksiz insan gücüne olan gereksinimi azaltırken iş güvenliği ve verimlilik açısından önemli avantajlar sunmaktadır [5]. Robot teknolojisinin ilerlemesiyle birlikte, robot kolları malzeme taşıma, kaynak, montaj, dağıtım, termal püskürtme boyama ve delme gibi birçok alanda yaygın olarak kullanılmaya başlanmıştır [6]. Bu gelişme, otomasyon ve robotiğin yakından ilişkili olduğu gerçeğini pekiştirirken, endüstriyel otomasyonda robot kollarının kullanımı, işlemlerin verimliliğini artırarak önemli bir rol oynamaktadır. Bu bağlamda, robot otomasyon sistemleri, zorlu çalışma koşulları altında bile yüksek hassasiyet ve doğrulukla görevlerini yerine getirebilen mekanizmalar olarak öne çıkmaktadır. Sonuç olarak, robot otomasyon sistemleri ve çizgi izleme teknolojisinin endüstrideki yaygın kullanımı, üretim süreçlerinin daha verimli, güvenli ve ekonomik

hale gelmesini sağlamıştır. Robotların geliştirilmesi ve kullanılmasıyla ilgili karşılaşılan teknik sorunlar [2], projenin yapım aşamasında çözüm yolları aranan zorluklardır. Ancak, bu zorlukların üstesinden gelindiğinde, robot teknolojisinin sağladığı avantajlar, onları endüstriyel uygulamalarda vazgeçilmez hale getirmektedir [7].

Bu tez çalışmasında çizgi izleyen robot tasarımı yapılmıştır bunun amacı fabrikalarda üretilen parçaların veya mamullerin operasyonları arasında malzeme taşınmasını kolay kılmak ve otomasyon haline getirmek amacıyla belirlenen istasyonlar arasında insan gücü yerine robotların kullanılmasıdır. Bu da işlem kolaylığı ve süre bakımından verim sağlayacağı açıktır. Bu tez çalışmasında tasarlanan robotun özelliği çizgileri algılayıp takip edebilen bir araç ve bu aracın üzerine monte edilmiş, kutuları belirli duraklardan alıp bırakabilen bir robot kol şeklindedir. Tasarım, Arduino mikrokontrolörü kullanarak mekatronik ve otomasyon prensiplerinin bir araya getirilmesiyle gerçekleştirilmiştir. Birden fazla çeşidi olan Arduino kart içerisinde Arduino Nano kart kullanılmış ve bu sayede devre kartı boyutunun küçük olması sağlanmıştır. Robot tasarlanırken ergonomik açıdan da tasarım kriterleri göz önüne alınmıştır. Robot kolu tasarlanırken malzemeyi en kolay biçimde alabileceği ve yerleştirebileceği durumlar göz önünde bulundurularak, kinematik açıdan avantajlı olarak dizayn edilmiştir. Sürücü kartları ve elektronik sistemler robotun montaj ve demontaj kolaylığı göz önünde bulundurularak her biri uygun yerlere konulmuştur. Robota düzgün hareket kabiliyeti (sarsılmaması) sağlamak için PID kontrol sistemi ile desteklenmiştir. LDR kontrol mekanizması olarak kullanılmıştır. Değişik ışık şiddetlerinde ani değişim ile motor aktif edilmiş, mobil robotun ilerleme hareketi durduğu anda LDR'nin algıladığı ışık seviyesine bakılarak robot kolunun nesneyi alma ya da bırakma durumu kontrol altına alınmıştır.

2. ROBOT KAVRAMI

Robot, çeşitli görevleri otomatik olarak yerine getirebilen, genellikle programlanabilir olan bir mekanik cihaz olarak tanımlanabilir. Robotlar, sensörler (algılayıcılar), işlemciler (kontrol sistemleri), hareket mekanizmaları (aktüatörler) ve yazılımlar gibi karmaşık sistemlerden meydana gelir. İnsan müdahalesi olmaksızın verilen görevleri başarılı bir şekilde tamamlayabilirler. Çevresel verilere dayanarak hareket etme, öğrenme ve karar verme yeteneklerine sahiptirler. Kullanım alanları endüstriyel otomasyondan ev içi yardımcılara, tehlikeli ortamlarda gerçekleştirilen araştırma ve kurtarma operasyonlarına kadar geniş bir yelpazeyi kapsar. Gelişen teknoloji ile robotlar, giderek daha zeki, bağımsız ve fonksiyonel hale gelmektedir. Fiziksel olarak etkileşimde bulunabilen, bu kabiliyetleri algılama, planlama, etkinlik gösterme ve kontrol sistemleri aracılığıyla algoritmik bir düzende işlem yürütebilen akıllı cihazlar olarak tanımlanabilirler. Her bir atanan işi başarıyla ve zamanında yerine getirmek üzere tasarlanmışlardır ve ayrıca verilen komutlara uygun şekilde hareket etmek üzere geliştirilmişlerdir. Robotların bazı temel özellikleri arasında programlanabilir olmaları, fiziksel bileşenlerin komutlarla hareket ettirilebilmesi ve belirli işler için tasarlanmış olmaları yer alır. Ayrıca, üzerindeki sensörler aracılığıyla çevresindeki ses, ışık, görüntü, titreşim ve ısı gibi etmenleri algılama yeteneğine sahip akıllı makinelerdir [8].

Sensörlerin verileri yeterli kalitede algılaması, algılanan verilerin toplanması ve kontrolünü sağlayan devrelerin bulunması, tasarlanan robotun belirlenen amaçlara uygun hareket edebilmesini sağlayan yazılımların bulunması, sonrasında yazılım tarafından belirlenen hareketleri gerçekleştirebilmeyi sağlayan mekanik yapıların bulunması bir makinenin robot olarak tanımlanabilmesi için sahip olması gereken temel özelliklerdir [8].

Robotik ise, makinelerin planlanması, inşa edilmesi, işletilmesi ve kullanılması ile ilgili bilimsel ve mühendislik disiplini [9]. Bu saha, bilgisayar teknolojileri, yapay zekâ, mekatronik, elektrik mühendisliği ve mekanik mühendislik gibi çeşitli disiplinlerin kesişiminden oluşur [10]. Bu sistemler, özellikle tehlikeli ya da tekrarlı görevleri otomatize ederek, yaşam kalitesini iyileştirmek amacıyla geliştirilmiştir. Robotlar, çeşitli

algılama ve kontrol sistemleri sayesinde bağımsız olarak hareket edebilir veya uzaktan yönetilebilir hale getirilmiş cihazlardır. Bu alan, günlük yaşamdan endüstriyel uygulamalara, sağlık sektöründen uzay keşiflerine kadar çok geniş bir uygulama yelpazesine sahiptir. Robotik, makinelerin ve kontrol sistemlerinin yanı sıra bilgisayar, elektronik ve uzay teknolojilerini içeren bir çalışma alanıdır. Bu alan, mekanizmaların tasarımı ve bunların elektronik donanımlar ile desteklenmesi, dijital platformlarda kontrol edilmesi ve özel koşullar altında işlev görmesi gibi süreçleri kapsar. Bu süreçler, çeşitli mühendislik dallarının iş birliğiyle yürütülmektedir [8].

Robotik, sadece insanların yapabildikleri işlevleri üstlenmekle kalmaz, aynı zamanda insanlara faydalı olacak özellikler geliştirmeye de odaklanır. İnsan ihtiyaçlarının sürekli olarak değişmesi, bu alandaki robotların gelişimini yönlendirmektedir. Özellikle riskli görevlerde (örneğin bomba imha, arama kurtarma, keşif görevleri) robotlardan giderek daha fazla yararlanılmaktadır. Robotik alanındaki bu ihtiyaçlara yanıt olarak, bilimsel ilerleme ve teknolojik yenilikler sürekli olarak artmaktadır. Robotik, dünyada büyük bir ilgi odağı olmanın yanı sıra Türkiye'de de bu alana olan ilgi dikkat çekicidir ve bu konsept Türkiye'de genellikle "mekatronik" olarak adlandırılır.

2.1. Robotların Çalışma Prensipleri

Robotlar, belirli görevleri otomatik olarak yerine getirebilen programlanabilir mekanik cihazlardır. Temelde, bir robotun bağımsız karar verebilmesi için sensörler, elektronik karar mekanizmaları ve mekanik uygulayıcılar gibi üç ana bileşene ihtiyacı vardır.

- **Algılama ve Sensörler:** Robotlar, çevrelerini algılamak için çeşitli sensörleri kullanır. Bu sensörler, ışık, ses, basınç, sıcaklık gibi fiziksel koşulları tespit eder ve bu verileri elektrik sinyallerine dönüştürerek robotun karar mekanizmasına aktarır [9]. Bu sayede robotlar, çarpma gibi istenmeyen durumları önleyebilir ve çevreleriyle etkileşimde bulunabilir.
- **Karar Verme Mekanizması:** Algılanan veriler, robotun merkezi işlem birimi olan işlemciye gönderilir. İşlemci, bu verileri değerlendirerek robotun ne yapması gerektiğine karar verir. Bu süreç, yazılım desteği ile yürütülür ve robotun karmaşık problemleri çözmesine, sosyal etkileşimde bulunmasına ve öğrenmesine olanak tanır.
- **Mekanik Uygulayıcılar ve Hareket:** Karar mekanizmasından çıkan komutlar, robotun hareket etmesini sağlayan aktüatörler veya motorlar tarafından fiziksel

harekete dönüştürülür. Bu mekanik uygulayıcılar, robotun çevresiyle etkileşimde bulunmasını sağlar ve beklenen görevleri yerine getirir.

- **Planlama ve Eylem:** Robotlar, belirlenen görevleri yerine getirmek için kendi içlerinde veya dışarıdan alınan bilgilere dayalı olarak eylem planları oluşturur. Bu planlama, basit bir hareket sekansından karmaşık bir problem çözme stratejisine kadar değişebilir.
- **Öğrenme ve Adaptasyon:** Gelişmiş robotlar, yapay zekâ ve makine öğrenimi teknikleri sayesinde deneyimlerinden öğrenip zamanla davranışlarını iyileştirebilir.

Bu bileşenlerin entegrasyonu, robotların bağımsız bir şekilde hareket etmesini ve çeşitli görevleri yerine getirmesini sağlar. Bu özellikler, robotların endüstriyel otomasyondan evdeki yardımcılara ve tehlikeli ortamlardaki araştırma ve kurtarma operasyonlarına kadar geniş bir kullanım alanında etkili olmasını mümkün kılar [8,9].

2.2. Robotların Sınıflandırılması

Robot teknolojileri, tarih boyunca çeşitli sınıflandırmalarla evrim geçirerek, endüstriyel ve mobil robotlar olarak iki ana kategoride toplanmıştır. Bu kategoriler, robotların işlevselliği, yapısal özellikleri, kullanım alanları ve etkileşim biçimleri açısından ayrıntılı bir şekilde incelenmiştir.[8]

- **Endüstriyel Robotlar:** Üretim ve imalat sektörlerinde yaygın olarak kullanılan endüstriyel robotlar, eklem sayılarına ve işlevsel özelliklerine göre sınıflandırılır. Bu robotlar genellikle sabit kurulumlarla, tekrar eden ve kesinlik gerektiren işlerde görev alır, üretim hatlarının hızını ve verimliliğini artırır.
- **Mobil Robotlar:** Daha dinamik ve çeşitli çevrelerde kullanılan mobil robotlar, işlevselliklerine, boyutlarına, ve uygulama alanlarına göre farklılık gösterir. Bu kategoride insansı robotlar, çoklu robot sistemleri, sürü robotları, mikro ve nano robotlar, biyo-ilhamlı robotlar ve işbirlikçi robotlar bulunur. Bu robotlar genellikle hizmet, keşif, sağlık hizmetleri ve ev otomasyonu gibi alanlarda etkin şekilde kullanılır.
- **Haptik Sistemler:** Hem endüstriyel hem de mobil robotlarda bulunan haptik sistemler, robotlarla insanlar arasındaki etkileşimi artıran teknolojilerdir. Bu sistemler, özellikle uzaktan kontrol edilen veya hassas operasyonlar gerektiren durumlarda, kullanıcının robot üzerinde daha iyi kontrol ve hassasiyet elde etmesini sağlar.

Bu çeşitlilik, robotların farklı endüstriyel ve sosyal ihtiyaçlara nasıl uyum sağladığını ve teknolojik ilerlemelerin robotik çözümleri nasıl şekillendirdiğini göstermektedir. Her bir sınıflandırma, robotların belirli bir problem setini çözme kapasitesini ve kullanıldıkları spesifik alanları vurgular, böylece teknolojinin insanların çalışma şekillerini nasıl dönüştürebileceğini ortaya koyar [7].

ISO 8373 standardına göre tanımlanan endüstriyel robotlar, günümüz sanayisinde kritik bir rol oynamaktadır. Bu tanım, robotları “üç veya daha fazla programlanabilir eksen olan, otomatik kontrollü, programlanabilir, çok amaçlı, bir yerde sabit duran veya tekerlekleri olan endüstriyel uygulamalarda kullanılan manipülatör” şeklinde ifade eder. Bu tanım, endüstriyel robotların esnekliğini ve geniş uygulama alanını da vurgular. Endüstriyel robotlar, imalat, kesme, taşıma, montaj gibi işlerde kullanılarak, sanayi üretimini hızlandırmış, işçilik maliyetlerini düşürmüş ve üretim miktarlarını artırmıştır. Ayrıca, insan hatası riskinin yüksek olduğu, insan hayatını tehlikeye atan ve yüksek dikkat ile hassasiyet gerektiren işlerde yüksek başarılı sonuçlar sağlamıştır. Bu robotlar sayesinde, tutarlı ve tekrarlanabilir kalitede ürünler elde edilirken, tehlikeli koşullarda insan güvenliği de maksimum seviyede korunmuştur. Endüstriyel robotlar, işlevlerine, boyutlarına, hassasiyetine, gücüne, kullandığı enerjiye, kontrol yöntemine, hareket yeteneğine ve eklem yapısına göre çeşitli şekillerde sınıflandırılmaktadır [11]. Bu sınıflandırma aşağıdaki gibi robot türlerini içerir:

- **Kartezyen Robotlar:** Üç doğrusal hareket eksenine sahip çalışan ve genellikle düzlemde hareket etme yeteneğine sahip robotlar.
- **Silindirik Robotlar:** Bir döner eksen etrafında silindirik bir çalışma alanı sağlayan robotlar.
- **Küresel Robotlar:** Küresel bir çalışma alanı yaratan ve genellikle bir döner eksen ile bir veya iki doğrusal eksen kombinasyonuna sahip robotlar.
- **Döner Mafsallı Robotlar:** En yaygın kullanılan endüstriyel robot türü olup, insan koluna benzer şekilde, bir veya daha fazla döner ekleme sahip robotlar.
- **Scara Robotlar:** Özellikle montaj işleri için tasarlanmış, yüksek hızlı ve yüksek hassasiyetli robotlar.

Bu çeşitlilik, endüstriyel robotların iş süreçlerine nasıl entegre edilebileceğini ve farklı ihtiyaçlara nasıl uyum sağlayabileceğini gösterir. Endüstriyel robotlar, sürekli gelişen teknolojilerle birlikte iş verimliliğini artırmanın yanı sıra, iş güvenliği ve kalite standartlarını da yükseltmektedir [7].



3. ENDÜSTRİYEL MOBİL ROBOTLARIN GELİŞİMİ

Endüstriyel mobil robotlar ilk olarak çizgi izleyen robotlar şeklinde 1970'lerin başlarında Stanford Üniversitesi'nde Shakey isimli mobil robot olarak tasarlanmıştır. Bunlar endüstride belirli noktalar arasında yük taşıma amacıyla kullanılmışlardır. 1980'lerden itibaren endüstriyel robotlar, bilgisayarlardaki ve görme sistemlerindeki ilerlemeler sayesinde çalışmalar ilerleyerek günümüzde yüksek teknolojinin en önemli araştırma alanlarından biri olmuştur. Mobil robotlarda gezinim, aracın hareketinin bir noktadan diğer bir noktaya başarılı bir şekilde ilerlemesi olarak tanımlanır. Gezinim sistemi mobil robotlar için en temel yapı olarak ele alınır çünkü hareketi sırasında sürekli olarak çalıştığı alanda ve etrafında bulunan nesnelere göre bulunduğu yeri bilmesi, gideceği hedefe sorunsuz bir şekilde ilerleyebilmesi için gerekmektedir [12].

Günümüzde ise endüstriyel mobil robotlarla ilgili literatürde birçok çalışma yapılmıştır. Bu tez çalışmasında çizgi izleyen robot tasarımı yapılmıştır bunun amacı fabrikalarda üretilen parçaların veya mamüllerin operasyonları arasında malzeme taşınmasını kolay kılmak ve otomasyon haline getirmek amacıyla belirlenen istasyonlar arasında insan gücü yerine robotların kullanılmasıdır. Buda işlem kolaylığı ve süre azaltıcılığı bakımında verim sağlayacağı açıktır. Bu tez çalışmasında tasarlanan robotun özelliği çizgileri algılayıp takip edebilen bir araç ve bu aracın üzerine monte edilmiş, kutuları belirli duraklardan alıp bırakabilen bir robot kol şeklindedir. Tasarım, Arduino mikrokontrolörü kullanarak mekatronik ve otomasyon prensiplerinin bir araya getirilmesiyle gerçekleştirilmiştir. Birden fazla çeşidi olan Arduino kart içerisinde Arduino Nano kart kullanılmış ve bu sayede devre kartı boyutunun küçük olması sağlanmıştır.

Harish vd. (2017) yaptıkları çalışmalarında "Arduino Kullanarak Al ve Yerleştir Robot Kol" başlıklı bir projeye ait bilgiler yer almaktadır. Bu projede, otomasyon alanında ve günlük işlerde kullanılan al ve yerleştir robot kollarının önemi vurgulanmaktadır. Projede, Robo-Arduino kullanılarak herhangi bir al ve yerleştir işlevi için tasarlanmış bir robot kolun nasıl uygulanabileceği anlatılmaktadır. Robot kol RF sinyali ile kontrol edilmekte ve dört Omni tekerlek üzerinde desteklenen şasi sayesinde robot kolun yer değiştirmesi sağlanmaktadır. Robot kolun iki serbestlik derecesi bulunmaktadır [13].

Shaibu vd. (2019) yaptıkları çalışmalarında beş serbestlik derecesinde (5-DoF) sahip bir Arduino kontrollü robot kolun geliştirilmesini kapsamaktadır. Robot kol, öğretim ve eğitim amaçları için tasarlanmıştır ve al ve yerleştir operasyonları gerçekleştirebilmektedir. Projede, robot kolun çeşitli eklemlerini hareket ettirmek için servo motorlar kullanılmış ve her bir eklem için gerekli tork hesaplamaları yapılarak uygun servo motor seçilmiştir. Robot kolun bileşenleri, hafifliği ve dayanıklılığı nedeniyle alüminyum malzemeden üretilmiştir. Kontrol için Arduino UNO R3 kartı ve Arduino IDE yazılımı tercih edilmiştir. Ek olarak, eklem hareketlerinin kontrolü için Bluetooth modülü ve bir Android cihazı kullanılmıştır [14].

Pawar vd. (2018) yaptıkları çalışmalarında "Robotic Arm Generation-1'in Tasarımı ve Geliştirilmesi Üzerine İnceleme" başlıklı bir çalışma sunulmaktadır. Çalışma, insan elinin hareketlerini taklit ederek bir nesneyi tutabilen Arduino kontrollü bir robot kolun tasarımını içermektedir. Robot kol, tehlikeli alanları analiz edebilen ve malzeme taşıyabilen bir robota entegre edilebilecek şekilde tasarlanmıştır. Bu çalışmanın amacı, yüksek hassasiyetli aktüatörler ve özel parça işleme maliyetlerinin yüksek olması nedeniyle pahalı olan robotik kontrollerin maliyetlerini düşürmek ve böylece robotik kontrol araştırmalarının daha hızlı ilerlemesini sağlamaktır [15].

Shanta vd.(2020) yaptıkları çalışmalarında bu robotun geliştirilmesinin ana amacı, depo ayarlarında seçilen nesnelerin alınması ve yerleştirilmesine yardımcı olarak çıktıyı artırmaktır. Robot, farklı renklere sahip ürünleri veya malları algılamak, bu öğeleri almak ve onları renge göre önceden belirlenen farklı yerlere yerleştirmek için tasarlanmıştır [16].

Aksu (2020) çalışmasında deprem, kaza analizi ve araştırma gibi insan hayatını tehlikeye atan durumlarda yardımcı olacak, kolay yönlendirilebilir ve birden fazla özelliğe sahip bir robotun üretimini kapsar. Bu robot, bu tür zorluklar karşısında insanlığa destek olmak amacıyla geliştirilmiştir [17].

Baby vd. (2017) yaptıkları çalışmalarında, Arduino kullanılarak tasarlanmış ve inşa edilmiş bir al ve yerleştir robotik kolun uygulanması ele alınmıştır. Bu robotik kol, kullanıcı komutlarına göre nesneleri bir kaynaktan hedefe güvenli bir şekilde taşıyabilen, yumuşak tutuşlu bir gripper kullanarak nesnelere ekstra baskı uygulamadan çalışan bir sistemdir. Robot, Android tabanlı akıllı telefonlar aracılığıyla Bluetooth üzerinden kontrol edilebilir. Kullanıcının komutlarına göre robot, dört motordan oluşan bir mikro

denetleyici arayüzü ile hareket eder; iki motor araç hareketi için ve diğer iki motor kol ve gripper hareketleri için kullanılır. Robot kontrolü için Blue Control uygulaması kullanılmaktadır [18].

Myint vd. (2016) yaptıkları çalışmalarında, gelişmekte olan ülkelerdeki endüstrilerin artmasıyla birlikte iş gücüne olan ihtiyacı azaltmak ve endüstriyel üretim kapasitesini artırmak amacıyla ileri robot kollarının önemini vurgulamaktadır. "Nesne Sıralama Sistemi İçin AI ve Yerleştir Robot Kolunun Konum Kontrol Yöntemi" başlıklı bu çalışma, nesne sıralama sistemleri için manuel kontrolü ortadan kaldırmayı amaçlamaktadır. Robot kol tasarımı, iki eklem, üç bağlantı ve servo motorları kullanarak gerçekleştirilmiştir. Bu araştırmada, robot kolunun konum kontrolü, kinematik kontrol yöntemleri kullanılarak tasarlanmıştır. İleri ve ters kinematik yöntemler olmak üzere iki tür kinematik kontrol yöntemi bulunmaktadır. İleri kinematik yöntemde, giriş parametreleri robot kolunun eklem açıları ve bağlantı uzunluklarıdır ve çıktı, aletin veya gripperin XYZ koordinatındaki konumudur. Ters kinematikte, giriş parametreleri gripperin XYZ koordinatındaki konumu ve robot kolunun bağlantı uzunluklarıdır ve çıktı parametreleri eklem açılarıdır. Bu çalışmada, sıralama sistemi için Metal dedektörü, metal veya metal olmayanı tespit etmek için kullanılmıştır. Robot kolunun konum kontrolü tam olarak test edilmiş ve sonuç daha kesin olarak elde edilmiştir [19].

Abdulkareem vd. (2019) yaptıkları çalışmalarında günlük hayatta karşılaşılan monoton ve tekrarlayan görevleri etkin ve verimli bir şekilde yerine getirebilen, uzaktan kontrol edilebilen düşük maliyetli bir "AI ve Yerleştir Robotu"nun tasarımı ve uygulanmasını kapsar. Robot, iki Arduino mikrodenetleyici kullanılarak geliştirilmiştir; biri kol diğeryse tabanı kontrol etmek üzere master/slave konfigürasyonunda bağlanmıştır. Robot kolunun hareketleri servo motorlar kullanılarak sağlanırken, taban hareketliliği için dc dişli motorlar ve paletli tekerlekler tercih edilmiştir. Robotun uygulamaları arasında depolama, fabrika hatlarında görev yapma ve engelliler için kişisel yardımcı olarak kullanım gibi çeşitli alanlar bulunmaktadır [20].

Rohman vd. (2021) yaptıkları çalışmalarında robot kol tabanlı aktüatörlerin eklemlerinin açı konumunun Diskret PID Algoritması kullanılarak nasıl modelleneceği ve kontrol edileceği üzerine odaklanır. Çalışma, DC motorunun robotik ve kontrol alanlarında sıklıkla kullanıldığını ve yüklerden kaynaklanan hatalar nedeniyle DC motorunun bazen doğruluğunun azalabildiğini belirtir. PID (Oransal İntegral Türev) kontrol sistemi, K_p , K_i

ve Kd kontrol değişkenlerine dayanarak beklenen set noktasına ulaşmak için hesaplamalar yapar. Çalışmanın amacı, PID kontrolörü kullanarak DC motor pozisyon kontrol cihazının kontrol edilebilmesini sağlamaktır. Bu araştırmada kontrol edilen değişken, dereceler cinsinden pozisyon kontrolüdür. Arduino Mega kontrolcüsü ve motor sürücüsü kullanılarak DC motora bir enkoder sensörü geri bildirimi verilmiş ve Arduino IDE yazılımı kullanılmıştır [21].

Surati vd. (2021) yaptıkları bu inceleme makalesinde, manipülatörler üzerine yapılmış çeşitli başarılı araştırma makalelerini gözden geçirerek robot kollarının çeşitli yönleri vurgulanmıştır. Günümüzde robot kolları, insan hatalarını en aza indirmek ve operasyonlarda verimliliği ve hassasiyeti artırmak için endüstrilerde kullanılmaktadır. Robot kolunun endüstrilere tanıtılmasının en önemli avantajlarından biri, yüksek sıcaklık ve basınç gibi insanların çalışması için riskli olan kritik koşullarda çalışabilmesidir. Manipülatör, Esnek Otomasyonun bir parçası olarak kabul edildiğinden, kolayca güncellenip modifiye edilebilir. Bu makalede, çeşitli kontrolörlerin kullanımı ve manipülatörün serbestlik derecelerini belirlemek için farklı yazarlar tarafından kullanılan farklı metodolojileri gözlemlemek üzere deneysel olarak doğrulanmış birçok araştırma makalesine atıfta bulunulmuştur [22].

Müftü (2023) çalışmasında robot kollarının endüstriyel öneminin artmasıyla birlikte, robot kol kontrolünün hassasiyeti de büyük önem taşımaktadır. Bu tezde, Arduino ve Matlab Destek Paketi kullanılarak iki uzuvlu bir robot kolunun kontrol optimizasyonu gerçekleştirilmiştir. DC motor için transfer fonksiyonu kullanılarak, alan etkili enkoder sayesinde geri bildirim alınmış ve her bir motor için ayrı konum kontrolü yapılmıştır. Kontrolcü parametreleri, parçacık sürü, yapay arı koloni ve kaos oyun optimizasyon algoritmaları ile belirlenmiştir. Ayrıca, kare ve daire yörünge takibi de bu robot kol ile başarıyla gerçekleştirilmiştir. Çalışmanın sonuçları, her bir yörünge için yapılan üç adet meta sezgisel optimizasyon algoritması sonucunda elde edilen kontrolcü parametreleri üzerinden değerlendirilmiştir [6].

Sarıaltın (2017) projesinde tasarladığı robot kolun amacı, bir nesneyi belirli bir noktadan alıp, istenilen başka bir noktaya taşıyabilmektir. Bu işlevsellik, robot kollarının endüstriyel uygulamalarda ne kadar önemli olduğunu ve otomasyon sistemlerinin geliştirilmesindeki rolünü vurgulamaktadır. Çalışmanın literatür incelemesi kısmında, robotik kol sistemleri üzerine yapılan çeşitli çalışmalara atıfta bulunulmuş, bu

alışmaların zetleri sunulmuştur. Bu atıflar, robot kol sistemlerinin tasarımı, uygulaması ve kontrol zerine geniř bir perspektif sunmakta ve robot teknolojisinin geliřimine dair nemli bilgiler iermektedir [7].

Yildiz vd. (2020) yaptıkları bu alışmada izgi izleyen robotların verimli bir řekilde bir yol izlemesini saėlayacak saėlam bir kayan mod(kipli) kontrol (SMC) tasarımı ve uygulanmasını ierir. Robot, dz veya kıvrımlı bir yolu kızıltesi sensrlerle algılayarak, bu sensrlerden gelen srekli akışı saėa veya sola dnerek yolu izleyecek řekilde motorları harekete geirir. Kontrol stratejisi, karmařık yolları doėru bir řekilde takip edebilmek ve hızlı, kararlı ve doėru bir izgi izleyen robot elde etmek iin hayati neme sahiptir. Bu nedenle, karřılařtırma amacıyla, robota geleneksel orantılı-integral-trevli (PID) kontrol de uygulanmıştır. alışmanın ana amacı, kayan mod kontrolnn yol izleme sırasında performansını incelemektir [23].

Syafalni vd. (2022) yaptıkları bu alışmada, Q-Learning tabanlı en kısa yol problemi zmnn test edilmesi iin tasarlanmış bir izgi izleyen robot ve engeller ieren bir test ortamının tasarımını detaylandırır. alışma, izgi izleyen robotun engellerle dolu bir arenada hareket ederken Q-Learning sonularını gsterme yntemine odaklanır. nerilen zm olarak, robotun DC motorları hareket ettiriciler olarak kullanır ve kontrol yntemi iin referans olarak kızıltesi sensr ve pusula sensr okumalarını tamamlar. Gerek arena kořulları ve robotun řekli, robotun hareketini engelleyebilir ve belirsizliklere neden olabilir. Bu durumda, robotun en kısa yol durumu iin test alanında doėru řekilde alışabilmesi iin daha fazla tasarım gereklidir [24].

Tayal vd. (2020) yaptıkları makalede, izgi izleyen robotların tasarımı ve donanım uygulamaları zerine odaklanılmaktadır. izgi izleyen robotlar, siyah bir izgiyi beyaz zemin zerinde ya da tersini takip edebilen, nceden belirlenmiş yolları izleyebilen robotlardır. Bu robotlar, endstriyel ve ev ii uygulamalarda malzeme taşıma, zemin temizleme, teslimat hizmetleri ve ulařım gibi eřitli iřler iin kullanılmaktadır [25].

Amorim vd. (2022) yaptıkları alışmada, siyah bir mat zerinde beyaz izgiyi takip eden bir izgi izleyen robotun ana amacını, mmkn olduėunca hızlı bir řekilde parkuru tamamlamak olarak belirtir. Robot, erevenin n tarafına monte edilen sensrler aracılığıyla izginin konumunu alır ve bir mikrokontrolr, ynlendirme mekanizmasına gerekli ıktıyı gndererek takip hareketini telafi eder. Robotik yarıřmaları, robotun mmkn olan en kısa srede yarıřı tamamlamasını gerektirir [26].

Kader vd. (2018) yaptıkları projede geliştirilen robot, ofis içerisinde belirlenen yollarda çizgiyi takip ederek dosya, çay, atıştırmalıklar gibi nesneleri ofis çalışanlarına otomatik olarak teslim edebilir. Kullanıcılar, robotu belirli bir masaya çağırmak için bir düğmeye basabilirler. Robot, PID (Oransal-İntegral-Türev) algoritması kullanarak yolu takip eder ve bir engelle karşılaştığında uyarı verir. Projenin temel bileşenleri arasında mikrodenetleyici, IR sensör modülü, RF Tx-Rx modülü, ultrasonik sensör, buzzer ve DC motorlar yer alır [27].

Andrian vd. (2020) yaptıkları bu araştırmanın amacı, robot kol kullanarak endüstriyel otomasyon süreçlerinde insan emeğini azaltacak teknolojik çözümler geliştirmektir. Robot kol, önceden belirlenmiş yerlere otomatik olarak malzeme taşıyabilme yeteneğine sahiptir [28].

Putri vd. (2022) yaptıkları çalışmada, nesneleri belirlenen yere taşımak için tasarlanmış bir prototip robot kolun geliştirilmesini içermektedir. Robot kol, Arduino Mega 2560 ve Pixy2 CMUCam5 kullanılarak tasarlanmıştır ve renkli nesneleri algılayıp onları önceden belirlenmiş yere taşıyabilme yeteneğine sahiptir. Bu robotik kol prototipinin geliştirilmesinde Arduino Mega 2560, Pixy2 CMUCam5, Servo MG90S, Servo SG90, jumper kablo ve bir dizüstü bilgisayar kullanılmıştır. Yazılım olarak Arduino IDE ve PixyMon kullanılmıştır. Bu çalışma, robotik kolun hareketini her bir eklemden bulunan 4 servo motor aracılığıyla sağlar [29].

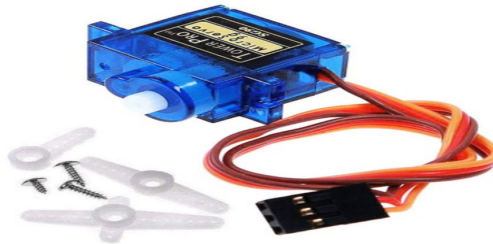
4. ENDÜSTRİYEL MOBİL ROBOTUN TASARIMI

Bu tezde tasarlanması amaçlanan endüstriyel mobil robot için kullanılan servo motor, kızılötesi sensör, sarhoş tekerlek, sürücü kartı, bağlantı durumları ve ultrasonik sensör, DC motor, LDR ve sürücü kartı, arduino nano kart ve PID algoritması hakkındaki bilgiler bu ana başlıkta alt başlıklar halinde sunulmuştur.

4.1.Servo Motor

Servo motorlar, mekanizmaların çalışma hatalarını tespit edip, geri bildirim mekanizmasıyla düzeltmeyi sağlayan cihazlardır. Bu motorlar içlerinde alternatif akım (AC), doğru akım (DC) veya step motorları barındırabilirler ve aynı zamanda sürücü ile kontrol devrelerine sahiptirler. Servo motorlar, verilen komutları hassas bir şekilde yerine getirebilir ve çok düşük veya çok yüksek hızlarda dahi istikrarlı çalışma performansı gösterebilirler. Bu motorlar, küçük boyutlarına rağmen yüksek tork değerleri sunabilir [30,31].

Kontrol sistemlerinde, hızlı işlemler, çok eksenli hareketler, durum denetimi gibi uygulamalarda tercih edilirler. Son kontrol elemanı olarak kabul edilen servo motorlar, yüksek hassasiyetleriyle öne çıkar ve genellikle elektronik ya da programlanabilir devrelerle birlikte kullanılırlar. AC ve DC olmak üzere iki temel tipe ayrılan servo motorlar, sırasıyla fırçasız ve fırçalı yapıdadırlar. Genel olarak, güç için kırmızı, topraklama için siyah ve kontrol (data) için sarı olmak üzere üç kablo ile donatılmışlardır. Projede kullanılan örnek bir servo motor ve kablo bağlantıları Şekil 4.1. ve Şekil 4.2’de gösterilmiştir.



Şekil 4.1. DC servo motor



Şekil 4.2. Kablo bağlantıları gösterimi

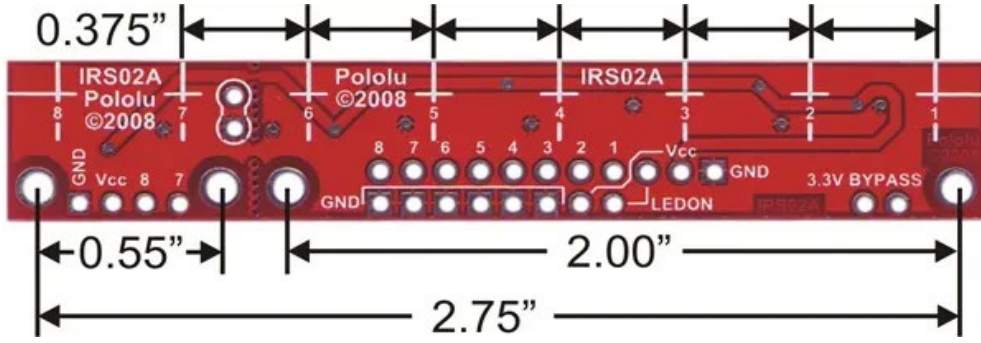
Bu motorun öne çıkan özellikleri arasında çok yönlü kullanımı, darbe genişlik kontrolünün $10\ \mu s$ olması, $VP - P$: $3 - 5\ V$ kare dalga ile uyumluluğu ve $4,8-6V$ arasında bir çalışma gerilimi bulunmaktadır. Bu servo motor, düşük çalışma geriliminde bile, yüksüz durumda $0,12\ s/60^\circ$ dönüş hızına ve $1,2-1,6\ kg/cm$ torka sahiptir. Servo motorlar, gelen sinyale göre kontrol edilirler ve bu kontrol için veri hattı üzerinden gönderilen darbe genişlik modülasyonu (PWM) sinyalleri kullanılır ve genellikle $10-20\ ms$ aralığında ve $0,5-1,5\ ms$ sürelerinde PWM sinyalleriyle yönetilir. Motor milinin pozisyonu, sinyalin lojik1’de kaldığı süreye (tk) bağlı olarak değişir [30,31]:

- $tk = 0,5\ ms$ olduğunda, motor milinin en uç konuma döndüğü,
- $tk = 0,5- 1\ ms$ arasında olduğunda, milin orta konumda olduğu,
- $tk = 1- 1,5\ ms$ arasında olduğunda, milin sağa döndüğü,
- $tk = 10-20\ ms$ olduğunda (aynı sinyal tekrarlandığında) milin önceki konumunu koruduğu belirlenir.

4.2. Kızılötesi Sensör

Pololu QTR-8RC Çizgi Algılayıcı, çizgi takip eden robotlar için tasarlanmış bir yansıma türü sensör dizisidir. QTR-8RC, $9.525\ mm$ aralıklarla düzenli olarak yerleştirilmiş sekiz adet IR yayan ve algılayan (fototransistör) çifte sahiptir. Gerektiğinde algılayıcı iki bölüme ayrılabilir. QTR algılayıcı, düşük IR yansıtma özelliğine sahip koyu renk bir yüzey ile yüksek IR yansıtma özelliğine sahip açık renk bir yüzey arasındaki farkları saptayabilen sekiz kızılötesi yayan/alıcıya sahiptir [32-34].

Şekil 4.3’te Pololu QTR-8RC Çizgi Algılayıcının boyutları belirtilmiştir. QTR'nin açılımı: Q "Charge" (Şarj), T "Transfer", R ise "Resistance" (Direnci) ifade eder.



Şekil 4.3. Pololu QTR-8RC çizgi algılayıcının boyutları

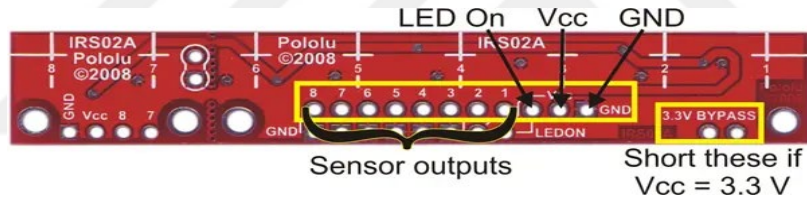
Her algılayıcı birimi, QRE1113GR fototransistör ile seri bağlantılı 10nF'lik bir kapasitöre sahiptir. Kapasitörün bir ucu VCC'ye, diğer ucu ise GND'ye bağlanmıştır. Kapasitör ve fototransistör arasındaki bağlantı noktası, 220 Ω 'lık bir direnç üzerinden bir mikrokontrolcünün GPIO'suna bağlanır ve bu bağlantı OUT pinini oluşturur [32-34].

Fototransistör, bir akımın iki terminali arasında geçişini düzenleyen ve üçüncü terminal ile bu akımın miktarını ayarlayan bir kapak görevi gören bir transistör türüdür. Transistör tipine göre, akım akışı voltaj, akım veya fototransistörde olduğu gibi ışık ile kontrol edilebilir ve bu, üçüncü terminalin işlevini görür. QTR algılayıcısı durumunda, ışık Kızılötesidir (IR). Pololu QTR, altında bulunan yüzeyin yansıtma özelliğini belirlemek için QRE1113GR kızılötesi (IR) yansıtıcı sensörü kullanır. Algılayıcı koyu bir zemin üzerindeyken, yansıtıcılık değeri çok düşük olur; QTR açık renkli bir zemin üzerindeyken ise, yansıtıcılık değeri yüksektir ve bu durum sensörden alınan okumanın değişmesine neden olur [32-34].

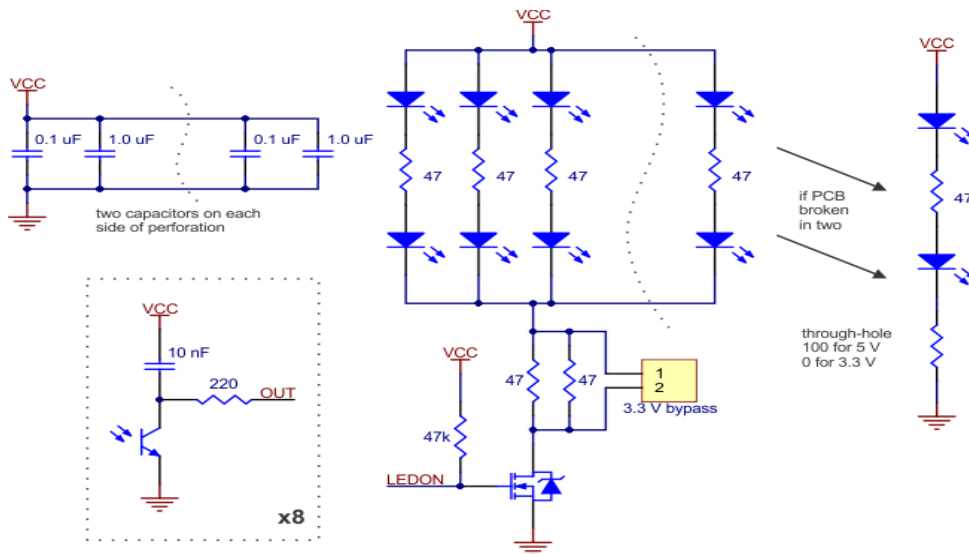
Pololu QTR-8RC sensörün pin çıkışları Şekil 4.4'te ve hat sensör şeması Şekil 4.5'de gösterilmiştir. QTR algılayıcı, VCC pinine 5V verilerek etkinleştirilir. LEDON pini YÜKSEK (5V) konumundayken, QRE1113GR'nin LED tarafına 47 Ω 'luk bir direnç üzerinden akım sağlanır. Yüzeyden yansıyan IR ışık, QRE1113GR'nin fototransistör tarafındaki akım akış kabiliyetinde değişiklik yapar. Böylece transistör, etkili bir şekilde bir IR kontrolü altındaki direnç gibi işler. QTR aktif olduğunda ve OUT pini YÜKSEK durumda tutulduğunda, kapasitörün her iki ucu da VCC'ye kısa devre yapılır ve kapasitör, 220 Ω 'luk direnç üzerinden 2.2 μ s'luk bir zaman sabiti ile deşarj olur. GPIO'nun yaklaşık 10 μ s süreyle YÜKSEK tutulması, tamamen deşarj olduğundan emin olunmasını sağlar. Kapasitör deşarj olduktan sonra, OUT pinine bağlı mikrokontrolcünün GPIO'su Yüksek

Empedans Girişi olarak ayarlanır. Pin giriş moduna geçtiği anda kapasitör yeniden şarj olmaya başlar. Mikrokontrolcü içerisindeki zamanlayıcı, kapasitörün belirli bir düzeye kadar şarj olmasını gerektiren süreyi hesaplamak için kullanılır. Kapasitördeki şarj miktarı arttıkça, üzerindeki voltaj düşüşü de artar ve OUT pinindeki voltaj düşer. Tam şarjda, OUT pinindeki voltaj neredeyse sıfıra (DÜŞÜK) iner. Bu süreç, QRE1113GR'nin fototransistör tarafından geçen akım tarafından yönetilir. QTR, koyu bir zemin üzerindeyken, fototransistörden geçen akım düşük olduğundan, kapasitörün şarj olması uzun zaman alır. Açık renkli bir zemin üzerindeyken ise, fototransistörden geçen akım yüksek olduğundan, kapasitörün şarj süresi kısa olur [32-34].

Sonuç olarak Pololu QTR 8RC sensörlerin modüler tasarımı, robot tasarımcılarına büyük bir esneklik sunması sebebiyle bu projede tercih edilmiştir. Son iki sensörün ayrılabilir olması, sensörlerin robot üzerinde çeşitli konfigürasyonlarda kullanılabilmesine olanak tanır. Bu, özellikle farklı türdeki yarış pistlerine veya görev alanlarına uyum sağlamak isteyen geliştiriciler için büyük bir avantaj sağlamaktadır [32-34].



Şekil 4.4. Pin çıkışları



Şekil 4.5. Hat sensör şeması

4.3.Sarhoş Tekerlek

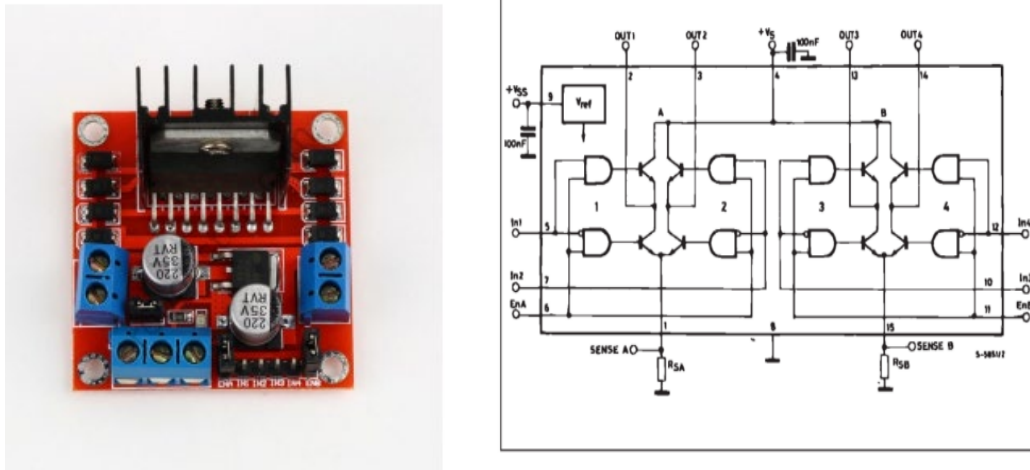
Sarhoş tekerlekler, çizgi takip eden robotlar gibi hassas manevra yapma yeteneği gerektiren platformlarda yaygın olarak kullanılır. Bu tekerleklerin başlıca özelliği, araç yön değiştirdiğinde ekstra sürtünmeye maruz kalmadan rahatça kayabilmesidir. Bu özellik, robotun motorlarından gelen gücün daha etkin yönlendirilmesini sağlayarak, aracın istenilen yöne doğru sorunsuz bir şekilde ilerlemesine imkân sağlar. Sarhoş tekerlekler genellikle robotun ön kısmına monte edilir ve yön değişiklikleri sırasında ana yükü taşırlar. Bu tekerleklerin kullanılması, robotun dönüş yaparken gereksiz sürtünmeye maruz kalmamasını sağlar ve bu sayede daha çevik ve hassas manevralar yapmasına olanak verir. Bu tekerlekler, hafiflik ve maliyet avantajları nedeniyle plastikten; sürtünmeyi daha fazla azaltma, daha fazla dayanıklılık ve hassasiyet sağlama gibi nedenlerle de metalden üretilebilir. Performansı iyileştirmek isteyen tasarımcılar genellikle metal sarhoş tekerlekleri tercih ederler. Bu sebeplerden dolayı projemizde metal sarhoş tekerlek kullanılmıştır. Sarhoş tekerleklerin kullanımı, arka tekerlekler arasında oluşturulan hız farkıyla aracın yönlendirilmesini sağlar. Bu düzenek “diferansiyel dönüş” olarak adlandırılır ve aracın daha etkili bir şekilde manevra yapmasını mümkün kılar. Eğer pasif tekerlekler kullanılsaydı, aracın dönüş yapması zorlaşacak ve daha fazla motor gücüne ihtiyaç duyulacaktı. Ancak bu tekerlekler sayesinde, motor gücü doğrudan yönlendirilerek aracın istenilen yolda ilerlemesi sağlanır. Bu özellikle dar alanlarda manevra yapılmasını kolaylaştırır [35]. Sarhoş tekerlek parçaları Şekil 4.6’da gösterilmiştir.



Şekil 4.6. Sarhoş tekerlek parçaları

4.4.Sürücü Kartı

DC motorları Arduino ile çalışır hale getirmek sandığımız kadar basit değildir. Bunun için mutlaka harici bir güç kaynağına (6-12V aralığında) ve uygun bir motor sürücü modülüne ihtiyacımız vardır. Bu ek donanımların gerekliliği, DC motorların yüksek güç tüketiminden kaynaklanmaktadır. Arduino'nun pin çıkışları, bir DC motorun ihtiyaç duyduğu ortalama akımın yalnızca küçük bir kısmını karşılayabilir. Motor sürücü modülleri ise, Arduino'dan gelen sinyalleri güçlendirerek motorlara aktarır ve böylece motorların Arduino pinleri aracılığıyla kontrol edilmesini sağlar. L298N, eski ve yaygın bir motor sürücü entegresidir. Arduino dünyasında genellikle, soğutucu, koruma diyotları, motor, güç ve kontrol bağlantıları ile birlikte bir modül formunda kullanılır. Bu modülde, harici güç kaynağından alınan güçle Arduino'yu beslemek için bir 5 Volt gerilim regülatörü de bulunmaktadır. L298N, aynı anda iki DC motoru kontrol edebilecek şekilde tasarlanmıştır ve bu yüzden modül üzerinde çoğu bileşen ikişer adet bulunur. Motorların bağlantıları modülün iki yanında yer alırken, motor başına üç giriş pini olmak üzere toplam altı giriş pini ön bölümde yer alır [36-39]. Şekil 4.7'de L298N motor sürücü kartı ve blog diyagramı verilmiştir.



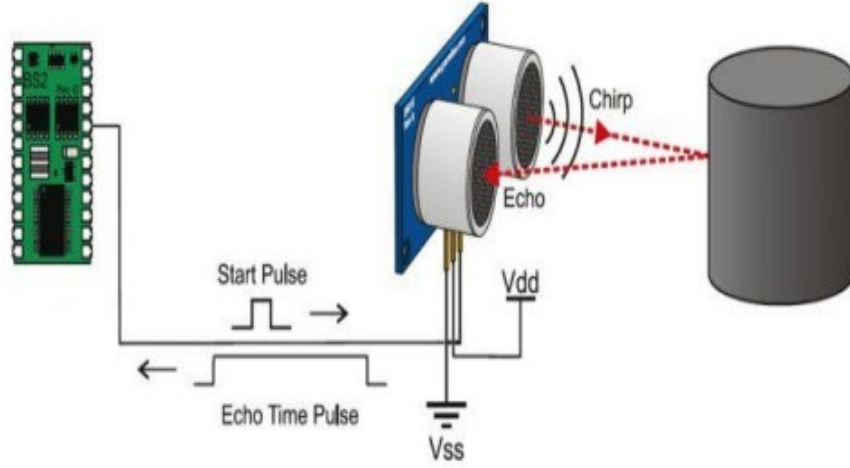
Şekil 4.7. L298N motor sürücü kartı ve blog diyagramı

4.5. Bağlantılar ve Ultrasonik Sensör

DC motor bağlantı pinleri, modülün üzerinde A motoru için OUT1-OUT2 ve B motoru için OUT3-OUT4 olarak işaretlenmiştir. DC motorların iki ucu, bu çıkışlara bağlanır. Motorları bağlarken yönlendirme çok kritik değildir; motorun dönüş yönünü kod

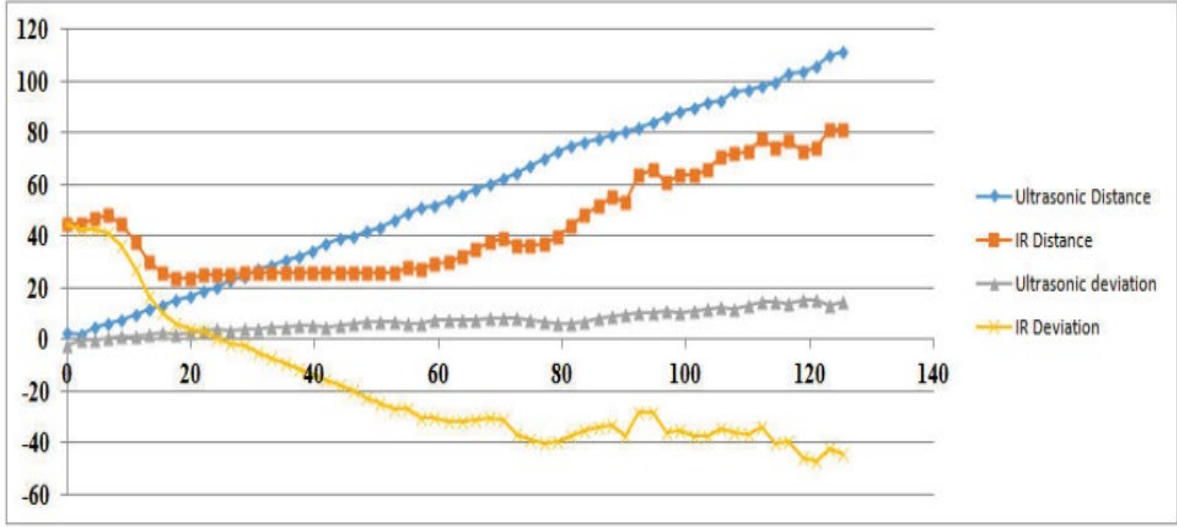
üzerinden ayarlayabiliriz. Yön değiştirmek gerektiğinde, bağlantıları ters çevirebiliriz. Her motor için üç adet giriş pini bulunmaktadır: A motoru için ENA, IN1 ve IN2. ENA (ENABLE MOTOR A), motorun hız ayarını yapmak için kullanılırken, IN1 ve IN2 motorun çalışıp çalışmamasını ve hangi yönde döneceğini kontrol eder. B motoru için bağlantılar ise ENB, IN3 ve IN4 şeklindedir. Bu da modülün iki bağımsız motor sürücü gibi çalışmasını sağlar. ENA ve ENB pinleri yanında bulunan ve jumper ile bağlanmış 5V bağlantıları, motor hız ayarı gerekmeyen durumlarda motorların sürekli tam güçte çalışması için kullanılır. Bu durumda, ENA ve ENB pinlerini Arduino'ya bağlamamıza gerek kalmaz. Ancak bu durumda hız kontrolünden mahrum kalırız. Bu nedenle, kontrolümüzde olsun diye bu jumper'ları çıkarıp pinleri Arduino'ya bağlarız. Motorlar yüksek akıma ihtiyaç duyduğu için, bir harici güç kaynağı gerekir. Bu, seri bağlı kalem piller, 9 Volt pil, Li-Po pil veya kuru bir akü olabilir. Güç kaynağı, L298N modülünün 12V (+) ve GND (-) bağlantılarına bağlanır. Arduino'nun da çalışması için güç gerekir. L298N modülü üzerinde bulunan küçük 5 Volt gerilim regülatörü ile Arduino'yu direkt olarak modül üzerinden çalıştırabiliriz. Ancak dikkat edilmesi gereken, modüle 12 Volt'tan fazla voltaj verilmemesi ve modülle Arduino'nun GND bağlantılarının ortak olması gerektiğidir. Bu, Arduino'nun modülden aldığı HIGH ve LOW sinyallerinin doğru olarak tanınmasını sağlar [36-39].

Ultrasonik sensörler, belirli bir mesafedeki nesneleri tespit etmek ve bu nesnelerin konumlarını belirlemek için yaygın olarak kullanılan cihazlardır. Bu sensörlerin seçimi, ölçüm yapılacak maksimum mesafe ve sensörün ölçüm açısına bağlı olarak değişir. Özellikle bu çalışmada, yüksek verimlilik-maliyet oranı nedeniyle HC-SR04 ultrasonik sensörü tercih edilmiştir. HC-SR04 ultrasonik sensörü, 40 kHz frekansta ultrasonik dalgalar yayarak çalışır. Bu dalgalar, çevredeki nesnelere çarparak yansır ve sensörün başka bir alıcısı tarafından bu yankılanan dalgalar algılanır. Algılanan yankı süresi, nesnenin sensöre olan uzaklığını hesaplamak için kullanılır. Bu ölçüm, ultrasonik dalgaların nesneye çarpıp geri dönmesi esasına dayanır ve bu sensörlerin menzili genellikle 2 cm ile 4 metre arasında değişir. Sensörün çalışma prensibi Şekil 4.8'de gösterilmiştir.



Şekil 4.8. Sensörün çalışma prensibi

HC-SR04 ultrasonik sensör, 5 V DC ile çalışır ve Arduino gibi mikrodeneleyicilere dört dijital pin aracılığıyla bağlanır. Bu pinler arasında, ultrasonik sinyallerin gönderilmesi ve alınması için gerekli bağlantılar bulunur. Sensör, bir ultrasonik verici, bir alıcı ve bir izleme devresinden oluşur. Ultrasonik sensörlerin yüksek doğrulukta mesafe ölçme kabiliyeti, onları robotik (çizgi izleyen robotlar, denge robotları gibi), otomasyon ve güvenlik sistemleri gibi birçok farklı alanda ideal kılar. Bu sensörler, yankı süresi üzerinden hesaplanan mesafeleri kullanarak, hareketli robot projelerinde çevresel engellerden kaçınma ve nesne yerleştirme gibi karmaşık görevleri yerine getirebilir. Bu bilgiler ışığında, HC-SR04 ultrasonik sensörün kullanımı, projelerde yüksek doğrulukta ve düşük maliyetli bir çözüm sunarak, teknolojik uygulamaların geniş bir yelpazesinde değer kazanmaktadır. Bu sensörler, basit yapıları ve kolay programlanabilirlikleri ile mühendisler ve hobi kullanıcıları tarafından sıkça tercih edilen bileşenler arasındadır. Ultrasonik sensörlerden alınan veriler plastik engellerde daha kararlı ve doğrusaldır. Bunun nedeni, plastik yüzeyin düz ve sert olmasıdır, bu da üzerine düşen ses dalgalarının çoğunu yansıtır. Kızılötesi sensör, plastik diğer engel türlerine kıyasla şeffaf olduğu ve üzerine düşen ışığın çoğunu ilettiği için gerçek değerden büyük bir sapma gösterir. Ayrıca, çift kırılma da sensör okumalarını etkiler [40-42]. Plastik engele karşı sensör verileri Şekil 4.9’ da verilmiştir.



Şekil 4.9. Plastik engelle karşı sensör verileri

4.6. DC Motor

DC motorlar, robotik projelerde en sık kullanılan motor türlerinden biridir. Bunlar, maliyet açısından uygun, küçük boyutlu ve verimli olmaları nedeniyle tercih edilir. Ayrıca, çeşitli boyut, şekil ve güç seçenekleriyle gelmeleri, DC motorların kullanımını daha da yaygınlaştırır.

DC motorlar, robotlarda veya farklı sistemlerde doğrudan veya dişli kutularıyla (redüktörlü veya redüktörsüz) kullanılabilir. DC motorların robotik uygulamalardaki temel özellikleri aşağıda detaylandırılmıştır:

Yön: DC motorlara elektrik bağlandığında, motorun dönüş yönü akımın yönüne bağlı olarak değişir. Akımın yönünün tersine çevrilmesi, motorun dönüş yönünü de tersine çevirir.

Hız: Motor hızı, RPM (dakikadaki devir sayısı) cinsinden ölçülür ve hız, uygulanan voltaja ve motorun üzerindeki yüke bağlı olarak değişir. Motorun hızı, yüksüzken veya sabit yükte çalıştığında voltajla doğru orantılıdır. Yüke bağlı olarak hız, yük arttıkça azalan bir eğilim gösterir.

Voltaj: DC motorlar, genellikle 1,5 V ile 48 V arasında değişen voltajlarla çalışabilir. Her motorun belirlenen voltaj değeri, o motorun en stabil çalıştığı, hız, güç ve akım

değerlerine ulaştığı voltajı ifade eder. Robotik uygulamalarda, bu voltaj değerleri, motorun alabileceği maksimum çalışma voltajını belirler.

Akım: Motor, belirlenen voltajda çalıştırıldığında çektiği akım, yüke bağlı olarak değişir. Yük arttıkça, motorun çektiği akım da artar. Motor, maksimum akım sınırlarını aşacak şekilde yüklenirse, kısa devre olabilir ve bu durum motorun zarar görmesine neden olabilir.

Güç: Motorun gücü, akım ile voltajın çarpımıyla hesaplanır. Ancak, robot projeleri ve mekanik sistemlerde motorun ürettiği güç, çoğunlukla tork cinsinden değerlendirilir.

Tork: Tork, motorun dönme kuvvetidir ve motorun mekanik ve elektriksel özelliklerine bağlı olarak değişir. Tork, motor şaftının yarıçapı ile doğrudan ilişkilidir. Motorun torku, dişli kutuları kullanılarak ayarlanabilir. Bu kutular, motorun hızını düşürüp, gücünü artırarak çalışma verimliliğini artırır. Örneğin, motor şaftına göre on kat daha büyük yarıçapa sahip bir dişli, motor hızını on kat düşürürken, gücünü on kat artırır.

Robotikte, motorların özelliklerini ihtiyaç duyulan işe uygun hale getirmek için çeşitli redüksiyon oranlarındaki dişli kutuları sıklıkla kullanılır. Bir motorun torkunu ve kullanılan dişli kutusunun redüksiyon oranını bilmek, sistemin son çıkış gücünü belirlemede kritik öneme sahiptir. Tez çalışması kapsamında, Plastik redüktörlü 6 volt ve 250 RPM özelliklerine sahip DC motorlar kullanılmıştır. Projede kullanılan motorun görseli Şekil 4.10'de gösterilmiştir ve özellikleri görselin altında belirtilmiştir [43,44].



Şekil 4.10. DC motor

6 V 250 RPM Motor Özellikleri:

- Redüksiyon Oranı: 1:48
- Hız: 250 Rpm(@6V)
- Akım: 95mA (maks. 160mA)

- Ağırlık: 29gr
- Çalışma Voltajı: 3-12V

Basitliği ve değişken hız kontrolünün uygulanması kolay olduğundan, DC motorlar endüstriyel uygulamalarda yaygın olarak kullanılmaktadır. Her fiziksel sistem, giriş ve çıkış arasındaki ilişkiyi veren transfer fonksiyonu formunda ifade edilebildiğinden, DC motorlar da dört denklem kullanılarak transfer fonksiyonu yapısında modellenir. Ayrıca, durum değişkenleri, durum değişkenlerinin türevleri, girişler ve çıkışlar arasındaki ilişkiyi veren durum uzayı denklemleri kullanılarak da modellenirler. DC motorun elektriksel kısmına ait denklem Eş. 4.1’de verilmiştir [43-47] :

$$v_a = R_a i_a + L_a \frac{di_a}{dt} + e \quad (4.1)$$

ile verilir. Denklem (1) düzenlenirse

$$L_a \frac{di_a}{dt} + R_a i_a = V_a - e$$

elde edilir. Elde edilen son bağıntının Laplace dönüşümü alınır

$$(R_a + L_a(s))i_a = V_a - e$$

olur. Dolayısıyla, DC motorun elektrik kısmına ait transfer fonksiyonu aşağıdaki gibi olur.

$$\frac{i_a}{V_a - e} = \frac{1}{L_a s + R_a} \quad (4.2)$$

Mekanik kısma ait transfer fonksiyonu türetmek için Newton denklemlerinden faydalanılır:

$$J\omega + f\omega = t_{em} - t_L$$

Burada $t_L = t_d + f\omega_m$ yukarıdaki denklemde yerine yazılıp Laplace dönüşümü alınır aşağıdaki bağıntı bulunur.

$$\frac{\omega}{t_{em} - t_d} = \frac{1}{Js + f} \quad (4.3)$$

Denklem (2) ve (3) kullanılarak, çıkış açısal hız değeri ile giriş gerilimi arasındaki transfer fonksiyonu Eş. 4.4’de bulunabilir:

$$G(s) = \frac{\omega(s)}{V_a(s)} = \frac{K/JL_a}{s^2 + \left[\frac{JR_a + fL_a}{JL_a}\right]s + (fR_a + K^2)/JL_a} \quad (4.4)$$

Benzer şekilde, açısal hız değeri ile yük torku arasındaki transfer fonksiyonu Eş. 4.5’de hesaplanabilir:

$$H(s) = \frac{\omega(s)}{t_d(s)} = \frac{(R_a + L_a)}{(R_a + L_a s)(f + Js) + K^2} \quad (4.5)$$

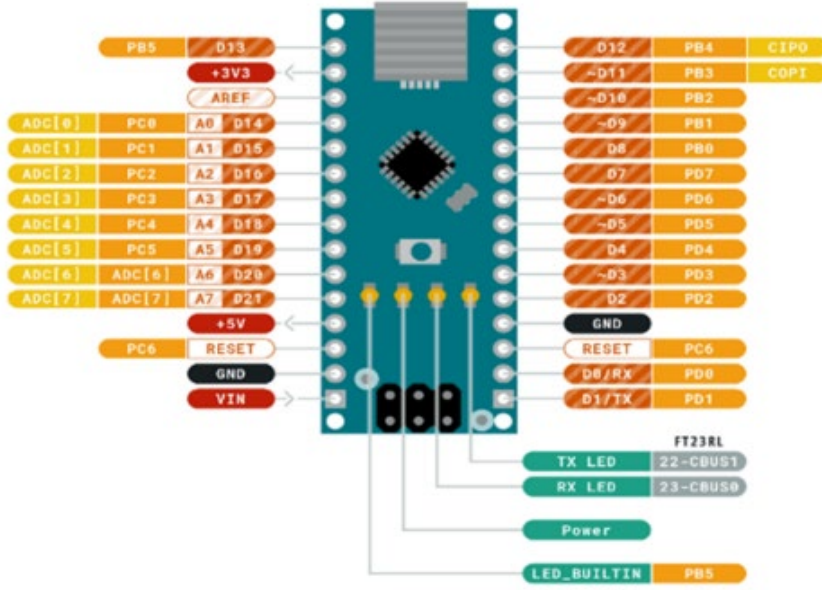
Sonuçta, DC motor açısal hız değeri aşağıdaki gibi hesaplanır:

$$\omega(s) = G(s)V_a - H(s)t_d \quad (4.6)$$

4.7. Arduino Nano

2008 yılında piyasaya sürülen Arduino Nano, Arduino dünyasının en küçük geliştirme kartlarından biridir ve aynı zamanda breadboard veya PCB’ler üzerinde kullanılmak üzere özel olarak tasarlanmıştır. Arduino Uno’nun tüm özelliklerini barındıran bu kompakt platform, Uno ile aynı Atmel tarafından üretilen Atmega328 mikrokontrolcüyu kullanmaktadır. Nano, Uno ile karşılaştırıldığında bazı küçük elektriksel farklılıklar içerse de Uno için yazılmış olan ve fazla uzun olmayan sketch kodları, herhangi bir değişiklik yapmaya gerek kalmadan Nano üzerinde de çalışabilir. Nano, 30 pinli Dual Inline Package (DIP) yapıya sahiptir ve Uno ile karşılaştırıldığında boyut olarak çok daha küçüktür. Bu küçük boyut, bileşenlerin çoğunun kartın altına yerleştirilmesiyle mümkün olmuştur. Nano, sıkça USB harici disklerde gördüğümüz Mini-B tipi USB bağlantısına sahiptir ve 14 dijital pin ile 8 analog pine ev sahipliği yapmaktadır. Ayrıca, üzerinde güç aldığını gösteren bir ON LED’i, RX ve TX LED’leri bulunur ve Uno’da olduğu gibi 13. Dijital pine bağlı dahili bir LED de mevcuttur. Arduino Uno’nun sahip olduğu teknik özellikler göz önünde bulundurulduğunda, Nano’nun Uno’dan iki fazla analog pine sahip olduğu görülür. Uno’nun analog pinleri A0’dan A5’e kadar devam ederken, Nano A6 ve A7 olarak iki ek pin sunar. Bu ek pinlerin kullanımıyla ilgili bazı farklılıklar bulunmaktadır ve bu konu daha sonraki bölümlerde ele alınacaktır [48-50].

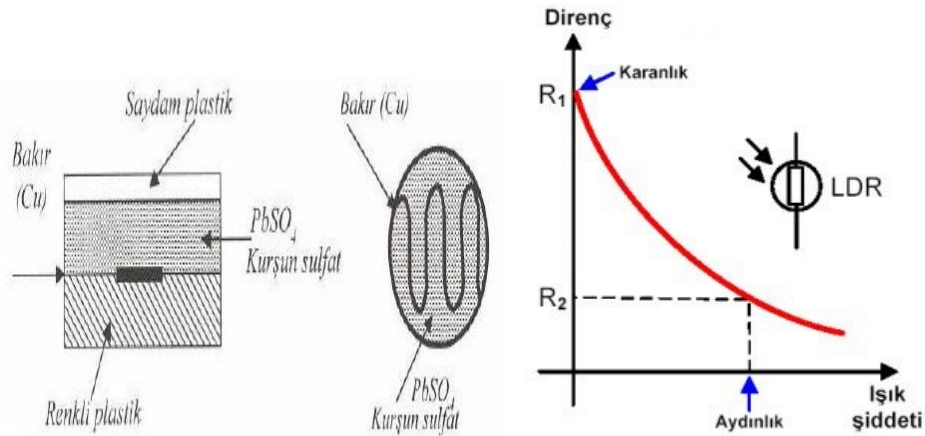
Nano, breadboard kullanımı için idealdir ve 45×18 mm boyutlarındadır. Altında lehimli erkek header pinleri sayesinde, bir entegre devre çipi gibi breadboard’a kolayca takılabilir. Arduino projelerinin çoğu prototip oluşturma veya kısa süreli eğlence amaçlı olduğundan, devrelerin breadboard üzerine kurulması projeyi hızla hayata geçirmemizi ve sonra parçaları başka projelerde kullanabilmemizi sağlar. Nano, tasarlanmış PCB’ler üzerinde de kullanılabilir. PCB’ye, Nano’yu yerleştirebileceğimiz bir soket oluşturmak için 2 adet 15 pinli dişi header lehimleyebiliriz. Bu sayede Nano’yu gerektiğinde söküp başka projelerde kullanmak mümkün olur. Eğer Nano’yu PCB üzerinde kalıcı olarak



Şekil 4.12. Arduino nano pin şeması

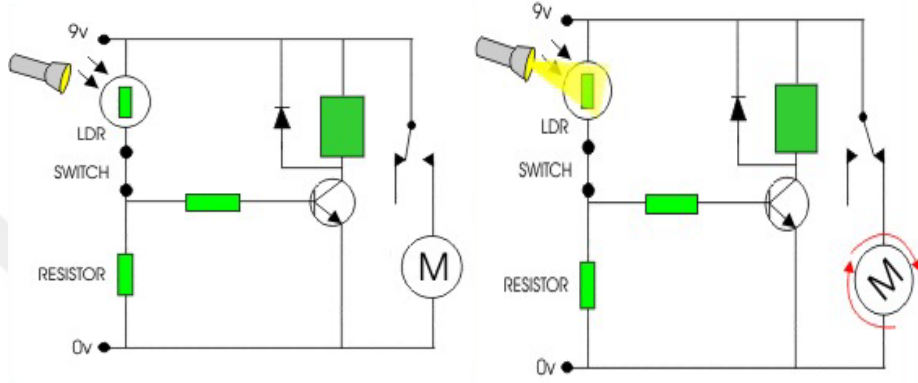
4.8. LDR ve Servo Sürücü Kartı

LDR (Light Dependent Resistor), Türkçede “Işığa Duyarlı Direnç” olarak bilinir ve foto direnç olarak da adlandırılır. LDR’ler, bir direnç türü olmalarına rağmen aynı zamanda pasif bir sensör işlevi görürler. Bulundukları devrelerde, dış ortamdan algıladıkları ışık değişikliklerine bağlı olarak direnç değerlerini değiştirerek bir çıkış sinyali üretirler. Bu özellikleri sayesinde, bir sensör görevi de üstlenmiş olurlar [51,52]. LDR’nin iç yapısı ve çalışma prensibi Şekil 4.13’te gösterilmiştir.



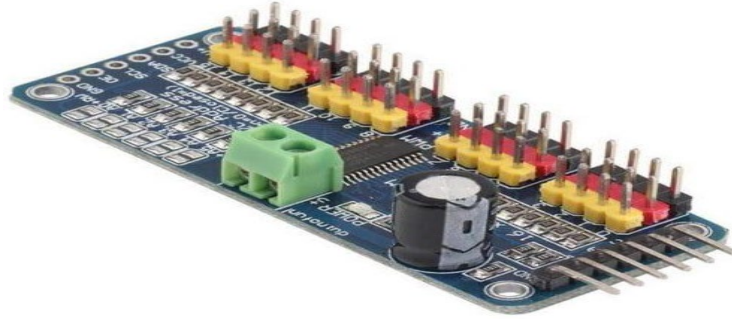
Şekil 4.13. LDR’nin iç yapısı ve çalışma prensibi

LDR, ışığa duyarlı bir komponenttir. Üzerine düşen ışığın yoğunluğu azaldıkça direnç değeri artar; ışık yoğunluğu arttıkça direnç değeri azalır. Bu özellik, LDR'nin ışık değişimlerine tepki vermesini ve bağlı olduğu devrede bir anahtarlama işlevi görmesini sağlar. Bu durum, LDR'yi bir tür optik sensör haline getirir. Ancak, aşırı sıcaklıklar LDR'nin performansını olumsuz etkileyebilir (Maksimum sıcaklık: 60°C) [51,52]. Şekil 4.14'te LDR'nin ışığa karşı tepkisel örneği gösterilmiştir.



Şekil 4.14. LDR'nin çalışma örneği

Servo sürücü kartı ile Arduino veya diğer mikrokontrollerlere 16 ekstra PWM çıkışı ekleyebiliriz. Bu özellik, özellikle çok sayıda LED ya da servo motor içeren projelerinizi kolayca hayata geçirmenizi sağlar. Kart, 12-bit çözünürlük sunar ve 3.3V ile 5V arasındaki lojik seviyelerde çalışabilir. I2C arayüzü kullanılarak, mikrokontrolcüde fazla pin kullanımını önler. Üzerindeki altı adet adres değiştirme jumper'ı ile tek bir I2C hattında bu karttan en fazla 62 adet kullanılabilir. Kart üzerinde standart servo konektörleriyle uyumlu çıkış dizilimi mevcuttur. Her iki uçtaki bağlantı pinleri sayesinde, kartlar birbirine eklenebilir veya projenize en uygun şekilde monte edilebilir. Ayrıca, V+ hattından harici güç beslemesi için klemens bağlantısı bulunur ve ters gerilim koruması sağlanmıştır [53-55]. Bu tez çalışmasında kullanılan PCA9685 servo sürücü kartı Şekil 4.15'te gösterilmiştir.



Şekil 4.15. PCA9685 servo sürücü kartı

Teknik Özellikler:

- Boyutlar (klemens ve header bağlantıları dışında): 62.5 mm x 25.4 mm x 3 mm
- Ağırlık: 5,5 gram (klemens ve header bağlantıları hariç), 9 gram (klemens ve header bağlantıları dahil)
- I2C adres aralığı: 0x60 – 0x80, kullanıcı tarafından seçilebilir
- Çözünürlük: 12-bit, 60 Hz güncelleme frekansı
- PWM frekansı: 1.6 kHz'e kadar ayarlanabilir
- Çıkış seçenekleri: Push-pull veya open-drain
- Output Enable pini ile tüm çıkışların kontrolü
- LED sürme işlemini kolaylaştıran seri bağlı 220 Ω direnç
- Güç kaynağının yeterliliğini gösteren yeşil renkli LED

4.9. PID Algoritması

Bir orantılı-integral-türev kontrolörü (PID kontrolörü veya üç terimli kontrolör) olarak bilinen bu mekanizma, sürekli olarak ayarlanması gereken kontrol gereksinimleri olan endüstriyel sistemlerde ve diğer birçok uygulamada kullanılan bir geri bildirim döngüsü mekanizmasıdır. Bu kontrolör, istenen bir set noktası (SP) ile ölçülen işlem değişkeni (PV) arasındaki fark olan hata değeri $e(t)$ 'yi sürekli olarak hesaplar. Hata üzerinde orantılı (P), integral (I) ve türev (D) düzeltmeler yaparak müdahalede bulunur; bu nedenle adını bu terimlerden alır. PID algoritmasına ait denklem Eş. 4.7'de verilmektedir [43].

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{de(t)}{dt} \quad (4.7)$$

Burada:

- K_p orantılı kazanç
- K_i integral kazanç
- K_d türev kazanç

Bu sistemler, kontrol fonksiyonlarına otomatik ve hassas düzeltmeler uygulayarak hızlı yanıt verirler. Örneğin, bir otomobilin hız sabitleyici sistemi, yokuş çıkarken sabit motor gücü uygulandığında hız düşer. PID algoritması, motor gücünü kontrol ederek ölçülen hızı istenilen hızla az gecikme ve aşma ile geri yükler. Orantılı kontrolde, hata sinyali e ve bu hatayı etkileyen K_p orantılı kazancı bulunur. Hata miktarı arttıkça, orantılı bir kontrol çıkışı elde edilir. Hata sabit kaldığında kontrol çıkışı da sabit kalır. Orantılı kontrolde, çıkış değeri ile referans giriş değeri arasında bir fark oluşur ve bu durum kalıcı durum hatası olarak adlandırılır. Kalıcı durum hatasını azaltmak amacıyla orantılı bant daraltılabilir; ancak bu, aç-kapa kontrolüne benzer bir durum yaratır ve referans değeri etrafında dalgalanmalar oluşur. Orantılı kontrol katsayısını artırarak, çıkış değerini istenen değere daha yakın hale getirebiliriz. Ancak, bu durum sistemde salınımlar ve aşmalar yaratarak dengesizliklere yol açabilir. Bu nedenle orantılı kontrol, birçok kontrol sistemi için tek başına yeterli değildir.

İntegral (I) kontrol, orantılı kontrolün yarattığı kalıcı durum hatasını gidermek için kullanılır. integral kontrol, referans giriş değeri ile çıkış değeri arasındaki kalıcı durum hatasını ortadan kaldırmaya çalışır. Ancak, PI denetleyicinin sistemin yerleşme zamanını uzatması ve aşma olasılığını artırması göz ardı edilmemelidir.

$$u_I(t) = K_i \int_0^t e(\tau) d\tau \quad (4.8)$$

Türev (D) kontrol, sistem çıkışının her noktada türevini alarak, sistemin gelecekteki davranışlarını tahmin etmeye çalışır. Bu denetleyicinin amacı, çıkış değerinin kontrol altında tutulması ve aşmaların önlenmesidir. Türev kontrol, hızlı ve kısa süreli değişikliklere, anlık gürültülere ve çıkış değerindeki değişikliklere hızla tepki verir.

$$u_D(t) = K_d \frac{de(t)}{dt} \quad (4.9)$$

PID kontrolörü, Orantılı, İntegral ve Türev kontrol özelliklerini bir arada sunar. Kontrolü zor ve karmaşık sistemlerde tercih edilir. PI ve PD kontrolörlerinin sistem üzerindeki farklı etkilerinin olumsuz yönleri de bulunabilir. Bu sorunları gidermek için, her iki kontrolör birleştirilerek sistem davranışlarının düzeltilmesi sağlanabilir. PID kontrolörü

kullanılırken, bir değişkenin diğer iki değişken üzerinde olumlu veya olumsuz etkiler yaratabileceği unutulmamalıdır [43].



5. MATERYAL VE YÖNTEM

Bu çalışmada; Arduino Nano'da A3 pin'i, dijital I/O olarak kullanılabilen analog pinlerden biridir. Ultrasonik sensörler için hem tetikleme (TRIGGER) hem de yankı (ECHO) pinlerinin kullanımı gerekir. Eğer tek bir pin kullanılarak tetikleme ve yankı işlevleri gerçekleştirilecekse, bu işlevler için hızlı yanıt verme yeteneği olan bir analog pin tercih edilir. A3 pinini kullanarak, yer tasarrufu sağlanmış ve donanım karmaşıklığı azaltılmıştır. Bu sayede ultrasonik sensörden gelen yankı süresini doğru bir şekilde ölçmek ve buna göre mesafe hesaplamak mümkün olur.

A7 pin'i, Arduino Nano'da kullanılabilen son analog giriş pinidir ve bu pin diğer analog pinlerle birlikte genellikle sensör okumaları için ayrılır. LDR gibi pasif bir sensör için analog okuma yapabilen bir pin gereklidir çünkü LDR'den alınan analog değer, ışık şiddetiyle doğru orantılıdır. LDR'nin bağlandığı A7 pin'i, ışık şiddetine bağlı olarak değişen voltajı okuyarak ortam ışığının yoğunluğunu ölçer. Bu bilgi, robotun çevresel aydınlatma koşullarına göre adaptasyonunu sağlar.

Sağ Motor (12, 10, 11), Sol Motor (8, 7, 9) için kullanılan bu pinler, Arduino Nano'da PWM çıkışı sağlayabilen dijital pinlerdir. Motor sürücüler genellikle motor hızını kontrol etmek için PWM sinyallerini kullanır. Bu nedenle, PWM desteği olan pinler motor kontrolü için idealdir. PWM pinleri (11 ve 9), motor hızını ayarlamak için kullanılırken, diğer pinler (12, 10, 8, 7) motorun dönüş yönünü kontrol eder. Bu pinler üzerinden motorlara uygulanan yüksek ve düşük sinyaller, motorların ileri veya geri dönüşünü sağlar.

Arduino Nano'da I2C iletişimi için özel olarak tahsis edilmiş olan A4 ve A5 pinleri, seri veri (SDA) ve seri saat (SCL) işlevlerini destekler. Bu pinler, birden fazla cihazın aynı hat üzerinde iletişim kurabilmesini sağlayarak donanım bağlantılarını basitleştirir. PCA9685 servo kontrolcüsü, bu I2C hatları üzerinden Arduino ile iletişim kurar. Bu, servo motorların kontrolünü merkezileştirir ve Arduino'nun pin sayısını verimli bir şekilde kullanmasını sağlar.

2, 3, 4, 5, 6, A0, A1 ve A2 pinleri, çizgi izleme sensörleri için gerekli olan hızlı dijital giriş çıkış işlemlerini desteklemek için seçilmiştir. Çizgi izleme sensörleri, yüzey üzerindeki renk değişimlerini hızlı bir şekilde algılamalıdır. QTR sensörleri bu pinlere bağlanarak, robotun üzerinde hareket ettiği zemin üzerindeki izi takip etmesini sağlar. Sensörlerden gelen dijital veriler, robotun çizgi üzerinde kalmasını ve yönlendirilmesini sağlayacak şekilde işlenir.

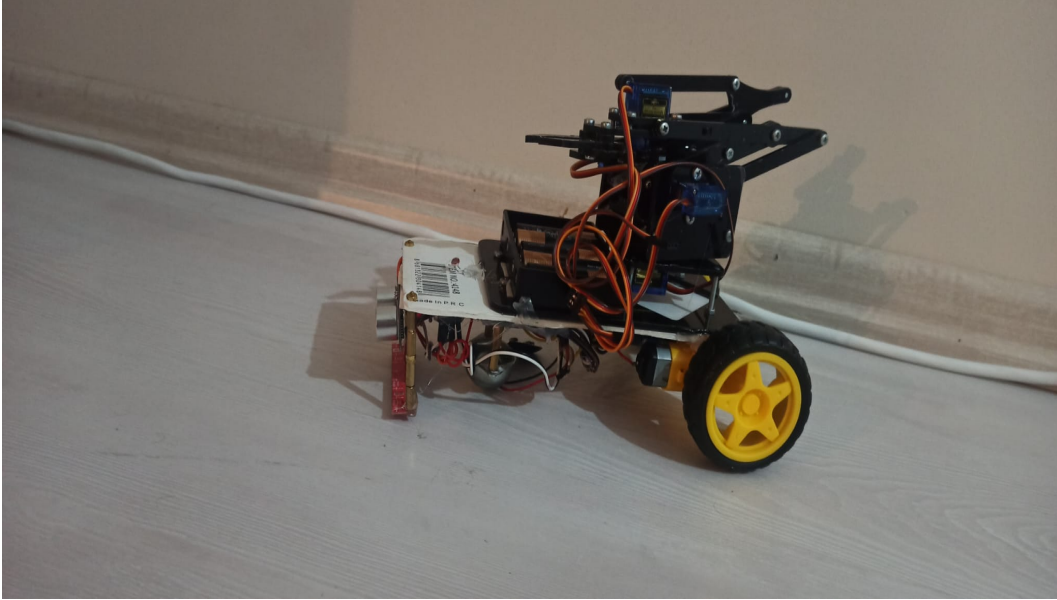
Robotun ön kısmında yer alan HC-SR04 ultrasonik mesafe sensörü, çevresindeki nesnelerin mesafesini ölçmek için ses dalgaları yayar. Bu dalgalar, bir nesneye çarptığında geri yansır ve sensör bu yansıyan dalgaların süresini ölçerek mesafeyi hesaplar. Eğer sensör, programda belirlenen kritik bir mesafe olan 7 cm veya daha azını algırsa, motorControl(false) fonksiyonu aracılığıyla motorlar durdurulur, böylece robot güvenli bir şekilde durakta durur. Motorlar durduktan sonra, LDR (Işık Bağımlı Direnç) sensörü devreye girer. Bu sensör, ortamdaki ışık şiddetini ölçer ve elde edilen analog değer, Arduino tarafından işlenir. Değerlendirme sürecinde; LDR'deki ışık şiddeti belirli bir eşik değeri aştığında veya altına düştüğünde bu değişiklik algılar. Eğer LDR değerinde belirlenen bir değişim eşiği (örneğin 300 birim) aşılsa, bu robotun yeni bir göreve veya konuma hareket etmesi gerektiğini gösterir ve motorControl(true) fonksiyonu ile motorlar yeniden başlatılır. Eşik değerlere göre duraksama ve görev ataması, LDR'den alınan ışık şiddeti bilgisine göre yapılır ve robot belirli görevleri yerine getirir. Düşük ışık şiddeti belirli eşik değerinin altındaysa, bırak() fonksiyonu çağrılır ve servo motorlar durak 1'de bir nesneyi almak için uygun konuma getirilir. Yüksek ışık şiddeti belirli bir değer üzerindeyse, al() fonksiyonu ile durak 2'de bir nesneyi almak için harekete geçilir. Servo motorlar, nesneleri kavramak veya bırakmak için belirli açılarda konumlandırılmalıdır. Bu açıların PWM (Pulse Width Modulation) değerlerine dönüştürülmesi, servo motorların teknik özelliklerine göre yapılır. Örneğin, 60 derecelik bir açının PWM değerinin hesaplanması, servo motorun minimum ve maksimum PWM değerleri ile çalışma aralığını dikkate alarak yapılır. Tipik bir servo motor için, 0 derece 150 PWM ve 180 derece 600 PWM olarak ayarlanabilir. Bu durumda, 60 derece için PWM değeri lineer bir interpolasyonla hesaplanır:

$$PWM = 150 + (60/180) \times (600 - 150) \quad (5.1)$$

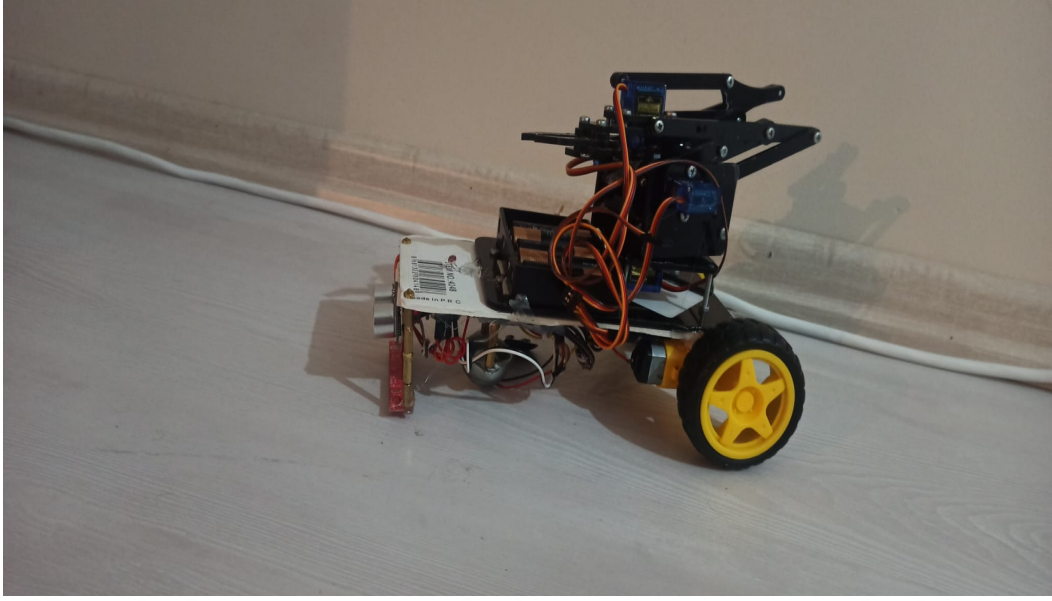
$$PWM = 150 + (180/60) \times (600 - 150) \quad (5.2)$$

Bu formül, 60 derecenin servo motor üzerindeki karşılığını bulmak için kullanılır.

Robot, çizgi üzerinde stabil bir şekilde ilerlemek için PID kontrol mekanizması kullanır. QTR sensörleri çizginin konumunu belirler ve 3500 değeri çizginin sensör dizisinin tam ortasında olduğunu gösterir. Bu değerden sapma, çizginin merkezden sağa veya sola kaydığını gösterir ve bir hata sinyali olarak PID kontrolörüne beslenir. PID kontrolörü, bu hata sinyaline dayanarak orantılı, integral ve türev kontrol sinyalleri üretir ve bu sinyaller motor hızlarını ayarlamak için kullanılır, böylece robot çizgi üzerinde merkezi olarak kalır. Projede tasarlanan robot Resim 5.1’de ve Resim 5.2’de gösterilmiştir. Robotun yazılımı için gerekli kodlar Ek A’da verilmiştir. Çizgi takip yöntemlerini karşılaştırılması, robot hareket yöntemlerini karşılaştırılması, diferansiyel sürüş yöntemlerini karşılaştırılması ve mikro işlemcilerin karşılaştırılması Ek B’de verilmiştir.



Resim 5.1. Projede tasarlanan endüstriyel mobil robot



Resim 5.2. Projede tasarlanan endüstriyel mobil robot

6. BULGULAR VE TARTIŞMA

6.1. Sensör Entegrasyonu ve Algılama Kapasitesi

Bulgular: Başlangıç kodunda, bir LDR sensörü ve bir ultrasonik sensör kullanılmaktadır. LDR sensörü, ışık seviyelerine bağlı olarak nesne algılama işlevi görürken, ultrasonik sensör mesafe ölçümünde kullanılmaktadır. Bu sensörler aracılığıyla yapılan temel algılamalar, robotun çevresiyle etkileşimini sınırlı tutmaktadır.

Tartışma: Mevcut literatür [56,57], özellikle mobil robotikte, çoklu sensör entegrasyonunun önemini vurgulamaktadır. Çalışmanın başlangıcında kullanılan LDR ve ultrasonik sensörlerin kombine edilmesi, temel engel algılama ve ışık tabanlı sınıflandırma işlevleri için yeterli olsa da, daha karmaşık görevler ve çevresel etkileşimler için yetersiz kalmaktadır. Örneğin, nesne tanıma ve detaylı çevresel modelleme gibi işlevler bu sensörlerle sınırlıdır.

6.2. Motor Kontrolü, Navigasyon ve İşlevselliği

Bulgular: Robot, PID kontrol parametreleri kullanılarak programlanmış motorlarla donatılmıştır. Bu kontrol sistemi, robotun belirli bir hızda ve istikamette hareket etmesini sağlar. Motorlar, robotun hattı izlemesine ve belirlenen yollar üzerinde manevra yapmasına olanak tanır. Servo motorda ise robotun çeşitli mekanik işlevleri, dört adet servo motor tarafından yürütülmektedir. Bu servo motorlar, belirli görevler için programlanabilir ve robotun nesneleri manipüle etmesini sağlar. Servo motorlar, belirlenen başlangıç ve durak pozisyonları arasında hareket ederken, PWM sinyalleri ile kontrol edilmektedir.

Tartışma: PID kontrol mekanizması, robotun hareket stabilitesini sağlamak için kritik bir yöntemdir; fakat, statik PID parametreleri, çevresel değişikliklere adaptasyon konusunda kısıtlamalar getirebilir. Dinamik çevrelerde robotun etkinliğini artırmak için adaptif kontrol sistemleri önerilmektedir. Bu sistemler, robotun farklı zemin tiplerine, engellere ve diğer değişkenlere real-time olarak uyum sağlamasını mümkün kılar. Örneğin, zeminin kayganlığı veya eğimi değiştiğinde, adaptif sistemler motor hızını ve yönünü bu yeni koşullara göre otomatik olarak ayarlayabilir. Literatürde benzer uygulamalar [58-60],

robotların çok daha esnek ve etkili bir şekilde navigasyon yapmalarını sağlamıştır. Servo motorların kullanımında ise robotun çeşitli mekanik görevleri yerine getirmesi açısından önemlidir, ancak mevcut kontrol sistemi sınırlı hassasiyet sunar. Servo Motor Kalitesi vs. PWM ile servo kontrolü, basit açık döngü kontrolüne dayanır ve motor pozisyonlarını geri bildirimle düzeltemez. Bu, özellikle hassas manipülasyon gerektiren uygulamalarda limitasyon oluşturabilir. Gelişmiş servo kontrol sistemleri, geri bildirim döngüleri ve daha hassas hareket planlaması içerir. Bu sistemler, servo motorların konumunu sürekli olarak izleyerek herhangi bir sapmayı düzeltebilir ve çok daha doğru hareketler sağlar. Literatürde [61-63], robotik kollar ve hassas montaj sistemlerinde bu tür gelişmiş kontrol tekniklerinin başarıyla kullanıldığı belirtilmektedir.

6.3. Hata Yönetimi, Güvenlik Protokolleri ve Kullanıcı Arayüzü

Bulgular: Robotta hata yönetimi ve güvenlik protokollerinin eksik olduğu görülmektedir. Sensör arızası, motor hataları veya beklenmeyen çevresel değişiklikler gibi durumlar, robotun işlevselliğini ciddi şekilde etkileyebilir ve güvenliğini riske atabilir. Robotun kullanıcı arayüzü ve sistem durumunun izlenmesi kapasiteleri sınırlıdır. Kod, bazı temel seri çıkışlar sunar, ancak bu, robotun durumunu anlamak ve yönlendirmek için yeterli bilgi sağlamaz.

Tartışma: Mevcut sistemde, sensör okumalarının geçerliliğini kontrol etmek veya hatalı durumları yönetmek için herhangi bir mekanizma bulunmamaktadır. Literatürde, mobil robotlarda hata algılama, hata düzeltme ve güvenlik protokolleri, sistemlerin güvenilirliğini artırmak için vazgeçilmezdir. Örneğin, sensör verilerinde ani ve mantıksız değişiklikler algılandığında, bir hata yönetimi sistemi robotun güvenli bir "bekleme" veya "kurtarma" moduna geçmesini sağlayabilir. Ayrıca, motorların aşırı yüklenmesini önlemek için akım sınırlayıcılar veya termal korumalar gibi donanım güvenlik önlemleri de entegre edilebilir. Bu tür önlemler, robotun uzun ömürlü ve güvenli bir şekilde çalışmasını sağlar.

Etkili bir kullanıcı arayüzü (UI), operatörlerin robotu kolayca kontrol etmelerini ve sistem durumunu anlamalarını sağlar. Gelişmiş UI tasarımları, robotun tüm sensör verilerini, motor durumlarını ve diğer kritik sistem bilgilerini gerçek zamanlı olarak gösterir. Ayrıca, operatörlerin robot üzerinde daha doğrudan kontrol ve ayarlamalar yapmasına olanak tanır. Literatürde [64-66], özellikle karmaşık robotik sistemlerde, detaylı ve interaktif

kullanıcı arayüzlerinin kullanılmasının, operasyonel verimliliği ve kullanıcı memnuniyetini artırdığı gözlemlenmiştir.

6.4. Güç Yönetimi, Verimlilik, Çevresel Adaptasyon ve Ağ-İletişim Kapasitesi

Bulgular: Robotun güç yönetimi stratejileri üzerine durulmamıştır. Motorlar ve diğer aktüatörler enerji tüketimi açısından optimize edilmesine önem verilmemiştir. Robotun çevresel değişikliklere adaptasyon kapasitesi oldukça sınırlıdır. Sensörlerden gelen veriler statik bir mantıkla işlenmekte ve çevresel değişimlere dinamik bir yanıt verilememektedir. Robotun diğer cihazlar veya sistemler ile haberleşme kabiliyetlerini içermemektedir.

Tartışma: Enerji verimliliği, özellikle batarya ile çalışan mobil robotlar için hayati öneme sahiptir. Enerji tüketimini optimize etmek, robotun çalışma süresini uzatabilir ve operasyonel maliyetleri azaltabilir. Motorların ve servo mekanizmaların enerji verimliliğini artıracak şekilde programlanması, güç tüketimini düşürebilir. Enerji verimliliği stratejileri arasında, yüksek verimli motorlar kullanmak, enerji geri kazanım sistemleri entegre etmek ve işlemci gücünü ihtiyaca göre ayarlamak gibi yöntemler öne çıkmaktadır.

Robotların çeşitli çevre koşullarına uyum sağlaması, özellikle açık ve değişken ortamlarda operasyonel başarı için kritik öneme sahiptir. Adaptif algoritmalar kullanarak sensör verilerinin çevresel koşullara göre yorumlanması, robotun daha esnek ve bağımsız çalışmasını sağlayabilir. Örneğin, ışık seviyelerinin değiştiği bir ortamda LDR sensörü ayarlarının dinamik olarak güncellenmesi veya farklı zemin tiplerine göre motor tork ayarlarının adaptasyonu gibi. Çevresel adaptasyon yeteneği robotların karşılaştıkları zorluklara daha etkin bir şekilde cevap vermelerini sağlayarak, kullanım alanlarının genişlemesine olanak tanır.

Günümüz robotik sistemlerinde, özellikle endüstriyel ve araştırma uygulamalarında, birden fazla robotun ve cihazın birlikte çalıştığı senaryolar yaygındır. İletişim protokolleri ve ağ bağlantıları, bu tür sistemlerde koordinasyonu ve verimliliği artırır. Örneğin, bir merkezi kontrol sistemi üzerinden robotlara görev ataması yapılabilir veya robotlar arası durum bilgisi paylaşımı gerçekleştirilebilir. Literatürde [67,68], IoT (İnternet üzerinden Nesneler) teknolojilerinin ve M2M (Makineden Makineye) iletişiminin, robotik sistemlerin daha etkili bir şekilde entegre edilmesini sağladığı görülmektedir.

6.5. Algoritmik Zorluklar ve İşlem Kapasitesi

Bulgular: Robot, temel işlem kapasitesi ile donatılmıştır ve mevcut kod, sınırlı sayıda görevi yerine getirecek şekilde tasarlanmıştır. İşlem kapasitesinin sınırlı olması, algoritmik karmaşıklığı artıran görevlerde performans sınırlamalarına yol açar.

Tartışma: Mobil robotik sistemlerde işlem kapasitesi, verilen komutların hızlı ve verimli bir şekilde yürütülmesi için kritik bir faktördür. Mevcut sistemde kullanılan mikrodenetleyici, basit görevler için yeterli olabilir, ancak daha karmaşık algoritmalarda, örneğin görüntü işleme veya karmaşık yol planlama algoritmalarında yetersiz kalabilir. Özellikle otonom araçlar ve gelişmiş navigasyon sistemleri için kullanılan güçlü işlemciler, bu tür karmaşık işlemleri saniyeler içinde gerçekleştirebilmekte ve sistem yanıtlarını önemli ölçüde hızlandırmaktadır. Güçlü işlemcilerin entegrasyonu, robotun çevresel adaptasyon kabiliyetini ve çoklu görev yönetimini önemli ölçüde artırabilir.

6.6. Çoklu Öğrenme Yönetimi ve Öğrenme Kapasitesi

Bulgular: Mevcut sistem, sınırlı sayıda görevi eş zamanlı olarak yürütebilmekte ve karmaşık görev yönetimi konusunda zorluklar yaşatabilir. Bu, özellikle birden fazla görevin veya eylemin koordinasyonu gerektiğinde, sistemin etkinliğini sınırlar. Sistem, yapay zekâ (AI) algoritmalarını veya öğrenme mekanizmalarını içermemektedir, bu da robotun çevresel etkileşimlerden öğrenme ve bu bilgileri gelecekteki eylemlerine uygulama kapasitesini sınırlar.

Tartışma: Modern robotik sistemlerde, çoklu görev yönetimi, robotun farklı işlevleri aynı anda yürütebilmesi için önemlidir. Bu kapasite, özellikle endüstriyel robotik uygulamalarda, robotun birden fazla iş istasyonunda eş zamanlı olarak çalışabilmesini sağlar. Mevcut sistemdeki gibi basit kontrol yapıları yerine, gelişmiş işletim sistemleri ve çok görevli yazılım mimarileri, bu tür karmaşık koordinasyonları mümkün kılar. Literatürde [69,70], özellikle real-time işletim sistemleri (RTOS), birden fazla görevin zamanlamasını ve önceliklendirilmesini yönetebilir ve böylece robotun genel iş akışı verimliliğini artırabilir. Çok görevli sistemlerin entegrasyonu, robotun hem verimliliğini hem de işlevsel kapasitesini artıracaktır. Yapay zekâ ve makine öğrenimi (ML) teknolojilerinin robotik sistemlere entegrasyonu, bu sistemlerin adaptasyon kabiliyetini ve işlevselliğini önemli ölçüde artırabilir. Derin öğrenme modelleri, çevresel verileri analiz ederek robotun karşılaştığı durumlar hakkında öğrenmesini ve bu bilgileri benzer durumlarda uygulamasını sağlar. Literatürde [71-73], AI ve ML'in, robotların karmaşık

görevleri yerine getirme yeteneğini önemli ölçüde geliştirdiği ve operasyonel verimliliği artırdığı belgelenmiştir. Bu tür sistemler, robotun karar verme süreçlerini otomlaştıracak ve daha dinamik ve esnek operasyonel yetenekler sunar.





7. SONUÇ VE ÖNERİLER

Bu projede, bir robot kolun tasarımı, montajı ve test edilmesi süreçleri detaylı bir şekilde incelenmiştir. Bu kapsamda, başlangıçta kullanılan LDR sensörü ve ultrasonik sensör ile ilgili bazı önemli zorluklar tespit edilmiştir. Geliştirilen algoritmanın testi, bu zorlukların üstesinden gelmek için ev ortamında denenmiştir. Kullanılan bileşenlerin seçimi, sistem entegrasyonu, sensör kalibrasyonu, motor performansı ve kontrol sistemi optimizasyonu gibi çeşitli aşamalar ele alınmıştır. Deneyler ve ölçümler, robot kolun performansını değerlendirmek ve potansiyel iyileştirme alanlarını belirlemek için kritik öneme sahip olmuştur.

- 1. Sensör Kalibrasyonu ve Algılama Hassasiyeti:** Sensörlerin doğru şekilde kalibre edilmesi, robot kolun çevresini doğru bir şekilde algılaması için hayati önem taşımaktadır. Çeşitli yüzey türleri ve aydınlatma koşulları altında yapılan testler, sensör kalibrasyonunun önemini ortaya koymuştur. LDR sensörü, değişken ışık koşullarında istikrarlı sonuçlar üretememiş ve durak noktalarını doğru bir şekilde algılayamamıştır. Bu, ışık şiddetine duyarlı olduğundan dış mekân uygulamalarında veya değişken aydınlatmalı ortamlarda kullanımını sınırlamaktadır. Ultrasonik sensör, özellikle plastik gibi bazı yüzeyleri algılamada zorluk çekmiştir. Yansıtma özellikleri düşük olan malzemeler, yanlış mesafe okumalarına yol açmıştır, bu da robotun engelleri yanlış algılamasına ve potansiyel çarpışma risklerine neden olmuştur.
- 2. Motor Performansı ve Kontrol Sistemi:** Motorların ve hareket kontrol sistemlerinin, beklenen yükler ve hızlarda etkili bir şekilde çalıştığı gözlemlenmiştir. Ancak, yüksek tork gerektiren durumlar ve hız kontrolü, motor seçimi ve kontrol algoritmalarında dikkatli bir planlama gerektirmiştir. Robotun motorları, zorlu arazi koşullarında yetersiz kalmıştır. Bu, robotun hareket kabiliyetini ve genel performansını olumsuz yönde etkilemiştir.
- 3. Enerji Verimliliği ve Güç Yönetimi:** Robotun enerji tüketimi, özellikle uzun süreli görevlerde önemli bir faktördür. Enerji verimliliği stratejileri ve uygun pil seçimi, projenin genel verimliliğini artırmıştır. Robotun enerji yönetimi yetersiz kalmış, bu da operasyonel süresini kısaltmış ve sık şarj gereksinimine yol açmıştır.

4. **Mekanik Dayanıklılık ve Güvenilirlik:** Dayanıklılık ve güvenilirlik testleri, robot kolun uzun süreli kullanımı sırasında mekanik ve elektronik bileşenlerin performansını doğrulamıştır. Bu testler, potansiyel zayıf noktaların ve iyileştirilmesi gereken alanların belirlenmesine yardımcı olmuştur.
5. **Robot Kolunun Manipülasyon Sorunları:** Robot kolunun, belirtilen pozisyonlarda doğru bir şekilde konumlanamaması ve nesneleri tam olarak kavrayamaması, montaj ve manipülasyon görevlerinde ciddi performans sorunlarına neden olmuştur.

Öneriler

1. **Gelişmiş Sensör Entegrasyonu:** Robot kolun çevresel algılamasını iyileştirmek için, daha gelişmiş sensörlerin entegrasyonu ve çoklu sensör füzyon tekniklerinin kullanılması önerilir. Bu, robotun daha karmaşık görevleri yerine getirmesine olanak tanır. QR kodları veya RFID etiketleri kullanarak, LDR sensörünün yerine daha güvenilir durak tanıma sistemi kurulmalıdır. Görüntü işleme teknolojileri ve daha ileri sensörler (LIDAR veya TOF kameralar) kullanılarak ultrasonik sensörün yetersiz kaldığı durumlar için alternatif çözümler geliştirilmelidir.
2. **Motor Gücünün Arttırılması ve Kontrol Mekanizmalarının İyileştirilmesi:** Yüksek tork kapasiteli motorlar kullanılarak, robotun arazi adaptasyonu ve yük taşıma kapasitesi artırılmalıdır. Adaptif kontrol algoritmaları geliştirilerek, motorların çeşitli yüzey ve yük koşullarına dinamik olarak uyum sağlaması sağlanmalıdır.
3. **Kontrol Sistemi İyileştirmeleri:** Hareket kontrol algoritmalarının daha da geliştirilmesi, robot kolun hareket hassasiyetini ve tepki süresini iyileştirebilir. Yapay zekâ ve makine öğrenimi teknikleri, kontrol sistemlerinin optimizasyonunda kullanılabilir.
4. **Enerji Verimliliği Odaklı Tasarım:** Enerji tüketimini daha da azaltmak için, enerji verimliliği odaklı tasarım yaklaşımları benimsenmelidir. Bu, düşük güç tüketimli bileşenlerin kullanılması ve yazılımda enerji tasarrufu sağlayacak optimizasyonların yapılmasını içerir. Dinamik enerji yönetim sistemleri ve verimliliği artıracak teknolojiler geliştirilerek robotun çalışma süresi uzatılmalıdır.
5. **Robot Kol Mekanizmalarının Geliştirilmesi:** Servo motorlar için geri bildirim döngülerine dayanan kontrol sistemleri entegre edilerek, robot kolun hareket

hassasiyeti ve kavrama gücü artırılmalıdır. Ayrıca, esnek veya modüler robotik kolların kullanılması, farklı manipölasyon görevleri için daha geniş bir uygulanabilirlik sağlayabilir.

- 6. Acil Durum Durdurma Sistemi:** Robotun anlık durdurulması için acil durum durdurma mekanizmaları entegre edilmelidir. Bu, operatörlerin güvenliği ve sistem güvenilirliği açısından önemlidir.





KAYNAKLAR

- [1] İren, K. (2013). *Endüstriyel Çizgi Takip Eden Robot Cihazı Geliştirilmesi* (Yüksek Lisans Tezi), Hacettepe Üniversitesi, Fen Bilimleri Enstitüsü, Ankara.
- [2] Buyrukbilen, Y. (2023). *Hastanelerde Hastaların ve İlaçların Robotlarla Taşınması* (Yüksek Lisans Tezi), Necmettin Erbakan Üniversitesi, Fen Bilimleri Enstitüsü, Konya.
- [3] Yalman, S., & Haşiloğlu, A. (2015). Hastanelerde Eş Zamanlı İlaç Dağıtımı Yapan Hemşire Hemşir Robotun Geliştirilmesi, *Tıp Teknolojileri Ulusal Kongresi*, Muğla, Türkiye, Ekim 15-18.
- [4] Turan, S. (2017). Akülü Engelli Araçları İçin Engel Algılayan Çizgiler Arasında Giden Robot Tasarımı ve Gerçeklenmesi, *Gaziosmanpaşa Bilimsel Araştırma Dergisi*, 6 (Özel Sayı (ISMSIT2017)), 21-29.
- [5] Barua, V., Neloy, M. A. I., Rahat, S. U., Das, M., Joy, M. S. I., Nahar, N., & Pathak, A. (2020). An ultrasonic line follower robot to detect obstacles and edges for industrial and rescue operations. *International Journal of Computer Applications*, 176(31), 31.
- [6] Müftü, S. (2023). *İki Uzunlu Bir Robot Kolunun Pozisyon Kontrolü İçin Optimize Edilmiş Pid Kontrolcü Tasarımı* (Doktora Tezi), Necmettin Erbakan Üniversitesi, Fen Bilimleri Enstitüsü, Konya.
- [7] Sarıaltın, E. (2017). *5 Eksenli Endüstriyel Robot Kol* (Yüksek Lisans Tezi), İstanbul Gelişim Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul.
- [8] Ozan, E. (2020). *Robotlar ve uygulamaları* (Yüksek Lisans Tezi), Batman Üniversitesi, Fen Bilimleri Enstitüsü, Batman.
- [9] Url-1 <<https://www.bilgiustam.com>>, erişim tarihi 05.04.2022.
- [10] Url-2 <<https://www.gercekbilim.com>>, erişim tarihi 05.04.2022.
- [11] Gürgöze, G., Türkoğlu, İ. (2019). Kullanım alanlarına göre robot sistemlerinin sınıflandırılması, *Fırat Üniversitesi Mühendislik Bilimleri Dergisi*, 31(1), 53-66.
- [12] Aksun, B. (1996). *Mobile Robots* (Yüksek Lisans Tezi), İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul.
- [13] Harish, K., Megha, D., Shuklambari, M., Amit, K., & Chaitanya, K. J. (2017). Pick and place robotic arm using Arduino. *Int. J. Sci. Eng. Technol. Res.(IJSETR)*, 6(12), 1568-1573.
- [14] Shaibu, H. A. A., Ogakwu, P. A., Binfa, B., & Ibrahim, A. A. (2019). Development of an Arduino Controlled Robotic Arm. *Journal of Good Governance and Sustainable Development in Africa*, 5(2), 73-82.

- [15] Pawar, V., Bire, S., More, S., More, K., & Mule, R. (2018). Review on design and development of robotic arm generation-1, *Int J Innov Sci Res Technol*, 3(3).
- [16] Shanta, F. A., Islam, E., & Ahammad, R. (2020). Design, Construction and Performance Test of an Autonomous Low-Cost Pick and Place Robot Based on Color Detection, *International Conference on Mechanical, Industrial and Energy Engineering*, (ICMIEE20), 1-6.
- [17] Aksu, C. (2020). *Arduuno ile Çalışan Çok Fonksiyonlu Robot* (Yüksek Lisans Tezi), Haliç Üniversitesi, Fen Bilimleri Enstitüsü, Bilgisayar mühendisliği, İstanbul.
- [18] Baby, A., Augustine, C., Thampi, C., George, M., AP, A., & Jose, P. C. (2017). Pick and place robotic arm implementation using Arduino. *IOSR Journal of Electrical and Electronics Engineering*, 12(2), 38-41.
- [19] Myint, K. M., Htun, Z. M., & Tun, H. M. (2016). Position control method for pick and place robot arm for object sorting system. *International journal of scientific & technology research*, 5(6), 57-61.
- [20] Abdulkareem, A., Ladenegan, O., Agbetuyi, A. F., & Awosope, C. O. A. (2019). Design and implementation of a prototype remote-controlled pick and place robot. *International Journal of Mechanical Engineering and Technology*, 10(2).
- [21] Rohman, M., & Saputra, D. I. (2021). Modeling and controlling the actuator joint angle position on the robot arm base using discrete PID algorithm. *Ultima Computing: Jurnal Sistem Komputer*, 13(2), 50-56.
- [22] Surati, S., Hedao, S., Rotti, T., Ahuja, V., & Patel, N. (2021). Pick and place robotic arm: a review paper. *Int. Res. J. Eng. Technol*, 8(2), 2121-2129.
- [23] Yildiz, H., Korkmaz Can, N., Ozguney, O. C., & Yagiz, N. (2020). Sliding mode control of a line following robot. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, 42(11), 561.
- [24] Syafalni, I., Firdaus, M. I., Yugansyah, C., Sutisna, N., Hadi, Y. W., & Adiono, T. (2022). Design of testing environment for line-follower robot with obstacles. In *2022 International Symposium on Electronics and Smart Devices (ISESD)*, 1-5.
- [25] Tayal, S., Rao, H. P. G., Bhardwaj, S., & Aggarwal, H. (2020). Line follower robot: Design and hardware application. In *2020 8th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)(ICRITO)*, 10-13.
- [26] Amorim, J. S., Fernandes, J. H., Machado, M. S., Canella, A. L., & Pinto, M. F. (2022). Design of a control approach to assist the performance of a competitive line follower robot. In *2022 Latin American Robotics Symposium (LARS), 2022 Brazilian Symposium on Robotics (SBR), and 2022 Workshop on Robotics in Education (WRE)*, 354-359.
- [27] Kader, M. A., Islam, M. Z., Al Rafi, J., Islam, M. R., & Hossain, F. S. (2018). Line following autonomous office assistant robot with pid algorithm. In *2018 International Conference on Innovations in Science, Engineering and Technology (ICISSET)*, 109-114.

- [28] **Andrian, A., Rahmadewi, R., & Bangsa, I. A.** (2020). Arm Robot Pemindah Barang (Atwor) Menggunakan Motor Servo MG995 Sebagai Penggerak Arm Berbasis Arduino. *Electro Luceat*, 6(2), 142-155.
- [29] **Putri, S. F. M., Mardiaty, R., & Setiawan, A. E.** (2022). The prototype of arm robot for object mover using Arduino Mega 2560. *In 2022 8th International Conference on Wireless and Telematics (ICWT)*, 1-6.
- [30] **Aydoğan B.** (2016) *Arduino ile Robot Kol Kontrolü* (Lisans Bitirme Tezi) Karabük Üniversitesi, Mühendislik Fakültesi, Mekatronik Mühendisliği, Karabük.
- [31] **Url-3** <<https://www.robotistan.com>>, erişim tarihi 01.02.2023.
- [32] **Meşedilci, Ö.** (2019). *Şerit İzleme ve Araç-Araç Haberleşmeden Faydalanarak Trafik Kaza Önleme Sistemlerinin Geliştirilmesi* (Yüksek Lisans Tezi), İstanbul Gelişim Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul.
- [33] **Url-4** <<https://www.arnabkumardas.com>>, erişim tarihi 05.02.2023.
- [34] **Url-5** <<https://www.pololu.com/product>>, erişim tarihi 05.02.2023.
- [35] **Kadioğlu F., Eraslan Y. B.** (2018) *Çizgi İzleyen Robot* (Lisans Bitirme Projesi), Hava Harp Okulu, Bilgisayar Mühendisliği, İstanbul.
- [36] **Yavuz, O. İ.** (2017). *EOG (Elektrookülografi) Kullanarak Göz Hareketleri ile Robot Kontrolü* (Yüksek Lisans Tezi), İstanbul Gelişim Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul.
- [37] **Url-6** <<https://www.datasheetcatalog.com>>, erişim tarihi 17.03.2023.
- [38] **Bektaş Ö.** (2015) *Arduino UNO RC CAR* (Lisans Bitirme Tezi), İstanbul Aydın Üniversitesi, Mühendislik Fakültesi, Yazılım Mühendisliği, İstanbul.
- [39] **Url-7** <<https://www.metehoca.com>>, erişim tarihi 23.03.2023.
- [40] **Bashir, R. H.** (2019) *Bir Hız Kontrol Sisteminin Yazılım ve Donanım Gerçeklemesi* (Yüksek Lisans Tezi), 19 Mayıs Üniversitesi, Fen Bilimleri Enstitüsü, Samsun.
- [41] **Adarsh, S., Kaleemuddin, S. M., Bose, D., & Ramachandran, K. I.** (2016). Performance comparison of Infrared and Ultrasonic sensors for obstacles of different materials in vehicle/robot navigation applications. *In IOP Conference Series: Materials Science and Engineering*, 149(1), 012141. IOP publishing.
- [42] **Katu, U., Pallu, S., Tola, M., & Hasanuddin, Z.** (2017). Flood early warning system and distribution of information communication using radio link. *Int. J. Civ. Eng. Technol*, 8(7), 1255-1262.
- [43] **Aciman, T.** (2019) *PID Kontrol ile DC Motorun Hassas Konum Kontrolü* (Yüksek Lisans Tezi), Tokat Gaziosmanpaşa Üniversitesi, Fen Bilimleri Enstitüsü, Tokat.
- [44] **Url-8** <<https://www.robotistan.com>> erişim tarihi 23.03.2023.
- [45] **Kaya, İ., & Bawdaka, A.** (2019). DC Motor Sürücüsü İçin Model Öngörülü Denetleyici Tasarımı, *Dicle Üniversitesi Mühendislik Fakültesi Mühendislik Dergisi*, 10(3), 899-910.
- [46] **Url-9** <<https://www.robotiksistem.com>>, erişim tarihi 25.06.2023.

- [47] **Krishnan, R.** (2001). Electric motor drives: modeling, analysis, and control. (No Title).
- [48] **Buyrukbilen, Y., Uzun, Y., & Buyrukbilen, E.** Bölüm 5 Engelli Hastaların Taşınması ve İlaçlarının Dağıtımında Robotların Kullanımı. *Engelsiz Bilişim 2022 Teknoloji Yaşam Işığında*, 63.
- [49] **Url-10** <<https://www.metehoca.com>>, erişim tarihi 4.07.2023.
- [50] **Idhom, M., Budijanto, A., Mufti, N., Alamsyah, M. R., Kristiawan, K. Y., & Arinda, U. Y.** (2022). Making a Mobile Educational Robot with a Practical Approach Using Arduino. *Nusantara Science and Technology Proceedings*, 253-262.
- [51] **Şengün, B.** (2018) *Hastane İçi Yardımcı Servis Robotu* (Yüksek Lisans Tezi), Haliç Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul.
- [52] **Url-11** <<https://maker.robotistan.com>>, erişim tarihi 06.10.2023.
- [53] **Semiconductors, N. X. P.** (2010). PCA9685 16-channel, 12-bit PWM Fm+ I2C-bus LED controller. *Document identifier: PCA9685, Rev, 3(2)*.
- [54] **Url-12** <<https://www.robotistan.com>>, erişim tarihi 18.03.2024.
- [55] **Url-13** <<https://cdn-shop.adafruit.com>>, erişim tarihi 18.03.2024.
- [56] **Çağlay, E., İlk, HG, Özel, RM, & Yılmaz, AE.,** (2005, Mayıs). Karar destek Mekanizmalarında Özellik Fuzyonu İçin Çoklu Sensör Özeti. *IEEE 13. Sinyal İşleme ve İletişim Uygulamaları Konferansı Bildirileri*, 2005. (s. 380-383). IEEE.DOI: 10.1109/SIU.2005.1567700.
- [57] **Çengiz, B., Daş, R.,** (2022). Veri Füzyonu: Veri Kaynakları, Mimariler, Zorluklar ve Çözüm Yaklaşımları, *Fırat Üniversitesi Mühendislik Bilimleri Dergisi*, 34(2), 899-922.
- [58] **Bölük, N., Uçar, Ö., Inner, A. B.,** (2019). Mobil Robotlarda Navigasyon Problemi İçin Pekiştirmeli Öğrenme, *Türkiye Robotbilim Konferansı*,40-44.
- [59] **Url-14** <<https://www.researchgate.net>>, erişim tarihi 9.04.2024.
- [60] **Yaşar, E.,** (2020). Sürü Robotların Hareket Planlamada Kullanılması, *European Journal of Science and Technology*, No. 20, 24-29.
- [61] **Coskun, İ., Isık M.F.,**(2004) Servo Motorun Mikrodenetleyici ile Konum Ve Hız Denetimi, *Gazi Üniversitesi Fen Bilimleri Dergisi*, 17(3), 115-125.
- [62] **Top, A.,** (2016). *Servo Sistemlerin Kontrolü İçin Endüstriyel Amaçlı Bir Mikrokontrolör Tasarımı* (Yüksek Lisans Tezi), Fırat Üniversitesi, Fen Bilimleri Enstitüsü, Elazığ.
- [63] **Url-15** <<https://openaccess.firat.edu.tr>>, erişim tarihi 10.04.2024.
- [64] **Gürgüze, G., Türkoğlu, İ.,** (2019). Kullanım Alanlarına Göre Robot Sistemlerinin Sınıflandırılması, *Fırat Üniversitesi Mühendislik Bilimleri Dergisi* 31(1), 53-66.
- [65] **Yıldırım, A., Kuncan, M., Yıldırım, B.,** (2023). Raspberry Pi ile Uzaktan Kontrollü Bomba İmha Robotu, *Disiplinlerarası Güncel Mühendislik Çalışmaları*, 2.
- [66] **Url-16** <<https://iksadyayinevi.com>>, erişim tarihi 11.04.2024.

- [67] **Görkem, L., Bozuklu, M.,** (2016). Nesnelerin İnterneti: Yapılan Çalışmalar ve Ülkemizdeki Mevcut Durum, *Gaziosmanpaşa Bilimsel Araştırma Dergisi*, 2146-8168 (13), 47-68.
- [68] **Toylan, H., Türkeş, E., Çağlar, E.,** (2017). Real-Time Control Of Mobile Robot Using Hmm-Based Speech Recognition System, *Anadolu University Journal of Science and Technology A- Applied Sciences and Engineering* 18(5), 897-907.
- [69] **Canbaz, S., Erdemir, G.,**(2021) Açık Kaynak Kodlu Gerçek Zamanlı İşletim Sistemlerinin İncelenmesi, *İstanbul Sabahattin Zaim Üniversitesi Fen Bilimleri Enstitüsü Dergisi*, 3(1), 30-37.
- [70] **Kutlu, E., Gençtürk, C.,**(2021). Uydu Uçuş Yazılımında Gerçek Zamanlı İşletim Sistemleri Zamanlama Algoritmaları, *Türkiye Bilişim Vakfı Bilgisayar Bilimleri ve Mühendisliği Dergisi*, 14(1),1-14.
- [71] **Altun, M., Nacar, M., Çakar, O.,**(2022). Yapay Zeka ve Makine Öğrenmesi, *TC Milli Eğitim Bakanlığı*, 210.
- [72] **Url-17**<<https://www.researchgate.net>>, erişim tarihi 15.04.2024.
- [73] **Okutkan, O.** (2006). *Yapay Zeka ile Mobil Robot Kontrolü* (Yüksek Lisans Tezi), İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul.



EKLER

EK A. Projede Yazılan Kod

```
// Gerekli kütüphaneleri dahil et (sensörler, iletişim ve servo kontrol için)

#include <NewPing.h> // Ultrasonik sensör için

#include <QTRSensors.h> // Çizgi izleme sensör dizisi için

#include <Wire.h> // I2C iletişimi için

#include <Adafruit_PWMServoDriver.h> // PWM aracılığıyla servo motorlarını kontrol etmek için

// Pin numaraları, parametreler ve ayarlar için sabitler tanımla

#define TRIGGER_PIN A3 // Ultrasonik sensörü tetiklemek için kullanılan pin

#define MAX_DISTANCE 300 // Ultrasonik ölçümler için maksimum mesafe (cm cinsinden)

#define LDR_PIN A7 // LDR'nin bağlandığı pin

#define DELTA_THRESHOLD 200 // LDR değişikliği için eşik değeri

#define IGNORE_TIME 10000 // Motorların yeniden başlamasını önlemek için geçersiz sayılacak zaman (ms)

// Motor hızları ve pinleri için sabitler

#define rightMaxSpeed 93 // Sağ motor için maksimum hız

#define leftMaxSpeed 93 // Sol motor için maksimum hız

#define rightBaseSpeed 57 // Sağ motor için temel hız

#define leftBaseSpeed 57 // Sol motor için temel hız

#define rightMotor1 12 // Sağ motor için ileri yönlendirme pini
```

```

#define rightMotor2 10 // Sağ motor için geri yönlendirme pini

#define rightMotorPWM 11 // Sağ motor hız kontrol pini

#define leftMotor1 8 // Sol motor için ileri yönlendirme pini

#define leftMotor2 7 // Sol motor için geri yönlendirme pini

#define leftMotorPWM 9 // Sol motor hız kontrol pini

Adafruit_PWMServoDriver pwm = Adafruit_PWMServoDriver(); // PWM servo
sürücüsü nesnesi oluştur

// Servo motorlar için min-max darbe uzunlukları ve frekans ayarı

#define SERVOMIN 150 // Servo için minimum darbe uzunluğu

#define SERVOMAX 600 // Servo için maksimum darbe uzunluğu

#define SERVO_FREQ 50 // Analog servolar için genellikle 50 Hz kullanılır

const uint8_t SensorCount = 8; // Sensör sayısı

uint16_t sensorValues[SensorCount]; // Sensör değerlerini saklamak için dizi

int lastError = 0; // Son hata değeri

QTRSensors qtr; // Çizgi izleme sensörü nesnesi

// PID Katsayıları

float Kp = 0.09; // Oransal katsayı

float Ki = 0.0008; // İntegral katsayı

float Kd = 0.7; // Türev katsayı

NewPing sonar(TRIGGER_PIN, TRIGGER_PIN, MAX_DISTANCE); // Ultrasonik
sensör nesnesi

int initialLDRValue = 0; // İlk LDR değeri

unsigned long lastMotorStart = 0; // Motorların son çalışma zamanı

bool motorRunning = true; // Motor çalışma durumu

```

```
void setup() {  
  
    Serial.begin(9600); // Seri haberleşmeyi başlat  
  
    pwm.begin(); // PWM başlat  
  
    pwm.setPWMFreq(SERVO_FREQ); // Servo frekansını ayarla  
  
    delay(1000); // 1 saniye bekle  
  
    // İlk servo açılarını ayarla  
  
    setServoAngle(0, 85);  
  
    delay(1000); // 1 saniye bekle  
  
    setServoAngle(1, 100);  
  
    delay(1000); // 1 saniye bekle  
  
    setServoAngle(2, 60);  
  
    delay(1000); // 1 saniye bekle  
  
    setServoAngle(3, 50);  
  
    delay(1000); // 1 saniye bekle  
  
    pinMode(LDR_PIN, INPUT); // LDR pinini giriş olarak ayarla  
  
    pinMode(rightMotor1, OUTPUT); // Motor pinlerini çıkış olarak ayarla  
  
    pinMode(rightMotor2, OUTPUT);  
  
    pinMode(rightMotorPWM, OUTPUT);  
  
    pinMode(leftMotor1, OUTPUT);  
  
    pinMode(leftMotor2, OUTPUT);  
  
    pinMode(leftMotorPWM, OUTPUT);  
  
    initialLDRValue = analogRead(LDR_PIN); // İlk LDR değerini oku  
  
    lastMotorStart = millis(); // Son motor başlama zamanını kaydet
```

```

qtr.setTypeRC(); // Sensör tipini ayarla

qtr.setSensorPins((const uint8_t[]){A1, A2, A0, 6, 5, 2, 4, 3}, SensorCount); // Sensör
pinlerini ayarla

calibration(); // Kalibrasyon yap

}

void loop() {

if (millis() - lastMotorStart > IGNORE_TIME) { // Belirli bir süre motorlar çalışmadıysa

    unsigned int distance = sonar.ping_cm(); // Mesafe ölç

    if (motorRunning && distance > 0 && distance <= 7) { // Engel varsa motorları durdur

        motorControl(false);

    }

}

if (!motorRunning) { // Motorlar durduysa

    int currentLDRValue = analogRead(LDR_PIN); // LDR değerini oku

    if (currentLDRValue > 300) { // LDR değeri belirli bir değerden büyükse

        al(); // Nesneyi al

        delay(5000); // 5 saniye bekle

    } else {

        bırak(); // Nesneyi bırak

        delay(5000); // 5 saniye bekle

    }

    checkLDRChange(); // LDR değişikliğini kontrol et

}
}

```

```

if (motorRunning) { // Motorlar çalışıyorsa

    motorControl(true); // Motor kontrolünü sağla

}

}

void checkLDRChange() { // LDR değişikliğini kontrol et

    int currentLDRValue = analogRead(LDR_PIN); // LDR değerini oku

    if (abs(currentLDRValue - initialLDRValue) >= DELTA_THRESHOLD) { // Eşik
değerinden büyük bir değişiklik varsa

        Serial.println("Significant LDR change detected - Evaluating action");

        initialLDRValue = currentLDRValue; // Başlangıç LDR değerini güncelle

        motorControl(true); // Motorları yeniden başlat

    }

}

void motorControl(bool state) { // Motor kontrol fonksiyonu

    motorRunning = state; // Motor durumunu ayarla

    if (!state) { // Eğer motor durdurulduysa

        digitalWrite(rightMotor1, LOW); // Motorları durdur

        digitalWrite(rightMotor2, LOW);

        digitalWrite(leftMotor1, LOW);

        digitalWrite(leftMotor2, LOW);

    } else { // Motorlar çalışıyorsa

        uint16_t position = qtr.readLineBlack(sensorValues); // Çizgi pozisyonunu oku

        int error = position - 3500; // Hata değerini hesapla

```

```

int P = error; // Oransal hata

static int I = I + error; // İntegral hata

int D = error - lastError; // Türev hata

lastError = error; // Son hatayı güncelle

int motorSpeed = P * Kp + I * Ki + D * Kd; // Motor hızını PID katsayıları ile hesapla

int rightMotorSpeed = rightBaseSpeed + motorSpeed; // Sağ motor hızını ayarla

int leftMotorSpeed = leftBaseSpeed - motorSpeed; // Sol motor hızını ayarla

rightMotorSpeed = constrain(rightMotorSpeed, 0, rightMaxSpeed); // Sağ motor
hızını sınırla

leftMotorSpeed = constrain(leftMotorSpeed, 0, leftMaxSpeed); // Sol motor hızını
sınırla

forward_brake(rightMotorSpeed, leftMotorSpeed); // İleri hareketi sağla
}

}

void calibration() { // Kalibrasyon fonksiyonu

for (uint16_t i = 0; i < 400; i++) { // Belirli sayıda kalibrasyon döngüsü yap

qtr.calibrate(); // Kalibre et

}

}

void forward_brake(int posa, int posb) { // İleri hareket fonksiyonu

digitalWrite(rightMotor1, HIGH); // Sağ motoru ileri harekete geçir

digitalWrite(rightMotor2, LOW);

digitalWrite(leftMotor1, HIGH); // Sol motoru ileri harekete geçir

digitalWrite(leftMotor2, LOW);

```

```

    analogWrite(rightMotorPWM, posa); // Sağ motor PWM ayarla

    analogWrite(leftMotorPWM, posb); // Sol motor PWM ayarla

}

void setServoAngle(uint8_t n, double angle) { // Servo açısını ayarla

    double pulseLength = map(angle, 0, 180, SERVOMIN, SERVOMAX); // Açıyı darbe
    uzunluğuna çevir

    pwm.setPWM(n, 0, pulseLength); // PWM sinyali ayarla

}

void moveServoSlowly(uint8_t n, double startAngle, double endAngle, int stepDelay) {
    // Servoyu yavaşça hareket ettir

    if (startAngle < endAngle) { // Başlangıç açısı bitiş açısından küçükse

        for (double angle = startAngle; angle <= endAngle; angle++) { // Açıyı arttırarak döngü

            setServoAngle(n, angle); // Açıyı ayarla

            delay(stepDelay); // Belirli süre bekle

        }

    } else { // Başlangıç açısı bitiş açısından büyükse

        for (double angle = startAngle; angle >= endAngle; angle--) { // Açıyı azaltarak döngü

            setServoAngle(n, angle); // Açıyı ayarla

            delay(stepDelay); // Belirli süre bekle

        }

    }

}

void al() { // Nesneyi alma fonksiyonu

    Serial.println("al çalıştı");

```

```

delay(2000);

moveServoSlowly(0, 85, 150, 20); // 20 ms gecikme ile servoyu hareket ettir

delay(2000); // 2 saniye bekle

    moveServoSlowly(1, 100, 150, 20); // 20 ms gecikme ile servoyu hareket ettir

delay(2000); // 2 saniye bekle

moveServoSlowly(2, 60, 100, 20); // 20 ms gecikme ile servoyu hareket ettir

delay(2000); // 2 saniye bekle

moveServoSlowly(3, 50, 0, 20); // 20 ms gecikme ile servoyu hareket ettir

delay(2000); // 2 saniye bekle

moveServoSlowly(1, 150, 100, 20); // 20 ms gecikme ile servoyu hareket ettir

delay(2000); // 2 saniye bekle

moveServoSlowly(2, 100, 60, 20); // 20 ms gecikme ile servoyu hareket ettir

delay(2000); // 2 saniye bekle

moveServoSlowly(0, 150, 85, 20); // 20 ms gecikme ile servoyu hareket ettir

delay(2000); // 2 saniye bekle

moveServoSlowly(3, 0, 0, 20); // 20 ms gecikme ile servoyu hareket ettir

delay(2000); // 2 saniye bekle

}

void birak() { // Nesneyi bırakma fonksiyonu

    Serial.println("birak çalıştı");

    delay(2000);

moveServoSlowly(0, 85, 150, 20); // 20 ms gecikme ile servoyu hareket ettir

delay(2000); // 2 saniye bekle

```

```
moveServoSlowly(1, 100, 150, 20); // 20 ms gecikme ile servoyu hareket ettir
delay(2000); // 2 saniye bekle

moveServoSlowly(2, 60, 100, 20); // 20 ms gecikme ile servoyu hareket ettir
delay(2000); // 2 saniye bekle

moveServoSlowly(3, 50, 0, 20); // 20 ms gecikme ile servoyu hareket ettir
delay(2000); // 2 saniye bekle

moveServoSlowly(1, 150, 100, 20); // 20 ms gecikme ile servoyu hareket ettir
delay(2000); // 2 saniye bekle

moveServoSlowly(2, 100, 60, 20); // 20 ms gecikme ile servoyu hareket ettir
delay(2000); // 2 saniye bekle

moveServoSlowly(0, 150, 85, 20); // 20 ms gecikme ile servoyu hareket ettir
delay(2000); // 2 saniye bekle

moveServoSlowly(3, 0, 0, 20); // 20 ms gecikme ile servoyu hareket ettir
delay(2000); // 2 saniye bekle

}
```

EK B. Robot Özellik Tabloları

ÇİZGİ TAKİP YÖNTEMLERİNİN KARŞILAŞTIRILMASI												
KRİTER	SAGLAMLIK	ÇEVRE KOŞULLARINDAN BAĞIMSIZ ÇALIŞMA	MALİYET	BAKIM	İŞ GÜVENLİĞİ	MÜDHALE KOLAYLIĞI	DONANIM TEDARİK KOLAYLIĞI	YAZILIM GELİŞTİRİLEBİLİRLİĞİ	TOLERANS	HASSASİYET	D ₁	D ₂
KATSAYI	3	1	3	3	1	2	2	1	2	2		
CCD	0	-1	0	1	-1	-1	0	1	1	1	8	-4
ULTRASONİK	0	0	0	1	-1	-1	1	0	1	-1	7	-5
MANYETİK	-1	0	1	1	0	1	0	0	0	0	8	-3
OPTİK	1	1	1	1	0	1	1	0	-1	0	14	-2
											4	12

Tablo B.1. Çizgi Takip Yöntemlerini Karşılaştırılması [1]

ROBOT HAREKET YÖNTEMLERİNİN KARŞILAŞTIRILMASI											
ÖZELLİKLER	HAREKET KAPABİLİYETİ	MALZEME TAŞIMA KAPASİTESİ	STABİL VE DENGELİ ÇALIŞMA	SAĞLAMLIK	MALİYET	HASSASİYET	MÜDAHALE KOLAYLIĞI	BAKIM	ÇEVRE KOŞULLARINDAN BAĞIMSIZ ÇALIŞMA	YÜKSEKLİK (Z) EKSENİNDE ÇALIŞMA	Z
KATSAYI	4	2	3	3	3	2	2	2	1	1	
PALETLİ	-1	1	1	1	0	0	0	-1	1	-1	2
TEKERLEKLİ	1	0	1	0	1	1	1	1	0	-1	15
BACAĞLI	1	0	0	-1	-1	0	0	0	1	1	0
EKLEMSEL YAPILI	1	0	0	-1	-1	0	0	0	1	1	0

Tablo B.2. Robot Hareket Yöntemlerini Karşılaştırılması [1]

ÖZELLİKLER	KAT SAYI	ACKERMAN SÜRÜŞ YÖNTEMİ	SENKRON SÜRÜŞ YÖNTEMİ	3 TEKERLEKLİ SÜRÜŞ YÖNTEMİ	DİFERANSİYEL SÜRÜŞ YÖNTEMİ
HAREKET KABİLİYETİ	4	0	1	-1	1
MALZEME TAŞIYMA KAPASİTESİ	2	1	1	0	0
STABİL VE DENGELİ ÇALIŞMA	2	1	1	0	0
SAĞLAMLIK	3	1	0	0	0
MALİYET	3	0	-1	1	1
KONTROL SİSTEMLERİNİN KARMASIZLIĞI	3	-1	-1	0	1
HAREKET SİSTEMLERİNİN KARMASIZLIĞI	3	-1	-1	1	0
BAKIM	2	0	-1	1	1
ÇEVRE KOŞULLARINDAN BAĞIMSIZ ÇALIŞMA	1	-1	1	0	1
Σ		0	-2	4	13

Tablo B.3. Robot Sürüş Yöntemlerini Karşılaştırılması [1]

MİKRO İŞLEMCİLERİN KARŞILAŞTIRILMASI											
ÖZELLİKLER	PROGRAM HAFIZASI	MONTAJ DEMONTAJ KOLAYLIĞI	BAĞLANTI KART YAPISI	MALİYET	PROTOTİP ÜRETİMİNE UYGUNLUĞU	PROGRAMLAMA KOLAYLIĞI	TEDARİK EDİLEBİLİRLİK	MUDAHALE EDİLEBİLİRLİK	ELEKTRONİK NİTELİKLERİ		
KATSAYI	3	2	3	3	4	3	2	2	2		
PIC MİKRO İŞLEMCİ	-1	-1	-1	1	1	1	1	1	0	6	
ARDUİNO	0	1	1	0	1	1	1	1	1	18	
PLC	1	0	-1	-1	-1	0	0	0	1	-5	

Tablo B.4. Mikro İşlemcilerin Karşılaştırılması [1]

ÖZGEÇMİŞ

Kişisel Bilgiler		
Adı Soyadı		
Doğum Yeri		
Doğum Tarihi		
Uyruğu		
Telefon		
E-Posta Adresi		

Eğitim Bilgileri	
Lisans	
Üniversite	
Fakülte	
Bölümü	
Mezuniyet Yılı	

Yüksek Lisans	
Üniversite	
Enstitü Adı	
Anabilim Dalı	
Program Adı	

Yüksek Lisans	
Üniversite	
Enstitü Adı	
Anabilim Dalı	
Programı	