



T.C.  
NECMETTİN ERBAKAN NİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ



**ENDÜSTRİYEL AMAÇLI  
KÜRESEL ROBOT KOLU VE TUTUCUNUN  
TASARIMI VE KONTROLÜ**

**Muhammet Mevlüt KARACA**

**YÜKSEK LİSANS TEZİ**

**Makine Mühendisliği Anabilim Dalı**

**Mayıs-2020  
KONYA  
Her Hakkı Saklıdır**

## TEZ KABUL VE ONAYI

Muhammet Mevlüt KARACA tarafından hazırlanan “Endüstriyel Amaçlı Küresel Robot Kolu ve Tutucunun Tasarımı ve Kontrolü” adlı tez çalışması 14/05/2020 tarihinde aşağıdaki jüri tarafından oy birliği / oy çokluğu ile Necmettin Erbakan Üniversitesi Fen Bilimleri Enstitüsü Necmettin Fen Bilimleri Enstitüsü Makine Mühendisliği Anabilim Dalı'nda YÜKSEK LİSANS olarak kabul edilmiştir.

### Jüri Üyeleri

**Başkan**  
Prof. Dr. Ziya ŞAKA

**Danışman**  
Dr. Öğr. Üyesi Ziya ÖZÇELİK

**Üye**  
Dr. Öğr. Üyesi Ümit ÖNEN

### İmza


Fen Bilimleri Enstitüsü Yönetim Kurulu'nun .../.../20.. gün ve ..... sayılı kararıyla onaylanmıştır.

Prof. Dr. S. Savaş DURDURAN  
FBE Müdürü

Bu tez çalışması BAP tarafından 171231004 nolu proje ile desteklenmiştir.

## **TEZ BİLDİRİMİ**

Bu tezdeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edildiğini ve tez yazım kurallarına uygun olarak hazırlanan bu çalışmada bana ait olmayan her türlü ifade ve bilginin kaynağına eksiksiz atıf yapıldığını bildiririm.

## **DECLARATION PAGE**

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Muhammet Mevlüt KARACA

Tarih: 14/05/2020

## ÖZET

### YÜKSEK LİSANS TEZİ

### ENDÜSTRİYEL AMAÇLI KÜRESEL ROBOT KOLU VE TUTUCUNUN TASARIMI VE KONTROLÜ

**Muhammet Mevlüt KARACA**

**Necmettin Erbakan Üniversitesi Fen Bilimleri Enstitüsü  
Makine Mühendisliği Anabilim Dalı**

**Danışman: Dr.Öğr.Üyesi Ziya ÖZÇELİK**

**2020, 151 Sayfa**

**Jüri**

**Dr. Öğr. Üyesi Ziya ÖZÇELİK  
Prof. Dr. Ziya ŞAKA  
Dr. Öğr. Üyesi Ümit ÖNEN**

Endüstrinin önemli bir parçası olan robot kolları teknoloji ile paralel bir şekilde gelişim göstermektedir. Genellikle tekrarlı işlerde ve insan sağlığını riske atacak alanlarda çalışan robot kolları endüstride kalite ve hızı arttırmaktadır. Bu çalışmada 2 eksenli, küresel ve özel silindirik ve kulaklı cisimleri kavramak için geliştirilen pnömatik uç işlevcili (gripper) robot kolunun tasarımı yapıldı, CAD programında montajı yapılarak simülasyonu yapıldı. Montaj hataları giderildikten sonra imalatı yapıldı. Mekanik ve elektriksel donanımın montajı tamamlandı. Sonra mikrodenetleyici bağlantıları yapılarak robot kolu çalışır hale getirildi. Uç işlevcinin koordinat değerlerinin bulunabilmesi ve robot kolunun düz kinematik hesapları için Denavit-Hartenberg yöntemi kullanıldı. Ters kinematik hesaplamaları ile de uç işlevcinin hedef koordinat değerlerine ulaşabilmesi için eklemlerin açı değerleri hesaplandı. Lagrange-Euler yöntemiyle robot kolundaki her bir eklemin torku, robot kolunun genel kütle matrisi, uzuvlara ait atalet tensörü, yerçekimi ivmesi ve Coriolis ve merkezkaç kuvvet vektörleri elde edildi. Lagrange-Euler yöntemiyle bulunan dinamik denklemleri ve yörünge planlaması ile oluşturulan denklemleri Matlab- Simulink programına tanımlandı. Simulink kısmında grafikleri çizdirildi. Solidworks programında tasarlanan katı model, Matlab- SimMechanics'e yüklenerek görsel simülasyonu gerçekleştirildi. Arduino-Uno mikrodenetleyicisinin yazılımı ile robot kolun kontrol programı yazıldı. Elde edilen kinematik denklemler ve planlanan yörünge denklemleri program içine yerleştirildi. Robot kolu sürücüsüne bağlanan Arduino'dan Matlab ile seri(com) port kullanılarak enkoder verileri okundu. Programdaki denklemlerin sayesinde mikro denetleyici ile robot kolunun kontrolü sağlandı. Mikrodenetleyici ile kontrolü sağlanan robot kolunun tekrarlanabilirlik ve doğruluk testi yapıldı. Deneysel sonuçlar esnasında enkoderden okunan eş zamanlı veriler kaydedildi. Bu veriler ile teorik veriler karşılaştırıldı.

**Anahtar Kelimeler:** Endüstriyel Robot Kolu, Küresel Robot, Robot Dinamiği, Robot Kinematığı, Uç İşlevci Tasarımı, Yörünge Planlaması

## **ABSTRACT**

### **MS THESIS**

#### **DESIGN AND CONTROL OF INDUSTRIAL PURPOSE SPHERICAL ROBOT ARM AND GRIPPER**

**Muhammet Mevlüt KARACA**

**THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCE OF  
NECMETTİN ERBAKAN UNIVERSITY  
THE DEGREE OF MASTER OF SCIENCE  
IN MECHANICAL ENGINEERING**

**Advisor: Dr. Ziya ÖZÇELİK**

**2020, 151 Pages**

**Jury**

**Advisor Dr. Ziya ÖZÇELİK**

**Prof. Dr. Ziya ŞAKA**

**Dr. Ümit ÖNEN**

Robot arms, which are an important part of the industry, are developing in parallel with technology. Robot arms, which generally work in repetitive jobs and in areas that will put human health at risk, increase the quality and speed in the industry. In this study, the robot arm with pneumatic end effector (gripper) developed to grasp 2-axis, spherical and special cylindrical and eared bodies was designed, it was assembled in CAD program and simulated. It was manufactured after fixing assembly errors. Installation of the mechanical and electrical equipment has been completed. Then, by making microcontroller connections, the robot arm was made operational. The Denavit-Hartenberg method was used to find the coordinate values of the end effector and the straight kinematic calculations of the robot arm. Inverse kinematic calculations, the angle values of the joints were calculated so that the end operator can reach the target coordinate values. With the Lagrange-Euler method, the torque of each joint in the robot arm, the general mass matrix of the robot arm, the inertial tensor of the limbs, the gravitational acceleration, and the Coriolis and centrifugal forces vectors were obtained. Dynamic equations found using Lagrange-Euler method and equations created by path planning are defined in Matlab-Simulink program. Graphics were drawn in the Simulink section. The solid model designed in the Solidworks program was loaded into Matlab- SimMechanics and its visual simulation was performed. The control program of the robot arm was written with the software of the Arduino-Uno microcontroller. The obtained kinematic equations and planned trajectory equations were placed in the program. Encoder data was read by using serial (com) port with Matlab from Arduino connected to the robot arm driver. Thanks to the equations in the program, the control of the robot arm was achieved with the micro controller. Repeatability and accuracy testing of the robot arm, controlled by a microcontroller, was performed. Simultaneous data read from the encoder was recorded during the experimental results. These data and theoretical data were compared.

**Keywords:** Cylindrical Robot, Gripper Design, Industrial Robot Arm, Path Planning, Robot Dynamics, Robot Kinematics

## ÖNSÖZ

Bu çalışmada endüstride kullanılan robot kollarının tasarım, imalat ve analiz süreci ortaya konmuştur. BAP projesi kapsamında olan bu çalışma Necmettin Erbakan Üniversitesi Seydişehir Ahmet Cengiz Mühendislik Fakültesi Makine Mühendisliği Bölümü Makine Teorisi ve Dinamiği Anabilim dalı başkanlığı bünyesinde gerçekleştirilmiştir.

Çalışmalarımın tüm aşamalarında değerli bilgi ve yardımlarını benden esirgemeyen danışman hocam Dr. Öğr. Üyesi Ziya Özçelik'e, çalışma boyunca teşvik ve yönlendirmelerinden oldukça istifade ettiğim arkadaşım Hasan Özçelik'e, maddi-manevi hiçbir desteğini esirgemeyen beni yüreklendirip emek veren sevgili eşime, bugünlere gelmemde büyük pay sahibi olan değerli anne ve babama şükranlarımı sunarım.

Muhammet Mevlüt KARACA  
KONYA-2020

## İÇİNDEKİLER

ÖZET .....	iv
ABSTRACT.....	v
ÖNSÖZ .....	vi
İÇİNDEKİLER.....	vii
ŞEKİLLER LİSTESİ .....	ix
TABLolar LİSTESİ .....	xiii
SİMGELER VE KISALTMALAR.....	xiv
<b>1. GİRİŞ.....</b>	<b>1</b>
<b>2. KAYNAK ARAŞTIRMASI .....</b>	<b>3</b>
<b>3. ROBOT KOLU TEMEL KAVRAMLAR.....</b>	<b>6</b>
3.1. Robot Kolu Tanımı ve Tarihi.....	6
3.2. Endüstriyel Robot Sistemleri ve Kullanım Alanları .....	7
3.3. Endüstriyel Robot Kolu Temel Bileşenler .....	11
3.3.1. Manipülator .....	11
3.3.2. Uç İşlevci (Gripper).....	15
3.3.3. Aktüatörler.....	18
3.3.4. Sensörler .....	27
3.3.5. Kontrol Ünitesi .....	27
3.3.6. Kontrol ve programlama el cihazı (Teach Pendant).....	29
3.4. Endüstriyel Robotların Sınıflandırılması .....	30
3.4.1. Kartezyen robotlar .....	31
3.4.2. Silindirik robotlar .....	32
3.4.3. Küresel robotlar .....	32
3.4.4. SCARA robotlar .....	33
3.4.5. Dikey mafsallı robotlar.....	33
3.4.6. Paralel robotlar .....	34

3.5. Endüstriyel Robotlarda Önemli Parametreler .....	34
3.5.1. Hassasiyet .....	34
<b>4. MATERYAL VE YÖNTEM.....</b>	<b>36</b>
4.1. Robot Kolu Tasarımı .....	36
4.1.1. Dış gövde sanal ve fiziksel montaj .....	37
4.1.2. Malzeme seçimi .....	44
4.1.3. Motor hesaplamaları .....	44
4.1.4. Pano ve mikrodenetleyicinin montajı .....	47
4.2. Kinematik Modelleme .....	49
4.2.1. Düz (İleri) kinematik .....	51
4.2.2. Ters kinematik .....	59
4.2.3. Yörünge planlaması .....	66
4.3. Dinamik Modelleme .....	70
4.4. Tasarlanan Robot Kolunun Blok Diyagramları .....	89
<b>5. ARAŞTIRMA SONUÇLARI VE TARTIŞMA.....</b>	<b>91</b>
5.1. Mikroişlemci (Arduino Uno) ile Matlab Seri Haberleşme .....	91
5.1.1. PID katsayılarının belirlenmesi .....	92
5.2. Enkoderden Veri Alma .....	93
<b>6. SONUÇLAR VE ÖNERİLER .....</b>	<b>98</b>
<b>KAYNAKLAR .....</b>	<b>101</b>
<b>EKLER .....</b>	<b>108</b>
<b>ÖZGEÇMİŞ .....</b>	<b>135</b>



## ŞEKİLLER LİSTESİ

Şekil 3.1. Filli su saati (“Al Jazari-The Elephant Clock”, t.y.) .....	6
Şekil 3.2. Unimate endüstriyel robotu (“George Devol”, t.y.) .....	7
Şekil 3.3. 2018 yılındaki imalat sanayisindeki robot yoğunluğu (“IFR”, t.y.).....	8
Şekil 3.4. Sektörlere göre yıllık (2016-2018) endüstriyel robot kurulumu (“IFR”, t.y.)..	8
Şekil 3.5. 2013-2022 yıllık endüstriyel robot kurulumu yıllara göre artış miktarı (“IFR”, t.y.).....	9
Şekil 3.6. Otomatik yönlendirmeli araç sistemi (Micieta, Zavodska, Rakyta ve Binasova, 2015) .....	9
Şekil 3.7. Endüstriyel robot kolu temel bileşenler (“Awareness In The Fields of Robotics”, 2015) .....	11
Şekil 3.8. Robot kolu ve insan kolu.....	12
Şekil 3.9. Kinematik çiftler.....	12
Şekil 3.10. Döner ve prizmatik eklem çiftleri (Dişlitaş, 2015).....	13
Şekil 3.11. Robot kolu çalışma alanı (“Industrial Robots”, t.y.) .....	14
Şekil 3.12. Mitsubishi PA10-7CE (“Mitsubishi PA10-7CE robot”, t.y.).....	14
Şekil 3.13. Gereğinden fazla serbestlik derecesi (“Robot Teknolojisi Ders Notları”, t.y.) .....	15
Şekil 3.14. Endüstride kullanılan kısaçap tipi uç işlevci (“Grasp, Sight, and Picking Evolve In Autonomous Robots”, 2019).....	15
Şekil 3.15. Pnömatik aktüatör sistemi .....	16
Şekil 3.16. Açısal ve paralel pnömatik uç işlevci (Bernier, t.y.) .....	17
Şekil 3.17. Pnömatik silindir temel bileşenler (“Pneumatic Cylinders”, t.y.) .....	17
Şekil 3.18. Tek ve çift etkili silindir .....	17
Şekil 3.19. Mafsal aktüatör sisteminde gücün dağılımı.....	18
Şekil 3.20. Servo mekanizması yapılandırılması.....	20
Şekil 3.21. Standart bir servo motor yapısı (“Servo Motor Technology”, t.y.).....	21
Şekil 3.22. Servo motor eklentisi enkoder(“What is an Encoder?”, t.y.) .....	22
Şekil 3.23. Alternatif veya doğru akımla beslenen servo motor (“Servo Motor”, t.y.)..	22
Şekil 3.24. Fırçalı veya fırçasız servo motorlar (“Servo Motor”, t.y.) .....	23
Şekil 3.25. Senkron Motor (“Servo Motor”, t.y.) .....	23
Şekil 3.26. AC servo bağlantı şeması (“Servo Motor Sürücü Bağlantısı”, t.y.).....	25
Şekil 3.27. Bant, kablo, kayış ve zincir tahrikleri (Craig, 2004) .....	26

<b>Şekil 3.28.</b> Açık döngü kontrol sistemi .....	28
<b>Şekil 3.29.</b> Kapalı döngü kontrol sistemi (Ben-Ari ve Mondada, 2018).....	29
<b>Şekil 3.30.</b> Kontrol ve programlama el cihazı (“KUKA smartPAD-2”, t.y.).....	29
<b>Şekil 3.31.</b> Döner ve prizmatik mafsalsal sembolik olarak gösterimi (Spong, Hutchinson ve Vidyasagar, 2004) .....	31
<b>Şekil 3.32.</b> Kartezyen robot (a) Kinematik zincir (b) Çalışma uzayı (c) Uygulama örneği EPSON marka Kartezyen robot (Spong ve diğerleri, 2004).....	31
<b>Şekil 3.33.</b> Silindirik robot (a) Kinematik zincir (b) Çalışma uzayı (c) Uygulama örneği Seiko RT3300 Robot (Spong ve diğerleri, 2004) .....	32
<b>Şekil 3.34.</b> Küresel robot (a) Kinematik zincir (b) çalışma uzayı (c) Uygulama örneği Stanford Arm Robotu (Spong ve diğerleri, 2004) .....	32
<b>Şekil 3.35.</b> SCARA robot (a) Kinematik zincir (b) çalışma uzayı (c) Uygulama örneği Epson E2L653S SCARA Robot (Spong ve diğerleri, 2004).....	33
<b>Şekil 3.36.</b> Dikey Mafsallı robot (a) Kinematik zincir (b) çalışma uzayı (c) Uygulama örneği ABB IRB1400 Robot (Spong ve diğerleri, 2004) .....	33
<b>Şekil 3.37.</b> Uygulama örneği ABB IRB940 Tricept Parallel Robotu (Spong ve diğerleri, 2004).....	34
<b>Şekil 3.38.</b> Tekrarlanabilirlik ve Doğruluk: a) düşük tekrarlanabilirlik ve düşük doğruluk, b) düşük tekrarlanabilirlik ve yüksek doğruluk, c) yüksek tekrarlanabilirlik ve düşük doğruluk, d) yüksek tekrarlanabilirlik ve yüksek doğruluk (Joubair, 2014).....	35
<b>Şekil 4.1.</b> Robot kolu genel görünüm .....	36
<b>Şekil 4.2.</b> a) Robot kolu genel sanal görünüm b) Robot kolu patlatılmış montaj görünüm c) Eksenler ve uç işlevci .....	37
<b>Şekil 4.3.</b> Kayış kaskas Sistemi.....	38
<b>Şekil 4.4.</b> Robot kolu dış gövde montajlanmış hali .....	38
<b>Şekil 4.5.</b> Robot kolu dış ebatlar.....	39
<b>Şekil 4.6.</b> Robot kolu bileşenleri teknik resim.....	40
<b>Şekil 4.7.</b> Robot kolu eksenler sınır şartları: a) 1. eksen, b) 2. eksen ve c) genel .....	41
<b>Şekil 4.8.</b> Uç işlevcinin yapısı .....	41
<b>Şekil 4.9.</b> Robot kolu tutucunun hareket esnasında zeminle paralelliği.....	42
<b>Şekil 4.10.</b> Robot kolu farklı uç işlevci seçenekleri .....	42
<b>Şekil 4.11.</b> Robot kolunun Matlab Simulink blok diyagramı.....	43
<b>Şekil 4.12.</b> Robot kolu simülasyonu: a) başlangıç konumu ve b) bitiş konumu .....	43
<b>Şekil 4.13.</b> St 52 malzemeden üretilen robot kolu gövdesi .....	44

Şekil 4.14. Robot kolu maksimum moment oluştuğu pozisyon.....	45
Şekil 4.15. Pano elemanları yerleşim şeması .....	47
Şekil 4.16. Pano montajlanmış hali .....	48
Şekil 4.17. Sürücü ve mikroişlemcinin bağlantı şeması.....	49
Şekil 4.18. Endüstriyel robot koordinat sistemleri (“Simatic S7-1500”, t.y.).....	50
Şekil 4.19. Ters ve düz kinematik örnek (Boggus, t.y.).....	50
Şekil 4.20. P noktasının {A} koordinat sistemine göre tanımlanması(Bingül ve Küçük, 2017a).....	52
Şekil 4.21. Uç işlevcinin yöneliminin referans koordinat sistemine göre tanımlanması (Craig, 2004).....	52
Şekil 4.22. Denavit-Hartenberg kinematik parametreler (“Denavit–Hartenberg Parameters”, 2019).....	54
Şekil 4.23. Tasarlanan robot kolu eksen atamaları.....	55
Şekil 4.24. Robot kolu başlangıç konumu ( $\theta_1 = \theta_2 = 0$ ).....	57
Şekil 4.25. Birinci eksene $-30^\circ$ , ikinci eksene $-60^\circ$ verildiğindeki uç işlevcideki konum .....	59
Şekil 4.26. Eksenler pozitif dönme yönleri .....	59
Şekil 4.27. Tasarlanan robot kolu ters kinematik geometrik gösterim (Spong ve diğerleri, 2004).....	66
Şekil 4.28. Kartezyen ve eklem uzayında hareket.....	67
Şekil 4.29. Üçüncü dereceden kübik yörüngeye ait a) konum b) hız ve c) ivme grafiği (Bingül ve Küçük, 2017a).....	70
Şekil 4.30. Robot kolu i. bağın kinetik enerjisi .....	71
Şekil 4.31. Robot kolu uzuvların potansiyel enerjisi ve yerçekimi ivmesi .....	74
Şekil 4.32. Robot kolunda birinci eksenin kütlesi, kütle merkezi ve atalet momenti ....	76
Şekil 4.33. Robot kolunda ikinci eksenin kütlesi, kütle merkezi ve atalet momenti.....	76
Şekil 4.34. Robot kolunda uç işlevcinin kütlesi, kütle merkezi ve atalet momenti.....	77
Şekil 4.35. Robot kolu Matlab -Simulink benzetim modeli.....	78
Şekil 4.36. Matlab -Simulink yörünge bloğu kodları .....	79
Şekil 4.37. Matlab -Simulink dynNS bloğu kodları-1 .....	80
Şekil 4.38. Matlab -Simulink dynNS bloğu kodları-2.....	81
Şekil 4.39. Birinci ekleme ait a) konum, b) hız ve c) ivme zaman grafiği.....	82
Şekil 4.40. İkinci ekleme ait a) konum, b) hız ve c) ivme zaman grafiği.....	83
Şekil 4.41. Üçüncü ekleme ait a) konum, b) hız ve c) ivme zaman grafiği.....	84

<b>Şekil 4.42.</b> Coriolis ve Merkezkaç Kuvvetleri: a) C1, b) C2 ve c) C3 .....	85
<b>Şekil 4.43.</b> Yer çekimi vektörleri: a) G1, b) G2 ve c) G3 .....	86
<b>Şekil 4.44.</b> Tork değerleri: a) Tork 1, b) Tork 2 ve c) Tork 3 .....	87
<b>Şekil 4.45.</b> Genel kütle matrisi elemanları: a) M11, b) M12, c) M13, d) M21, e) M22, f) M23, g) M31, h) M32 ve ı) M33 .....	88
<b>Şekil 4.46.</b> Robot kolu pnömatik aktüatör blok diyagramı.....	89
<b>Şekil 4.47.</b> Kontrol şeması.....	89
<b>Şekil 4.48.</b> Robot kolu genel blok diyagramı .....	90
<b>Şekil 5.1.</b> Matlab komut ekranı .....	92
<b>Şekil 5.2.</b> $-30^\circ$ ve $-60^\circ$ çalışma aralığında a) konum, b) hız, c) ivme ve d) uç işlevciye ait konum zaman grafiği .....	93
<b>Şekil 5.3.</b> $37^\circ$ ve $-43^\circ$ çalışma aralığında a) konum, b) hız, c) ivme ve d) uç işlevciye ait konum zaman grafiği .....	94
<b>Şekil 5.4.</b> Birinci ve ikinci eksen konum-zaman grafiği (Tekrar Sayısı = 10) ( $\theta_1 = 45^\circ, \theta_2 = -30^\circ$ ).....	95
<b>Şekil 5.5.</b> Birinci ve ikinci eksen konum-zaman grafiği (Tekrar Sayısı = 30) ( $\theta_1 = 45^\circ, \theta_2 = -45^\circ$ ).....	96

## TABLULAR LİSTESİ

<b>Tablo 3.1.</b> Farklı tahrik yöntemlerine göre uç işlevçiler .....	16
<b>Tablo 4.1.</b> Robot kolu bileşenleri ve özellikleri .....	40
<b>Tablo 4.2.</b> St52 çeliğinin mekanik özellikleri (Karanjule, Bhamare ve Rao, 2017) .....	44
<b>Tablo 4.3.</b> St52 çeliğinin kimyasal özellikleri (Karanjule ve diğerleri, 2017).....	44
<b>Tablo 4.4.</b> Kullanılan AC Servo motor özellikleri .....	45
<b>Tablo 4.5.</b> Kullanılan redüktör özellikleri .....	45
<b>Tablo 4.6.</b> AC Servo motor sürücü bilgileri.....	48
<b>Tablo 4.7.</b> Düz ve ters kinematik karşılaştırılması.....	51
<b>Tablo 4.8.</b> D-H yöntemi dört ana değişken .....	54
<b>Tablo 4.9.</b> D-H parametreleri tablosu .....	55
<b>Tablo 4.10.</b> D-H parametreleri tablosu .....	65
<b>Tablo 5.1.</b> PID katsayıları ve sistem cevabı.....	92
<b>Tablo 5.2.</b> Tekrar deneyi-1 hata değerleri .....	96
<b>Tablo 5.3.</b> Tekrar deneyi-2 hata değerleri .....	97

## SİMGELER VE KISALTMALAR

### Simgeler

P	: Güç
r	: Yarıçap
$\theta$	: Konum
$\dot{\theta}$	: Hız
$\ddot{\theta}$	: İvme
$\eta$	: Verim
$\tau$	: Tork
$\emptyset$	: Çap
kW	: Kilowatt
$^{\circ}$	: Derece
$\pm$	: Artı eksi toplam fark işareti
mm	: Milimetre
//	: Paralel
$\omega$	: Açısal hız
M	: Moment
kg	: Kilogram
T	: Tork
F	: Kuvvet
V	: Voltaj
$^{\circ}\text{C}$	: Santigrat derece
${}^{i-1}_i\text{T}$	: Dönüşüm matrisi
${}^A_B\text{R}$	: Yönelim matrisi
${}^A\text{P}_{\text{BORG}}$	: Konum vektörü
{A}	: A koordinat sistemi
$\alpha_{i-1}$	: $Z_{i-1}$ ile $Z_i$ arasında $X_{i-1}$ boyunca ölçülen açı
$a_{i-1}$	: $Z_{i-1}$ ile $Z_i$ arasında $X_{i-1}$ boyunca belirlenen uzunluk
$d_i$	: $X_{i-1}$ ile $X_i$ arasında $Z_i$ boyunca belirlenen uzunluk
$\theta_i$	: $X_{i-1}$ ile $X_i$ arasında $Z_i$ boyunca ölçülen açı
$P_x$	: Uç işlevcinin bulunduğu x koordinatı
$P_y$	: Uç işlevcinin bulunduğu y koordinatı

$P_z$	: Uç işlevcinin bulunduğu z koordinatı
$L_i$	: Uzuv uzunluğu
$t_0$	: Başlangıç zamanı
$t_f$	: Bitiş zamanı
$q$	: Döner eklemlerde açı
$\dot{q}$	: Eklem hızı
$K$	: Kinetik enerji
$P$	: Potansiyel enerji
$L$	: Lagrange operatörü
$X$	: X koordinatı
$Y$	: Y koordinatı
$Z$	: Z koordinatı
$v_i$	: Doğrusal hız
$\omega_i$	: Açısal hız
$m$	: i. bağıın kütlesi
$I$	: i. bağıın kütle merkezinin ana koordinat sistemine göre atalet tensörü
$h_i$	: i. bağıın kütle merkezinin ana koordinat sistemine göre konumu
$\xi_i$	: Eklem tip değişkeni
$z^i$	: i. koordinat sisteminin üçüncü kolon birim vektörü
$J_i$	: Jakobiyen matrisi
$D(q)\ddot{q}$	: Kütle matrisi
$C(q, \dot{q})$	: Coriolis ve Merkezkaç kuvvet vektörü
$G(q)$	: Yerçekimi vektörü
$\tau_i$	: Tork vektörü
ms	: milisaniye

### **Kısaltmalar**

AC	: Alternatif Akım (Alternative Current)
DC	: Doğru Akım (Direct Current)
D-H	: Denavit Hartenberg
GPS	: Küresel Konumlama Sistemi (Global Positioning System)
IFR	: Uluslararası Robotik Federasyonu (International Federation of Robotics)
IP67	: Uluslararası Koruma Kodu 67 (International Protection Code 67)

- ISO : Uluslararası Standartlar Örgütü (International Organization for Standardization)
- JIS : Japonya Standartlar Enstitüsü (Japanese Industrial Standart)
- PID : Oransal İntegral Türevsel denetleyici (Proportional Integral Derivative)
- PLC : Programlanabilir Mantıksal Denetleyici (Programmable Logic Controller)
- PWM : Darbe Geniřlięi Modülasyonu (Pulse Width Modulation)





## 1. GİRİŞ

Endüstrinin vazgeçilmez elemanı olan robot kolları kaliteli ve seri üretim alanında büyük kolaylıklar sağlamaktadır. Kullanım alanı oldukça geniş olan robot kollarını otomotiv, gıda, tekstil, kimya sektörlerinden uzay, tıp ve askeri alanlara kadar kullanımına rastlamak mümkündür. Robot kollarının ileri teknoloji ekipmanı ile donatılmış olması geliştirmekte olan ülkelerin sanayisinde kullanılmasını ve üretimini zorlaştırmaktadır. Dışarıdan alınan hazır olarak satılan sistemlerde maliyet açısından zorlamaktadır. Proses olarak incelendiğinde kaynak, boyama, yapııştırma, montaj, taşıma, paketleme vb. özellikleriyle ön plana çıkmaktadır. Kendi imkanlarımız doğrultusunda ülkemiz sanayisinde üretmek önem taşımaktadır. Bu kapsamda robot kolu tasarım, imalat ve analiz adımları birçok yönüyle ele alınmıştır.

Bu çalışmada iki eksenli, küresel yörünge çizen ve özel pnömatik uç işlevciye sahip endüstriyel robot kolunun tasarımı, imalatı ve analizi yapılmıştır. Bu robot kolu literatürdeki robot kollarından farklı tasarım içermektedir. Bu farklılık tasarımsal olarak özel uç işlevci ve kasnak sistemi olarak iki ayrı noktada ortaya çıkmaktadır. Robot kolu ikinci eksen den itibaren özel kasnak sistemiyle donatılmıştır. Bu sayede uç işlevciye tutulan parçanın zeminle paralelliği sağlanmış olup aynı zamanda bir nevi üçüncü eksen görevi görmektedir. Uç işlevci olarak silindirik ve kulaklı cisimleri kavramak için açışal pnömatik tasarım yapılmıştır. Robot kolunda kullanılan iki adet AC servo motor sistemde eksen hareketlerini elde etmek için kullanıldı ve Arduino Uno mikroişlemcisi ile PID kontrollü olarak PWM sinyali ile sürüldü.

Üç boyutlu katı modelleme programı olan Solidwork programında robot kolu tasarlanmıştır. İmalat ve montajı gerçekleştirilen robot kolunun dış gövdesinde St52 malzeme kullanılmıştır. Tasarımı yapılan robot kolunun İlk olarak kinematik ve dinamik hesaplamaları yapılmıştır. Düz kinematik problemi için Denavit Hartenberg yöntemi kullanılmıştır. Robot kolunun açış değerleri girdi olarak verilip çıkış değeri olarak uç işlevcinin yönelim ve konumu bulunmuştur. Ters kinematikte ise uç işlevcinin yönelim ve konum değeri verilip sisteme verilen açış değeri bulunmuştur. Ters kinematikte analitik ve geometrik yaklaşım ele alınmıştır. Robot kolunun titreşimlerden uzak, rezonansa girmeden ve yumuşak bir şekilde ilerlemesi için yörünge planlaması yapılmıştır. Dinamik denklemlerinin oluşturulmasında Lagrange Euler metoduna

başvurulmuştur. Robot kolu mikroişlemci ile sürüldükten sonra tekrarlanabilirlik ve doğruluk testi deneysel olarak yapılmıştır.

Bu proje kapsamında, sanayinin birçok alanında ihtiyaç duyulan tut yerleştir türü robot kollarını tasarım aşamasında, karşılaşılan problemlerin teorik ve sayısal çözümlerini elde edilmiştir. Çalışma ortamının sağlığa zararlı ve yangına elverişli olduğu ortamlarda dahi çalışabilme özelliğine sahiptir. Ayrıca robot kolu uç işlevcisi geliştirmeye açık olup yapılan işe göre farklı tip uç işlevcilerle de değiştirilerek farklı sanayi uygulamalarında kullanılabilir.



## 2. KAYNAK ARAŞTIRMASI

Homojen dönüşümler ve Lagrange mekaniği formülasyonu kullanımı, karmaşık robot dinamiği denklemlerinin türetilmesini, analizini ve basitleştirilmesini sağlar(Bejczy ve Paul, 1981).

Lagrange-Euler yaklaşımının avantajı, nihai kapalı form denklemini manipülatör ataleti, yerçekimi, sürtünme ve Coriolis ve merkezkaç kuvvetleri gibi terimler kullanarak basitçe fiziksel olarak ifade etmesidir(Schilling, 2003).

Denavit ve Hartenberg 1955 yılında, her bir ekleme koordinat sistemi yerleştirerek komşu eklemler arasındaki dönüşüm ve dönme ilişkisini gösteren matris tanımlamıştır(Fu, Gonzalez ve Lee, 1987).

Mitsubishi firması tarafından geliştirilmiş Movemaster EX (RV-M1) endüstriyel robotun, bilgisayar ortamında 3-boyutlu simülasyonu gerçekleştirilmiştir. Bu simülasyon robot kolunun hareket davranışlarının incelenmesinde, olası hataların tespit edilmesinde ve yapılan değişikliklerin etkisini gözlemlenmesinde kullanılır(Adalı, 2001).

Nesnenin konumu tespit edilip robot kolunun nesneye ulaşabilmesi için gerekli eklem açı değerleri ters kinematik yoluyla hesaplanıp mikroişlemci üzerinden robot koluna gönderilmiştir. Kontrol yazılımı ise Arduino Mega 2560 mikrodenetleyici kartı ile gerçekleştirilmiştir(Yavuz, Alıcı ve Uyar, 2015).

Üç serbestlik dereceli robot koluna kuvvet etkisi altındaki kontrol performansları PID ve bulanık mantık yöntemi ile Matlab -Simulink benzetim ortamında kıyaslanmıştır(Kayışlı ve Uğur, 2017).

Altı serbestlik derecesine sahip endüstriyel robot kolu, kontrol çalışmalarının uygulanabileceği platform olarak CAD ortamında tasarımı yapıldıktan sonra imalat ve montaj aşamaları gerçekleştirildi(Eren ve Kaftanoğlu, 2001).

Scara tipi robotun tasarımı Solidworks programı üzerinde katı modellemesi yapılmıştır. Parçaların CNC işleme kodları çıkartılarak imal edilmiştir. Ters ve düz kinematik hesaplamalar yapılp Matlab programı üzerinde simüle edilmiştir(Saygılı, 2006).

Endüstriyel robot olan ABB firmasına ait IRB 140 robotunun kinematik ve dinamik hesaplamaları ve kontrolleri yapılmıştır. Uyarlamalı kontrol sistemi üzerinde

çalışılmıştır. Denavit-Hartenberg yöntemi Matlab programı üzerinden kullanılmıştır(Öztürk, 2014).

Robotun meyveyi daldan koparma işlemi sırasında meyveye doğru bir şekilde yönelim için kinematik hesaplamalar ve hareket kontrolü yapıldı(Kahya ve Arın, 2015).

Beş serbestlik dereceli robot kolunun dinamik modellemesi yapıldı. Modellemesi yapılan robot kolu 3 uzuvlu ve 5 dönel mafsala sahiptir. Dinamik denklemleri Lagrange-Euler metodu ile matris formunda çıkarıldı. Geri beslemeli endüstride sıkça kullanılan kontrol tipi olan PID kontrol algoritması kullanıldı. Robot kolu uzuvlarına serbest düşme hareketi verilerek PID değerleri doğrulandı(Adar, Ören ve Kozan, 2013).

Puma tipi üç serbestlik dereceli robot kolunun manipülatörüne PID kontrol uygulanmıştır. Öncelikle Denavit-Hartenberg metoduyla düz kinematik hesapları yapılmıştır. Elde edilen denklemlerden matris cebri formuyla ters kinematik denklemleri elde edilmiştir. Dinamik denklemleri Lagrange-Euler metoduyla bulunmuştur. Sikloid, harmonik ve polinom şeklinde yol alma fonksiyonu sisteme giriş olarak verilmiştir. Analiz sonucunda harmonik yol alma fonksiyonunun daha avantajlı olduğu görülmüştür(Duran ve Ankaralı, 2010).

İlk olarak Denavit Hartenberg metoduyla düz kinematik denkleme ilişkin parametreler elde edildi. Sonra robot koluna lineer olmayan (nonlinear) dinamik denklemleri oluşturuldu. Robot koluna denge noktasında doğrusallaştırmaya dayalı model, Matlab / Simulink programı kullanarak ve son olarak robotun geribeslemeli olarak doğrusallaştırması olmak üzere PID üç farklı şekilde uygulandı. Bu üç kontrolör arasında karşılaştırma yapıldı(Mustafa ve Al-Saif, 2014).

Robot uygulamalarında genelde hareket redüktörlü servo motorlar ile sağlanır. Bu çalışmada redüktör yerine itme çekme kuvveti sağlayan rulmanlı vidalı mil kullanıldı. Bu sayede atalet etkisi azalıp robotun kaldıracağı yük miktarı artmıştır. Robot kolunun ileri ve ters kinematik hesaplamaları yapıldı. Matlab/Gui ortamında arayüz oluşturulan robot kolunun verilen yörüngeyi G kodlarıyla yönelimi ve mach3 kartıyla kontrolü gerçekleştirildi(Koç ve Doğan, 2017).

Beş eksenli edubot robotun ters kinematik hesaplamaları yapıldı. Bu sayede robot kolunun ulaşacağı koordinat belirlenmiş ve bu koordinata gitmesi için robota verilmesi gereken açı değerleri bulunmuştur. Robotun hedefe düzgün ve titreşimsiz gidebilmesi için

5. dereceden polinom oluşturularak yörünge planlaması yapılmıştır. Ters kinematik hesaplaması ve yörünge planlaması Matlab 5.02 programında oluşturuldu. Ayrıca robotun hedefe ulaşmasında istenilen noktalardan geçmek koşuluyla yörünge planlaması yapıldı(Tonbul ve Sarıtaş, 2013).

Beş eksenli ve manipülatöre sahip robot kolunun tasarımı Solidworks programında gerçekleştirildi. Robot kolunda hobi amaçlı kullanılan RC servo motor kullanıldı. Ters ve düz kinematik hesaplamaları Matlab Programında yapıldı (Uzuner, 2012).



### 3. ROBOT KOLU TEMEL KAVRAMLAR

#### 3.1. Robot Kolu Tanımı ve Tarihi

Amerika Robot Enstitüsünün kabul görmüş tanımına göre robot, çeşitli görevleri yerine getirmek için programlanmış hareketlerle maddeleri, parçaları, aletleri veya özel cihazları taşımak için tasarlanmış yeniden programlanabilir çok işlevli bir manipulatördür(Siciliano, Sciavicco, Villani ve Oriolo, 2009).



Şekil 3.1. Filli su saati (“Al Jazari-The Elephant Clock”, t.y.)

Robot ve Siberetik biliminin kurucusu olan El-Cezerî(1136-1206) tarafından tasarlanmış ve Artuklu hükümdarına sunulmuş Şekil 3.1’de gösterilen filli su saati, robot alanında ilk çalışma olarak görülebilir(Kumar, 2010).

Endüstriyel robot kolunun en kapsamlı tanımı ISO 8373 stardardına göre, endüstriyel otomasyon uygulamalarında kullanım için sabitlenmiş veya mobil olarak sabitlenebilen otomatik veya kontrollü, yeniden programlanabilir, üç veya daha fazla ekseninde programlanabilir, çok amaçlı manipulatördür(“ISO 8373:2012(en), Robots and robotic devices—Vocabulary”, t.y.).

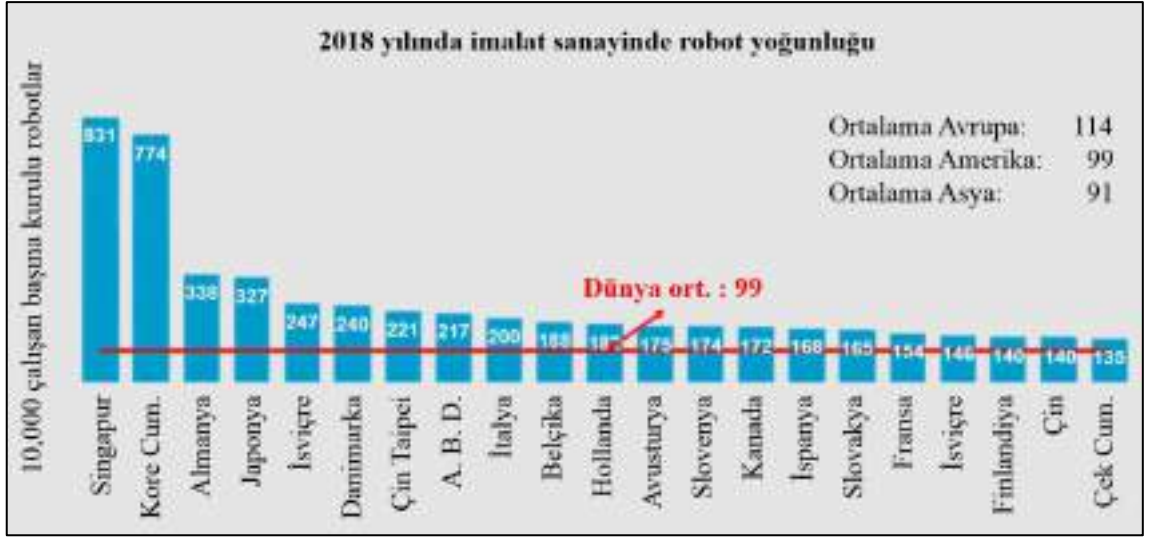


Şekil 3.2. Unimate endüstriyel robotu (“George Devol”, t.y.)

Endüstriyel robot tarihinin başlangıcı olarak kabul edilen 1954 yılında Amerikalı bir bilim adamı olan George Devol’un patent olarak başvurduğu “programlanabilir parça aktarımı (programmable article transfer(Devol, 1961))” adlı patent başvurusudur. 1963 yılında onaylanan bu patentte “Unimate” adı verilen endüstriyel robotun gelişme adımları başlamıştı. 1956'da Connecticut'taki bir kokteyl partisi sırasında Devol, bir uzay sanayi mühendisi olan Joseph Engelberger ile karşılaştı. Devol patentli makinenin olası kullanımı hakkında tartışılar ve sektörde kullanılacak manipülatörleri tasarlamak ve kurmak için bir şirket kurma fikrini konuştular. İlerleyen yıllarda Devol ve Engelberger, üretim tesislerinin ihtiyaçlarını daha iyi anlamak için (özellikle otomotiv sektöründe) birçok fabrikayı ziyaret etti. 1961'de, ilk endüstriyel robot olarak kabul edilen, Unimate adlı bir hidrolik olarak işletilen manipülatör olan Unimaton şirketini kurdular (Şekil 3.2), Aynı yıl, ilk Unimate bulunan General Motors fabrikasında kuruldu. Trenton'da (ABD): tek bir görevi yerine getirebilir, yani döküm makinesinden parçaları çıkarmak. Unimate'in ileri versiyonları, takip eden yıllarda iş parçası kullanımı ve araç gövdelerinin nokta kaynağı için kullanılmıştır(Gasparetto ve Scalera, 2019).

### 3.2. Endüstriyel Robot Sistemleri ve Kullanım Alanları

Nesnelerin internetle haberleştiği ve akıllı üretimin gerçekleştiği Endüstri 4.0'ın vazgeçilmez bileşeni endüstriyel robotlardır. Endüstriyel robotların kullanımını artırarak küresel ekonomide pay almak isteyen ülkeler kendilerini Endüstri 4.0 devrimine hazırlamaktadırlar.



Şekil 3.3. 2018 yılındaki imalat sanayisindeki robot yoğunluğu (“IFR”, t.y.)

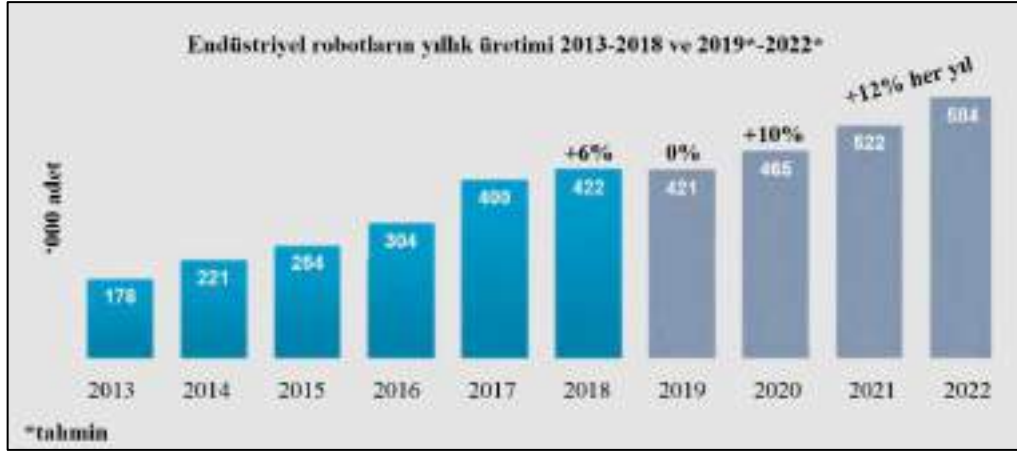
Uluslararası Robot Federasyonunun istatistiklerine göre İmalat sanayisinde 2018 yılına ait 10000 kişi başına düşen robot sayısı Şekil 3.3’te gösterilmiştir. İstatistikler göz önüne alındığında ortalama adet 99 olduğu grafikte ilk sırada Singapur 831 adet ile, ikinci sırada 774 adet ile Kore göze çarpmaktadır.



Şekil 3.4. Sektörlere göre yıllık (2016-2018) endüstriyel robot kurulumu (“IFR”, t.y.)

Endüstriyel robot kullanımını sektör bazında incelendiğinde Şekil 3.4’teki IFR istatistikleri göz önüne alındığında büyük oranda otomotiv alanında kullanıldığı görülmektedir. Daha sonra sırasıyla elektrik/elektronik, metal ve makine sanayileri, plastik ve kimya endüstrisi ve yemek sektörü yer almaktadır.





**Şekil 3.5.** 2013-2022 yıllık endüstriyel robot kurulumu yıllara göre artış miktarı (“IFR”, t.y.)

Şekil 3.5’teki IFR raporunda göre 2013 ile 2022 yılları arasında yıllık üretilen endüstriyel robot sayısı 1000 in katları şeklinde gösterilmiştir. 2019, 2020, 2021 ve 2022 yılları tahmin artış değerleri gösterilmiştir.

Bir ürünün imalatı aşamasında üretim, birleştirme, paketlenme ve depolama vb. safhalarına geçebilmesi için bir yerden başka bir yere taşınması gerekir. Robot nesneyi taşıma aşamasında, nesneyi herhangi bir fiziksel değişikliğe uğratmadan önceden belirlenmiş rotada hareket ettirmesi sonucu serbest bıraktığı başlıca uygulamalar şunlardır(Siciliano ve diğerleri, 2009):

- Paletleme (nesneleri palet üzerine sıralı şekilde yerleştirmek)
- İşleme tezgahına parça besleme
- Parça sıralama
- Paketleme



**Şekil 3.6.** Otomatik yönlendirmeli araç sistemi (Micieta, Zavodska, Rakyta ve Binasova, 2015)

Bu uygulamaların yanı sıra bir üretim hattından diğer üretim hattına taşıma işlemi için Otomatik Yönlendirmeli Araçlar (Automated Guided Vehicles) sayesinde gerçekleştirilmektedir. Şekil 3.6’da gösterildiği gibi vagonlar önceden tanımlanmış yolda otomatik olarak ilerler (Micieta ve diğerleri, 2015). Geleneksel otomatik yönlendirmeli araçlarda kılavuz olarak endüktif kılavuz teli, manyetik bant veya optik görünür çizgi ile gerçekleştirirken modern araçlarda yerleşik mikroişlemciler ve sensörlerine (lazer, kilometre sayacı, GPS) sahip yerel teknolojilere sahip yüksek teknoloji sistemler kullanılmaktadır. Gelişmiş uygulamalarda kullanılan mobil robotlar, otomatik yönlendirmeli araçların doğal evrimi olarak düşünülebilir (Siciliano ve diğerleri, 2009).

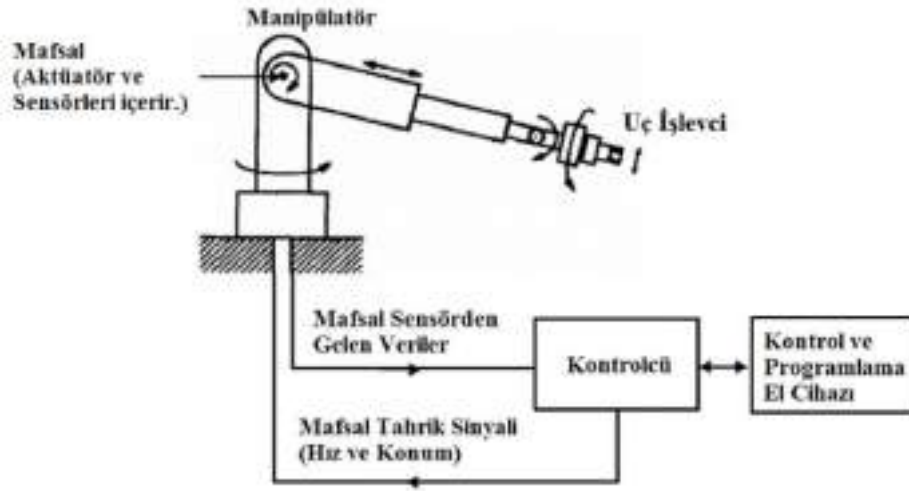
Parçaların fiziksel özelliğinin değiştiği veya başka parçalarla montajı yapılarak yeni kimlikler kazandığı başlıca uygulamalar (Siciliano ve diğerleri, 2009):

- Ark ve nokta kaynağı
- Boyama ve kaplama
- Yapıştırma ve sızdırmazlık
- Lazer ve su jeti ile kesme
- Frezeleme ve delme
- Döküm vb. uygulamalarda
- Çapak alma ve öğütme
- Vidalama, kablolama ve sabitleme
- Mekanik ve elektrik grupların montajı
- Elektronik panoların montajı

Ayrıca imalat aşamaları boyunca ürün kalitesini test etmek için ölçümler yapmak gerekir. Robot kolu hareket kabiliyeti sayesinde parçaları 3 boyutlu olarak rahatlıkla ölçüm yapabilme imkânı sunar. Başlıca ölçüme dair uygulamalar(Siciliano ve diğerleri, 2009):

- Nesne incelemesi
- Kontur tespiti
- İmalat kusurlarının tespiti

### 3.3. Endüstriyel Robot Kolu Temel Bileşenler



Şekil 3.7. Endüstriyel robot kolu temel bileşenler (“Awareness In The Fields of Robotics”, 2015)

Endüstriyel robotlar şekil 3.7’de gösterildiği gibi temel olarak altı bileşenden oluşurlar. Bu bileşenler:

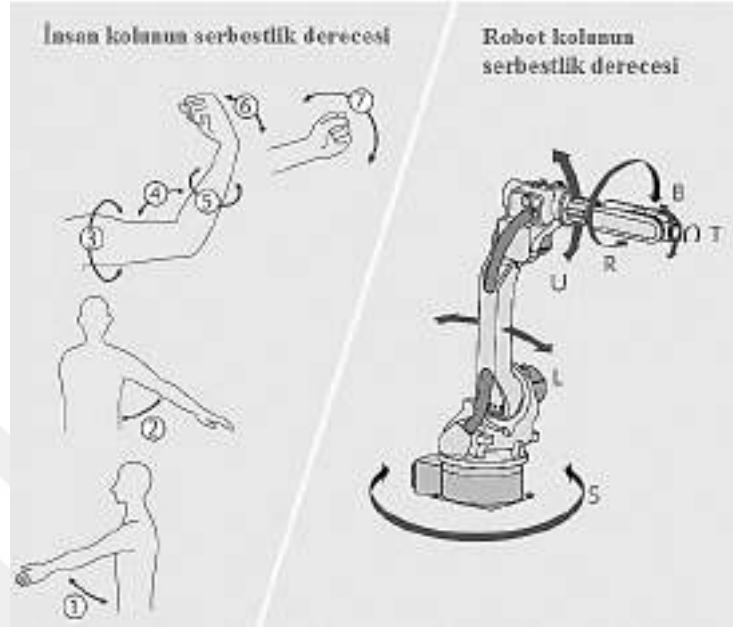
- Manipülâtör (Manipulator)
- Uç İşlevci (Gripper)
- Aktüatörler (Actuator)
- Sensör (Sensor)
- Kontrolör (Kontroller)
- Kontrol ve Programlama El Cihazı (Teach Pendant)

#### 3.3.1. Manipülâtör

Manipülâtör başta uzuv ve mafsallar olmak üzere diğer yapısal elemanlardan oluşan robotun ana gövdesidir. İnsan kolu ve bir robot kolundaki serbestlik dereceleri Şekil 3.8’de gösterilmiştir. Birçok hareket yönünden benzerlik göstermektedirler. İnsan kolu yedi serbestlik derecesinden oluşmaktadır. Şekil 3.8’ incelendiğinde:

- 1) Omuzların önünde ve arkasında serbestlik derecesi
- 2) Omuzları sağa ve sola hareket ettirme serbestliği derecesi
- 3) Üst kolları bükme serbestisi derecesi
- 4) Bükme serbestliği derecesi ve dirseklerde esneme

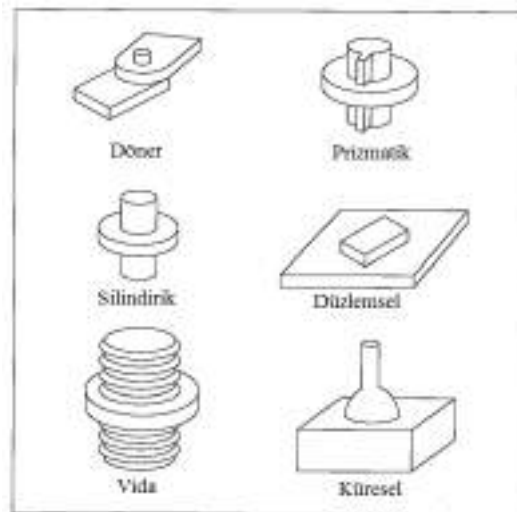
- 5) Kolun bükülmesi için serbestlik derecesi
- 6) Bileklerde elin avuç içine doğru bükülme serbestliği derecesi
- 7) Bilekleri yana doğru hareket ettirme serbestlik derecesi



Şekil 3.8. Robot kolu ve insan kolu

### 3.3.1.1. Eklem Tipi

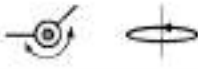

Manipülâtörün komşu uzuvlar arasındaki bağlantısı eklem (mafsâl) ile sağlanır. Kinematik çift terimi, Şekil 3.9’da gösterildiği bir çift gövdenin yüzeylerinin birbirleri arasında teması ile hareket eden bağlantılardır(Craig, 2004).



Şekil 3.9. Kinematik çiftler

Başlıca kinematik çiftler şunlardır (Şekil 3.9): döner mafsalsal, prizmatik mafsalsal, silindirik mafsalsal, düzlemsel mafsalsal, vida mafsalsal ve küresel mafsalsaldır.

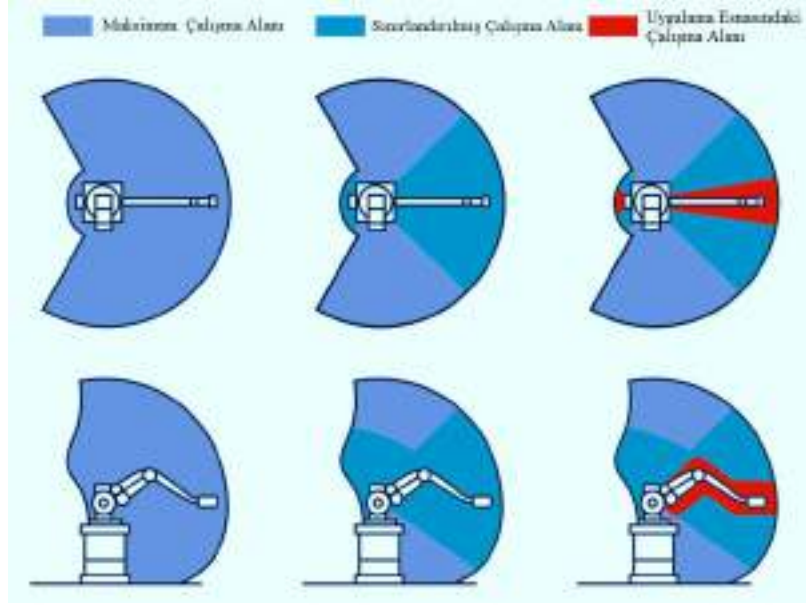
Eklem tipi olarak genelde Şekil 3.10'da gösterildiği gibi döner ve prizmatik mafsalsal çifti kullanılır.

Eklem Tipi	Sembolik Gösterim		Eklem Tanımı
	Harfsel	Şekilsel	
Rotational - Revolute (Dönel)	R		Eksen etrafında dönme hareketi
Prismatic - Translational (Prizmatik - Ötelemeli)	P ( veya T )		Eksen boyunca lineer hareket

Şekil 3.10. Döner ve prizmatik eklem çiftleri (Dişlitaş, 2015)

### 3.3.1.2. Çalışma Alanı

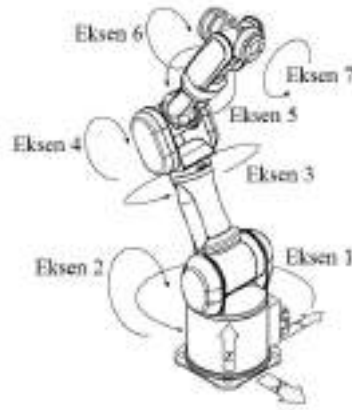
Robot manipülâtörün çalışma alanı, uç işlevci tarafından ulaşılabilir noktalar kümesi olarak tanımlanır(Cao, Lu, Li ve Zang, 2011). Nesnede yapılacak herhangi bir değişiklik (yani fiziksel bir model değişikliği) veya ortamda yapılan değişiklik robotun çalışma alanı içindeki hareketini değiştirir. Robot kolu çalışma zarfı Şekil 3.11'de incelendiği zaman gri renkli alanda maksimum çalışma zarfı yer almaktadır. Robot kolunun programlama ve bakım durumlarında müdahale etmek için çalışma alanı sınırlandırılır. Böylece robot koluna müdahale eden kişiye olası kaza durumu engellenmiş olunur. Sınırlandırılan alan mavi renkle gösterilmiştir. Robot kolunun çalışma anındaki hareketleri ise kırmızı renkle gösterilmiştir.



Şekil 3.11. Robot kolu çalışma alanı (“Industrial Robots”, t.y.)

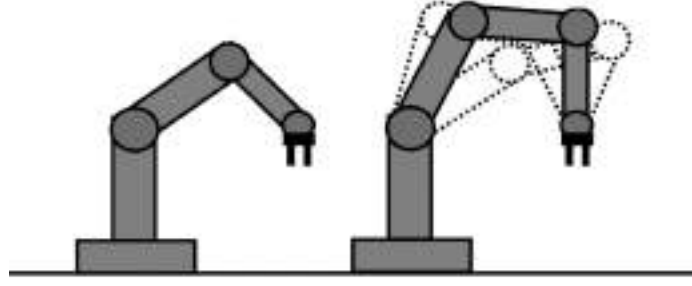
### 3.3.1.3. Serbestlik Derecesi (DOF-Degrees Of Freedom)

Robot kolunun hareketliliğinin ölçüsü olan serbestlik derecesi, robot kolunun bütün uzuvlarının uzayda yerini belirlemek için birbirinden bağımsız eklem değişkenlerinin sayısıdır. Sistemi kontrol etmek için minimum sayıda hareket ettirilen eklem sayısına eşittir (Briot ve Khalil, 2015). Şekil 3.12’de 7 serbestlik derecesine sahip olan Mitsubishi PA10-7CE robot kolu gösterilmektedir.



Şekil 3.12. Mitsubishi PA10-7CE (“Mitsubishi PA10-7CE robot”, t.y.)

İstenilen görevi yerine getirmek için robot kolunun serbestlik derecesi yeterli sayıda olması gerekir. Genellikle uzayda bir nesneyi istenilen konuma koymak için 3 serbestlik derecesi kullanılır. Görev değişkenlerinden daha fazla serbestlik derecesi varsa, manipülatörün kinematik bir bakış açısından gereksiz olduğu söylenir(Siciliano ve diğerleri, 2009).



Şekil 3.13. Gereğinden fazla serbestlik derecesi (“Robot Teknolojisi Ders Notları”, t.y.)

### 3.3.2. Uç İşlevci (Gripper)

Robot kolunun son elemanı olan uç işlevci, nesneye kuvvet veya tork uygulayarak robot kolunun nesneyi tutup bırakmasını sağlayan elemandır. Uç işlevçiler insan eline benzetilebilir. Robot kolları, farklı görevleri gerçekleştirmek için farklı tiplerde uç işlevçilerle donatılırlar. Farklı tahrik yöntemlerine göre uç işlevçiler Tablo 3.1’de sınıflandırılmıştır. İki parmaklı veya iki tırnaklı tutucular, kullanımı kolay, üretimi basit, fiyat açısından ekonomik ve birçok endüstriyel uygulama için uygun oldukları için en temel robot tutucular olarak kabul edilir. Bu tür uç işlevciye sahip robot kolları montaj, parça alıp yerleştirme ve basit manipülasyonlar gibi çeşitli görevleri yerine getirmede kullanılırlar(Samadikhoshkho, Zareinia ve Janabi-Sharifi, 2019).



Şekil 3.14. Endüstride kullanılan kısıkaç tipi uç işlevci (“Grasp, Sight, and Picking Evolve In Autonomous Robots”, 2019)

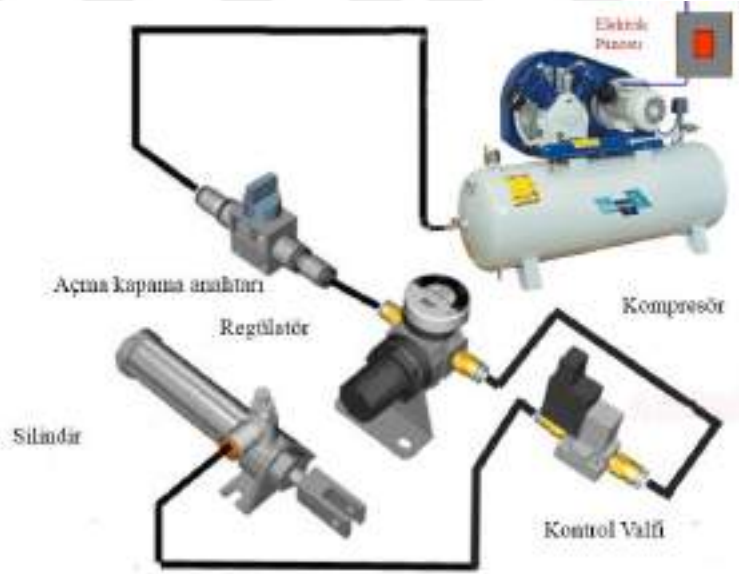
Şekil 3.14’te gösterilen robot kolu kısıkaç tipi uç işlevcisi ile nesneyi bir yerden alıp başka bir yere bırakmaktadır. Robotik sistem için yapılacak uygulamaya bağlı olarak, çeşitli tutucular gerekir. Bu nedenle, doğru olanı seçmek, tasarım sürecinde önemli bir husustur(Samadikhoshkho ve diğerleri, 2019).

**Tablo 3.1.** Farklı tahrik yöntemlerine göre uç işlevciler

Uç İşlevci (Gripper) Tipi	Avantajlar	Dezavantajlar
Kablo Mekanizmalı	Optimal ağırlık ve alan	Kontrol karmaşıklığı
Vakumlu	Son derece esnek Temiz	Bazı işlevsel problemler
Pnömatik	Küçük boyut Düşük ağırlık Temiz	Yeterince hassas değil Yüksek çalıştırma maliyeti
Hidrolik	Yüksek kuvvet	Yeterince temiz değil Yüksek bakım maliyeti
Servo Elektrik	Son derece esnek Düşük bakım maliyeti Kolayca kontrol edilebilir Temiz	Düşük kuvvet

### 3.3.2.1. Pnömatik Uç İşlevci

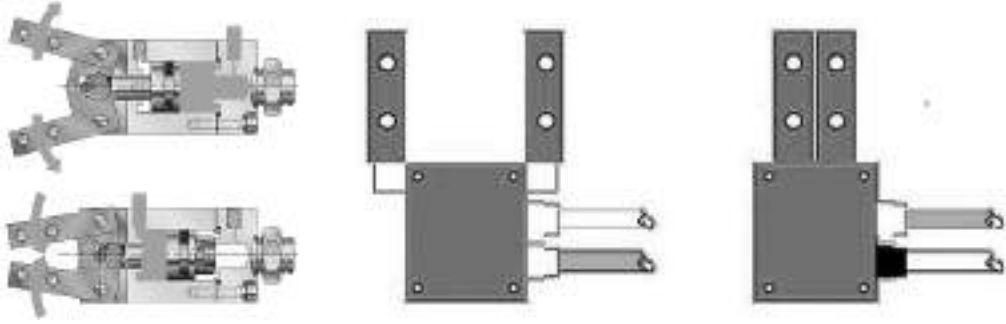
Pnömatik uç işlevcide sistemde basınçlı hava kullanılarak açma ve kapama hareketi yaptırılır. Pnömatik uç işlevci kompakt boyutu ve hafifliği nedeniyle popülerdir. İmalat endüstrisinde yardımcı olabilecek dar alanlara kolayca dahil edilebilir(“Grippers for Robots”, t.y.).



**Şekil 3.15.** Pnömatik aktüatör sistemi

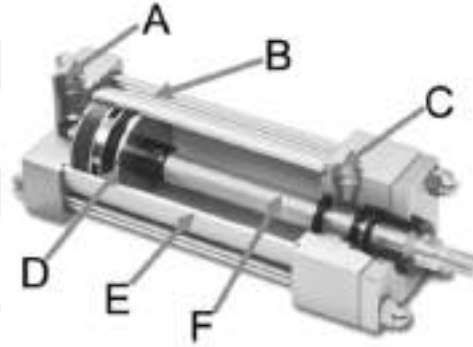
Şekil 3.15'te pnömatik aktüatör sistemi gösterilmektedir. Sistemde kompresörden sağlanan basınçlı hava regülatörden geçerek hem filtreden geçirilir hem de basınç istenilen düzeye getirilir. Daha sonra hava istenilen şekilde silindire yönlendirilmesi kontrol valfi sayesinde gerçekleşir. Yönlendirme sonucu sisteme ulaşan basınçlı hava pistonda açma kapama yaptırması sonucu uç işlevci çalıştırılmış olur.





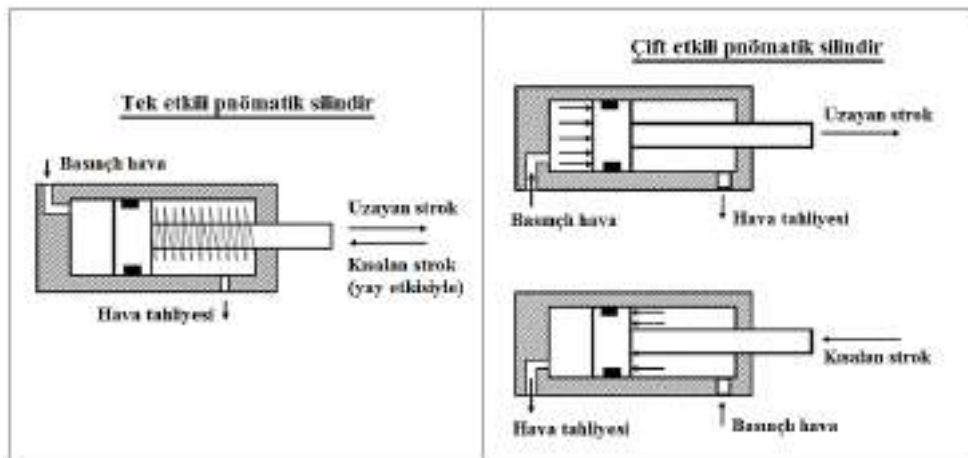
Şekil 3.16. Açısal ve paralel pnömatik uç işlevci (Bernier, t.y.)

Endüstride sık kullanılan açısal ve paralel pnömatik tipi Şekil 3.16’da gösterilmiştir. Açısal uç işlevcide pistonun ileri yönde hareket etmesiyle çeneler bir eksen etrafında dönerek açılır. Pistonun geri yönlü hareketinde ise çeneler kapanır. Paralel uç işlevcide, çenelerin tutma yüzeyi tüm hareketleri boyunca paralel kalır.



Şekil 3.17. Pnömatik silindir temel bileşenler (“Pneumatic Cylinders”, t.y.)

Pnömatik silindirin temel bileşenleri Şekil 3.17’de gösterildiği gibi ön hava portu (A), gerdirme mili (B), piston (D), arka hava portu (C), silindir (E) ve piston mili (F) şeklindedir.



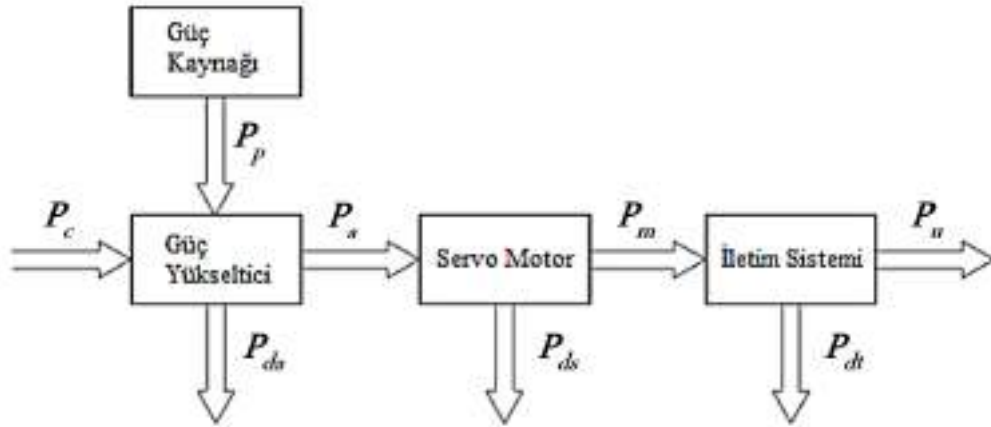
Şekil 3.18. Tek ve çift etkili silindir

Şekil 3.18’de tek ve çift etkili pnömatik silindirin yapısı ve çalışma mantığı gösterilmiştir. Tek etkili silindirde tek portta hava girişi sağlanarak piston ileri yönde hareket eder, ters yöndeki hareketi ise mildeki yay ile sağlanır. Çift etkili silindire göre daha az hava kullanılır. Çift etkili silindirde iki portta hava girişi vardır. İleri ve geri yönlü hareketi hava ile sağlanır.

### 3.3.3. Aktüatörler

Aktüatörler, robot koluna gönderilen komutlar neticesinde hareketi sağlayan tahrik elemanlarıdır. Aktüatör (actuator) kelimesi İngilizcede act kelimesinden türemiştir. Act kelimesi hareket etmek, etkilemek gibi anlamlarına karşılık gelir(Akı, t.y.). Manipülâtörün mafsasında hareketi gerçekleştiren ve işbirliği içinde çalışan başlıca elemanlar şunlardır(Siciliano ve diğerleri, 2009):

- Güç kaynağı (Power supply)
- Güç amplifikatörü (Power amplifier)
- Servo motor (Servo motor)
- İletim sistemi (Transmission)



Şekil 3.19. Mafsal aktüatör sisteminde gücün dağılımı

Mafsaldaki aktüatör sistemindeki güçlerin değişimi Şekil 3.19’da gösterilmiştir. Güç, bir akımın ürünü olarak ifade edilebildiği gibi fiziksel kavram olarak mekanik, elektrik, hidrolik veya pnömatik sistemlerden üretilen kuvvet miktarı olarak da ifade edilir. Sistemde  $P_c$  elektrik gücünü temsil ederken,  $P_u$  sistemi harekete geçirmek için gerekli olan mekanik gücü ifade eder. Motorun (elektrik, hidrolik veya pnömatik) besleme gücünü  $P_a$ , motor tarafından geliştirilen mekanik gücü  $P_m$  ile ifade edilmektedir.

Ayrıca  $P_{da}$ ,  $P_{ds}$  ve  $P_{dt}$ , sırasıyla amper, motor ve transmisyon tarafından kaybedilen güçleri belirtir(Siciliano ve diğerleri, 2009).

Mekanik sistemlerde hareketin gerçekleştirilmesini sağlayan motorlara verilen giriş gücü olarak verilen  $P_a$  üç gruba ayrılır:

- Pnömatik motorlar, kompresörün sağladığı pnömatik enerjiyi kullanan ve piston veya türbinler vasıtasıyla mekanik enerjiye dönüştürürler.
- Hidrolik motorlar, bir depoda depolanan hidrolik enerjiyi uygun pompalar vasıtasıyla mekanik enerjiye dönüştürürler.
- Elektrik motorları, elektrik enerjisini mekanik enerjiye dönüştürürler.

Giriş gücünün bir kısmı  $P_a$ , mekanik güç  $P_m$  olarak çıkışa dönüştürülür ve kalan kısım  $P_{ds}$ , mekanik, elektrik, hidrolik veya pnömatik kayıp nedeniyle dağılır.

Robotik sistemlerde kullanılan motorlar, yaklaşık 10 W ile yaklaşık 10 kW arasında değişen güce sahiptirler. Gerekli performans sağlayabilmesi için şu özelliklere sahip olması gerekir:

- Düşük atalet ve yüksek güç / ağırlık oranı,
- Darbe torklarını aşırı yüklenme ve dağıtma imkânı,
- Yüksek ivmelenme geliştirme kabiliyeti,
- Geniş hız aralığı (1 ila 1000 devir / dak),
- Yüksek konumlandırma hassasiyeti (en az dairenin 1 / 1000'i),
- Düşük devirde bile sürekli dönüşü garanti etmek için düşük tork dalgalanması.

### 3.3.3.1. AC Servo Motorlar

Kelime olarak “servo”, nesnelerin hedef konuma gitmesi veya hareket hedefini takip ettiği durumlarda kullanılır. Kelimenin kökeni Latince köle anlamına gelen “servus” kelimesinden gelmektedir. Servo mekanizmasına kısaca “servo” denilmektedir. Servo, istenilen komutları yerine getiren bir kontrol mekanizmasıdır. Servo mekanizmaları pozisyon, hız, tork ve bunların kombinasyonu ile kontrolü sağlar(“E-Learning Courses”, t.y.).

Konum kontrolünde, servo doğru bir şekilde hareket eder veya istenilen pozisyonda durur. Servo motorlar, tekrarlı durup ve başlama tekrarlı durumlarında mikronun altındaki ( $1\mu=1/1000$  mm) hassasiyeti yakalayabilirler.

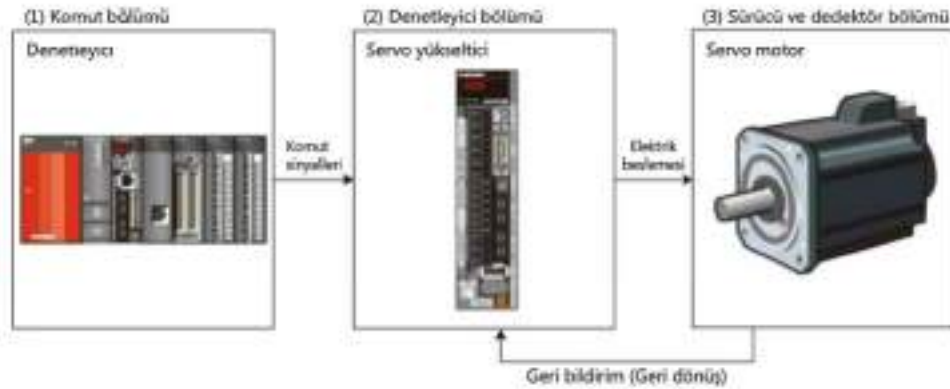
Hız kontrolünde servo istenilen hıza hızlı bir şekilde uyum sağlayabilir. Servo motorlar, yükün değişiklik gösterdiği zamanlarda da minimum hız farkıyla istenilen hıza ulaşırlar.

Tork kontrolünde hassas bir şekilde ve yükün değiştiği durumlarda istenilen tork değerlerini sağlarlar. Tork değerinin artması durumunda hız düşer ve elde edilen güçte artar("E-Learning Courses", t.y.).

Yüksek hız ve hassasiyet isteyen durumlarda servo motorlar sürekli geri besleme sağlayarak sistemde istenilen talimatları yerine getirirler. Geri bildirim sinyali ile kontrol sinyali arasındaki fark minimize etmek sistemi doğru bir şekilde kontrol etmek açısından önemlidir.

Japon endüstriyel standartlarına göre (JIS) servo mekanizmasının tanımı:

Objelerin kontrolü için hedef pozisyon, yönelme, duruş ve diğer faktörlerin koordineli olarak çalışmasıyla oluşan kontrol sistemidir("E-Learning Courses", t.y.).



Şekil 3.20. Servo mekanizması yapılandırılması

Servo mekanizmaları Şekil 3.20’de gösterildiği gibi komut bölümü, denetleyici bölümü ve sürücü ve detektör bölümü olmak üzere 3 ana bölümden oluşur("E-Learning Courses", t.y.).

- Komut bölümü, komut sinyallerinin üretir.
- Denetleyici bölümü, verilen komutlar neticesinde motor ve diğer elemanlar hareket ettirilir.

- Sürücü ve detektör bölümünde, hedefe doğru hareket gerçekleşir ve hedefe hakkında durum bilgileri alınır.

### Servo motorun yapısı ve çalışma prensibi



Şekil 3.21. Standart bir servo motor yapısı (“Servo Motor Technology”, t.y.)

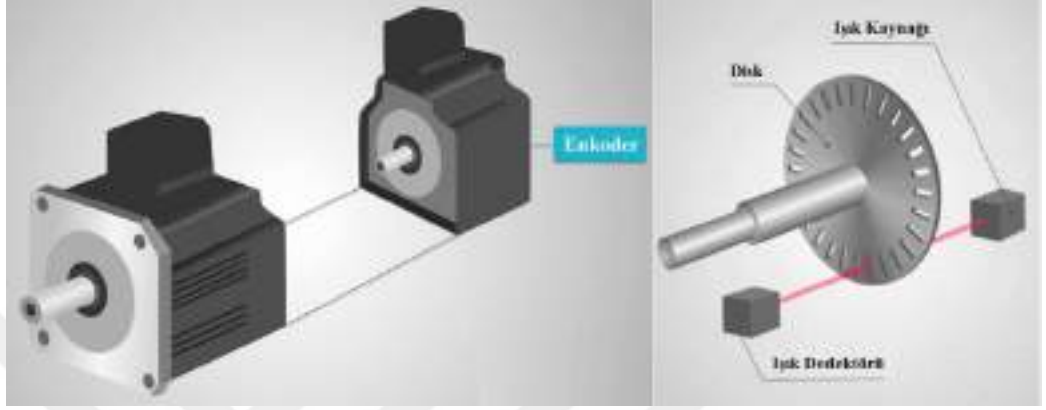
Standart bir servo motorun temel bileşenleri Şekil 3.21’de gösterilmiştir. Statorda bulunan mıknatıslar sabit bir manyetik alan oluşturur. Rotor dönen kısımdır ve bir çeşit bobindir. Rotor komütatör yardımıyla güç kaynağına bağlanır. Akım bobin üzerinden geçtiğinde Lorentz kanununa göre elektro manyetik kuvvet gerçekleşir. Lorentz kuvvet kanununa göre akım taşıyan iletken tel manyetik alanla etkileşirse iletken telde bir kuvvet indüklenmiş olur(Hasırcı, t.y.). Bu oluşan kuvvetin etkisiyle rotorda dönme hareketi gerçekleşir.

### Servo motor eklentisi enkoder

Servo motor, rotorun konumunu ve hızını tespit etmek için motorun arka şaft tarafına monte edilmiş bir dönme detektörüne (enkoder) sahiptir. Enkoder, motorun hızını ve konumunu tespit etmek ve geri bildirim sağlamak için kullanılan için bir sensördür. En temel terimlerle, “konum”, “yön”, “hız” veya “sayım” algılar. Enkoderler hareketi çeşitli teknolojiler altında kullanacak ve bir elektrik sinyaline çevirecektir. Bu sinyal daha sonra PLC gibi bir kontrol cihazına geri gönderilir ve program içinde kullanılacak bir değeri temsil etmek üzere yorumlanır, yani ölçeklenir(“What is an Encoder?”, t.y.).

Enkoderlerde yer alan bazı teknolojiler şunlardır:

- Manyetik
- Mekanik
- Dirençli
- Optik



Şekil 3.22. Servo motor eklentisi enkoder("What is an Encoder?", t.y.)

Optik en yaygın olarak kullanılan Enkoder türüdür. Optik Enkoderlerde Şekil 3.22'de gösterildiği gibi yarıklı desenli bir disk bulunur. Işık kaynağından gelen ışık demetleri yarıklardan geçtiğinde ışık kaynağının karşısında bulunan ışık dedektörüne gelir. Işık dedektörü yarıklardan geçen ışıkları okurken, yarık olmayan yerlerde ışıklar dedektöre ulaşmaz. Böylece enkoderler disk döndükçe desenler dijital veya darbe sinyali çıkışı üretirler("What is an Encoder?", t.y.).

### Servo Motorların Sınıflandırılması

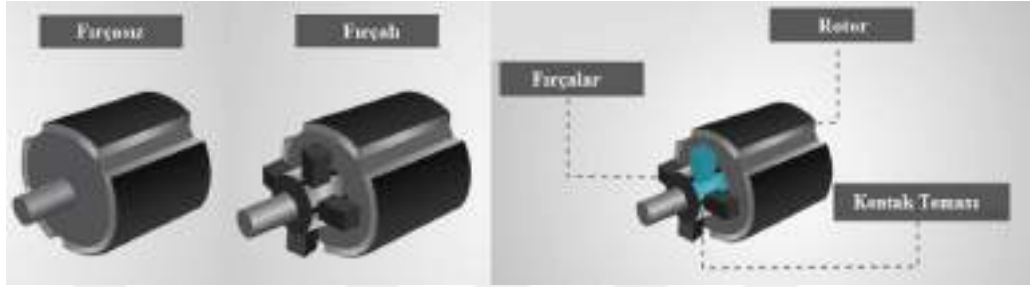
Servo motor tipleri, uygulama alanlarına göre AA servo motor ve DA servo motor gibi farklı tiplerde sınıflandırılır. Servo motorlar 3 ana başlıkta sınıflandırılabilirler:

- Akım türlerine göre (alternatif ve doğru akım)
- Komütasyon tipine göre (firçalı ve firçasız)
- Manyetik alan ile rotorun dönme hızına göre (senkron ve asenkron)



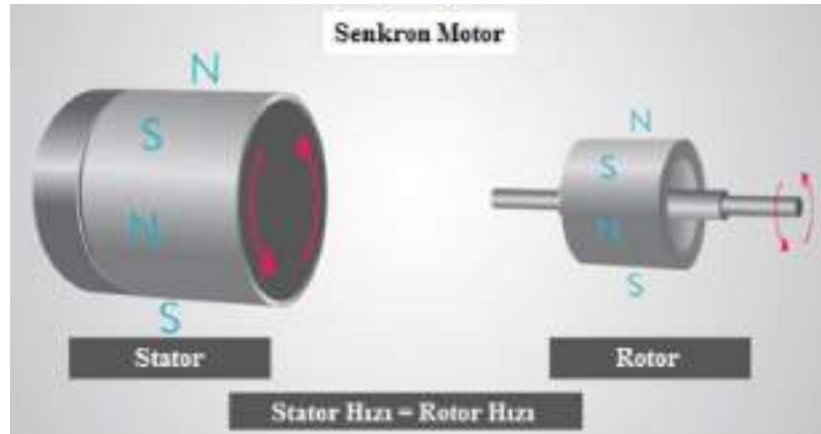
Şekil 3.23. Alternatif veya doğru akımla beslenen servo motor ("Servo Motor", t.y.)

Alternatif ve doğru akım, motorun kullanacağı akım türüne göre belirlenir. Ayrıca hız kontrolü bakımından farklılıklar vardır. Doğru akım (DA) servo motorlarda sabit yük altında besleme voltajı ile doğru orantılı olarak hızı artar. Alternatif akımda (AA) ise hız uygulanan voltajın frekansı ve manyetik kutupların sayısı ile belirlenir. Alternatif akımlı servo motorlar yüksek akımlarda, fazla hassasiyet istenen durumlarda ve tekrarlı iş yapılan endüstriyel uygulamalarda kullanılırlar (“Servo Motor”, t.y.).



Şekil 3.24. Fırçalı veya fırçasız servo motorlar (“Servo Motor”, t.y.)

Fırçalı servo motorlarda şekil 3.24’te gösterildiği gibi komütatör ve fırça bulunur. Komütatöre baskı yapan iletken malzemeden yapılmış fırçalar, komütatörün segmentleri arasında kayar bir temas kurarlar. Komütatör, rotor ile sürücü devresi arasındaki akım yönünü periyodik olarak tersine çeviren bir döner elektrik anahtarıdır. Fırçalı motor daha ucuz ve kullanımı kolaydır. Fırçasız motorlar ise daha az gürültülü, daha güvenilir ve daha yüksek verimliliğe sahiptir. AA servo motorlar genellikle fırçasızdır (“Servo Motor”, t.y.).



Şekil 3.25. Senkron Motor (“Servo Motor”, t.y.)

AA servo motorlar genellikle senkron ve asenkron motor olarak sınıflandırılırlar. Senkron bir motorda rotor, statorun dönen manyetik alanıyla aynı hızda döner (Şekil

3.25). Asenkron motorda ise rotor, statorun dönen manyetik alanından daha yavaş bir hızda döner(“Servo Motor”, t.y.).

### **Servo Motorların Kullanım Alanları**

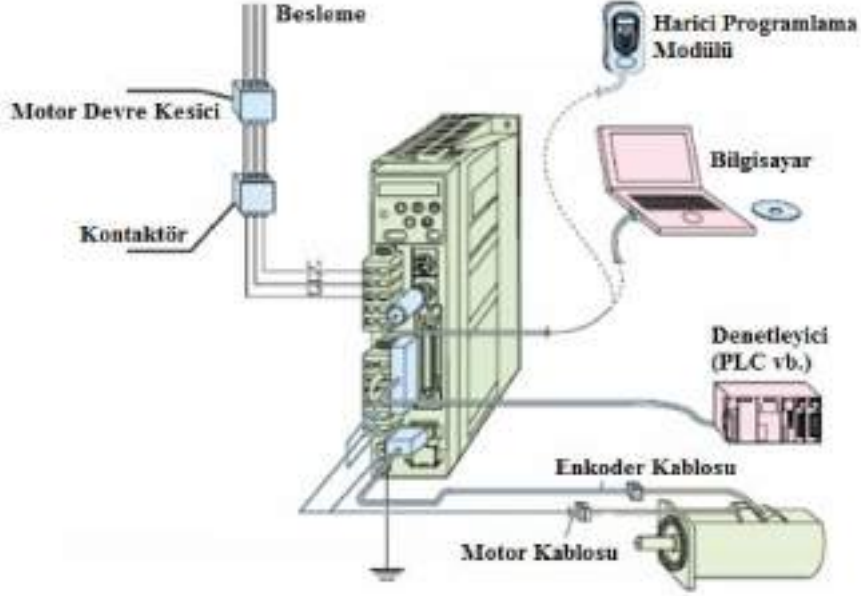
Servo motorlar birçok endüstriyel uygulamalarda kullanılır. Kullanım alanlarına örnek verilirse(“Servo Motor”, t.y.):

- Bir robotun her hareket noktasında hassas hareket açısını gerçekleştirmek için servo motor kullanılır.
- Kamera otomatik odaklaması, odak dışı görüntüleri netleştirmek için lensin konumunu tam olarak düzelteren kameraya yerleştirilmiş bir servo motor kullanır.
- Ulusal Radyo Astronomi Gözlemevi tarafından kullanılan servo motorlar hem azimutun hem de antenlerin ve teleskopların yükselme ekseninin konumlandırılmasında kullanılırlar.

### **AC Servo Motor Bağlantı Şeması**

AA servo motor bağlantı şeması Şekil 3.26’da gösterilmiştir. Sistemde besleme yapılırken servo motora uygun voltaj ve frekans değerleri seçilir. Devre kesici, sisteme aşırı akım gelmesi durumunda devreyi keserek güç hatlarını korur. Kontaktör, servonun gücünü açıp kapatma işlemini yapar. Motor besleme kablosu, servo motoru beslemede kullanılır. Enkoder kablosu, sistemde geri dönüş yapıp kapalı çevrim elemanı olan enkoderden veri alış verişini sağlar. Denetleyici kısmında sistemi kontrol etmek için PLC gibi kontrolcülerin bağlantısı bulunur. Bilgisayardan ise sistemi kontrol etmek için çeşitli sürücü programları yüklendikten sonra programlama yapılarak motoru sürme komutları oluşturulur.





Şekil 3.26. AC servo bağlantı şeması (“Servo Motor Sürücü Bağlantısı”, t.y.)

### İndirgeme ve Aktarma Sistemleri

Dişliler ve kayışlar, dönme hareketini bir şafttan diğerine iletir, bu da genellikle işlemdeki hızı ve torku değiştirir. Dişli takımları genellikle iki şaftın birbirine yakın olduğu yerlerde kullanılır. Kayışlar ve kasnaklar ise, birbirinden daha uzak olan şaftları bağlar(Berardinis, 2000).

Redüktörler, sistemde hızı istenilen düzey ve torka getirmek için kullanılırlar. Kompakt yapılandırmalarla büyük indirgeme sağlayabilirler. Dişliler, redüktörü oluşturan temel elemanlardır. Düz dişliler, konik dişliler, sonsuz dişliler, helisel dişliler ve dişli çeşitlerine örnek olarak verilebilir. Dişliler çeşitlerine göre farklı yük değerlerine, aşınma özelliklerine ve sürtünme özelliklerine sahiptir(Craig, 2004).

Dişli kullanmanın en büyük dezavantajları boşluk ve sürtünmedir. Dişlilerin tam olarak birbirine geçmemesinden kaynaklanan boşluk, giriş dişlisi sabit kaldığında çıkış dişlisinin maksimum açısal hareketi olarak tanımlanabilir. Dişlilerdeki dişler, boşluğu gidermek için sıkıca örülürse, aşırı miktarda sürtünme olabilir. Çok hassas dişliler ve çok hassas montajla bu sorunlar en aza indirilir fakat maliyeti de artırır.

Dişli oranı, dişli çiftinin hız azaltıcı ve tork arttırıcı etkilerini tanımlar. Hız düşürme sistemleri için  $\eta > 1$ ; giriş ve çıkış hızları ile momentler arasındaki ilişkiler denklem 3.1 ve denklem 3.2’de gösterilmiştir. Denklemde  $\dot{\theta}_0$  ve  $\dot{\theta}_i$  sırasıyla çıkış ve giriş hızları,  $\tau_0$  ve  $\tau_i$  ise çıkış ve giriş torklarıdır.

$$\dot{\theta}_0 = (1/\eta) \cdot \dot{\theta}_i \quad (3.1)$$

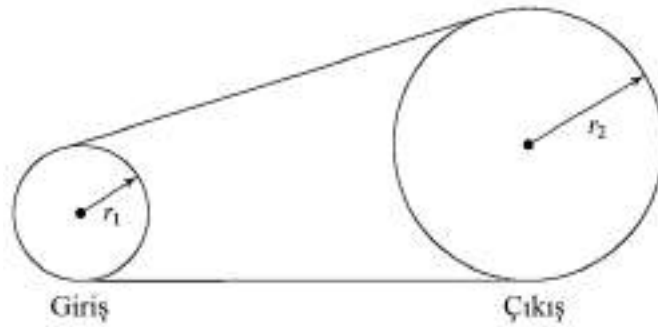
$$\tau_0 = \eta \cdot \tau_i \quad (3.2)$$

İkinci indirgeme sınıfı esnek bantlar, kablolar ve kayışlardır. Bu elemanlar kasnakların etrafında bükülebilecek kadar esnek olmaları gerektiğinden, uzunlamasına yönde de esnek olma eğilimindedirler. Esnekliği uzunlukları ile orantılıdır. Bu sistemler esnek olduğundan, kayışın veya kablonun kasnağa takılı kalmasını sağlamak için halkanın önyüklemesi için bazı mekanizmalar olmalıdır. Büyük ön yükler, esnek elemana gereksiz yükler ekleyebilir ve aşırı sürtünmeye neden olabilir.

Makaralı zincirler esnek bantlara benzer, ancak yüksek sertliği korurken nispeten küçük kasnakların etrafında bükülebilir. Bağlantıları bağlayan pimlerdeki aşınma ve yüksek yüklerin bir sonucu olarak, dişli kayış sistemleri belirli uygulamalar için makaralı zincirlerden daha kompakttır.

Bant, kablo, kayış ve zincir tahrikleri, aktarımı indirgeme ile birleştirebilir. Şekil 3.27'de gösterildiği gibi, giriş kasnağının yarıçapı  $r_1$  olduğunda ve çıkış kasnağının yarıçapı  $r_2$  olduğunda, iletim sisteminin dişli oranı denklem 3.3'teki gibi olur:

$$\eta = \frac{r_2}{r_1} \quad (3.3)$$



Şekil 3.27. Bant, kablo, kayış ve zincir tahrikleri (Craig, 2004)

### 3.3.4. Sensörler

Robotik sistemlerde yüksek performans elde etmek için sensörler kullanılır. Sensörler robot kolunun iç durumunu ölçen içsel sensörler ve çevre ortam hakkında bilgi veren dışsal sensörler olarak sınıflandırılır(Siciliano ve diğerleri, 2009).

Manipülâtörün iç durumunu ölçmede kullanılan başlıca içsel sensörler şunlardır:

- Mafsal pozisyonları ölçmede kullanılan sensörler
- Mafsal hızlarını ölçmede kullanılan sensörler
- Mafsal torklarını ölçmede kullanılan sensörler

Diğer taraftan başlıca dışsal sensörler şunlardır:

- Kuvvet sensörleri
- Dokunsal sensörler
- Yakınlık sensörleri
- Mesafe sensörleri
- Görüş sensörleri

Bu tür sensörlerin amacı, sistemi daha otonom hale getirerek, robotun ortamdaki nesnelere karakterize etmesi sonucu etkileşimini arttırmaktır. Ayrıca bu sensörlere ses, nem, duman, basınç ve sıcaklık sensörleri gibi robotik uygulamalar için kullanılan özel sensörler de dahil edilir(Siciliano ve diğerleri, 2009).

### 3.3.5. Kontrol Ünitesi

Robot çeşitli algoritmalar sayesinde karar verir. Robota bir görev verildiğinde, bu görevi yerine getirebilmesi için hareket etmesi gerekir. Hareket etme eylemi sensörler tarafından algılanan ortama bağlıdır. Örneğin, bir robot depodaki bir raftan teslimat paletine bir nesne getirecekse, uygun rafa gitmek, nesneyi algılamak ve kavramak ve daha sonra palete geri dönüp yüklemek için sensörleri kullanmalıdır. Yalnızca son derece iyi tanımlanmış ortamlarda çalışan robotlar bu tür işleri sensörlerden bilgi almadan gerçekleştirebilir. Örnek olarak, bir cihazı fabrikada monte eden robotik bir kol; parçalar hassas bir şekilde çalışma yüzeyine yerleştirilirse, robot parçaları algılamadan manipüle edebilir. Ancak çoğu ortamda sensörler kullanılmalıdır. Bir depoda rafa giden yollarda engeller olabilir, nesne rafa tam olarak yerleştirilmeyebilir ve forklift asla tam olarak aynı

yere park etmeyebilir. Robotun karar vermek için kontrol algoritmalarını kullanarak bu küçük deęişimlere uyum saęlaması gerekir(Ben-Ari ve Mondada, 2018).

Robotikte temel olarak iki çeşit kontrol yöntemi vardır. İlki algoritmanın parametrelerinin önceden ayarlandığı “açık döngü kontrolü” dięeri ise sensörlerden gelen verilerin algoritmanın davranışını etkilediğı “kapalı döngü kontrolü” dür. Bir robotun tasarımcısı yeterli performans saęlamak için bu kontrol algoritmalar arasından seçim yapmalıdır(Ben-Ari ve Mondada, 2018).

### 3.3.5.1. Açık döngü kontrol sistemi

Çıktının kontrol eylemi üzerinde etkisi olmayan sistemlere “açık döngü kontrol sistemleri” denir. Başka bir ifadeyle, bir açık döngü kontrol sisteminde çıktı, girdi ile karşılaştırmak için ne ölçülür ne de geri beslenir. Şekil 3.28’de açık kontrol sistemi modeli gösterilmiştir(Ben-Ari ve Mondada, 2018).

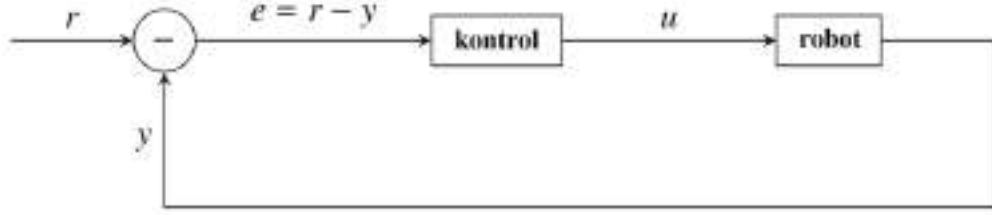


Şekil 3.28. Açık döngü kontrol sistemi

Açık döngü kontrol sistemine ekmek kızartma makinesi örnek verilebilir. Ekmek kızartma makinesi, eylemleri yarı otonom olarak gerçekleştiren bir makinedir. Ekmek kızartma makinesinde ekmek dilimleri yerleştirilir, zamanlayıcıyı ayarlanır ve kızartma işlemini başlatmak için kolu aşağı itilir. Sonuçlar garanti edilmez. Eğer zamanlayıcının süresi çok kısaysa, ekmeğı tekrar kızartmak gerekir; zamanlayıcının süresi çok uzunsa, tost yanar. Sonuç belirsizdir çünkü ekmek kızartma makinesi açık çevrim kontrol sistemidir. Gerekli sonuca ulaşıp ulaşılmadığını görmek için kızartma işleminin sonucunu kontrol etmez(Ben-Ari ve Mondada, 2018).

### 3.3.5.2. Kapalı döngü kontrol sistemi

En az hata deęeriyle istenilen deęeri elde etmek için çıkış sinyalinin bir kısmının girişe geri beslendiğı yani geri beslemenin olduğı sistemlere “kapalı döngü kontrol sistemleri” denir. Otonom davranış elde etmek için robotlarda kapalı döngü kontrol sistemi kullanılır(Ben-Ari ve Mondada, 2018).



Şekil 3.29. Kapalı döngü kontrol sistemi (Ben-Ari ve Mondada, 2018)

Kapalı döngü kontrol sistemi temel olarak Şekil 3.29’ da gösterilmiştir. Sistemde “r” parametresi, robota verilen görevin referans değerini temsil eder. Örneğin, bir depo robotunda referans değerleri, robotun belirlenen raf yığına göre konumunu ve tutucu kolun alınacak nesneden uzaklığını içerir. Referans değeri doğrudan robot tarafından kullanılamaz. Bunun yerine, kontrol değerine “u” parametresine dönüştürülür. Örneğin, referans değeri robotun belirlenen rafa göre konumu ise, kontrol değeri motorların güç ayarları ve motorların çalışma süresi olacaktır. Çıkış değeri olan “y” parametresi, yani robotun gerçek durumu, örneğin bir nesneye olan mesafesini temsil eder. Şekil 3.28’deki modele “geri besleme kontrol sistemi” de denir. Çünkü “y” çıkış değeri kontrol algoritmasının da geri beslenir ve kontrol değerini hesaplamak için tekrar tekrar sistemde kullanılır. Çıkış, “ $e = r - y$ ” değerini hesaplamak için referans değeri ile karşılaştırılır. Kontrol algoritması hatayı, robota giriş olan kontrol sinyalini “u” üretmek için kullanır. Sistemde en az hatalı değer elde edilinceye kadar döngü devam eder (Ben-Ari ve Mondada, 2018).

### 3.3.6. Kontrol ve programlama el cihazı (Teach Pendant)



Şekil 3.30. Kontrol ve programlama el cihazı (“KUKA smartPAD-2”, t.y.)

Kontrol ve programlama el cihazları, bir robotun belirli bir alandaki istenen noktalara ve noktalardan hareketlerini kontrol etmek için kullanılan anahtarlar ve kadranslarla donatılmış cihazlardır. İmalat, işleme ve tıbbi uygulamalarda otomatik veya robotik makinelerin hareket kontrolü için kullanılır(“Teach Pendants Information”, t.y.). Şekil 3.30’da dokunmatik ekran panelli Kuka firmasına ait “KUKA smartPAD-2” modeli gösterilmektedir.

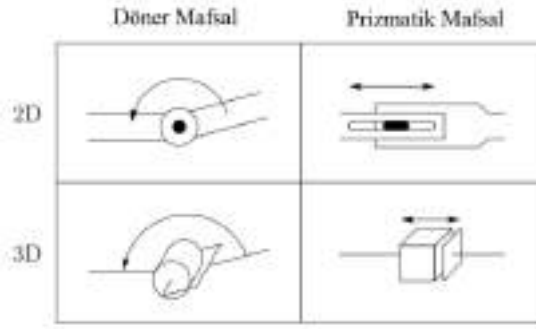
Endüstriyel uygulamalarda zorlu, tehlikeli veya tekrarlayan görevleri yerine getiren otomatik makineleri veya robotik sistemleri programlamak ve kontrol etmek için kullanılır. Operatörün robotik makinelerin eylemlerini güvenli bir şekilde manipüle etmesine izin veren taşınabilir kontrol cihazlarıdır. Otomatik veya robotik makinelerde genellikle şu amaçlar için kullanılırlar(“Teach Pendants Information”, t.y.):

- Malzeme taşıma ve montaj
- Kaynak
- Yükleme ve boşaltma
- Boyama ve kaplama işleri

Kontrol ve programlama el cihazları, robotu belirli bir görev, hareket aralığı veya hız için programlamak üzere kumanda kullanılarak öğretim ve tekrarlama tekniği ile robotu kontrol etmek üzere tasarlanmıştır. Operatör robotun hız, gecikme süreleri ve belirli işlevlerin yürütülmesi hakkında bilgi sağlamanın yanında robotun çalışma alanında makinelerle fiziksel ilişkisini tanımlamak için kullanılır(“Teach Pendants Information”, t.y.).

#### **3.4. Endüstriyel Robotların Sınıflandırılması**

Kinematik zincirler oluşturmak için prizmatik ve döner eklemleri kullanarak birçok robot tipi oluşturulmasına rağmen, pratikte bunlardan sadece birkaçı yaygın olarak kullanılmaktadır. Döner ve prizmatik mafsalsın sembolik gösterimi Şekil 3.31’de olduğu gibidir.

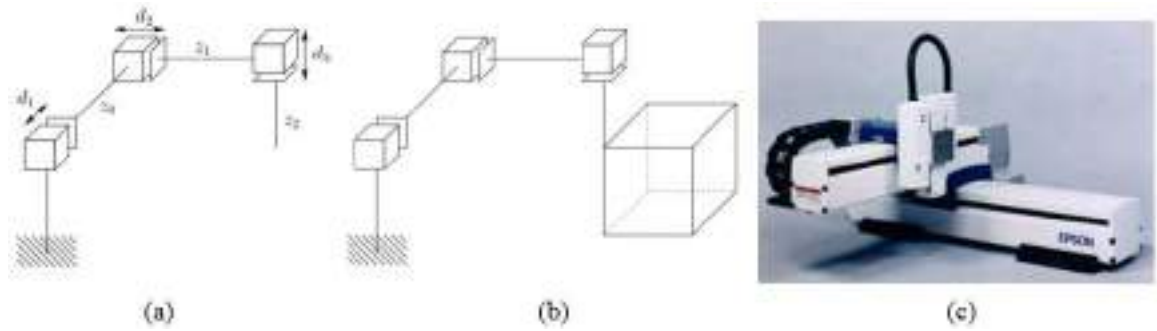


Şekil 3.31. Döner ve prizmatik mafsal sembolik olarak gösterimi (Spong, Hutchinson ve Vidyasagar, 2004)

Endüstriyel robotlar genel olarak şu şekilde sınıflandırılırlar:

- Kartezyen Robotlar
- Silindirik Robotlar
- Küresel Robotlar
- SCARA Robotlar
- Mafsalı Robotlar
- Paralel Robotlar

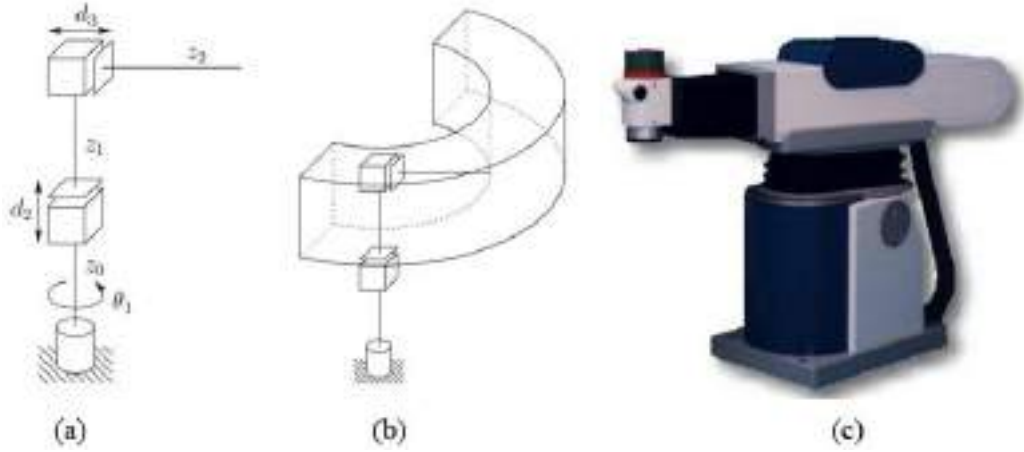
### 3.4.1. Kartezyen robotlar



Şekil 3.32. Kartezyen robot (a) Kinematik zincir (b) Çalışma uzayı (c) Uygulama örneği EPSON marka Kartezyen robot (Spong ve diğerleri, 2004)

Kartezyen robotun, 1'den 3'e kadar olan eklemleri prizmatik ve birbirine karşılıklı diktir. Bu konfigürasyon için ters kinematik çözüm önemsizdir. Çok büyük gezer köprülü vinçlere benzer robotlar inşa edilebilir. Dezavantajı, yapılacak olan uygulamaya ait tüm besleyicilerin ve armatürlerin robotun "içinde" bulunması gerektiğidir. Robotun destek yapısının boyutu, fikstürlerin ve sensörlerin boyutunu ve yerleşimini sınırlar. Bu sınırlamalar Kartezyen robotların mevcut iş alanlarına uyarlanmasını zorlaştırmaktadır (Craig, 2004). Şekil 3.32'de Kartezyen robota ait kinematik zincir, çalışma uzayı ve uygulama örneği olan EPSON robotu gösterilmektedir.

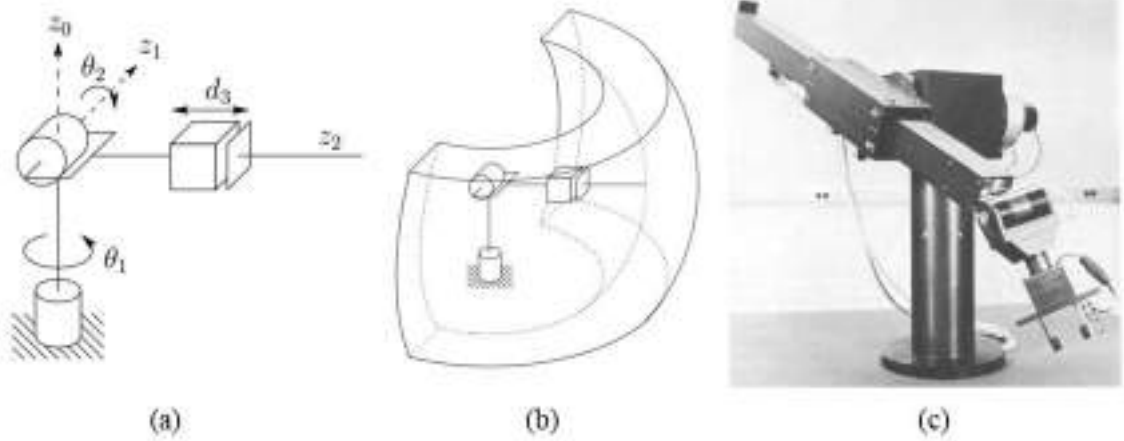
### 3.4.2. Silindirik robotlar



Şekil 3.33. Silindirik robot (a) Kinematik zincir (b) Çalışma uzayı (c) Uygulama örneği Seiko RT3300 Robot (Spong ve diğerleri, 2004)

Silindirik robotlarda, ilk eklem döner mafsalsal, ikinci ve üçüncü eklemler prizmatik mafsaldır. Çalışma uzayı olarak silindirik bir alan tararlar. Şekil 3.33'te Silindirik robota ait kinematik zincir, çalışma uzayı ve uygulama örneği olan Seiko RT3300 robotu gösterilmektedir.

### 3.4.3. Küresel robotlar

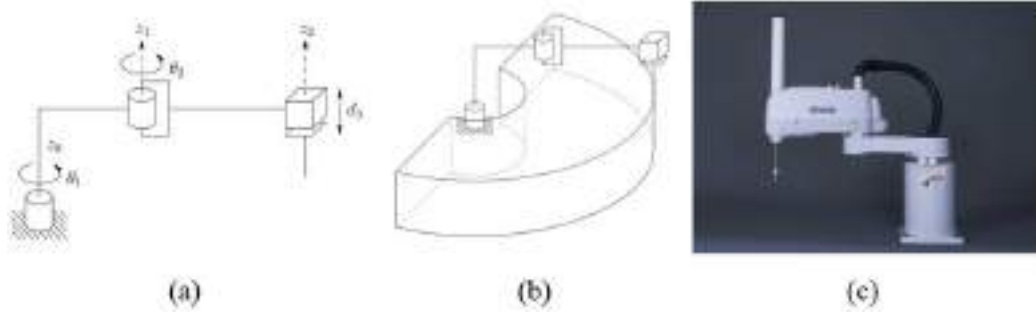


Şekil 3.34. Küresel robot (a) Kinematik zincir (b) çalışma uzayı (c) Uygulama örneği Stanford Arm Robotu (Spong ve diğerleri, 2004)

Küresel robotların ilk iki eklemi döner mafsalsal, üçüncü eklemi ise prizmatik mafsala sahiptir. Şekil 3.34 a'da eksenlere bakıldığı zaman  $z_0$ ,  $z_1$  ve  $z_2$  bir noktada kesişmektedir. Çalışma uzayı olarak küresel bir alan tararlar. Şekil 3.34'te Küresel robota ait kinematik zincir, çalışma uzayı ve uygulama örneği olan Stanford Arm robotu gösterilmektedir(Spong ve diğerleri, 2004).



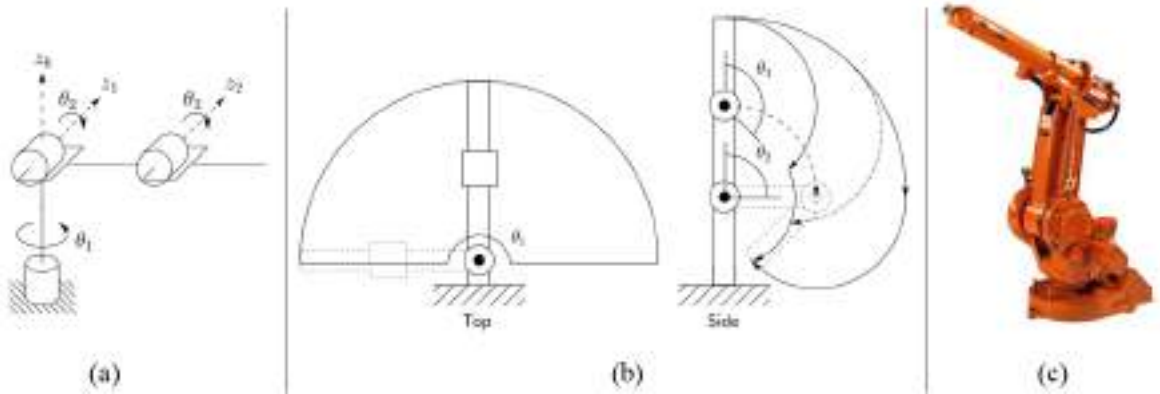
### 3.4.4. SCARA robotlar



Şekil 3.35. SCARA robot (a) Kinematik zincir (b) çalışma uzayı (c) Uygulama örneği Epson E2L653S SCARA Robot (Spong ve diğerleri, 2004)

SCARA Robotunun adı, “Selective Compliant Assembly Robot Arm (Seçici Uyumlu Montaj Robot Kolu)” veya “Selective Compliant Articulated Robot Arm (Seçici Uyumlu Eklemlili Robot Kol)” kelimelerinin kısaltılmış halinden gelmektedir (SCARA Robotu). SCARA robotunda ilk iki eklemler döner mafsaldan, üçüncü eklemler ise prizmatik mafsaldan oluşmaktadır. Şekil 3.35 a’da gösterildiği gibi eksenler ( $z_0$ ,  $z_1$  ve  $z_2$ ) birbirine paraleldir. Şekil 3.35’te SCARA robota ait kinematik zincir, çalışma uzayı ve uygulama örneği olan Epson E2L653S SCARA robotu gösterilmektedir(Spong ve diğerleri, 2004).

### 3.4.5. Dikey mafsallı robotlar



Şekil 3.36. Dikey Mafsallı robot (a) Kinematik zincir (b) çalışma uzayı (c) Uygulama örneği ABB IRB1400 Robot (Spong ve diğerleri, 2004)

Dikey mafsallı robotlara ayrıca “Revolute” veya “Antropomorfik “Robotlar da denir. Üç eksen de döner mafsala sahiptir. Şekil 3.36’da eksenlere bakıldığında  $z_1$ ,  $z_2$ ’ye paraleldir ve  $z_0$   $z_1$  ve  $z_2$ ’ye diktir. Çalışma uzayı olarak küresel şekle benzer bir alan taramaktadırlar. Şekil 1’de dikey mafsallı robota ait kinematik zincir, çalışma uzayı ve uygulama örneği olan ABB IRB1400 robotu gösterilmektedir(Spong ve diğerleri, 2004).

### 3.4.6. Paralel robotlar



Şekil 3.37. Uygulama örneği ABB IRB940 Tricept Paralel Robotu (Spong ve diğerleri, 2004)

Paralel robot, bağlantıların kapalı bir zincir oluşturduğu manipülatördür. Daha spesifik olarak, paralel bir manipülatörün, tabanı uç uca bağlayan iki veya daha fazla bağımsız kinematik zinciri vardır. Şekil 3.37, paralel bir konfigürasyona sahip ABB IRB 940 Tricept robotunu göstermektedir. Paralel robotların kapalı zincir kinematiği, açık zincirli robotlardan daha fazla yapısal rijitliğe sahip olduğundan daha fazla hassasiyete neden olabilir. Paralel robotların kinematik açıklaması temelde seri link robotlarından farklıdır ve bu nedenle farklı analiz yöntemleri gerektirir(Spong ve diğerleri, 2004).

## 3.5. Endüstriyel Robotlarda Önemli Parametreler

Endüstriyel bir robotun, görevlerinin yürütülmesi robotun etkinliğini belirleyecek başlıca performans ölçütü hassasiyettir.

### 3.5.1. Hassasiyet

Robot kollarının performans ölçütlerinden olan hassasiyet, hareket esnasındaki en küçük değişim aralığını ifade eder. Hassasiyet: doğruluk, çözünürlük ve tekrarlanabilirlik olarak alt başlıklarda incelenir.

### Doğruluk

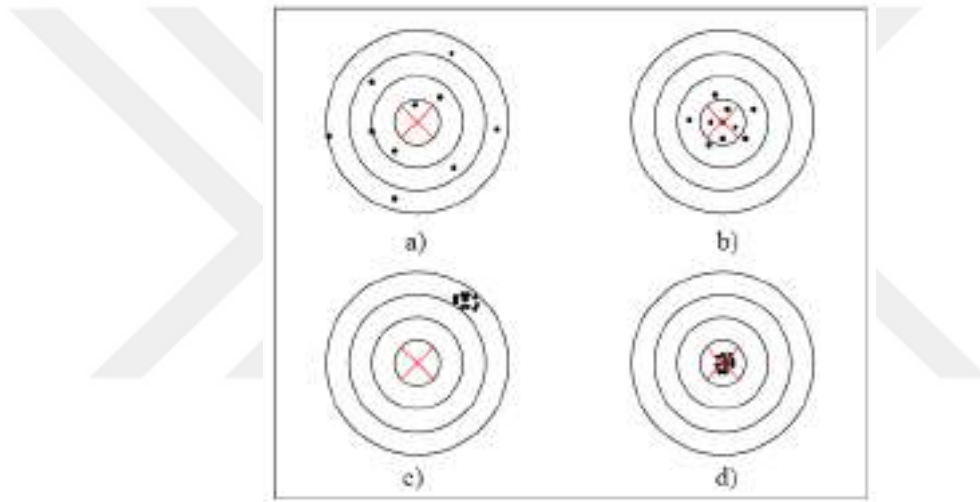
Robot koluna verilen komut ile elde edilen komut arasındaki farktır. Robot kolundaki uç işlevcinin çalışma uzayı içerisinde istenilen şekilde konumlandırma kabiliyetidir.

## Çözünürlük

Robot kolunun hareket adım aralığındaki en küçük değişimi ifade eder. Adım aralığı küçüldükçe çözünürlük artar. Dolayısıyla adım büyüklüğü ile çözünürlük ters orantılıdır.

## Tekrarlanabilirlik

Robot kolunun tekrarlı komut neticesinde aynı işi yapabilme yeteneğidir. Tekrarlanabilir testinde, robot koluna verilen görevde aynı koşullar altında, defalarca aynı pozisyona gitmesi istenir. Daha sonra istenilen pozisyonla gidilen pozisyon arasındaki hata miktarı belirlenir. Şekil 3.38’de tekrarlanabilirlik ve doğruluk arasındaki ilişki gösterilmektedir.



**Şekil 3.38.** Tekrarlanabilirlik ve Doğruluk: a) düşük tekrarlanabilirlik ve düşük doğruluk, b) düşük tekrarlanabilirlik ve yüksek doğruluk, c) yüksek tekrarlanabilirlik ve düşük doğruluk, d) yüksek tekrarlanabilirlik ve yüksek doğruluk (Joubair, 2014)

#### 4. MATERYAL VE YÖNTEM

Robot kolunun tasarlanmasındaki süreçte öncelikle robot kolunun hangi görevlerde kullanılacağı ve taşıyabileceği yük miktarı belirlendi. Robot kolunun gövdesinde kullanılan malzemenin cinsi belirlendi. Daha sonra sırasıyla:

- Bilgisayar destekli çizim programında gövdesi ve uç işlevci tasarlandı ve simülasyon edildi,
- Tasarlanan katı modelin mekanik ve elektriksel montajı yapıldı,
- Düz kinematik denklemleri “Denavit Hartenberg” yöntemiyle elde edildi,
- Ters kinematik denklemleri elde edildi,
- Yörünge planlaması yapıldı,
- Dinamik denklemleri “Lagrange Euler” yöntemiyle elde edildi,
- Mikrodenetleyici olarak “Arduino Uno” kullanılarak kontrolü sağlandı,
- Robot kolunun tekrarlanabilirlik ve doğruluk testi yapıldı.



Şekil 4.1. Robot kolu genel görünüm

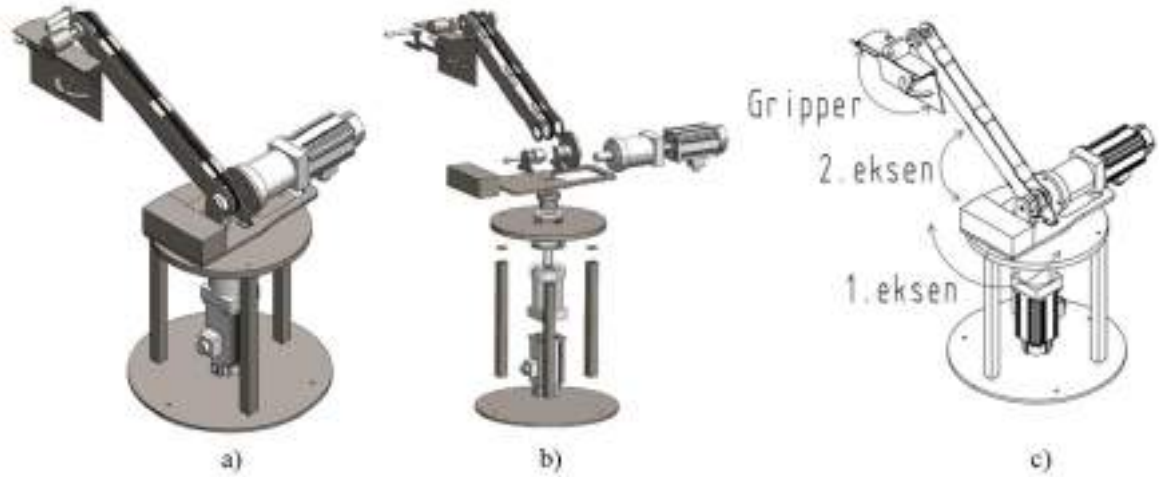
Çalışmada kullanılan robot kolunun katı modelinin genel görünümü Şekil 4.1’de gösterilmiştir.

##### 4.1. Robot Kolu Tasarımı

Bu bölümde tasarım hedefi belirlenen robot kolunun sırasıyla dış gövde tasarımı, pano ve mikro denetleyicinin montajı, kinematik modelleme ve dinamik modelleme üzerinde durulmuştur.

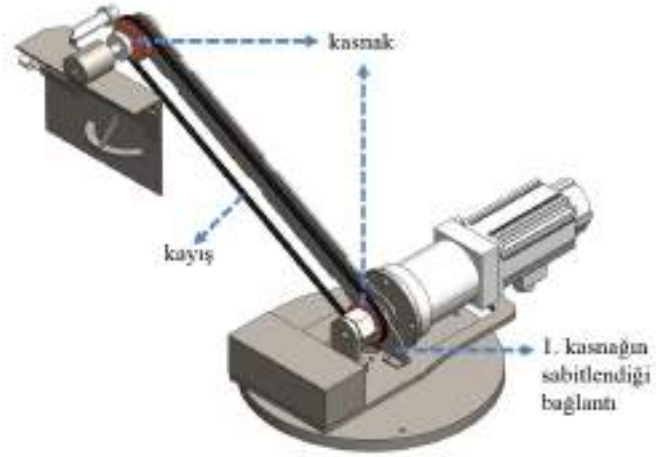
#### 4.1.1. Dış gövde sanal ve fiziksel montaj

Küresel çalışma uzayına sahip, tut-yerleştir görevini yapacak ve kulaklı ve konik cisimleri kavramak için özel açisal hareket mekanizmalı uç tutucu elemana sahip robot kolu tasarımı amaçlandı. Bilgisayar destekli katı modelleme programı olan Solidworks'te tasarımı yapıldı ve simülasyon yapılarak montaj hataları giderildi



Şekil 4.2. a) Robot kolu genel sanal görünüm b) Robot kolu patlatılmış montaj görünüm c) Eksenler ve uç işlevci

Bilgisayar ortamında katı modellemesi yapılan robot kolu Şekil 4.2'de gösterilmiştir. Eksenlerde döner mafsalları kullanıldı. Tahrik 2 adet AC sabit mıknatıslı senkron servo motorla yapılmaktadır. Ayrıca robot kolu endüstriyel ortamda kullanılacağından motorlar IP67 sertifikasına sahip olup suya, toza ve darbelere karşı dayanıklıdır. Servo motorun çıkış torkunu arttırmak ve robot kolunun daha fazla yük kaldırabilmesi için 2 adet planet dişli redüktör kullanıldı. Robot kolunda kullanılan redüktör ve servo motor hesaplamalarına sonraki bölümlerde değinilmiştir.



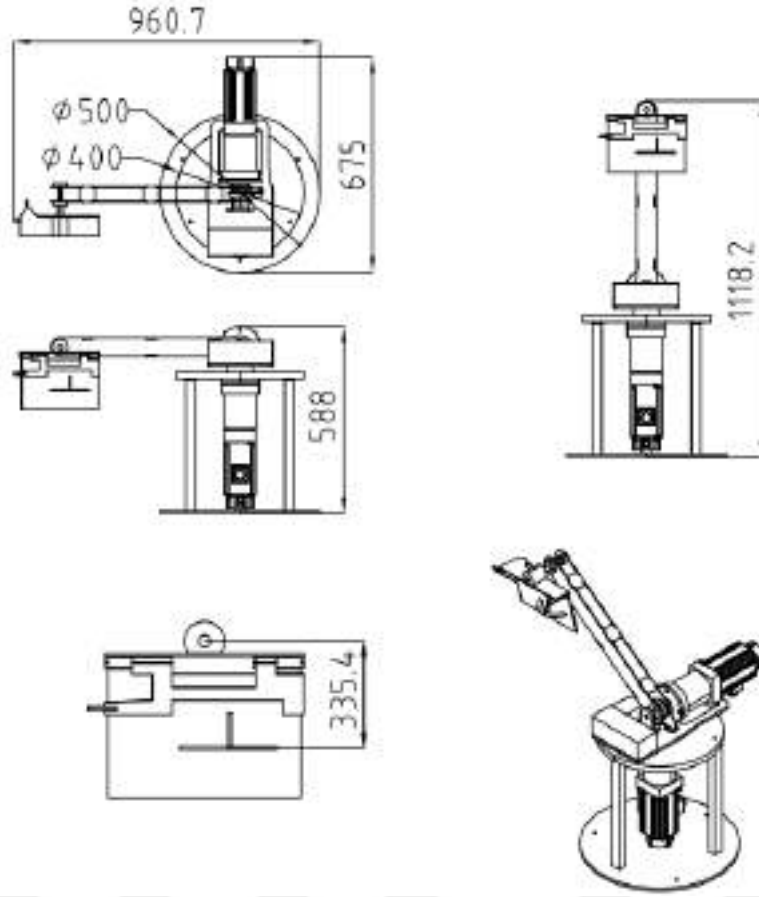
**Şekil 4.3.** Kayış kasnak Sistemi

Şekil 4.3'te gösterildiği gibi robotun 2. ekseninden uç işleveye kadar olan kısmında doğrudan aktarma sistemlerinden biri olan dişli kayış ve kasnak sistemi kullanıldı. Sistemde kullanılan birinci kasnak sabitken ikinci kasnak rulmanlarla yataklanmış olarak hareketlidir. Kayış kasnak sisteminin zincir dişli sistemine göre avantajı hafif ve sessiz çalışmasıdır.



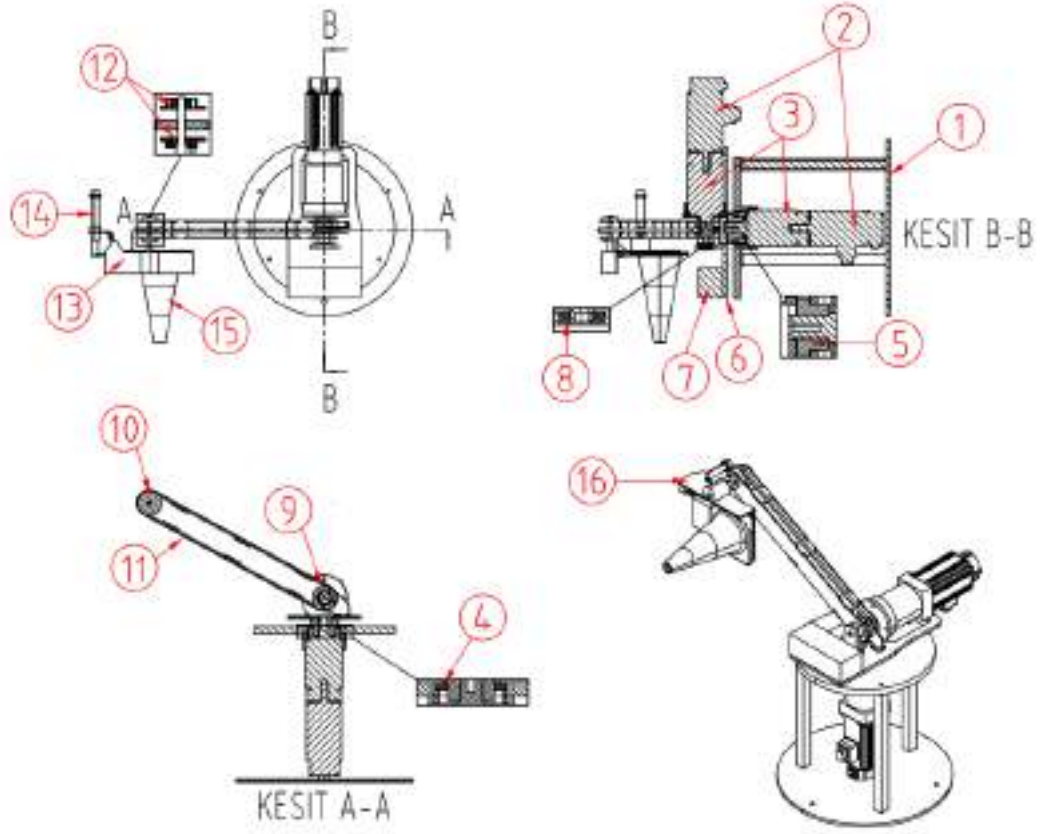
**Şekil 4.4.** Robot kolu dış gövde montajlanmış halı

Robot kolunun sanal ortamda montaj hataları giderildikten sonra imalat aşamasına geçildi. Gövde kısmında St52 malzeme tercih edildi. Kullanılan malzemenin özelliklerine sonraki bölümlerde değinilmiştir. Lazer kesim yapılan plakalar bağlantı elemanları ve kaynak işlemi uygulanarak montajlandı. Korozyonu önlemek ve estetik açıdan iyileştirmek için elektrostatik boya kullanıldı. Montajı tamamlanan robot kolunun gövdesi Şekil 4.4'te gösterilmiştir.



Şekil 4.5. Robot kolu dış ebatlar

Robot koluna ait genel ölçüler milimetre cinsinden Şekil 4.5'te teknik resimde gösterilmiştir. Şekil incelendiğinde robotun kapladığı en büyük hacim 960,7x675x1118.2'dir.



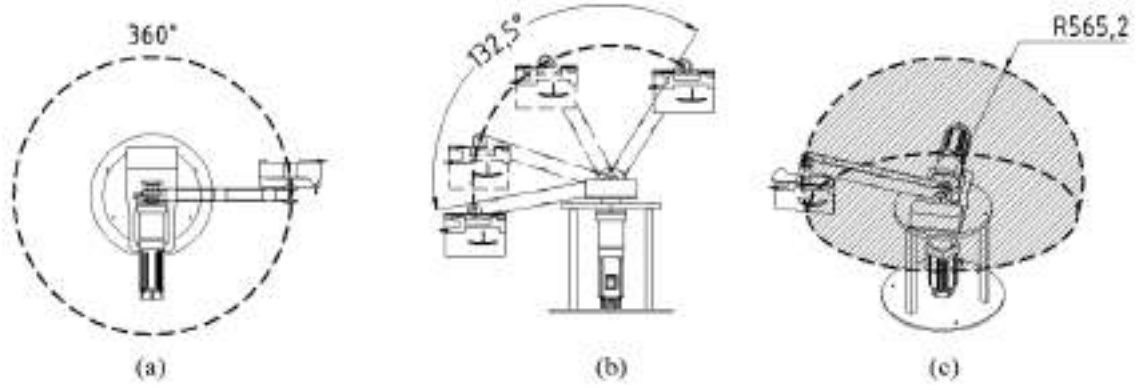
Şekil 4.6. Robot kolu bileşenleri teknik resim

Robot kolunun bileşenleri teknik resmi Şekil 4.6’da gösterilmiştir. Bileşenlerin adları ve özellikleri Tablo 4.2’de gösterilmiştir.

Tablo 4.1. Robot kolu bileşenleri ve özellikleri

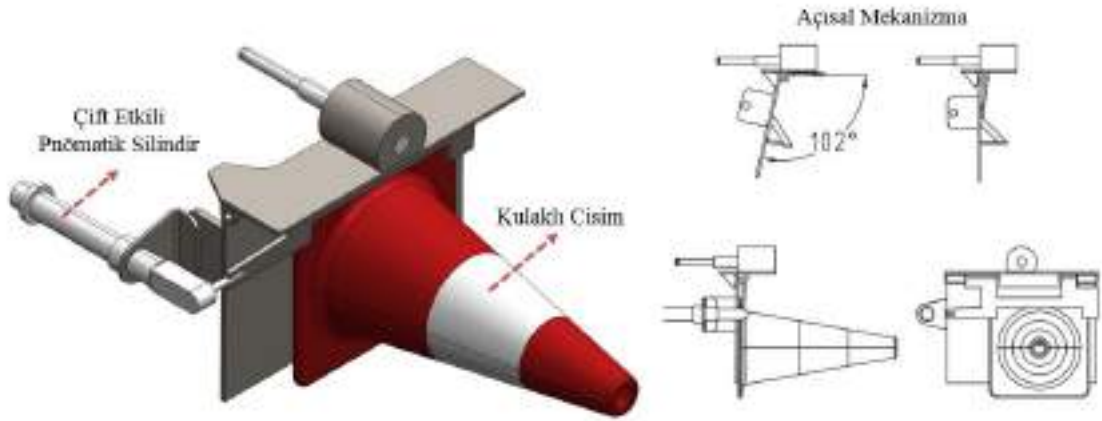
Bileşenler	Özellikleri	Bileşenler	Özellikleri
1) Ana Gövde	Taban Parçası, 3 Adet Taşıyıcı Kolon, Üst Parça	7) Dengeleyici Ağırlık	
2) AC Servo Motor	Panasonic Minas-A5, MSME204G1G, 2kW, 3000 dev/dakika, 6,37 N.m	8) Kol Rulman-1	15x32x8
3) Redüktör	1/25, APEX DYNAMICS, PE-120	9) Kol Kasnak-1	
4) 1. Eksen Rulman	65x100x11	10) Kol Kasnak-2	
5) 1. Eksen Ara Bağlantı Parçası		11) Kayış	1350 mm, 270 diş
6) 1. Eksen Tabla		12) Kol Rulman-2	10x25x8





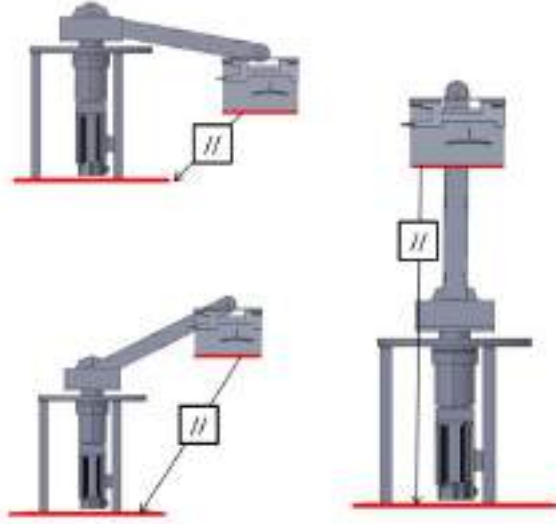
Şekil 4.7. Robot kolu eksenler sınır şartları: a) 1. eksen, b) 2. eksen ve c) genel

Robot kolunun sınır şartları Şekil 4.7’de gösterilmiştir. İlk eksen 360° dönebilirken ikinci eksen 132,5°'lik dönme kabiliyetine sahiptir. Sisteme genel olarak bakılırsa yarıçapı 565,2 mm’lik yarıçapta küresel bir çalışma alanına sahiptir.



Şekil 4.8. Uç işlevcinin yapısı

Robot kolunda açısal hareket edebilen aç kapa mantığıyla çalışan mekanizma kullanıldı. Bu mekanizma çift etkili silindir vasıtasıyla tahrik edilmektedir. Kulaklı cisimleri kavraması amaçlanmıştır. Şekil 4.8’de pnömatik uç işlevciye ait genel bir görsel ve mekanizma çalışma mantığı gösterilmektedir.



**Şekil 4.9.** Robot kolu tutucunun hareket esnasında zeminle paralellığı

Tasarlanan uç tutucu elemanın en önemli özelliği, Şekil 4.9’da gösterilen herhangi bir nesneyi bir yerden başka bir yere taşıırken hangi pozisyonda aldıysa o pozisyonda daima zemine paralel kalabilmesini sağlamasıdır. Bu sayede nesnenin pozisyonu bozulmadan istenilen hedefe yerleştirilebilmesi mümkün olmaktadır. Paralellik durumu kayış kassak sistemi ile gerçekleştirilmektedir. Ayrıca kayış kassak sistemi sayesinde aynı iş 3 eksen yerine 2 eksen kullanılarak bir eksen tasarruf sağlanmıştır.

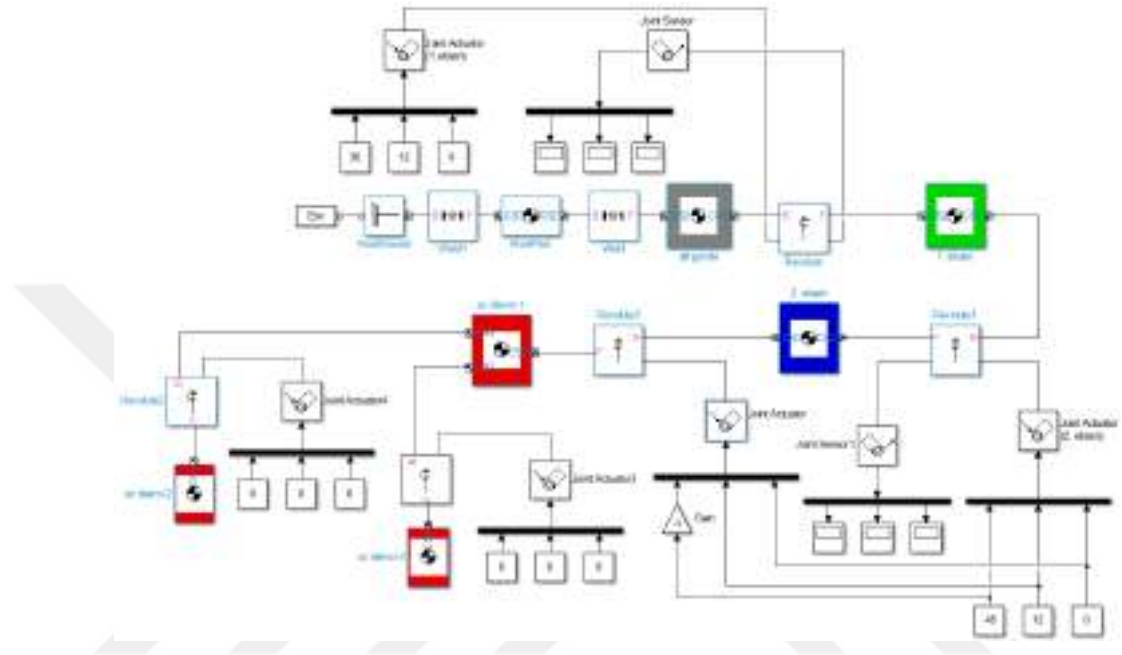


**Şekil 4.10.** Robot kolu farklı uç işlevci seçenekleri

Robot koluna istenilen işe göre Şekil 4.10’da gösterildiği gibi farklı uç işlevci modelleri uyarlanabilir. Şekilde sırasıyla paralel iki parmaklı, açışal üç parmaklı ve açışal iki parmaklı uç işlevci gösterilmiştir. Her biri farklı amaçlar için kullanılabilecek tasarıma sahiptir.

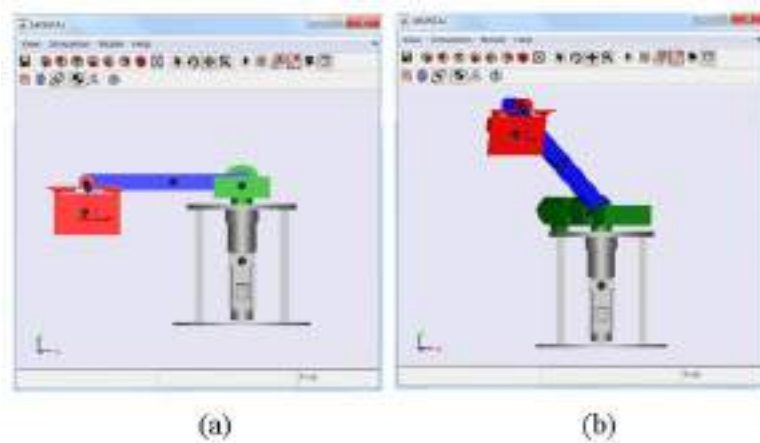
## Görsel simülasyon (Matlab Simulink-Simmechanics)

Tasarımı ve montajı biten robot kolu ayrıca Solidworks'ten Matlab Simulink kısmına SimMechanics Link'i kullanılarak entegre edildi. Matlab SimMechanics ortamında robot kolunun mafsal ve uzuvlarının blok diyagramları oluşturuldu. Oluşturulan blok diyagramı Şekil 4.11'de görülmektedir.



Şekil 4.11. Robot kolunun Matlab Simulink blok diyagramı

Blok diyagramında eklem yerlerine tahrik verilerek robot kolunun görsel simülasyonu oluşturuldu. Robot kolunun simülasyon çalıştırdıktan sonra oluşan görsel Şekil 4.12'de gösterilmiştir. Şekil birinci eksen için  $30^\circ$  ve ikinci eksen için  $-45^\circ$  verilerek oluşturulmuştur. Oluşturulan bu simülasyon daha sonra yapılacak farklı çalışmalar için hazır hale getirildi.



Şekil 4.12. Robot kolu simülasyonu: a) başlangıç konumu ve b) bitiş konumu

#### 4.1.2. Malzeme seçimi

Robot kolunun genel gövdesi için çelik türü olan St 52 malzeme seçildi. Bu malzemeye ait mekanik ve kimyasal özellikleri Tablo 4.2 ve 4.3'te gösterilmiştir. Bu çeliğin St52 olarak adlandırılmasının sebebi 52 kg/mm çekme dayanımına sahip olmasıdır. St52 çelik kalitesi düşük karbonlu, yüksek mukavemetli yapısal çeliktir. Düşük karbon miktarı soğuk şekillendirmede kolaylık sağlar. Ayrıca bu çeliklerin kaynak performansı yüksektir. Genellikle normal makine parçaları, vinç, şasi, köprüler ve mobil ekipmanlarda kullanılır. (St 52 Çeliği)

**Tablo 4.2.** St52 çeliğinin mekanik özellikleri (Karanjule, Bhamare ve Rao, 2017)

Malzeme	Yoğunluk (kg/mm <sup>3</sup> )	Elastise Modülü (kN/mm <sup>2</sup> )	Uzama (%)	Poisson Oranı	Çekme Dayanımı (N/mm <sup>2</sup> )	Akma Dayanımı (N/mm <sup>2</sup> )
St52	7.87	210	32.6	0.3	602	522

**Tablo 4.3.** St52 çeliğinin kimyasal özellikleri (Karanjule ve diğerleri, 2017)

Malzeme	C (%)	Fe (%)	Mn (%)	Si (%)	P (%)	S (%)
St52	0.2	-	1.6	0.5	0.025	0.025



**Şekil 4.13.** St 52 malzemenen üretilen robot kolu gövdesi

Şekil 4.13'te robot kolunun gövdesinde kullanılan St52 malzemenin montajı tamamlanıp boyama aşamasına geçmeden önceki hali gösterilmektedir.

#### 4.1.3. Motor hesaplamaları

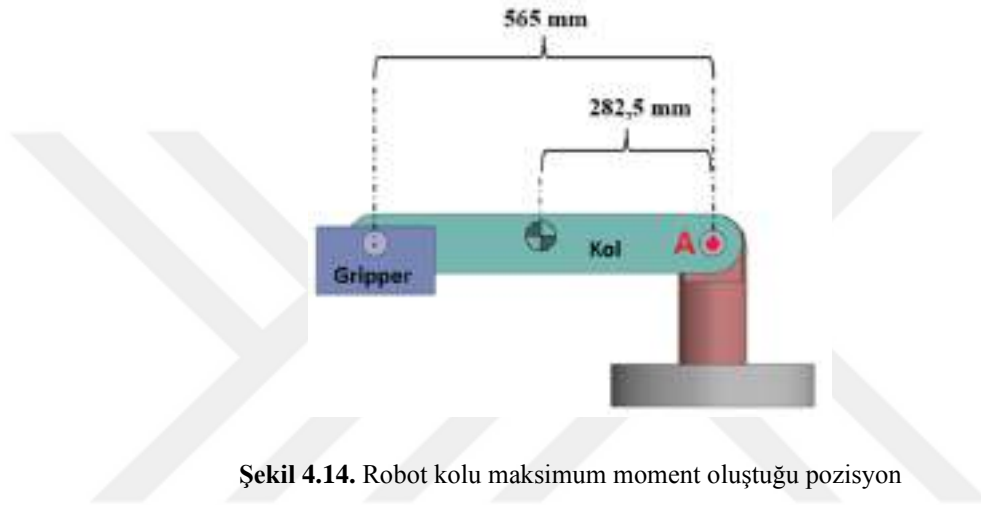
Motor seçiminde robot kolunun kaldıracağı yük miktarı, hassasiyet, endüstride kullanılabilirlik ve kontrol edilebilirlik parametreleri dikkate alınarak seçim yapılmıştır. Robot kolunda kullanılan ve özellikleri aynı olan iki adet AC servo motor ve redüktöre ait teknik bilgiler Tablo 4.4 ve 4.5'te gösterilmiştir.

**Tablo 4.4.** Kullanılan AC Servo motor özellikleri

Eksen	Motor Tipi	Motor Gücü (kW)	Hız (dev/dk)	Motor Torku (N.m)
1 ve 2	Panasonic Minas A5 (MSME204G1G)	2	3000	6,37

**Tablo 4.5.** Kullanılan redüktör özellikleri

Redüktör Tipi	Redüktör Oranı
Planet Dişli Redüktör (Apex Dynamics PEII120)	1/25



**Şekil 4.14.** Robot kolu maksimum moment oluştuğu pozisyon

Robot kolunun istenilen yükü ve gövdesinden kaynaklı motora binen yükleri kaldırabilmesi için eklemlere yeterli miktarda tork veren motor seçilmelidir. Şekil 4.14'te robot kolunun sadeleştirilmiş hali gösterilmektedir. Denklem 4.1'de ikinci motorun bağlandığı noktadan (A noktası) moment alınarak uç işlevcide oluşan maksimum moment hesaplandı. Hesaplama robot kolunu oluşturan elemanların kütleli dağılımı  $m_{kol}=3,2$  kg ve  $m_{gripper}= 3,8$  kg şeklindedir. Uç işlevcinin taşıyacağı parça  $m_{cisim}= 5$  kg olarak kabul edilmiştir.

$$\begin{aligned}\sum MA &= (0,285m) \cdot (3,2kg) \cdot (9,801) \\ &+ (0,565 m) \cdot (3,8 kg) \cdot (9,801) \\ &+ (0,565 m) \cdot (5 kg) \cdot (9,801) \\ &= 57,67 N.m\end{aligned}\quad (4.1)$$

Denklem 4.2'de motora ait açısal hız, Denklem 4.3'te motorun gücü gösterilmiştir.

$$\omega_{motor} = (2 \cdot \pi \cdot n / 60) = (2 \cdot \pi \cdot 3000) / 60 = 314,159 \text{ rad/s} \quad (4.2)$$

$$\begin{aligned} P_{motor} &= M \times \omega_{motor} = (6,37) \times (314,159) \\ &= 2001,19 \text{ watt} = 2KW \end{aligned} \quad (4.3)$$

Denklem 4.1 de elde edilen 57,67 (N.m) tork değeri olup motorun sağladığı 6,37 (N.m) tork değerinden büyük olduğundan motor tek başına istenilen ağırlığı kaldıramaz. Bundan dolayı 1/25 redüktör seçimi yapılarak motorun sağladığı tork değeri artırılmıştır. Redüktör eklenmesi sonucu oluşan açısız hız ve devir Denklem 4.4 ve Denklem 4.5'te gösterilmiştir.

$$\omega_{Planet Dişli Çıkış} = 314,159 / 25 = 12,566 \text{ rad/s} \quad (4.4)$$

$$n_{Planet Dişli} = 3000 / 25 = 120 \text{ dev/dk} \quad (4.5)$$

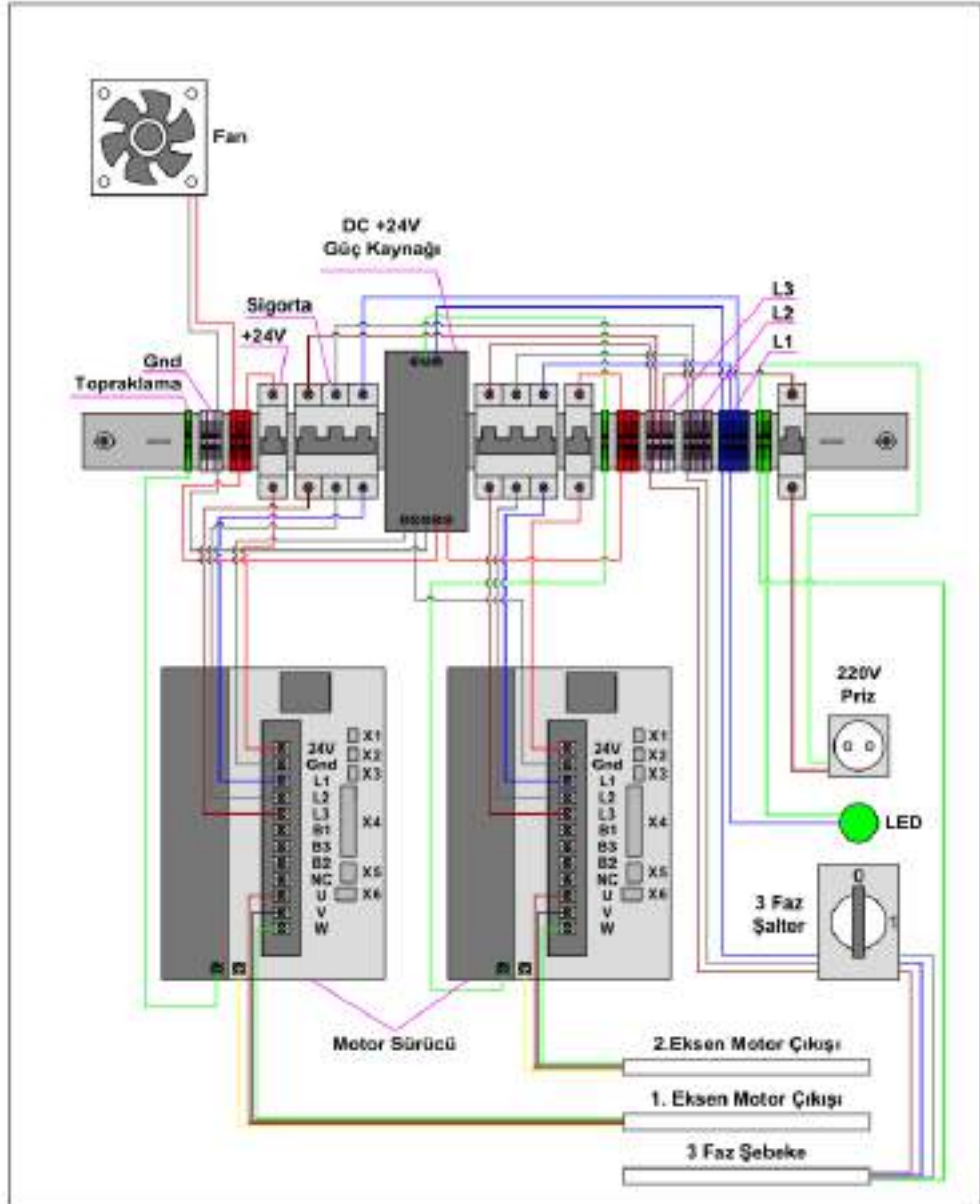
Redüktörün etkisiyle sistemin çıkış torku, Denklem 4.6'da gösterildiği gibi 6,37 (N.m)'den 159,254 (N.m) torka yükselmiştir. Robot kolunun ortalama uzunluğu 60 cm olarak kabul edilirse, Denklem 4.7'de gösterildiği gibi 265,423 N kuvvet oluşurken Denklem 4.8'de gösterildiği gibi yaklaşık 27 kg'lık yük için gerekli tork kapasitesini üretebilmektedir. Böylece motor 5 kg'lık yükü rahatlıkla kaldırabilecek tork sağlamaktadır.

$$T_{redüktör çıkış} = \frac{P_{motor}}{\omega_{Planet Dişli Çıkış}} = 2001,19 / 12,566 = 159,254 \text{ N.m} \quad (4.6)$$

$$F_{Kol(60cm=0,6m)} = \frac{T_{redüktör çıkış}}{L_{Kol(m)}} = 159,254 / 0,6 = 265,423 \text{ N} \quad (4.7)$$

$$Kolun taşıyabileceği kütle = 265,423 / 9,806 = 27,06 \text{ kg} \quad (4.8)$$

#### 4.1.4. Pano ve mikrodenetleyicinin montajı



Şekil 4.15. Pano elemanları yerleşim şeması

Robot kolunda AC servo motorlara ait sürücüler, güç kaynağı, sigortalar, led, 3 faz şalter ve fan gibi elektriksel elemanlarının sanal ortamda pano içerisinde yerleşimi Şekil 4.15'te gösterilmektedir.



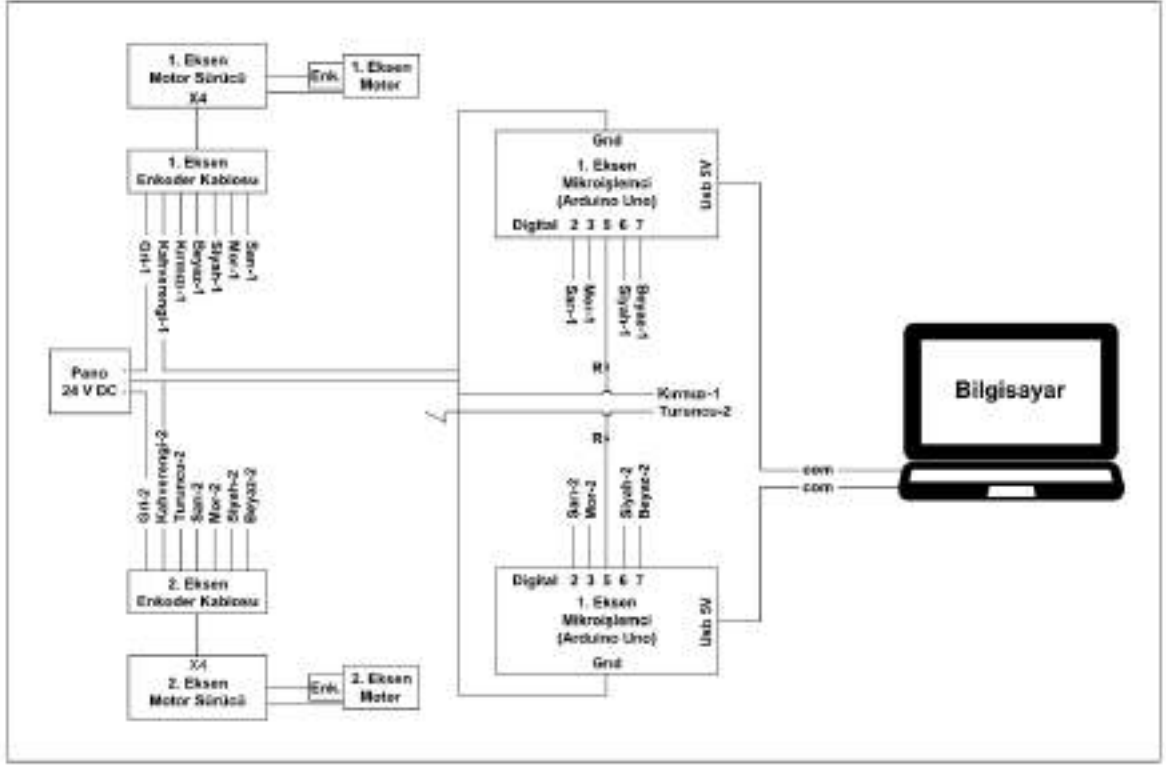
**Şekil 4.16.** Pano montajlanmış hali

Pano içerisindeki elemanların su, sıcaklık gibi etmenlerden korumak için Şekil 4.16'da gösterilen kapaklı plastik panoya montajlanmış hali gösterilmektedir. AC servo motora ait sürücünün bilgileri Tablo 4.6'da gösterilmektedir.

**Tablo 4.6.** AC Servo motor sürücü bilgileri

<b>Üretici Firma</b>	Panasonic
<b>Parça Numarası</b>	MFDHT5440
<b>Ağırlık</b>	5 kg
<b>Gerilim- Tedarik</b>	380 ~ 480 V AC
<b>Çalışma Sıcaklığı</b>	0°C ~ 55°C
<b>Kullanıldığı Ürün</b>	MINAS A5 Series
<b>Watt- Yük</b>	5 kW
<b>Gerilim- Yük</b>	480 V



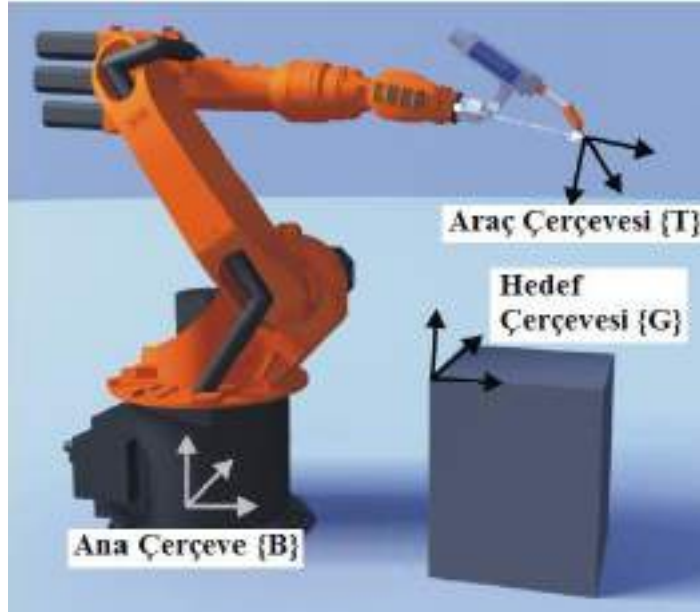


Şekil 4.17. Sürücü ve mikroişlemcinin bağlantı şeması

Motoru kontrol etmek için Arduino Uno mikroişlemcisi kullanıldı. Diğer mikroişlemciler yerine tercih edilmesinin sebebi açık kaynaklı olması, hesaplı olması, arayüzünün anlaşılır olması ve literatürde birçok kaynak bulunmasıdır. Mikro işlemcinin bağlantı şeması Şekil 4.17’de gösterilmektedir.

## 4.2. Kinematik Modelleme

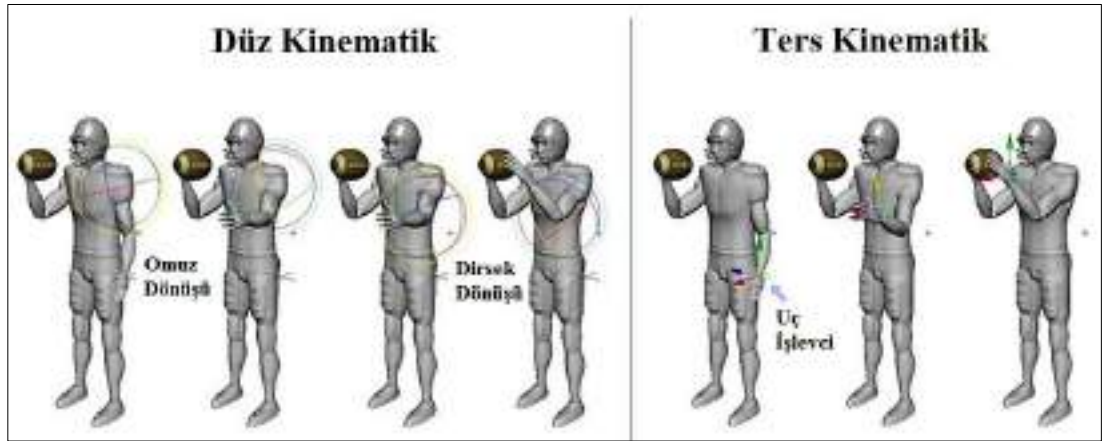
Kinematik, nesneye etki eden kuvvetlere bakmaksızın hareketi ele alan bilimdir. Kinematik bilimi, hareketli cisimlerin pozisyonunu, hızını, ivmesini ve tüm yüksek dereceli türevlerini (zamana veya diğer değişkenlere göre) inceler(Craig, 2004).



Şekil 4.18. Endüstriyel robot koordinat sistemleri (“Simatic S7-1500”, t.y.)

Kinematik hesaplamalar yapılırken koordinat sistemlerine gereksinim vardır. Robot kollarında kullanılan başlıca koordinatlar Şekil 4.18’de gösterilmiştir. Ana çerçeve robotun sabit yeri olan taban noktasında, araç çerçeve uç işlevcide ve hedef çerçevede işlem yapılacak parçanın üzerine tanımlanmaktadır.

Robotların düz kinematikinde ana çerçeveden araç çerçeveye olan vektör ele alınırken, ters kinematikte ise araç çerçevesinden ana çerçeveye olan vektör ele alınır.



Şekil 4.19. Ters ve düz kinematik örnek (Boggus, t.y.)

Düz kinematikte sisteme giriş verileri olarak eklemlerdeki açılar verilir çıkış verisi olarak uç işlevcinin pozisyonu ve yönelimi elde edilir. Ters kinematikte ise durum tam tersidir. Şekil 4.19’da bir eliyle topu tutan insan diğer eliyle topu kavramak için önce omuz sonra dirsek eklemine döndürmektedir. Eklemlere verilen açı neticesinde eli topa

ulaşmaktadır. Bu eylem düz kinematiğe örnektir. Diğer taraftan elin (uç işlevcinin) topu tutuğu konumunun bilinmesinden yola çıkarak eklemlere verilmesi gereken açı değerlerinin hesaplanması ters kinematiğe örnektir. Özetle Tablo 4.7’de karşılaştırılması gösterilmiştir.

**Tablo 4.7.** Düz ve ters kinematik karşılaştırılması

<b>Kinematik</b>	<b>Girdi</b>	<b>Çıktı</b>
<i>Düz Kinematik</i>	Eklem Açıları	Uç işlevci pozisyonu ve yönelimi
<i>Ters Kinematik</i>	Uç işlevci pozisyonu ve yönelimi	Eklem Açıları

Bu bölümün ana konusu, robot kolundaki uç işlevcinin robotun ana eksen takımına göre eklem değişkenlerinin bir fonksiyonu olarak konumunu ve yönünü hesaplamaya ilişkin ters ve düz kinematik yöntemleri içerir.

#### **4.2.1. Düz (İleri) kinematik**

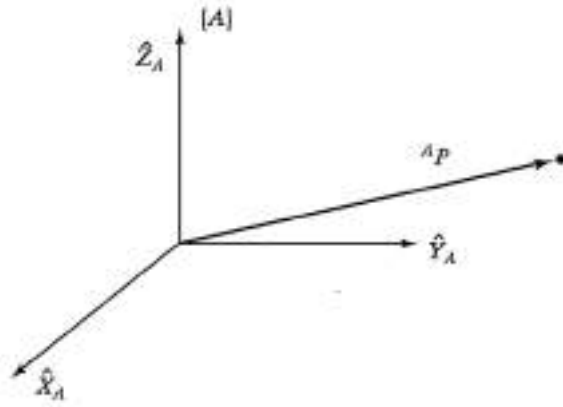
Robotların düz(ileri) kinematiğini hesaplanmasında kullanılan yöntemler arasında en çok tercih edilen yöntem Denavit-Hartenberg yöntemidir. Kısaca D-H yöntemi olarak gösterilir(Hartenberg ve Denavit, 1955).

Denavit ve Hartenberg 1955 yılında, her bir ekleme koordinat sistemi yerleştirerek komşu eklemler arasındaki dönüşüm ve dönme ilişkisini gösteren matris tanımlamıştır(Fu ve diğerleri, 1987). Elde edilen matris ile robot kolunun hareketi sonucunda uç işlevcinin pozisyonu ve yönelimi bulunmaktadır.

D-H yöntemine geçmeden önce dönüşüm matrisi kavramı bilinmesi gerekir. Bir robotun uzayda hareketi boyunca çevresindeki nesnelere olan konum ve yönelimini koordinat sistemlerine göre tanımlamak için dönüşümleri matrisleri kullanılır. “Dönüşüm matrisi” dönme matrisi ve konum vektöründen oluşur.

Konum vektörü, bir noktanın evrensel uzayda referans alınan eksen takımına göre tanımlanmasıdır. Şekil 4.20’de gösterilen P noktası A koordinat eksen takımına göre tanımlanmıştır. Gösterim olarak  ${}^A P$  şeklindedir. Matematiksel olarak  ${}^A P$  vektörünün ifade edilişi denklem 1.1 gösterilmiştir.

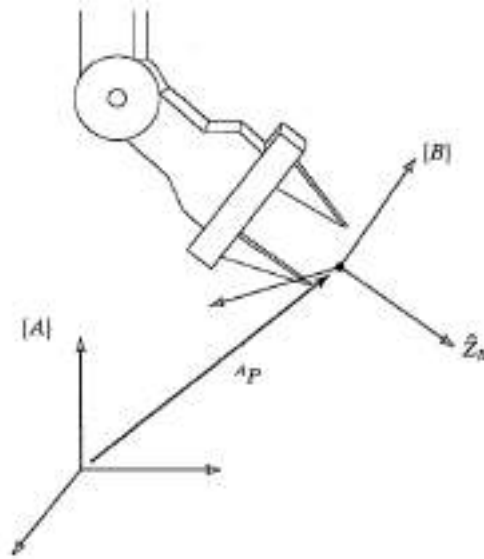
$${}^A P = \begin{bmatrix} A P_x \\ A P_y \\ A P_z \end{bmatrix} \quad (4.9)$$



**Şekil 4.20.** P noktasının {A} koordinat sistemine göre tanımlanması (Bingül ve Küçük, 2017a)

Yönelim ise bir koordinat sisteminin başka bir koordinat sistemini referans olarak ifade etmektir. Gösterim olarak  ${}^A R_B$  şeklindedir. Matematiksel olarak ifade edildiği denklem 4.10'da gösterilmiştir.

$${}^A R_B = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \quad (4.10)$$



**Şekil 4.21.** Uç işlevcinin yöneliminin referans koordinat sistemine göre tanımlanması (Craig, 2004)

Dönüşüm matrisinin matematiksel olarak ifade edilişi Denklem 4.11’de gösterilmektedir. Denklemde  ${}^A_B R$  yönelim matrisini,  ${}^A P_{BORG}$  ise konum vektörünü temsil eder. Şekil 4.21’de uç işlevciye ait koordinat sisteminin  $\{A\}$  koordinat sistemine göre tanımlanması gösterilmektedir.

$${}^A_B T = \begin{bmatrix} {}^A_B R & {}^A P_{BORG} \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & p_x \\ r_{21} & r_{22} & r_{23} & p_y \\ r_{31} & r_{32} & r_{33} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.11)$$

D-H yöntemindeki robot kolundaki her bir ekleme belirli kurallara göre eksen takımları eklenir. Komşu eklemler arasındaki ilişki  ${}^{i-1}_i T$  dönüşüm matrisleriyle ifade edilir. Ekleme oluşan dönüşüm matrislerinin art arda çarpılmasıyla uç işlevciye ait konum ve yönelim matrisi elde edilir (Bingül ve Küçük, 2017a).

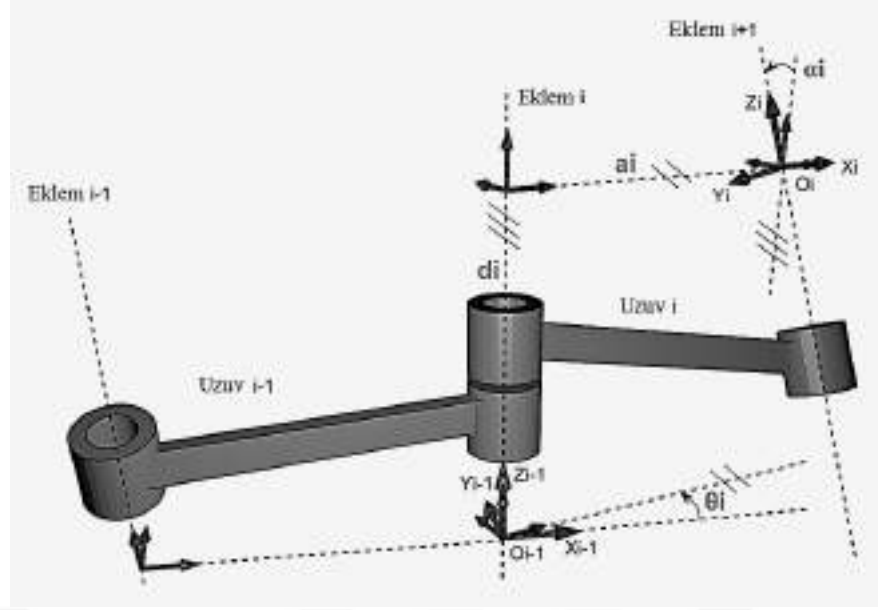
$${}^0_N T = {}^0_1 T {}^1_2 T \dots {}^{N-1}_N T \quad (4.12)$$

Konum matrisi (3x3) ve yönelim matrisinin (3x1) çarpılmasıyla elde edilen dönüşüm matrisi Denklem 4.13’te gösterilmiştir.

$${}^{i-1}_i T = R_x(\alpha_{i-1}) D_x(a_{i-1}) R_z(\theta_i) D_z(d_i)$$

$${}^{i-1}_i T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c\alpha_{i-1} & -s\alpha_{i-1} & 0 \\ 0 & s\alpha_{i-1} & c\alpha_{i-1} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a_{i-1} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} x \begin{bmatrix} c\theta_i & -s\theta_i & 0 & 0 \\ s\theta_i & c\theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} c\theta_i & -s\theta_i & 0 & a_{i-1} \\ s\theta_i c\alpha_{i-1} & c\theta_i c\alpha_{i-1} & -s\alpha_{i-1} & -s\alpha_{i-1} d_i \\ s\theta_i s\alpha_{i-1} & c\theta_i s\alpha_{i-1} & c\alpha_{i-1} & c\alpha_{i-1} d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.13)$$



**Şekil 4.22.** Denavit-Hartenberg kinematik parametreler (“Denavit–Hartenberg Parameters”, 2019)

D-H yönteminde kullanılan dört değişken Şekil 4.22’de gösterilmiştir. Bu değişkenlerin tanımı Tablo 4.8’de gösterilmiştir. Yöntemde kullanılan temel adımlar şunlardır(Bingül ve Küçük, 2017a):

- İlk olarak eklem yerleri tespit edilir. Döner eklemlerde z eksenini dönme yönü, prizmatik eklemlerde kayma yönü Z eksenini olarak atanır.
- Kol boyunca uzanan ve Z eksenine dik olarak X eksenini yerleştirilir.
- Y eksenini X ve Z eksenleri tanımlandıktan sonra sağ el kuralına göre tanımlanır.
- Genellikle Z eksenleri öyle bir yerleştirilir ki arka arkaya gelen eklemlerde X eksenini etrafında döndürüldüğünde bir sonraki eklemin Z eksenleri üst üste denk gelir.

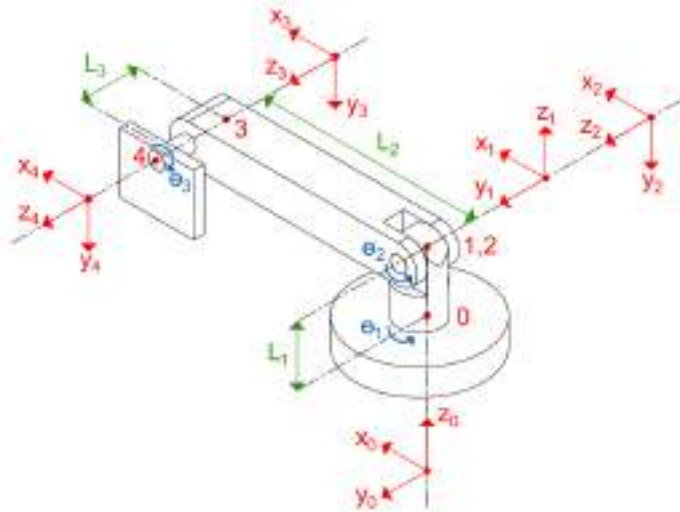
**Tablo 4.8.** D-H yöntemi dört ana değişken

$\alpha_{i-1}$	$Z_{i-1}$ ile $Z_i$ arasında $X_{i-1}$ boyunca ölçülen açıdır.
$a_{i-1}$	$Z_{i-1}$ ile $Z_i$ arasında $X_{i-1}$ boyunca belirlenen uzunluktur.
$d_i$	$X_{i-1}$ ile $X_i$ arasında $Z_i$ boyunca belirlenen uzunluktur.
$\theta_i$	$X_{i-1}$ ile $X_i$ arasında $Z_i$ boyunca ölçülen açıdır.

Değişkenler bulunduktan sonra D-H tablosu oluşturulur ve düz (ileri) ve ters kinematik hesaplamaları yapılır.

## Tasarlanan robot kolunun düz(ileri) kinematik hesaplamaları

Tasarlanan iki eksenli robot kolunun eksen atamaları daha rahat anlaşılabilmesi için Şekil 4.23'te sadeleştirilmiş hali gösterilmektedir. Eksenler döner eklemlerden oluşmaktadır. Birinci eklem ile ikinci eklemin dönme eksenini arasında  $90^\circ$  açı vardır. İkinci eklem yerinden üçüncü eklem yerine kasnaklı aktarma sistemi olduğundan  $\theta_2, \theta_3$ 'e bağlıdır ve birbirleri ile ters yönlü dönerler.



Şekil 4.23. Tasarlanan robot kolu eksen atamaları

Tasarlanan robot kolunun eksen atamaları yapıldıktan sonra D-H parametreleri Tablo 4.9'da yer almaktadır.

Tablo 4.9. D-H parametreleri tablosu

Eksen No	D-H Değişkenleri			
	$\alpha_{i-1}$	$a_{i-1}$	$d_i$	$\theta_i$
1	$0^\circ$	0	$L_1$	$\theta_1$
2	$-90^\circ$	0	0	$\theta_2$
3	$0^\circ$	$L_2$	0	$0^\circ$
4	$0^\circ$	0	$L_3$	$\theta_3 = -\theta_2$

Tablo oluşturulduktan sonra sırasıyla takip edilen işlemler şunlardır:

D-H değişkenleri bulunur.

$$\alpha_0 = 0, \alpha_1 = -\frac{\pi}{2}, \alpha_2 = 0, \alpha_3 = 0$$

$$a_0 = 0, a_1 = 0, a_2 = L_2, a_3 = 0$$

$$d_1 = L_1, d_2 = 0, d_3 = 0, d_4 = L_3$$

$$\theta_1 = \theta_1, \theta_2 = \theta_2, \theta_3 = 0, \theta_4 = -\theta_2 \quad (4.14)$$

Eklem dönüşüm matrisleri oluşturulur.

$${}^0_1T = \begin{bmatrix} \cos\theta_1 & -\sin\theta_1 & 0 & 0 \\ \sin\theta_1 & \cos\theta_1 & 0 & 0 \\ 0 & 0 & 1 & L_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^1_2T = \begin{bmatrix} \cos\theta_2 & -\sin\theta_2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -\sin\theta_2 & -\cos\theta_2 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^2_3T = \begin{bmatrix} 1 & 0 & 0 & L_2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^3_4T = \begin{bmatrix} \cos\theta_2 & \sin\theta_2 & 0 & 0 \\ -\sin\theta_2 & \cos\theta_2 & 0 & 0 \\ 0 & 0 & 1 & L_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.15)$$

Ana çerçeveden araç çerçeveye doğru T04 ileri yönlü robot kinematiği oluşturulur.

$${}^0_4T = \begin{bmatrix} \cos\theta_1 & 0 & -\sin(\theta_1) & -\sin(\theta_1)L_3 + \cos(\theta_1)\cos(\theta_2)L_2 \\ \sin\theta_1 & 0 & \cos\theta_1 & \cos(\theta_1)L_3 + \sin(\theta_1)\cos(\theta_2)L_2 \\ 0 & -1 & 0 & -\sin(\theta_2)L_2 + L_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.16)$$

T04 dönüşüm matrisindeki konum vektörleri, Denklem 4.16'daki matrisin üçüncü sütunundaki elemanlarıyla Denklem 4.17'de gösterildiği gibi oluşturulur.



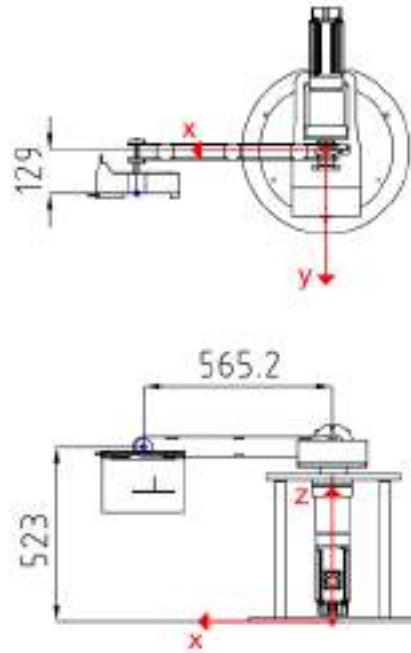
$$P_x = -\sin(\theta_1)L_3 + \cos(\theta_1)\cos(\theta_2)L_2$$

$$P_y = \cos(\theta_1)L_3 + \sin(\theta_1)\cos(\theta_2)L_2$$

$$P_z = -\sin(\theta_2)L_2 + L_1 \quad (4.17)$$

### İleri kinematik denklemlerinin doğruluğunun test edilmesi

Robot kolundaki uzunluk parametreleri sırasıyla  $L_1=523$  mm,  $L_2=565,2$  mm,  $L_3=129$  mm ve  $L_4=181,5$  mm şeklindedir. Bütün açılara  $0^\circ$  vererek robotun sıfır konumu elde edilir. Şekil 4.24'de katı modelleme programında birinci ve ikinci eksene  $0^\circ$  verildiğinde, uç tutucu elemana ait ölçüler gösterilmektedir. Bu ölçüler robot kolunun taban noktası referans alınarak hesaplanmıştır. Robot kolunun uç işlevcisine ait  $P_x$ ,  $P_y$  ve  $P_z$  konumları Şekil 4.24'te gösterilen teknik resim ve Denklem 4.18'de hesaplanan değerler ile örtüştüğünden ileri yön kinematiği doğrudur.



Şekil 4.24. Robot kolu başlangıç konumu ( $\theta_1 = \theta_2 = 0$ )

$$\theta_1 = \theta_2 = 0$$

$$P_x = -\sin(0) L_3 + \cos(0) \cos(0) L_2 = L_2$$

$$P_y = \cos(0) L_3 + \sin(0) \cos(0) L_2 = L_3$$

$$P_z = -\sin(0) L_2 + L_1 = L_1$$

$$(P_x, P_y, P_z) = (565.2, 129, 523) \quad (4.18)$$

Robot kolunun pozitif ve negatif dönme yönünü tespit etmek için örnek olarak birinci eksene -30 derece, ikinci eksene -60 derece verilirse Denklem 4.19'daki konum değerleri elde edilir.

$$\theta_1 = 30^\circ, \theta_2 = 90^\circ$$

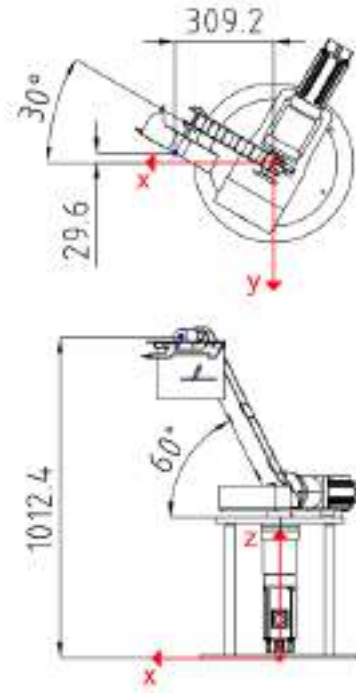
$$P_x = -\sin(-\pi/6) L_3 + \cos(-\pi/6) \cos(-\pi/3) L_2 = 309.2$$

$$P_y = \cos(-\pi/6) L_3 + \sin(-\pi/6) \cos(-\pi/3) L_2 = -29.6$$

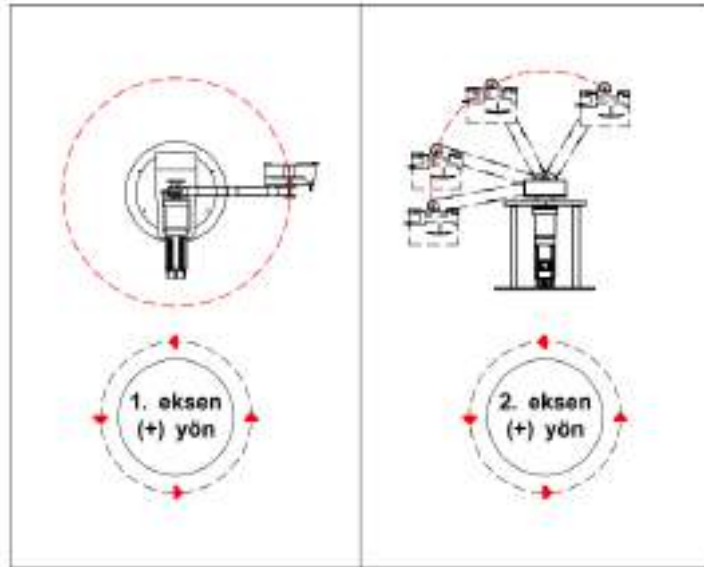
$$P_z = -\sin\left(-\frac{\pi}{3}\right) L_2 + L_1 = 1012.4$$

$$(P_x, P_y, P_z) = (309.2, -29.6, 1012.4) \quad (4.19)$$

İleri kinematik denklemlerinden çıkan değerler sonucunda robot kolunun pozisyonu görsel olarak Şekil 4.25'te gösterilmiştir. Robot kolunun ileri kinematikte taban noktasına atılan eksen takımı referans alınmıştır. Dönme yönleri Şekil 4.26'da gösterilmiştir. Birinci ve ikinci eksen için saat yönünün tersi pozitif olduğu görülmektedir.



Şekil 4.25. Birinci eksene  $-30^\circ$ , ikinci eksene  $-60^\circ$  verildiğindeki uç işlevcideki konum



Şekil 4.26. Eksenler pozitif dönme yönleri

#### 4.2.2. Ters kinematik

Bu bölümde, ters kinematik hesaplamasında kullanılan analitik ve geometrik çözüm olarak iki yöntem üzerinde durulmuştur.

##### Ters kinematik analitik çözüm

Ters kinematikte uç işlevciye ait konum ve yönelim bilgisine göre eklemlere verilmesi gereken açı değerleri hesaplanır. Ters kinematikten kullanılan hesaplamalar

doğrusal olmayıp karmaşık yapıda olabilirler. Bu nedenle birden fazla çözüm kümesi olabilir.

Altı serbestlik dereceli robot koluna ait ileri kinematik Denklem 4.20'de gösterilmiştir. Denklem 4.21'de  ${}^0T_6$  matrisinin konum ve yönelim bilgilerini içeren matris gösterilmektedir.

$${}^0T_6 = {}^0T_1 {}^1T_2 {}^2T_3 {}^3T_4 {}^4T_5 {}^5T_6 \quad (4.20)$$

$${}^0T_6 = \begin{bmatrix} r_{11} & r_{12} & r_{13} & p_x \\ r_{21} & r_{22} & r_{23} & p_y \\ r_{31} & r_{32} & r_{33} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.21)$$

Denklem 4.20'deki denklemin her iki tarafı  $[{}^0T_1]^{-1}$  çarpılmış hali denklem 4.22'de gösterilmiştir. Bir matrisin tersiyle çarpımı birim matrise eşit olacağından  $[{}^0T_1]^{-1} {}^0T_1 = I$  'ya eşit olur. Denklemin tekrar düzenlenmiş hali Denklem 4.23'te gösterilmiştir.

$$[{}^0T_1]^{-1} {}^0T_6 = [{}^0T_1]^{-1} {}^0T_1 {}^1T_2 {}^2T_3 {}^3T_4 {}^4T_5 {}^5T_6 \quad (4.22)$$

$$[{}^0T_1]^{-1} {}^0T_6 = {}^1T_2 {}^2T_3 {}^3T_4 {}^4T_5 {}^5T_6 \quad (4.23)$$

Ters kinematikte dönüşüm matrislerinin sağ ve sol taraftan sırasıyla çarpılıp eşitlenmesiyle çözüm yoluna gidilir. Bu işlemlerin tekrarlanmış hali Denklem 4.24'te gösterilmiştir.

$$\begin{aligned} [{}^0T_1 {}^1T_2]^{-1} {}^0T_6 &= {}^2T_3 {}^3T_4 {}^4T_5 {}^5T_6 \\ [{}^0T_1 {}^1T_2 {}^2T_3]^{-1} {}^0T_6 &= {}^3T_4 {}^4T_5 {}^5T_6 \\ [{}^0T_1 {}^1T_2 {}^2T_3 {}^3T_4]^{-1} {}^0T_6 &= {}^4T_5 {}^5T_6 \\ [{}^0T_1 {}^1T_2 {}^2T_3 {}^3T_4 {}^4T_5]^{-1} {}^0T_6 &= {}^5T_6 \end{aligned} \quad (4.24)$$

Ters kinematikte kullanılan başlıca trigonometrik formüller şunlardır:

$$\cos \theta = a \text{ ise } \theta = \text{Atan2}(\sqrt{1-a^2}, a) \quad (4.25)$$

$$\sin \theta = a \text{ ise } \theta = \text{arctan2}(a, \pm\sqrt{1-a^2}) \quad (4.26)$$

$$\cos \theta = a \text{ ve } \sin \theta = b \text{ ise } \theta = \text{arctan2}(b, a) \quad (4.27)$$

$$\begin{aligned} a \sin \theta + b \cos \theta = 0 \text{ ise} \\ \theta = \text{arctan2}(-b, a) \text{ veya } \text{arctan2}(b, -a) \end{aligned} \quad (4.28)$$

$$\begin{aligned} a \sin \theta + b \cos \theta = c \text{ ise} \\ \theta = \text{arctan2}(a, b) + \text{arctan}(\pm\sqrt{a^2 + b^2 - c^2}, c) \end{aligned} \quad (4.29)$$

Bir dönüşüm matrisinin tersini almak için öncelikle dönme matrisinin transpozesi alınır. Daha sonra konum vektörü ile transpozesi alınan matris çarpılır ve önüne eksi işareti atılır. Hesaplanan bu değerler Denklem 4.30'da yerlerine yazılarak matrisin tersi bulunur.

$$[{}^{N-1}T_N]^{-1} = \begin{bmatrix} ({}^{N-1}R_N)^T & -({}^{N-1}R_N)^T P_N \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.30)$$

Tasarımı yapılan robot kolunun ters kinematik hesaplamasında sırasıyla aşağıdaki adımlar takip edilmiştir:

Birinci ekleme ait dönme matrisi oluşturuldu.

$${}^0R_1 = \begin{bmatrix} \cos(\theta_1) & -\sin(\theta_1) & 0 \\ \sin(\theta_1) & \cos(\theta_1) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (4.31)$$

Dönme matrisinin transpozesi alındı.

$${}^0_1R^T = \begin{bmatrix} \cos(\theta_1) & \sin(\theta_1) & 0 \\ -\sin(\theta_1) & \cos(\theta_1) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (4.32)$$

Birinci ekleme ait konum vektörü oluşturuldu.

$${}^0_1P = \begin{bmatrix} 0 \\ 0 \\ L_1 \end{bmatrix} \quad (4.33)$$

Birinci ekleme ait dönme matrisinin transpozesi ile konum vektörü ile çarpıldı.

$${}^0_1R^T \cdot {}^0_1P = \begin{bmatrix} 0 \\ 0 \\ -L_1 \end{bmatrix} \quad (4.34)$$

Elde edilen veriler Denklem 4.30'da yerine yazılarak birinci ekleme ait dönüşüm matrisinin tersi Denklem 4.35'de gösterildiği gibi oluşturulur.

$${}^0_1T^{-1} = \begin{bmatrix} \cos\theta_1 & \sin\theta_1 & 0 & 0 \\ -\sin\theta_1 & \cos\theta_1 & 0 & 0 \\ 0 & 0 & 1 & -L_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.35)$$

Denklem 4.35'de oluşturulan matris, robot kolunun ileri yön kinematik denklemindeki eşitliğin her iki tarafıyla çarpılmasıyla Denklem 4.36 elde edilir.

$$[{}^0_1T]^{-1} {}^0_4T = [{}^0_1T]^{-1} {}^0_1T {}^1_2T {}^2_3T {}^3_4T \quad (4.36)$$

Denklem 4.36'daki eşitliğin sol tarafındaki hesaplama Denklem 4.37'de gösterilmiştir. Birinci eklem tersi ile ileri yön kinematiğini temsil eden matris çarpılmıştır.

$$\begin{aligned}
({}^0_1T)^{-1} \cdot {}^0_4T &= \begin{bmatrix} \cos\theta_1 & \sin\theta_1 & 0 & 0 \\ -\sin\theta_1 & \cos\theta_1 & 0 & 0 \\ 0 & 0 & 1 & -L_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & p_x \\ r_{21} & r_{22} & r_{23} & p_y \\ r_{31} & r_{32} & r_{33} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
&= \begin{bmatrix} \cos\theta_1 r_{11} + \sin\theta_1 r_{21} & \cos\theta_1 r_{12} + \sin\theta_1 r_{22} & \cdot & \cdot \\ -\sin\theta_1 r_{11} + \cos\theta_1 r_{21} & -\sin\theta_1 r_{12} + \cos\theta_1 r_{22} & \cdot & \cdot \\ r_{31} & r_{32} & \cdot & \cdot \\ 0 & 0 & \cdot & \cdot \\ \cdot & \cdot & \cos\theta_1 r_{13} + \sin\theta_1 r_{23} & \cos\theta_1 p_x + \sin\theta_1 p_y \\ \cdot & \cdot & -\sin\theta_1 r_{13} + \cos\theta_1 r_{23} & -\sin\theta_1 p_x + \cos\theta_1 p_y \\ \cdot & \cdot & r_{33} & p_z - L_1 \\ \cdot & \cdot & 0 & 1 \end{bmatrix} \quad (4.37)
\end{aligned}$$

Denklem 4.36'daki eşitliğin sağ tarındaki hesaplama Denklem 4.38'de gösterilmiştir.

$${}^{\frac{1}{2}}_1T \cdot {}^{\frac{2}{3}}_3T \cdot {}^{\frac{3}{4}}_4T = \begin{bmatrix} \cos\theta_2^2 + \sin\theta_2^2 & 0 & 0 & \cos\theta_2 L_2 \\ 0 & 0 & 1 & L_3 \\ 0 & -\sin\theta_2^2 - \cos\theta_2^2 & 0 & -\sin\theta_2 L_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.38)$$

Denklem 4.37 ve Denklem 4.38'deki matrislerin üçüncü sütunundaki elemanlarının birbirine eşitlenmesi Denklem 4.39-4.40-4.41'de gösterilmiştir. Bu eşitlemeler kullanılarak verilen  $P_x$ ,  $P_y$  ve  $P_z$  konum değerlerinden açı değerlerine ulaşılır. Denklemlerin yeterli olmadığı durumda Denklem 4.24'teki hesaplamalara devam edilir.

$$\cos\theta_1 p_x + \sin\theta_1 p_y = \cos\theta_2 L_2 \quad (4.39)$$

$$-\sin\theta_1 p_x + \cos\theta_1 p_y = L_3 \quad (4.40)$$

$$p_z - L_1 = -\sin\theta_2 L_2 \quad (4.41)$$

Denklem 4.39'deki eşitliğin her iki tarafının da karesinin alınma işlemi Denklem 4.42'de gösterilmiştir.

$$(\cos\theta_1 p_x + \sin\theta_1 p_y)^2 + (-\sin\theta_1 p_x + \cos\theta_1 p_y)^2 = (\cos\theta_2 L_2)^2 + (L_3)^2 \quad (4.42)$$

Denklem 4.34'teki ifade sadeleştirilip  $L_2$  ve  $L_3$  ifadesi yerine yazılırsa Denklem 4.43 elde edilir.

$$(p_x)^2 + (p_y)^2 = (3.1945104 * 10^5) \cos\theta_2^2 + 16641 \quad (4.43)$$

Denklem 4.43'deki eşitlik Denklem 4.25'teki trigonometrik ifade uygulanarak çözülür. Denklem 4.44'de gösterildiği gibi  $\cos \theta$  ifadesi yalnız bırakılarak çözüm yoluna gidilir. Böylece konum değerlerinden  $P_x$  ve  $P_y$  değişkenlerinin bilinmesi durumunda ikinci eksene verilmesi gereken dönme miktarı ( $\theta_2$ ) bulundu.

$$a = \cos \theta_2 = \sqrt{\frac{(p_x)^2 + (p_y)^2 - 16641}{3.1945104 * 10^5}}$$

$$\theta_2 = \text{Atan2}(\sqrt{1 - a^2}, a) \quad (4.44)$$

Birinci eksene ait açısal dönme miktarı Denklem 4.40'teki eşitlikten elde edilebilir. Bu eşitliğin çözümünde Denklem 4.25'teki trigonometrik ifade uygulanır. Denklem 4.45'te çözüm basamakları gösterilmiştir.

$$-\sin\theta_1 p_x + \cos\theta_1 p_y = L_3 = 129 = c$$

$$a = P_x, b = P_y$$

$$\theta_1 = \text{arctan2}(a, b) + \text{arctan}(\pm\sqrt{a^2 + b^2 - c^2}, c) \quad (4.45)$$



## Ters kinematik denklemlerinin doğruluğunun test edilmesi

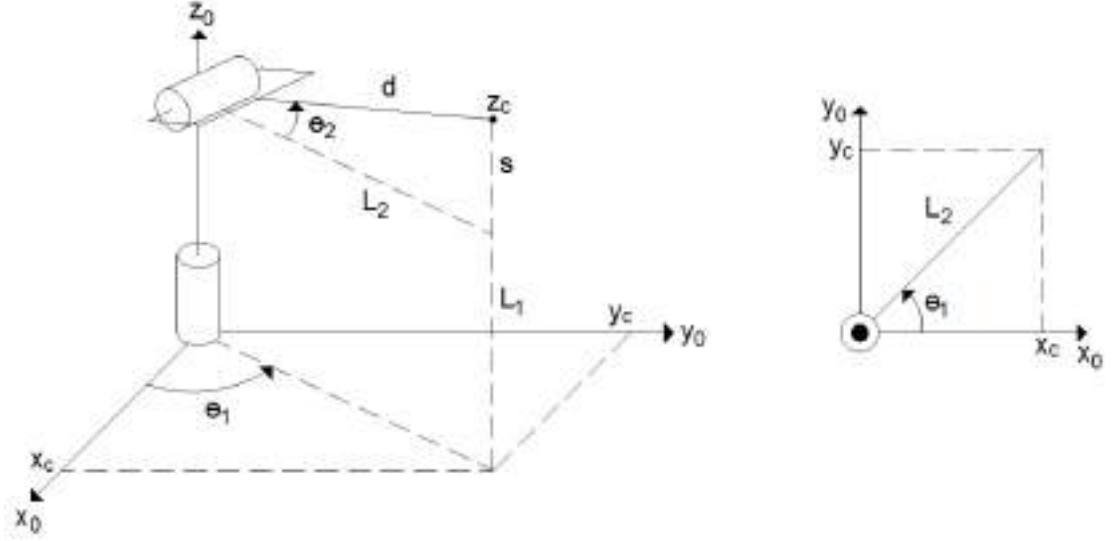
Robot koluna ait ters kinematik denklemler (Denklem 4.44 ve Denklem 4.45) elde edildikten sonra Tablo 4.10'da gösterildiği gibi doğruluğunu test etmek için üç adet  $P_x$ ,  $P_y$  ve  $P_z$  konumları verilip açı değerleri bulunmuştur. Açı değerlerinde ikişer adet ihtimal çıkmış olup doğru olan ihtimal kalın yazı formatında gösterilmiştir.

Tablo 4.10. D-H parametreleri tablosu

Ters Kinematik							
	Giriş			Çıkış			
	$P_x$	$P_y$	$P_z$	$\theta_1$	$\theta_1$	$\theta_2$	$\theta_2$
Deneme 1	309,24	-29,58	1012,48	<b>-29,99</b>	-160,00	60,00	<b>-60,00</b>
Deneme 2	88,11	410,61	922,66	-84,22	<b>59,99</b>	45,00	<b>-45,00</b>
Deneme 3	254,90	437,33	805,60	-105,47	<b>44,99</b>	29,99	<b>-29,99</b>

## Ters kinematik geometrik çözüm

Tasarlanan iki serbestlik dereceli robotun sembolik gösterimi Şekil 4.27'de gösterilmiştir. Denklem 4.46'da gösterildiği gibi robot ilk konumunda yani kolun zeminle paralel olduğu ve z ekseninde  $L_1$  boyunda olduğu konumdan  $z_c$  konumuna gittiği zaman robot kolu z ekseninde s kadar yer değiştirir. Robot kolunun ilk başlama konumundaki uzunluğu Denklem 4.47'de gösterilmiştir. Gösterime yukardan bakıldığı zaman ilk konumdan  $z_c$  konumuna gittiği zaman robot kolunun uzunluğunun iz düşüm olarak ifade edilişi Denklem 4.48'de gösterilmiştir. Birinci ve ikinci eklemin açılma açısı Denklem 4.49 ve 4.50'de ifade edilmiştir (Spong ve diğerleri, 2004).



Şekil 4.27. Tasarlanan robot kolu ters kinematik geometrik gösterim (Spong ve diğerleri, 2004)

$$s = z_c - L_1 \quad (4.46)$$

$$L_2^2 = x^2 + y^2 \quad (4.47)$$

$$d = \sqrt{L_2^2 + s^2} = \sqrt{x^2 + y^2 + (z_c - L_1)^2} \quad (4.48)$$

$$\begin{aligned} \theta_1 &= \text{atan2}(x_c, y_c) \\ \theta_1 \text{ diğer ç.k.} &= \pi + \text{atan2}(x_c, y_c) \end{aligned} \quad (4.49)$$

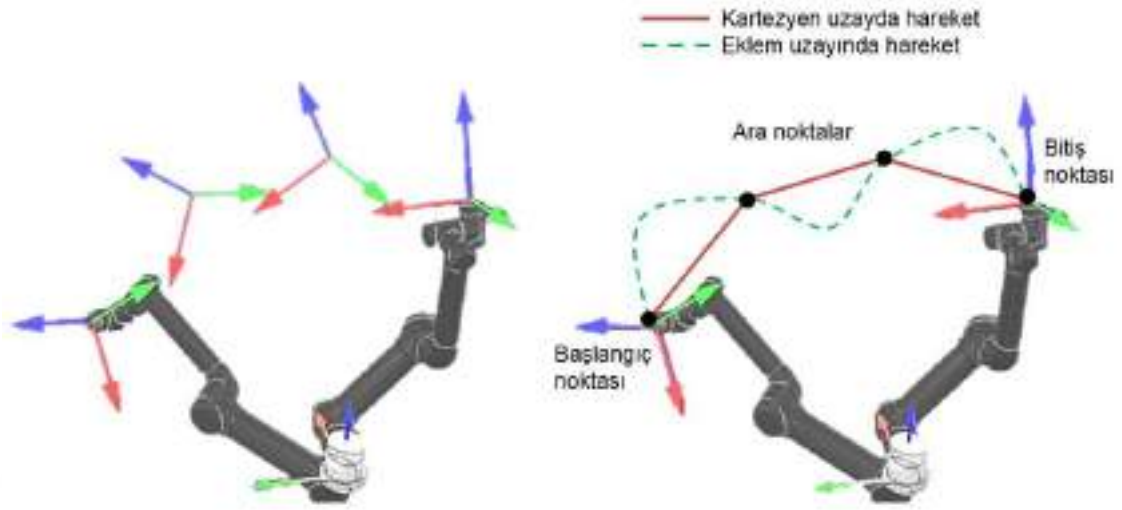
$$\theta_2 = \text{atan2}(r, s) + \frac{\pi}{2} \quad (4.50)$$

### 4.2.3. Yörünge planlaması

Robot kolunun ana çerçeveden araç çerçeveye olan hareketi boyunca takip ettiği noktalar kümesi yörünge olarak tanımlanır (Tonbul ve Sarıtaş, 2013). Yörüngede kullanılan her bir nokta ters kinematik hesaplamalarla açısal değere dönüştürülür.

Yörünge planlaması robot kolunun hareketi esnasında ani titreşim ve rezonansa karşı koruma, çevredeki cisimlere çarpmasını engelleme ve hareketini yumuşatmak için yapılır. Şekil 4.28’de gösterildiği gibi Kartezyen ve eklem uzayında hareket olmak üzere

iki çeşit yörünge planlaması vardır. Yörünge başlangıç, bitiş ve ara noktalardan oluşmaktadır(Bingül ve Küçük, 2017a).



Şekil 4.28. Kartezyen ve eklem uzayında hareket

Kartezyen uzayda robot kolu verilen başlangıç ve bitiş noktasında sabit hızlı olarak doğrusal olarak hareket eder. Endüstride basit çözüm olmasından dolayı genellikle Kartezyen uzayda hareket kullanılır.

Eklem uzayında hareket ederken hız ve ivme zamana bağlı değişir. Yörünge de bu değişime bağlı olarak parabolik şekil alır. Şekil 4.28’de gösterildiği gibi robot kolunun hareketi esnasında başlangıç, bitiş ve ara noktalar belirlenir. Belirtilen noktalardan pürüzsüz bir şekilde geçecek şekilde yörünge oluşturulur. Geçiş noktalarının ve zamanlarının seçimi yörüngeyi şekillendirir.

Üç veya daha yüksek dereceli polinomlar ile yörünge denklemi oluşturulur. Robot kolu  $t_0$  anında  $\theta(0) = \theta_0$  konumuna sahipken hedef konuma ulaştığında  $\theta(t_f) = \theta_f$  konumuna sahiptir. Buna göre başlangıç ve bitiş hızı  $\dot{\theta}(0) = 0$  ve  $\dot{\theta}(t_f) = 0$  olur(Bingül ve Küçük, 2017a).

$$\theta(0) = \theta_0 \quad (4.51)$$

$$\theta(t_f) = \theta_f \quad (4.52)$$

$$\dot{\theta}(0) = 0 \quad (4.53)$$

$$\dot{\theta}(t_f) = 0 \quad (4.54)$$

Denklem 4.51, 4.52, 4.53 ve 4.54'teki ifadeler kullanılarak polinomdaki diğer katsayılar bulunur. Üçüncü dereceden polinom kullanılarak yörünge oluşturulması Denklem 4.55'te verilmiştir.

$$\theta(t) = s_0 + s_1 t + s_2 t^2 + s_3 t^3 \quad (4.55)$$

Denklem 4.55'teki ifadeden hız ve ivmeye geçiş yapmak için birinci ve ikinci dereceden türevlerinin alınması Denklem 4.46 ve 4.57'de gösterilmiştir.

$$\dot{\theta}(t) = s_1 + 2s_2 t + 3s_3 t^2 \quad (4.56)$$

$$\ddot{\theta}(t) = 2s_2 + 6s_3 t \quad (4.57)$$

Denklem 4.51 ve 4.52, Denklem 4.55'te sırasıyla yerine yazılırsa Denklem 4.58 ve 4.59'daki ifadeler bulunur.

$$\begin{aligned} \theta(0) &= s_0 + s_1 0 + s_2 0^2 + s_3 0^3 \\ &\rightarrow s_0 = \theta(0) \text{ veya } s_0 = \theta_0 \end{aligned} \quad (4.58)$$

$$\begin{aligned} \theta(t_f) &= \theta_f \\ \theta(t_f) &= s_0 + s_1 t_f + s_2 t_f^2 + s_3 t_f^3 \end{aligned} \quad (4.59)$$

Denklem 4.53 ve 4.54, Denklem 4.56'da sırasıyla yerine yazılırsa Denklem 4.60 ve 4.61'deki ifadeler bulunur.

$$\begin{aligned} \dot{\theta}(0) &= s_1 + 2s_2 0 + 3s_3 0^2 = 0 \\ &\rightarrow s_1 = 0 \end{aligned} \quad (4.60)$$

$$\dot{\theta}(t_f) = s_1 + 2s_2 t_f + 3s_3 t_f^2 \quad (4.61)$$

Daha önceki denklemlerde  $s_0$  ve  $s_1$  bulunmuştu. Diğer iki katsayıdan  $s_2$  ve  $s_3$  parametreleri,  $\dot{\theta}(t_f) = 0$  ve  $s_1 = 0$  ifadelerinin Denklem 4.61’de yazılmasıyla Denklem 4.62’de gösterildiği gibi bulunur.

$$\begin{aligned}\dot{\theta}(t_f) &= s_1 + 2s_2t_f + 3s_3t_f^2 = 0 \rightarrow 0 = 0 + 2s_2t_f + 3s_3t_f^2 \\ \rightarrow s_3 &= -\frac{2s_2}{3t_f}, s_2 = -\frac{3s_3t_1}{2}\end{aligned}\quad (4.62)$$

Denklem 4.62’den bulunan  $s_3$  ifadesi Denklem 4.59’da yerine yazılmasıyla  $s_2$  ifadesi açı değerleri cinsinden Denklem 4.63’te gösterildiği gibi ifade edilir.

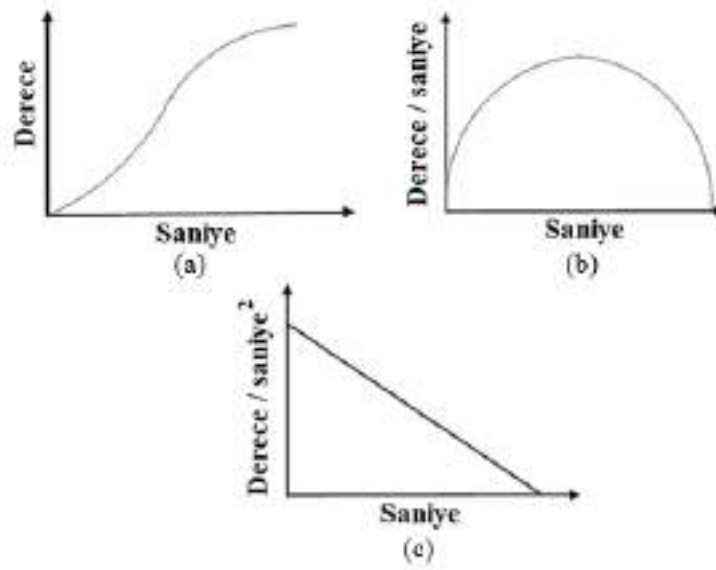
$$\begin{aligned}\theta(t_f) &= s_0 + s_1t_f + s_2t_f^2 + s_3t_f^3 \rightarrow \theta(t_f) = \theta_0 + 0t_f + s_2t_f^2 - \frac{2s_2}{3t_f}t_f^3 \\ \rightarrow s_2 &= \frac{3}{t_f^2}(\theta_f - \theta_0)\end{aligned}\quad (4.63)$$

Denklem 4.62’den bulunan  $s_2$  ifadesi Denklem 4.59’da yerine yazılmasıyla  $s_3$  ifadesi açı değerleri cinsinden Denklem 4.63’te gösterildiği gibi ifade edilir.

$$\begin{aligned}\theta_f &= s_0 + s_1t_f + s_2t_f^2 + s_3t_f^3 \rightarrow \theta_f = \theta_0 + 0t_f - \frac{3s_3t_1}{2}t_f^2 + s_3t_f^3 \\ \rightarrow s_3 &= -\frac{2}{t_f^3}(\theta_f - \theta_0)\end{aligned}\quad (4.64)$$

Denklemlerden elde edilen katsayılar Denklem 4.65’te gösterilmiştir. Bu katsayılar üçüncü dereceden polinomda yerine yazılmasıyla oluşan konum, hız ve ivme grafikleri Şekil 4.29’da gösterilmiştir.

$$s_0 = \theta(0), s_1 = 0, s_2 = \frac{3}{t_f^2}(\theta_f - \theta_0), s_3 = -\frac{2}{t_f^3}(\theta_f - \theta_0)\quad (4.65)$$



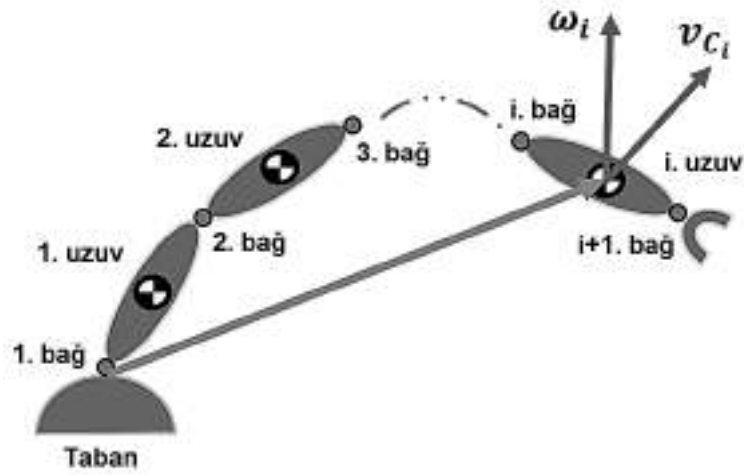
Şekil 4.29. Üçüncü dereceden kübik yörüngeye ait a) konum b) hız ve c) ivme grafiği (Bingül ve Küçük, 2017a)

### 4.3. Dinamik Modelleme

Bir robot kolunun dinamik denklemleri, robot kolunun hareketi boyunca eklemlere etki eden kuvvetler veya torklar arasındaki ilişkiyi ifade eder. Başka bir ifadeyle dinamik denklemler, kol eklemlerinin kinematik değişkenlerini (pozisyon, hız ve ivme), istenen kol hareketini elde etmek ve kontrol etmek için gerekli olan eklemlerdeki kuvvetler veya torklarla ilişkilendirir. Robot kollarının dinamiği, eklemler arasındaki dinamik etkileşim nedeniyle karmaşıktır. Rijit bir kol için bile, dinamik denklemler oluşturulurken atalet momenti, Coriolis ve merkezkaç kuvvetleri ve yerçekimi etkisi gibi birçok terim kullanılır(Bejczy ve Paul, 1981).

Karmaşık dinamik sistemler Lagrange Euler metodu kullanılarak basit bir şekilde ifade edilebilirler. Denklem 4.66'da gösterildiği gibi kinetik(K) ve potansiyel(P) enerji farkından hesaplanır. Denklemde yer alan  $q$  ifadesi döner eklemlerde açıyı ifade ederken  $\dot{q}$  eklem hızlarını gösteren vektördür(Schilling, 2003).

$$L(q, \dot{q}) = K(q, \dot{q}) - P(q) \quad (4.66)$$



Şekil 4.30. Robot kolu i. bağı kinetik enerjisi

Robot kolunda  $n$  tane bağı olduğu ve  $1 \leq i \leq n$  olduğu kabul edilirse  $i$ . bağı kütlesinde oluşan kinetik enerji doğrusal ( $v_i$ ) ve açısal hızdan ( $\omega_i$ ) oluşur. Şekil 4.30'da  $i$ . bağı kütlesindeki hız bileşenleri gösterilmiştir. Robot kolunda oluşan toplam kinetik enerji her bir bağıda oluşan kinetik enerjilerin toplanmasıyla Denklem 4.67'de gösterildiği gibi elde edilir. Denklemde  $m$   $i$ . bağı kütlesini, ( $I$ )  $i$ . bağı kütlesinin ana koordinat sistemine göre  $3 \times 3$  boyutunda atalet tensörü matrisini ifade eder (Bingül ve Küçük, 2017b).

$$K(q, \dot{q}) = \frac{1}{2} \sum_{i=1}^n (v_i)^T m_i + (\omega_i)^T I_i \omega_i \quad (4.67)$$

Atalet tensörü, katı bir cismin kütlesinin dağılımını karakterize eden  $3 \times 3$ 'lük bir matristir. Atalet tensörünü ana koordinat sistemine göre ifade etmek için öncelikle cismin kendi kütlesine göre ifade edilmesi gerekir. Bir katı cismin kütlesine ait atalet momentini ifade eden matris Denklem 4.68'de gösterilmiştir (Schilling, 2003).

$$I_m = \begin{bmatrix} I_{xx} & -I_{xy} & -I_{xz} \\ -I_{xy} & I_{yy} & -I_{yz} \\ -I_{xz} & -I_{yz} & I_{zz} \end{bmatrix} \quad (4.68)$$

Bir robot kolundaki  $i$ . bağı ana koordinat sistemine göre atalet tensörü, o bağı dönüşüm matrisindeki konum ve yönelim ifadeleri kullanılarak Denklem 4.69'deki gösterildiği gibi bulunur(Bingül ve Küçük, 2017b).

$$I_i = {}^0R I_m {}^0R^T \quad (4.69)$$

Robot kolunda  $i$ . bağı kütle merkezinin ana koordinat sistemine göre konumu Denklem 4.70'de verilmiştir(Bingül ve Küçük, 2017b).

$$h_i = {}^0T \Delta h_i \quad (4.70)$$

Robot kolunun kinetik enerjisinin bulunmasında kullanılan diğer adım Jakobiyen matrisinin bulunmasıdır. Denklem 4.71'de gösterilen Jakobiyen matrisi,  $A_i$  ve  $B_i$  bölümlenmiş alt matrislerinden oluşmaktadır. Matriste yer alan  $\xi_i$  eklem tip değişkeni temsil etmektedir. Bu değişkende döner eklemler için 1, prizmatik eklemler için 0 alınır. Jakobiyen matrisinde yer alan diğer değişken ( $z^i$ )  $i$ . koordinat sisteminin üçüncü kolon birim vektörünü temsil eder ve Denklem 4.72'de gösterilmiştir. Denklemde  $i^3 = [0 \ 0 \ 1]^T$  olarak alınır(Bingül ve Küçük, 2017b).

$$J_i = \begin{bmatrix} \frac{\partial h_1}{\partial q_1} \dots \frac{\partial h_i}{\partial q_i} & 0 \\ \xi_1 z^1 \dots \xi_i z^i & 0 \end{bmatrix} = \begin{bmatrix} A_i \\ B_i \end{bmatrix} \quad (4.71)$$

$$z^i = {}^0R i^3 \quad (4.72)$$

Denklem 4.67'de gösterilen toplam kinetik enerjisinin düzenlemiş hali, Denklem 4.73 ve 4.74'te yer alan doğrusal ve açısal hız ifadeleri kullanılarak Denklem 4.75'te gösterilmiştir.



$$v_i = A_i \dot{q} \quad (4.73)$$

$$\omega_i = B_i \dot{q} \quad (4.74)$$

$$K(q, \dot{q}) = \frac{1}{2} \dot{q}^T \sum_{i=1}^n [(A_i)^T m_i A_i + (B_i)^T I_i B_i] \dot{q} \quad (4.75)$$

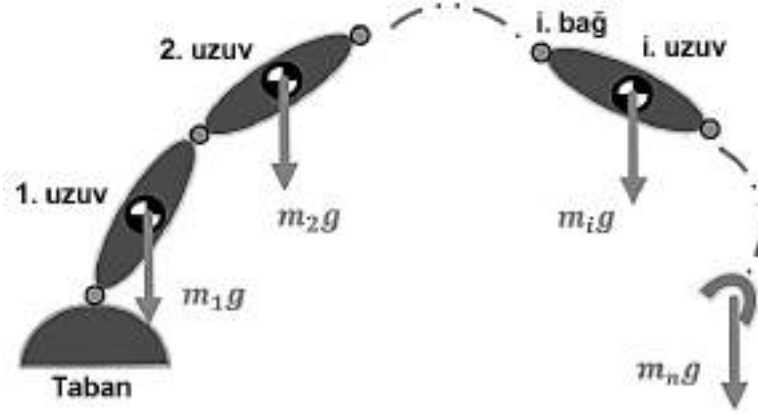
Denklem 4.75, Denklem 4.76'daki genel atalet tensörü  $D(q)$  ifadesi kullanılarak tekrar düzenlenirse Denklem 4.77'deki ifade elde edilir.

$$D(q) = \sum_{i=1}^n [(A_i)^T m_i A_i + (B_i)^T I_i B_i] \quad (4.76)$$

$$K(q, \dot{q}) = \frac{1}{2} \dot{q}^T D(q) \dot{q} \quad (4.77)$$

Robot kolunun toplam potansiyel enerjisi Denklem 4.78'de gösterilmiştir. Denklemden  $g$  yerçekimi ivmesini,  $h_i$  i. bağıın kütle merkezinin ana koordinat sistemine göre konumunu belirtir. Şekil 4.31'de robot kolunun her bir uzvun kütle merkezine etki eden potansiyel enerji ve yerçekimi ivmesi gösterilmiştir (Bingül ve Küçük, 2017b).

$$P(q) = \sum_{i=1}^n m_i g^T h_i \quad (4.78)$$



**Şekil 4.31.** Robot kolu uzuvların potansiyel enerjisi ve yerçekimi ivmesi

Lagrange fonksiyonunun toplam kinetik enerji ve potansiyel denklemleri kullanılarak düzenlenmiş hali Denklem 4.79’da gösterilmiştir(Bingül ve Küçük, 2017b).

$$L(q, \dot{q}) = \frac{1}{2} \dot{q}^T D(q) \dot{q} + mg^T h \quad (4.79)$$

### Lagrange Euler yöntemiyle robot kolu dinamik modelin oluşturulması

Robot kolunun hareketinden dolayı oluşan ifade Denklem 4.80’de gösterilmiştir. Bu ifadenin açılmış hali Denklem 4.81’de gösterilmiştir(Bingül ve Küçük, 2017b).

$$\frac{d}{dt} \left( \frac{\partial L}{\partial \dot{q}} \right) - \frac{\partial L}{\partial q} = \tau \quad (4.80)$$

$$\frac{d}{dt} \left( \frac{\partial K}{\partial \dot{q}} \right) - \frac{\partial K}{\partial q} + \frac{\partial P}{\partial q} = \tau \quad (4.81)$$

Kinetik ve potansiyel enerjileri ifade eden denklemden yerine konulursa ve sürtünme kayıpları ihmal edildiğinde Denklem 4.82’deki genel ifade elde edilir(Bingül ve Küçük, 2017b).

$$\sum_{j=1}^n D_{ij}(q) \ddot{q}_j + \sum_{k=1}^n \sum_{j=1}^n c_{kj}^i(q) \dot{q}_k + y_i(q) + b_i(q) = \tau_i \quad (4.82)$$

Denklem 4.82'deki ilk terim robotun hareketi esnasındaki eklemlere uygulanan tork ve içsel kuvvetlerden kaynaklanan ivmeyi temsil eder. Denklem 4.83'te gösterilen ikinci terim robot kolunun Coriolis ve Merkezkaç kuvvet vektörünü temsil eder(Bingül ve Küçük, 2017b).

Coriolis Etkisi, dönen bir platform üzerinde hareket eden bir nesnede oluşan sapmadır. Saat yönünde dönüşlü bir platformda, sapma, nesnenin hareketinin solundadır; saat yönünün tersine dönüşlü bir diğerinde sapma sağındadır. Robot kolundaki Coriolis etkisi, bir uzuv diğer dönen uzvun üzerinde dönerken hedefte sapma veya hata oluşturmasıdır(Thomas, 2016).

Merkezkaç Kuvveti, bir nesnenin kavisli bir yolda hareket ederken nesneyi eğrinin merkezinden dışına doğru çeken bir kuvvettir.

$$c_{kj}^i(q) = \frac{\partial}{\partial q_k} D_{ij}(q) - \frac{1}{2} \frac{\partial}{\partial q_k} D_{kj}(q) \quad 1 \leq i, j, k \leq n \quad (4.83)$$

Denklem 4.82'deki üçüncü terim yerçekimi ivmesini sembolize eder ve açılmış hali Denklem 4.84'te gösterilmiştir(Bingül ve Küçük, 2017b).

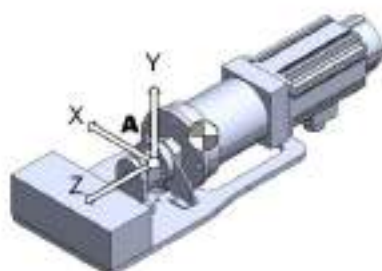
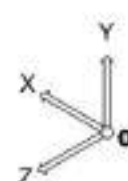
$$y_i(q) = - \sum_{k=1}^3 \sum_{j=1}^n g_k m_j A_{ki}^j(q) \quad (4.84)$$

Sonuç olarak sürtünme kuvvetleri ihmal edildiğinde robot kolunun dinamik denklemini Denklem 4.85'te ifade edilmiştir(Bingül ve Küçük, 2017b).

$$D(q)\ddot{q} + C(q, \dot{q}) + G(q) = \tau_i \quad (4.85)$$

### **Tasarlanan robot kolunun benzetim modeli**

Tasarımı yapılan robot kolunun her bir uzvuna ait kütle, kütle merkezi ve atalet momenti bilgileri Solidworks programında hesapla sekmesindeki kütle özellikler kısmından ulaşılmıştır. Elde edilen veriler Şekil 4.32-4.33 ve 4.34'te gösterilmiştir.

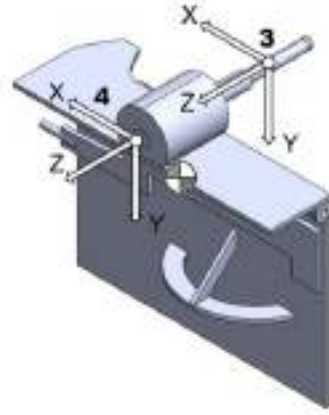
<b>Kütle (kg)</b>	 
44,59	
<b>Koordinat sistemi 1'e göre kütle merkezi (mm)</b>	
X = -0.10994 Y = -101.92463 Z = -12.72789	
<b>Koordinat sistemi A'ya göre atalet momenti (g * mm)</b>	
$I_{xx} = 2225208720.39875$ $I_{xy} = 13812.74074$ $I_{xz} = -323017.25576$ $I_{yx} = 13812.74074$ $I_{yy} = 319164026.05079$ $I_{yz} = -358428123.44543$ $I_{zx} = -323017.25576$ $I_{zy} = -358428123.44543$ $I_{zz} = 2050572752.42063$	

Şekil 4.32. Robot kolunda birinci eksenin kütlesi, kütle merkezi ve atalet momenti

<b>Kütle (kg)</b>	
3,16	
<b>Koordinat sistemi 3'e göre kütle merkezi (mm)</b>	
X = -231.86858 Y = 0 Z = 0.24344	
<b>Koordinat sistemi 2'ye göre atalet momenti (g * mm)</b>	
$I_{xx} = 2492993.29895$ $I_{xy} = 0$ $I_{xz} = 434860.44654$ $I_{yx} = 0$ $I_{yy} = 464842085.90099$ $I_{yz} = 0$ $I_{zx} = 434860.44654$ $I_{zy} = 0$ $I_{zz} = 464651324.85058$	

Şekil 4.33. Robot kolunda ikinci eksenin kütlesi, kütle merkezi ve atalet momenti

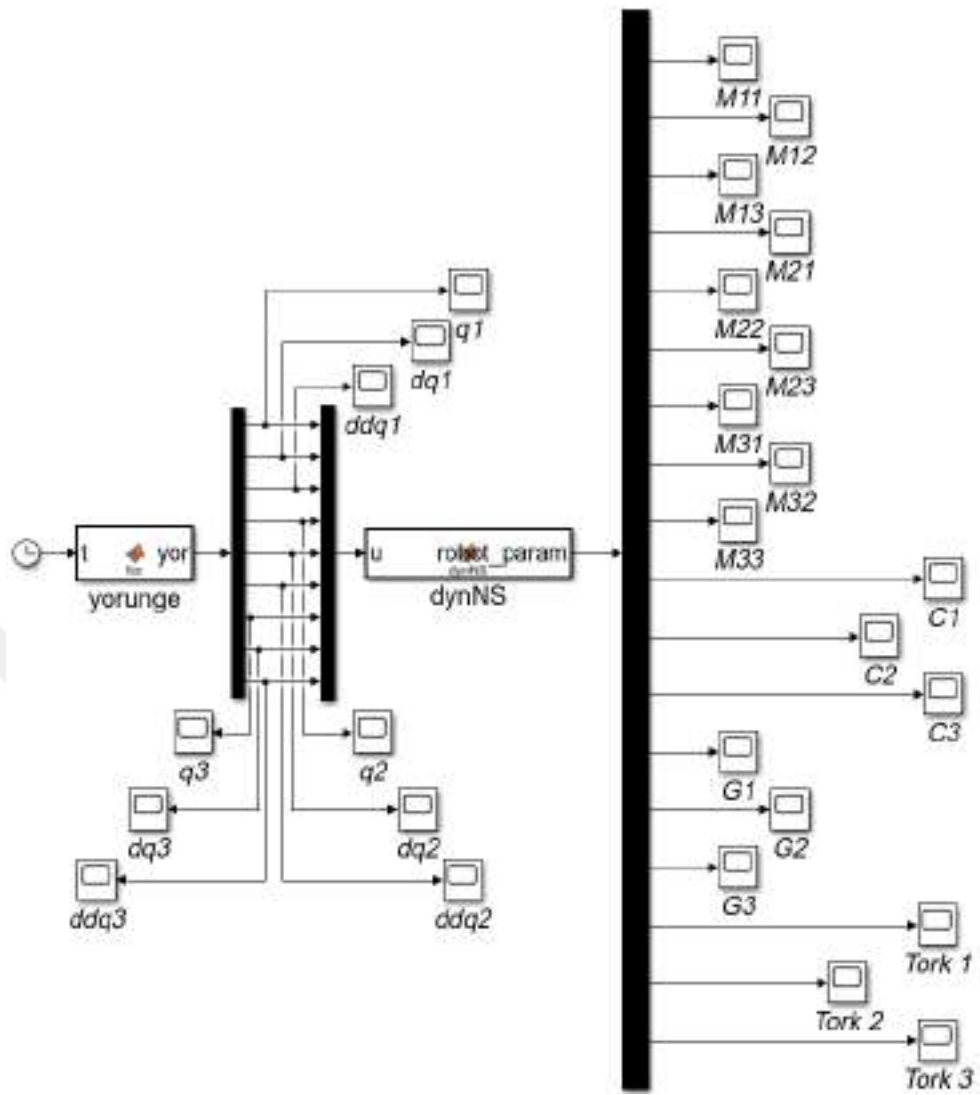
<b>Kütle (kg)</b>	
3,15	
<b>Koordinat sistemi 4'e göre kütle merkezi (mm)</b>	
X = 4.14070 Y = 58.96571 Z = -48.81630	
<b>Koordinat sistemi 3'e göre atalet momenti (g * mm)</b>	
I <sub>xx</sub> = 42206073.35148 I <sub>xy</sub> = 799543.71866 I <sub>xz</sub> = 391450.27876	
I <sub>yx</sub> = 799543.71866 I <sub>yy</sub> = 32438682.37391 I <sub>yz</sub> = 12215986.26972	
I <sub>zx</sub> = 391450.27876 I <sub>zy</sub> = 12215986.26972 I <sub>zz</sub> = 35484620.39720	



**Şekil 4.34.** Robot kolunda uç işlevcinin kütlesi, kütle merkezi ve atalet momenti

Robot kolunun kütle özelliklerinden elde edilen veriler Lagrange Euler denkleminde kullanılarak robot koluna ait genel kütle matrisi, Coriolis ve Merkezkaç Kuvvetleri ve eklemelere etki eden tork değerleri hesaplanmıştır. Bu hesaplamalardan EK 2'de verilmiştir. Robot kolunun dinamik benzetim modeli Matlab programı Simulink kısmında Şekil 4.35'te gösterildiği gibi blok diyagramlarıyla oluşturulmuştur.

Blok diyagramlarından *yorunge* isimli bloğa Şekil 4.36'da gösterildiği gibi yörünge planlamasına ait veriler girilmiştir. Yörünge planlamasında sırasıyla eklemlere 30°, -60° ve 60° verilmiştir. Üçüncü eklem, ikinci ekleme bağlı ve ters yönlü hareket ettiğinden ikinci ekleme verilen değerlerin eksi işaretli üçüncü ekleme verilmiştir. Şekil 4.37 ve 4.38'de gösterildiği gibi Lagrange Euler denkleminde elde edilen veriler *dynNS* bloğunun içine girilmiştir.



Şekil 4.35. Robot kolu Matlab -Simulink benzetim modeli

```

function yor = fcn(t)

q1i=0;
q1f=30;
q2i=0;
q2f=-60;
q3i=0;
q3f=60;
tf=10;

%%% Birinci Eklem %%%

a10=q1i;
a11=0;
a12=(3/tf^2)*(q1f-q1i);
a13=(-2/tf^3)*(q1f-q1i);
konum1=a10+a11*t+a12*t^2+a13*t^3;
hiz1=a11+2*a12*t+3*a13*t^2;
ivme1=2*a12+6*a13*t;

%%% İkinci Eklem %%%

a20=q2i;
a21=0;
a22=(3/tf^2)*(q2f-q2i);
a23=(-2/tf^3)*(q2f-q2i);
konum2=a20+a21*t+a22*t^2+a23*t^3;
hiz2=a21+2*a22*t+3*a23*t^2;
ivme2=2*a22+6*a23*t;

%%% Üçüncü Eklem %%%

a30=q3i;
a31=0;
a32=(3/tf^2)*(q3f-q3i);
a33=(-2/tf^3)*(q3f-q3i);
konum3=a30+a31*t+a32*t^2+a33*t^3;
hiz3=a31+2*a32*t+3*a33*t^2;
ivme3=2*a32+6*a33*t;

yor=[konum1, hiz1, ivme1, konum2, hiz2, ivme2, konum3, hiz3, ivme3];

end

```

Şekil 4.36. Matlab -Simulink yörünge bloğu kodları

```

function robot_param = dynNS(u)

q1=u(1); %Birinci Eklemin Konumu
q2=u(4); %İkinci Eklemin Konumu
q3=u(7); %Üçüncü Eklemin Konumu
dq1=u(2); %Birinci Eklemin Hızı
dq2=u(5); %İkinci Eklemin Hızı
dq3=u(8); %Üçüncü Eklemin Hızı
ddq1=u(3); %Birinci Eklemin İvmesi
ddq2=u(6); %İkinci Eklemin İvmesi
ddq3=u(9); %Üçüncü Eklemin İvmesi

s1=sin(q1/pi/180); s2=sin(q2/pi/180);
c1=cos(q1/pi/180); c2=cos(q2/pi/180);

%%% Kütle Matrisi %%%

M11=(2.569057201*10^9)+(1.818509043*10^9)*(c2^2)+(1.472508526*10^7)*c2;
M12=(-1.221598627*10^7)+(1.423955558*10^8)*s2;
M13=(-1.221598627*10^7)-(3.243868237*10^7)*s1+(1.428304163*10^8)*s2;
M21=(-1.221598627*10^7)+(1.423955559*10^8)*s2;
M22=(1.856295895*10^9);
M23=(1.391644570*10^9)+(1.221598627*10^7)*s1;
M31=(-1.221598627*10^7)-(3.243868237*10^7)*s1+(1.428304164*10^8)*s2;
M32=(1.391644570*10^9)+(1.221598627*10^7)*s1;
M33=(2.443197254*10^7)*s1+(1.424083253*10^9)-(3.2438682*10^7)*(c1^2);

M=[M11 M12 M13; M21 M22 M23; M31 M32 M33];

%%% Coriolis ve Centrifugal Kuvvetler %%%

C1=-0.04*dq1*dq2*c1*s2*s1-
0.0006*(c1^2)*(c2^2)*s1*(dq3^2)+0.0003*dq2*dq3*s1*(c1^2)-0.02*(dq3^2)
*c1*(c2^2)*s1+0.0001*(dq3^2)*(c1^2)+0.0001*(dq2^2)*(c1^2)-0.0001*(c1^4)
*(dq3^2)-0.0001*(c1^4)*(dq2^2)-0.01*(dq3^2)*(c1^3)+0.0003*(dq3^2)*s1
*(c1^2)-0.01*dq2*dq3*(c1^3)+0.0001*(c1^4)*c2*(dq2^2)-0.0002*(c1^4)*dq2
*dq3+0.0002*dq2*dq3*(c1^2)-0.0006*(c1^2)*(c2^2)*s1*dq2*dq3+0.02*(dq3^2)
*c1*(c2^4)*s1+0.0006*(dq3^2)*s1*(c1^2)*(c2^4)-(3.637018086*10^9)*dq1*dq2
*s2*c2+0.0006*dq2*dq3*s1*(c1^2)*(c2^4)+0.05*(dq1^2)*c2+(1.423955558*10^8)
*(dq2^2)*c2+(1.428304163*10^8)*dq2*dq3*c2-(1.221598626*10^7)*(dq3^2)*c1-
(1.221598626*10^7)*dq2*dq3*c1-(3.243868236*10^7)*(dq3^2)*c1*s1-
(1.472508526*10^7)*dq1*dq2*s2-0.10*(dq1^2)*(c1^2)*c2;

C2=-0.001*(dq1^2)*(c1^2)*s2+(7.362542629*10^6)*(dq1^2)*s2-
(1.428304163*10^8)*dq1*dq3*c2+0.01*dq1*dq2*c2-(dq1^2)*(c1^2)*s2*c2-
0.01*(dq1^2)*c1*s2*s1+(1.221598627*10^7)*c1*dq1*dq3+0.0002*(c1^4)*dq1
*dq2+0.0002*(c1^4)*dq1*dq3+(1.818509044*10^9)*(dq1^2)*s2*c2-
0.0002*dq1*dq2*(c1^2)-0.0002*dq1*dq3*(c1^2);

C3=(1.428304164*10^8)*dq1*dq2*c2+(2.443197254*10^7)*c1*dq1*dq3+0.0002
*(c1^4)*dq1*dq2+0.0002*(c1^4)*dq1*dq3+(6.487736474*10^7)*dq1*dq3*c1*s1-
0.0002*dq1*dq2*(c1^2)-0.0002*dq1*dq3*(c1^2)+(1.221598627*10^7)*dq1*dq2*c1-
(3.243868237*10^7)*(dq1^2)*c1;

C=[C1;C2;C3];

```

Şekil 4.37. Matlab -Simulink dynNS bloğu kodları-1



```

XXX Yerçekimi Vektörü XXX

G1=0;
G2=(-2.777844329*10^7)*c2;
G3=(-1.744387561*10^7)*c2;

G=[G1;G2;G3];

XXX Eklem Torkları XXX
ddq=[ddq1;ddq2;ddq3];
tork=M*ddq+C*G;

tork1=tork(1,1);
tork2=tork(2,1);
tork3=tork(3,1);

XXX çıktı XXX

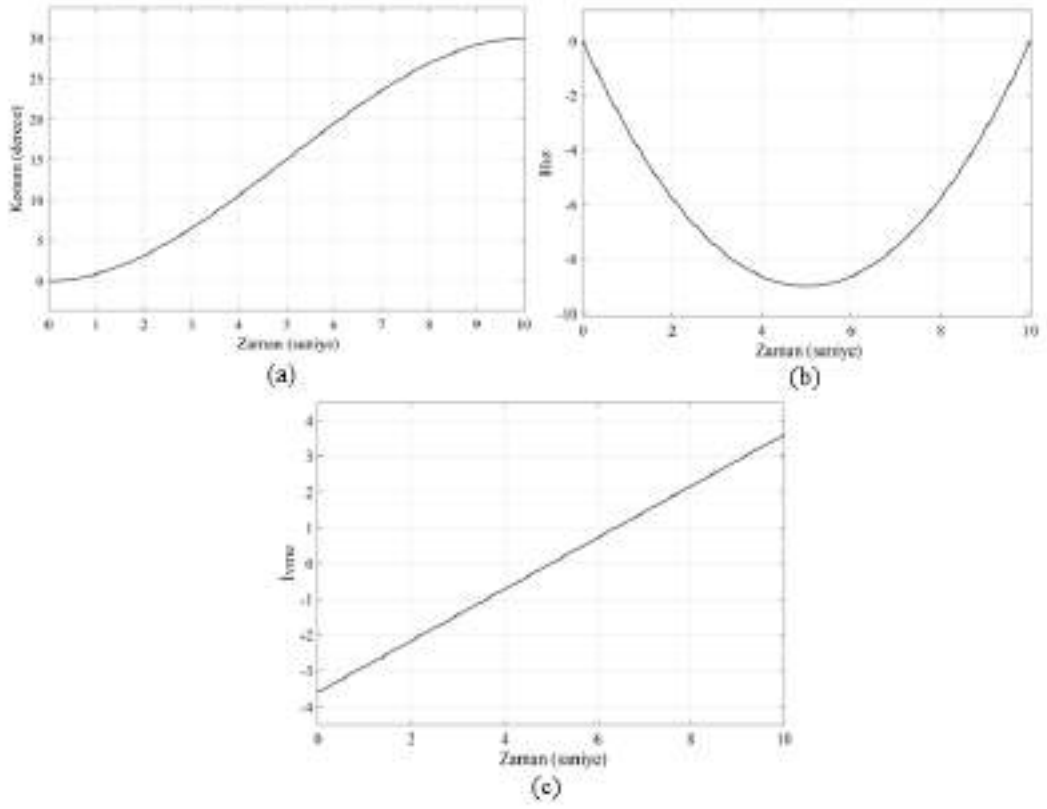
RP=[M11, M12, M13, M21, M22, M23, M31, M32, M33, C1, C2, C3, G1, G2, G3, tork1,
tork2, tork3];
robot_param=[RP];
end

```

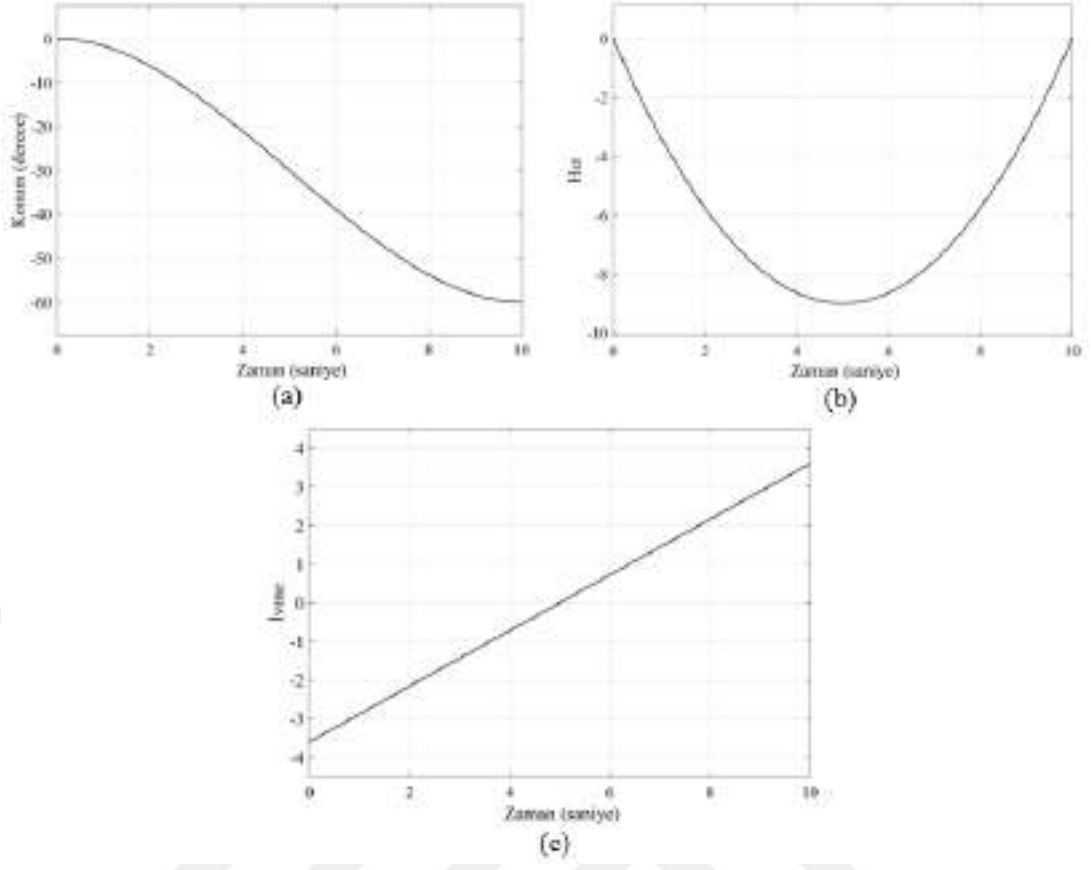
Şekil 4.38. Matlab -Simulink dynNS bloğu kodları-2

Blok diyagramı oluşturulup ve kodlar Matlab programına girildikten sonra simülasyon 10 saniye çalıştırıldı. Simülasyon sonucunda:

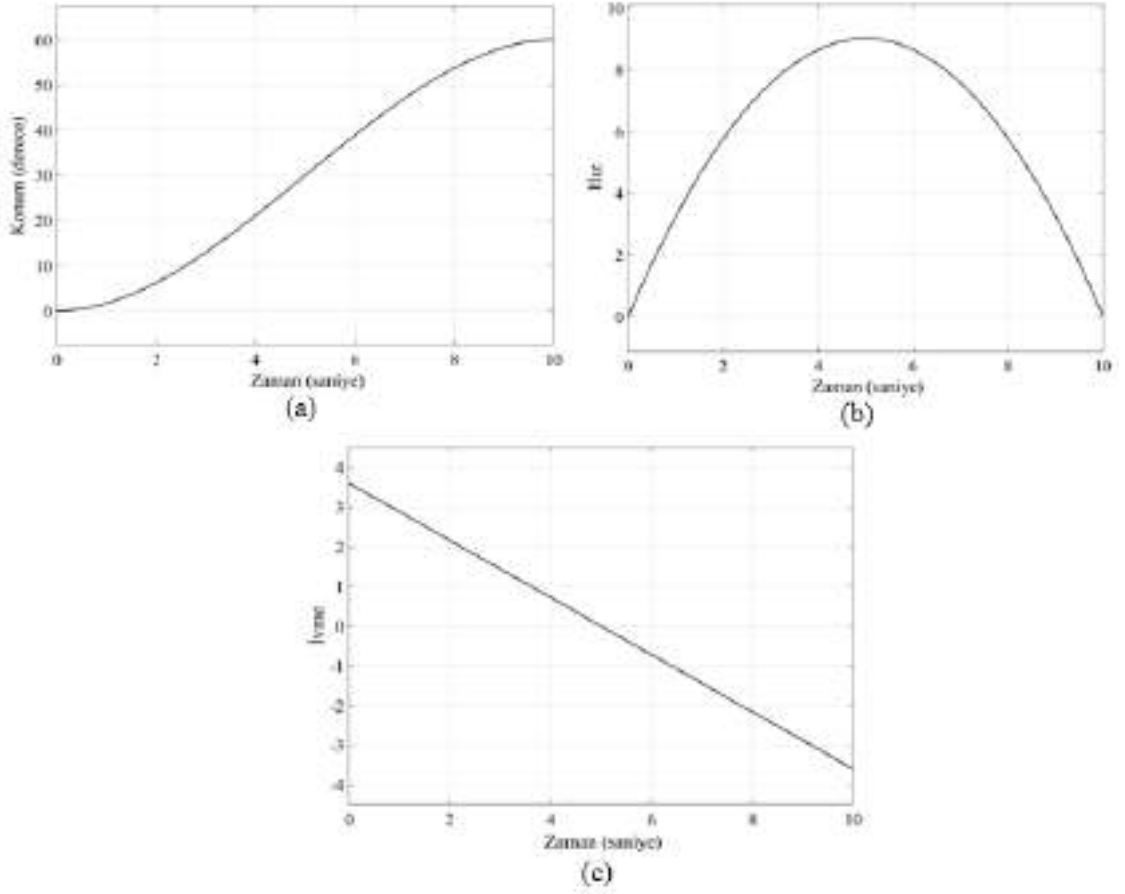
- Eklemlere ait konum hız ve ivme grafikleri Şekil 4.39-40 ve 41’de,
- Coriolis ve Merkezkaç Kuvvetler Şekil 4.42’de,
- Yer çekimi kuvvetleri Şekil 4.43’te,
- Tork kuvvetleri Şekil 4.44’de
- Genel kütle matrisi elemanları Şekil 4.45’te verilmiştir.



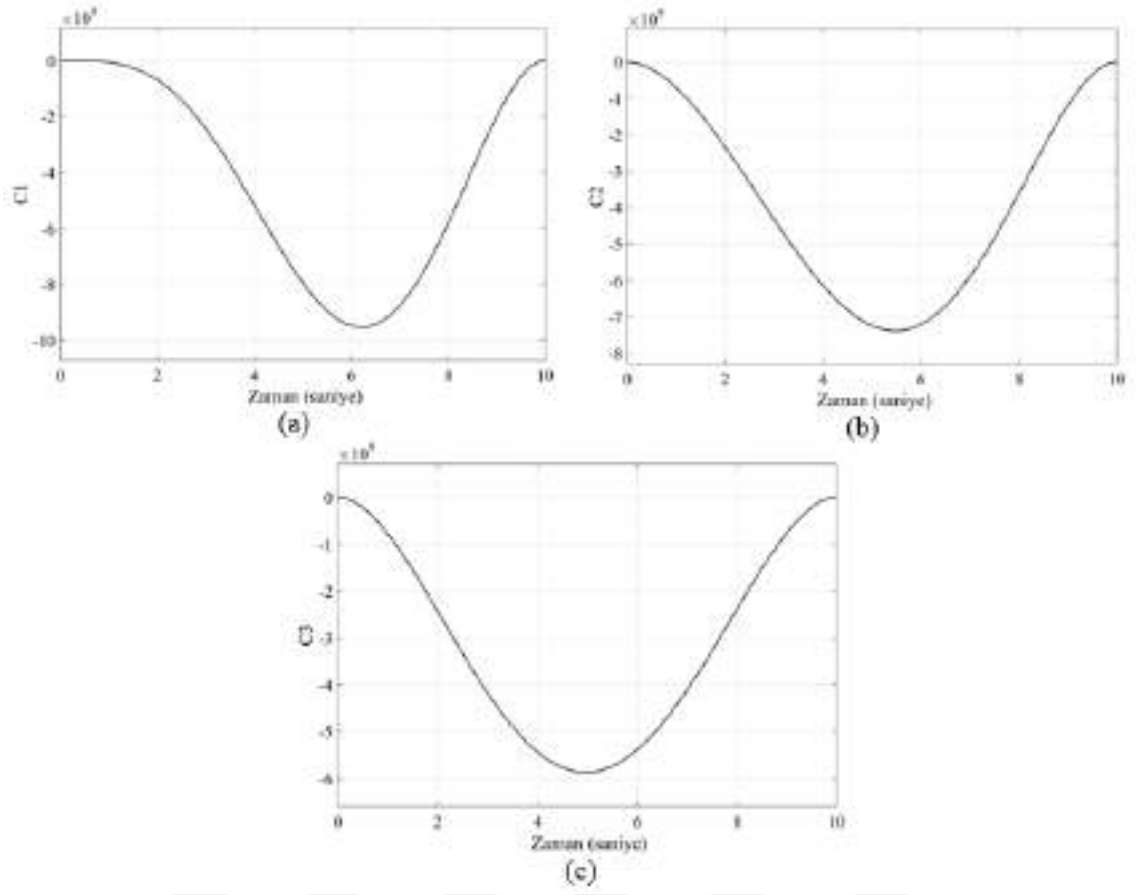
Şekil 4.39. Birinci ekleme ait a) konum, b) hız ve c) ivme zaman grafiği



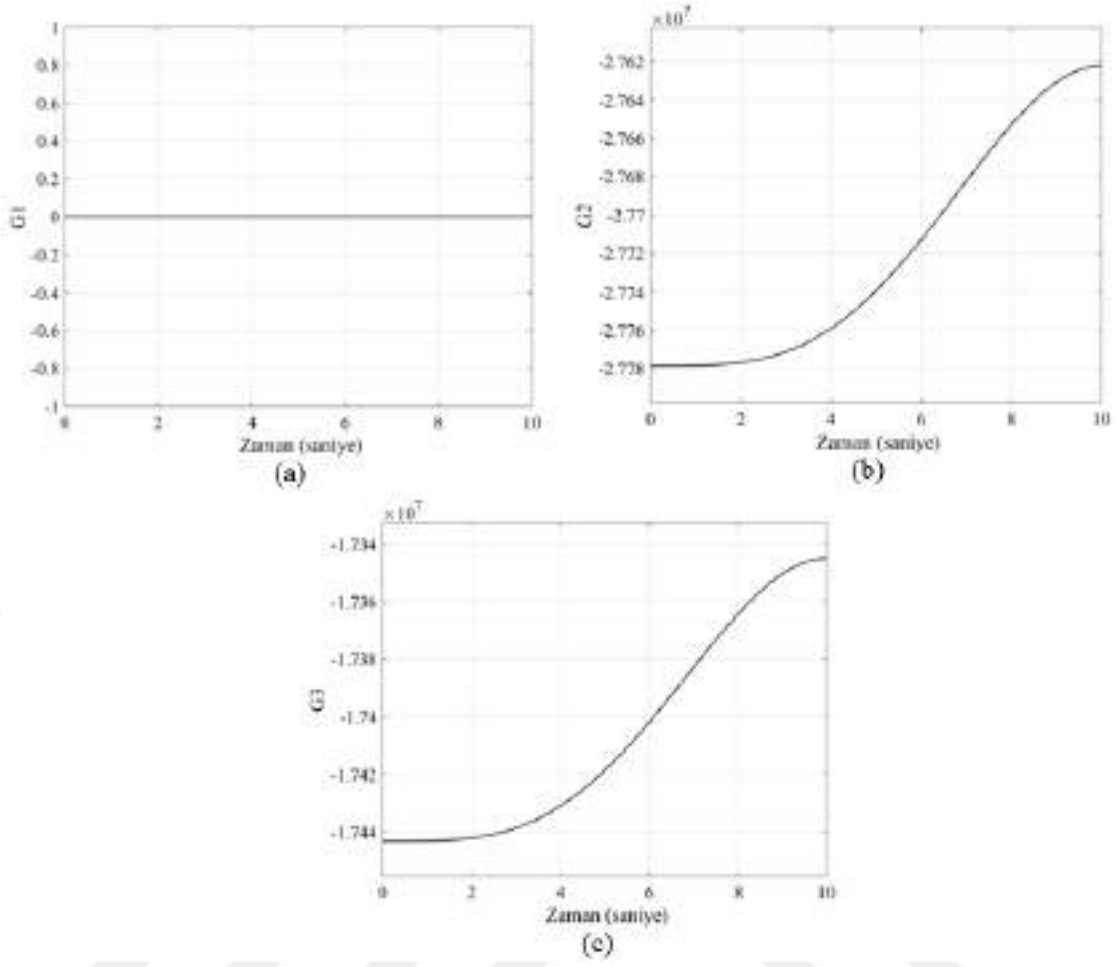
Şekil 4.40. İkinci ekleme ait a) konum, b) hız ve c) ivme zaman grafiği



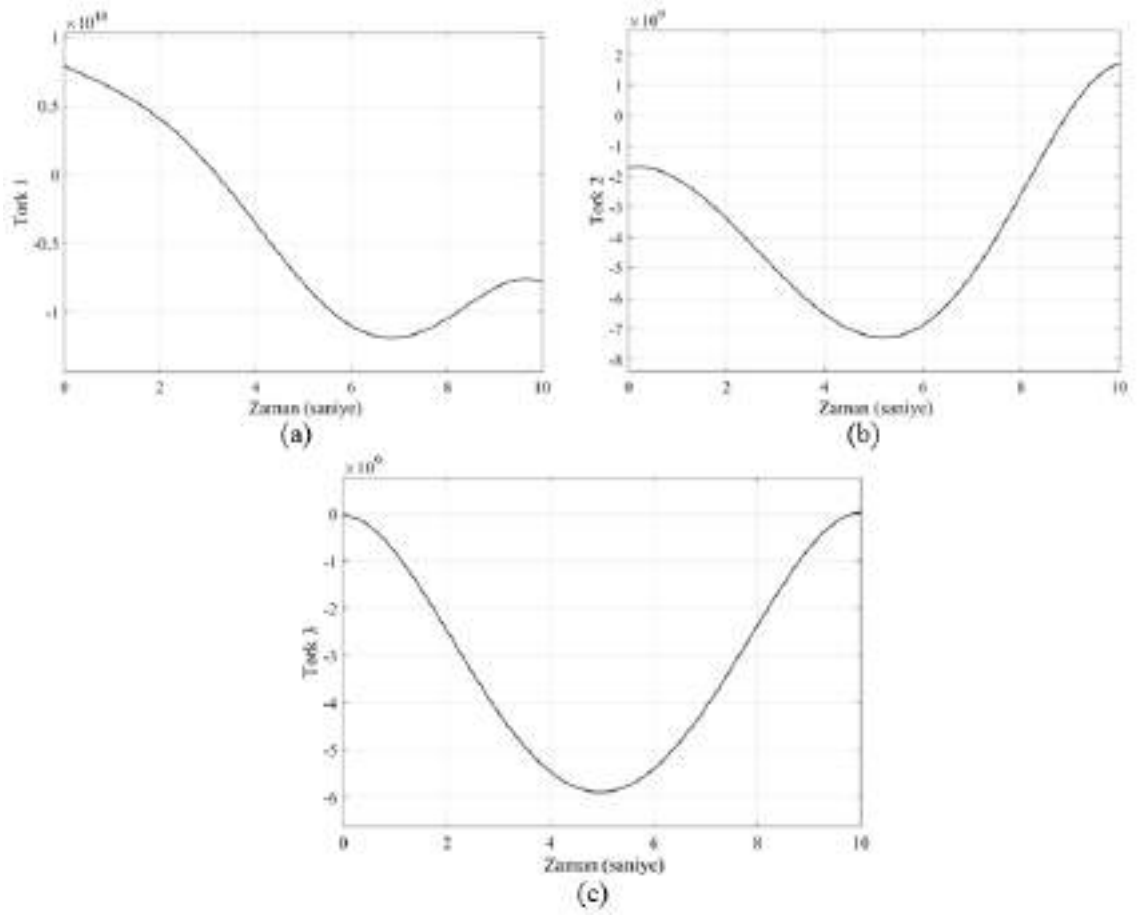
Şekil 4.41. Üçüncü ekleme ait a) konum, b) hız ve c) ivme zaman grafiği



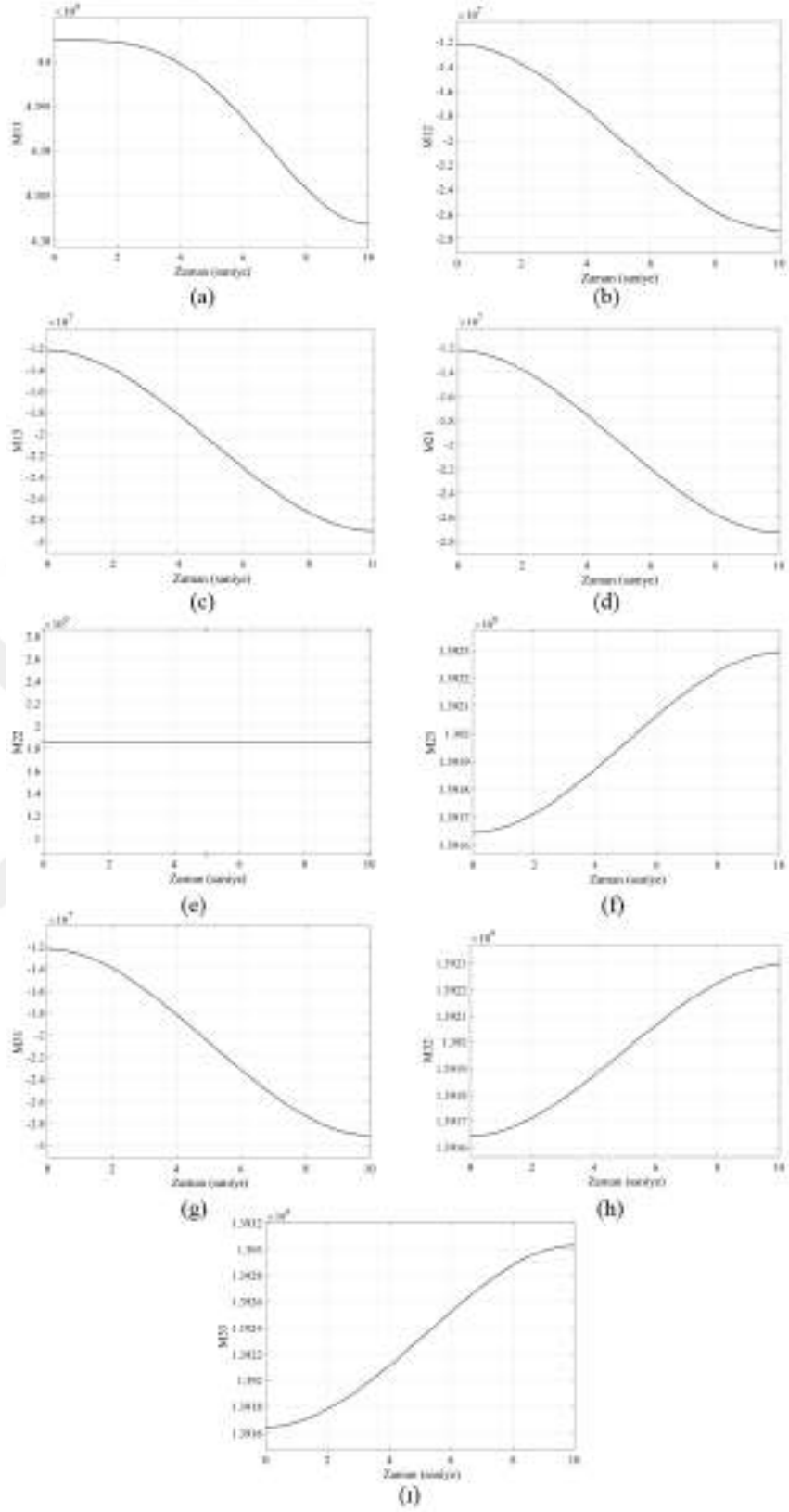
Şekil 4.42. Coriolis ve Merkezkaç Kuvvetleri: a)  $C_1$ , b)  $C_2$  ve c)  $C_3$



Şekil 4.43. Yer çekimi vektörleri: a) G1, b) G2 ve c) G3



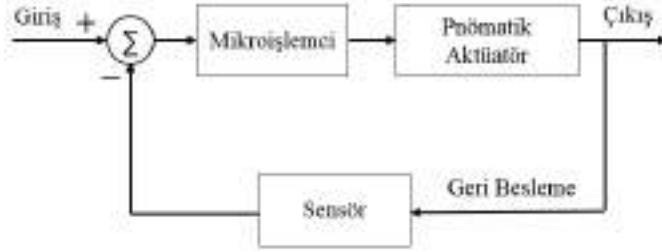
Şekil 4.44. Tork değerleri: a) Tork 1, b) Tork 2 ve c) Tork 3



Şekil 4.45. Genel kütle matrisi elemanları: a)  $M_{11}$ , b)  $M_{12}$ , c)  $M_{13}$ , d)  $M_{21}$ , e)  $M_{22}$ , f)  $M_{23}$ , g)  $M_{31}$ , h)  $M_{32}$  ve i)  $M_{33}$

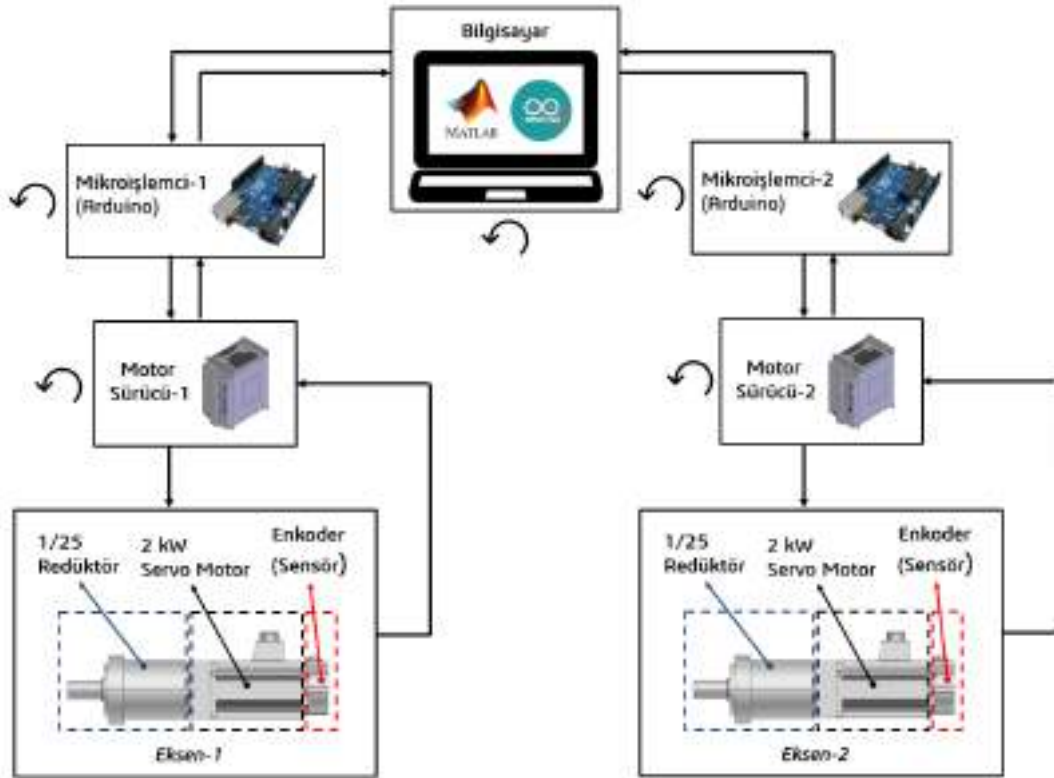


#### 4.4. Tasarılan Robot Kolunun Blok Diyagramları

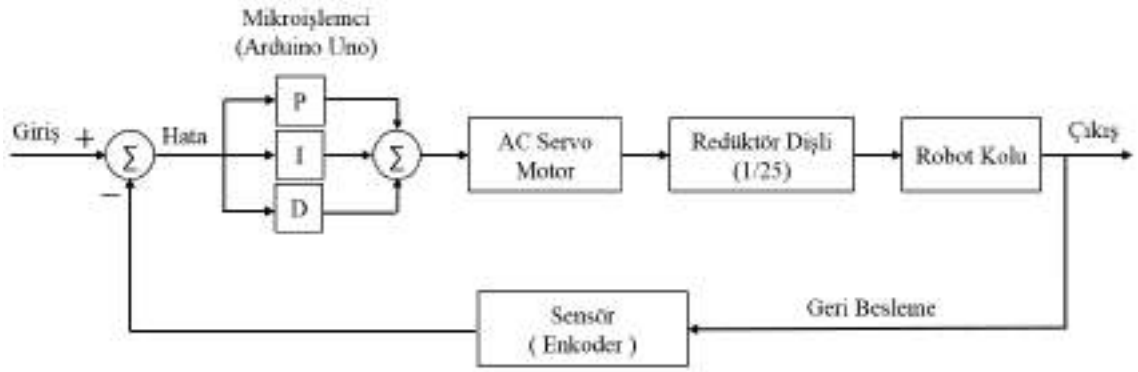


Şekil 4.46. Robot kolu pnömatik aktüatör blok diyagramı

Şekil 4.46’da robot kolunun uç işlevcisine ait pnömatik aktüatör blok diyagramı gösterilmektedir. Sistemde bulunan röle, mikro işlemci üzerinden kontrol edilerek pnömatik silindirin ileri ve geri yönlü doğrusal hareket yapması sağlanır.



Şekil 4.47. Kontrol şeması



**Şekil 4.48.** Robot kolu genel blok diyagramı

Şekil 4.47 ve 4.48’de sırasıyla sistemin kontrol şeması ve blok diyagramı gösterilmektedir. Bilgisayardan Matlab arayüzü ile gönderilen konum bilgileri Arduino ile motor sürücüyeye iletilir. Motor sürücü motoru konumlandırdıktan sonra enkoderden Arduino’ya geri besleme sağlanır. Enkoder bilgisi Arduino’da işlendikten sonra tekrar bilgisayarda Matlab arayüzüne geri bildirim yapılır. Gelen bilgiler Matlab’da veri (data) saklama yapılarak grafiğe dönüştürülür.

## 5. ARAŞTIRMA SONUÇLARI VE TARTIŞMA

### 5.1. Mikroişlemci (Arduino Uno) ile Matlab Seri Haberleşme

Robot kolu Matlab ve Arduino programı arasında seri iletişim kurularak sürülmüştür. Deney aşamasında Lagrange Euler denklemleri ve yörünge planlaması denklemleri Matlab ortamına dahil edilmemiştir. Robot kolu sabit hızda sürülmüştür. Robot kolunun hareket mesajları ve grafik oluşturma işlemi Matlab üzerinden yapılırken enkoder verisi okuma ve servo motorların PID kontrolü işlemi Arduino üzerinden yapılmıştır. Matlab programının komut ekranında kullanılan kodlar Ek-A bölümünde Şekil EK 1.1-1.8'de verilmiştir. Motorların PID kontrolü Arduino kütüphanesinden yararlanılarak oluşturulmuştur. Arduino kodları Ek-A bölümünde Şekil EK 1.9-1.16'da verilmiştir. Arduino bağlantıları bölüm 4.1.4'te verilmiştir. Robot kolunu sürme sonucunda elde edilen grafikler bölüm 5.2'de gösterilmiştir.

Robot kolunu sürmede kolaylık olması için Şekil 5.1'de gösterildiği gibi Matlab programının komut ekranı bölmesinde harekete dair bilgiler giriliyor. Öncelikle Ek-A bölümü Şekil 1'deki gösterilen komut çalıştırıldığında sistem düz veya ters kinematikte çalışmak istenildiğini soruyor. Eğer cevap evet ise 1 değilse 0 olarak giriliyor. Daha sonra eksenlerin dönme miktarı, yönü ve bu hareket neticesinde oluşacak grafiklerin örnekleme periyodu sisteme giriliyor. İstenilen hareketi yapıldıktan sonra grafikler oluşturuluyor. Grafik oluşturulurken örnekleme periyodu işlemci hızı ve grafiğin doğruluğu (stabilite) baz alınarak deneysel olarak 100 ms olarak belirlenmiştir. Hareket mesajına örnek verilirse:

- Hangi eksenler hareket edecek mesajına 11 girildiğinde iki eksen de hareket edecektir.

1. eksen hareket mesajına 110100015 girildiğinde yani birinci eksen pozitif yönde  $15^\circ$  açı yapacak ve grafik 100 ms örnekleme periyodu ile oluşturulacaktır.

2. eksen hareket mesajına -110100035 girildiğinde birinci eksen negatif yönde  $35^\circ$  açı yapacak ve grafik 100 ms örnekleme periyodu ile oluşturulacaktır.

```

Command Window
Workspace
Variables - encoded1deg

Düz kin mi ters kin mi (1/0)
1
Hangi eksen veya eksenler hareket edicek (11/10/01/00)
11
1. eksen Hareket mesajını giriniz
110100015
2. eksen Hareket mesajını giriniz örnekleme hizi 1.eksen ile aynı
-110100035
devam etmek istiyor musunuz (1/0)
1
Düz kin mi ters kin mi (1/0)
1
Hangi eksen veya eksenler hareket edicek (11/10/01/00)
11
1. eksen Hareket mesajını giriniz
110100050
2. eksen Hareket mesajını giriniz örnekleme hizi 1.eksen ile aynı
-110100060
devam etmek istiyor musunuz (1/0)
1

```

Şekil 5.1. Matlab komut ekranı

### 5.1.1. PID katsayılarının belirlenmesi

Robot kolunun Ziegler Nichols yöntemi ile belirlenen değerlerde alınan sistem cevabı uygun olmamıştır. Bu sebeple deneme yanılma yöntemiyle yaklaşık kp, ki ve kd değerleri bulundu. Deneme aşamasındaki birinci eksen için 45° ve ikinci eksen için -30° değerleri baz alındı. Deneme sırasında kullanılan bazı değerler Tablo 5.1'de gösterilmiştir. Her bir eksendeki konum farkları alınarak hata vektörü oluşturuldu. Hata vektörü minimize edilerek uygun kp, ki ve kd değerleri bulundu.

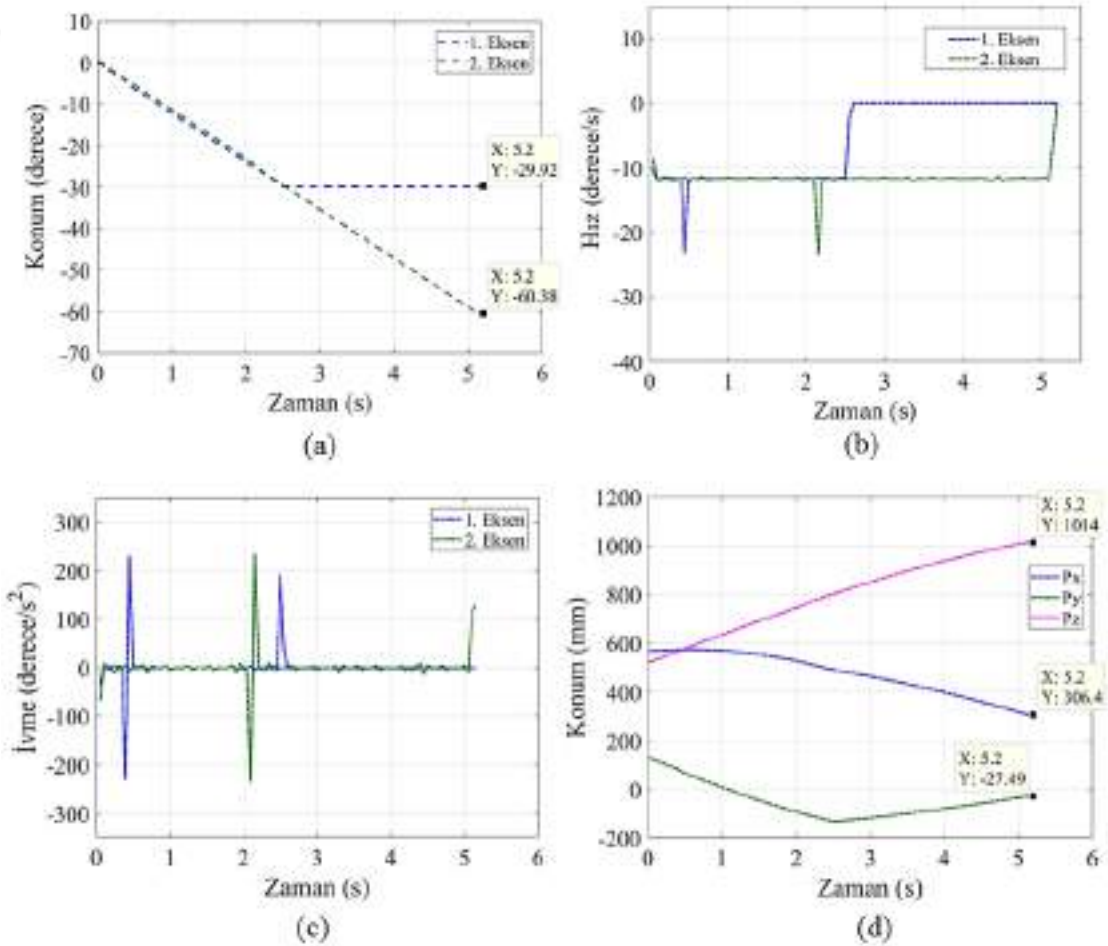
Tablo 5.1. PID katsayıları ve sistem cevabı

1.Eksen 45°			2. Eksen -30°			Sistem Cevabı									
Kp	Ki	Kd	Kp	Ki	Kd	Px-Teo. (XG) mm	Px-Den. (XD) mm	Px-Hata (mm)	Py-Teo. (XG) mm	Py-Den. (XD) mm	Py-Hata (mm)	Pz-Teo. (XG) mm	Pz-Den. (XD) mm	Pz-Hata (mm)	Hata Vektörü (mm)
0,0473	1E-05	1E-05	0,0253	1E-05	1E-05	254,896	256,614	1,718	437,329	437,449	0,12	805,6	803,855	-1,745	2,452
0,0473	3E-05	1E-05	0,0253	1E-05	1E-05	254,896	256,206	1,31	437,329	437,622	0,293	805,6	803,958	-1,642	2,121
0,0473	5E-05	1E-05	0,0253	1E-05	1E-05	254,896	256,022	1,126	437,329	437,73	0,401	805,6	803,958	-1,642	2,031
0,0473	6E-05	1E-05	0,0253	1E-05	1E-05	254,896	255,93	1,034	437,329	437,783	0,454	805,6	803,958	-1,642	1,993
0,0473	6E-05	5E-05	0,0253	1E-05	1E-05	254,896	255,93	1,034	437,329	437,783	0,454	805,6	803,958	-1,642	1,993
0,0473	6E-05	1E-04	0,0253	1E-05	1E-05	254,896	256,114	1,218	437,329	437,676	0,347	805,6	803,958	-1,642	2,074
0,0473	2E-04	1E-04	0,0253	1E-05	1E-05	254,896	254,646	-0,25	437,329	438,532	1,203	805,6	803,958	-1,642	2,051
<b>0,0473</b>	<b>1E-04</b>	<b>1E-04</b>	0,0253	1E-05	1E-05	<b>254,896</b>	<b>255,288</b>	<b>0,392</b>	<b>437,329</b>	<b>438,156</b>	<b>0,827</b>	<b>805,6</b>	<b>803,958</b>	<b>-1,642</b>	<b>1,880</b>
0,0473	1E-04	1E-04	0,0253	2E-05	1E-05	254,896	255,247	0,351	437,329	438,116	0,787	805,6	804,061	-1,539	1,764
0,0473	1E-04	1E-04	0,0253	3E-05	1E-05	254,896	255,297	0,401	437,329	438,021	0,692	805,6	804,164	-1,436	1,644
0,0473	1E-04	1E-04	0,0253	4E-05	1E-05	254,896	255,164	0,268	437,329	438,033	0,704	805,6	804,266	-1,334	1,532
0,0473	1E-04	1E-04	0,0253	5E-05	1E-05	254,896	255,122	0,226	437,329	437,991	0,662	805,6	804,369	-1,231	1,416
0,0473	1E-04	1E-04	0,0253	8E-05	1E-05	254,896	255,005	0,109	437,329	437,729	0,4	805,6	804,882	-0,718	0,829
0,0473	1E-04	1E-04	<b>0,0253</b>	<b>1E-04</b>	<b>1E-05</b>	<b>254,896</b>	<b>254,88</b>	<b>-0,016</b>	<b>437,329</b>	<b>437,604</b>	<b>0,275</b>	<b>805,6</b>	<b>805,19</b>	<b>-0,41</b>	<b>0,494</b>

## 5.2. Enkoderden Veri Alma

Motor sürücüsünün arka kısmına entegre olan 14 bitlik enkoder 1 devirde 30000 pulse olacak şekilde ayarlandı. Robot kolunda istenilen açı değerleri sisteme girildiğinde eksenler aynı anda hareket etmekte ve istenilen açı dercesine ulaşan eksen durmaktadır. Robot kolunu test etmek için örnek açı değerleri verilerek mutlak hata, bağıl hata oranları ve standart sapmalar hesaplanmıştır.

Robot kolunun birinci eksenine  $-30^\circ$ , ikinci eksenine  $-60^\circ$  verildiğinde oluşan grafikler sırasıyla konum, hız, ivme ve uç işlevciye ait konum zaman grafikleri Şekil 5.2'de verilmiştir.



Şekil 5.2.  $-30^\circ$  ve  $-60^\circ$  çalışma aralığında a) konum, b) hız, c) ivme ve d) uç işlevciye ait konum zaman grafiği

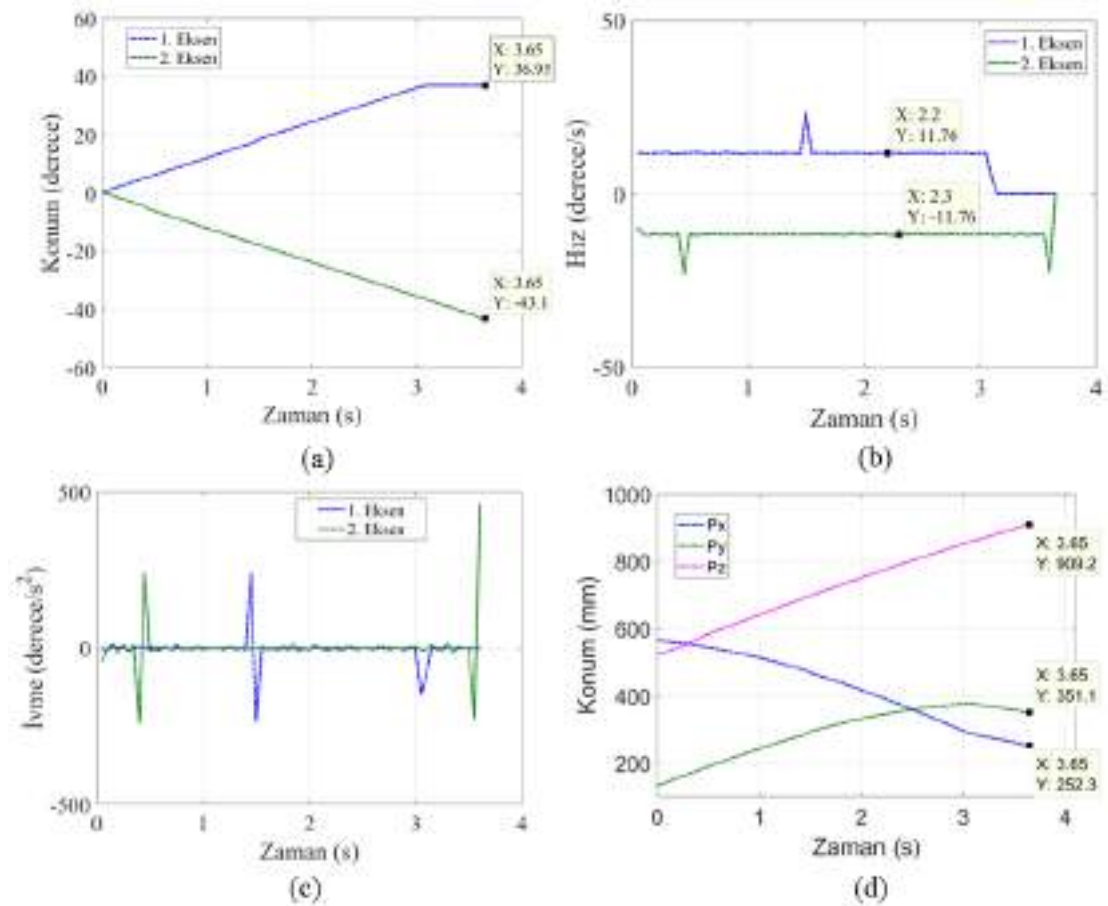
Konum zaman grafiği incelendiğinde mutlak hata değerleri birinci ve ikinci eksenle sırasıyla (0,08) ve (0,38) bağıl hata değerleri sırasıyla (% -0,267) ve (% -0,629) olarak hesaplanmıştır. Hız zaman grafiği incelendiğinde robot kolunun bazı noktalarda pik yaptığı ve her iki eksenle de ortalama 11,76 derece/saniye hızla hareket ettiği

görülmektedir. İvme zaman grafiği incelendiğinde robot kolu sabit hızla hareket ettiği halde bazı noktalarda pik yaptığı görülmüştür. Grafiklerde oluşan ara ara pikler aşağıdaki sebeplerden kaynaklanabilir:

- Yataklama problemi
- Tasarımda stabilite (kararlı olma) problemi
- Devreden kaynaklanan parazit
- Kullanılan redüktördeki diş boşluğu

Uç işlevciye ait konum zaman grafiği incelendiğinde x, y ve z eksenlerindeki mutlak hata değerleri sırasıyla (2,838), (2,092) ve (1,523) iken bağıl hata değerleri (% 0,926), (% -7,91) ve (% 0,15)'dir.

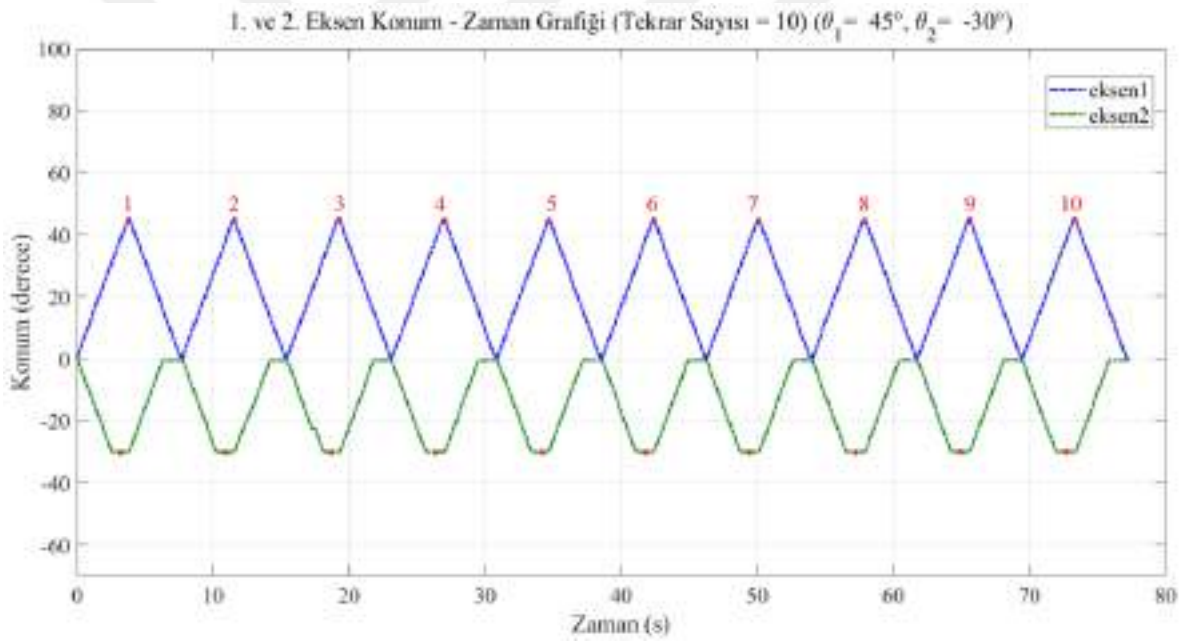
Robot kolunun birinci eksenine  $37^\circ$ , ikinci eksenine  $-43^\circ$  verildiğinde oluşan grafikler sırasıyla konum, hız, ivme ve uç işlevciye ait konum zaman grafikleri Şekil 5.3'de verilmiştir.



Şekil 5.3.  $37^\circ$  ve  $-43^\circ$  çalışma aralığında a) konum, b) hız, c) ivme ve d) uç işlevciye ait konum zaman grafiği

Konum zaman grafiđi incelendiđinde mutlak hata deđerleri birinci ve ikinci eksenle sırasıyla (0,05) ve (0,1) bađıl hata deđerleri sırasıyla (% 0,135) ve (% -0,232) olarak hesaplanmıřtır. Hız zaman grafiđi incelendiđinde robot kolu her iki eksenle de ortalama 11,76 derece/saniye hızla hareket etmektedir. İvme zaman grafiđi incelendiđinde robot kolu sabit hızla hareket ettiđi halde bazı noktalarda pik yaptıđı gorulmuřtur. Grafiklerde oluřan ara ara pikler aynı řekilde onceli grafikteki sebeplere dayanmaktadır. U iřlevciye ait konum zaman grafiđi incelendiđinde x, y ve z eksenlerindeki mutlak hata deđerleri sırasıyla (0,19), (0,69) ve (0,735) iken bađıl hata deđerleri (% 0,075), (% 0,197) ve (% 0,081)'dir.

Robot koluna birinci eksene  $45^\circ$  ve ikinci eksene  $-30^\circ$  deđerleri verildiđinde ve bu sure tekrar sayısı 10 olacak řekilde ayarlandıđında oluřan konum zaman grafiđi řekil 5.4'te verilmiřtir.



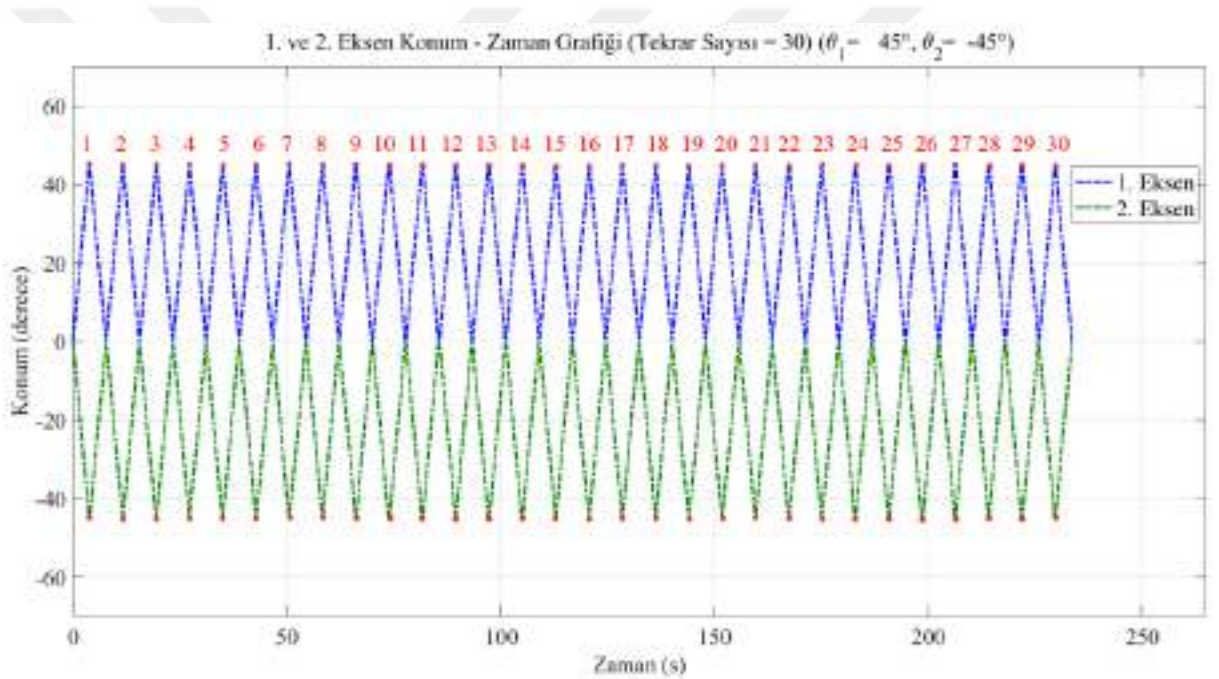
**řekil 5.4.** Birinci ve ikinci eksen konum-zaman grafiđi (Tekrar Sayısı = 10) ( $\theta_1 = 45^\circ, \theta_2 = -30^\circ$ )

Robot kolunun bu tekrarlanma sonucunda enkoderden olulen aı deđerleri ve bu aı deđerlerine bađlı olarak hesaplanan u iřlevciye ait konum deđerlerinin ortalama, mutlak hata, bađıl hata ve standart sapma deđerleri Tablo 5.2'de gosterilmiřtir.

**Tablo 5.2.** Tekrar deneyi-1 hata değerleri

Deney-1	$\theta_1$	$\theta_2$	Px	Py	Pz
gerçek değer	45,000	-30,000	254,896	437,330	805,600
ortalama değer	45,021	-29,995	254,753	437,441	805,557
mutlak hata (min.)	0,010	0,000	0,007	0,054	0,000
mutlak hata (maks.)	0,040	0,040	0,375	0,184	0,342
bağıl hata (min.)	0,022	0,000	0,003	0,012	0,000
bağıl hata (maks.)	0,089	-0,134	0,147	0,042	0,043
standart sapma	0,011	0,019	0,145	0,033	0,162

Robot koluna birinci eksene  $45^\circ$  ve ikinci eksene  $-45^\circ$  değerleri verildiğinde ve bu süreç tekrar sayısı 10 olacak şekilde ayarlandığında oluşan konum zaman grafiği Şekil 5.5'te verilmiştir.



**Şekil 5.5.** Birinci ve ikinci eksen konum-zaman grafiği (Tekrar Sayısı = 30) ( $\theta_1 = 45^\circ$ ,  $\theta_2 = -45^\circ$ )

Robot kolunun bu tekrarlanma sonucunda enkoderden ölçülen açı değerleri ve bu açı değerlerine bağlı olarak hesaplanan uç işlevciye ait konum değerlerinin ortalama, mutlak hata, bağıl hata ve standart sapma değerleri Tablo 5.3'de gösterilmiştir.



**Tablo 5.3.** Tekrar deneyi-2 hata deęerleri

<b>Deney-2</b>	<b><math>\theta_1</math></b>	<b><math>\theta_2</math></b>	<b>P<sub>x</sub></b>	<b>P<sub>y</sub></b>	<b>P<sub>z</sub></b>
<i>gerçek deęer</i>	45,000	-45,000	191,383	373,817	922,657
<i>ortalama deęer</i>	45,006	-45,051	191,094	373,587	923,010
<i>mutlak hata (min.)</i>	0,000	0,000	0,016	0,016	0,000
<i>mutlak hata (maks.)</i>	0,040	0,140	0,952	0,559	0,975
<i>baęıl hata (min.)</i>	0,000	0,000	0,008	0,004	0,000
<i>baęıl hata (maks.)</i>	0,089	-0,310	0,500	0,150	0,106
<i>standart sapma</i>	0,014	0,046	0,315	0,180	0,317

## 6. SONUÇLAR VE ÖNERİLER

Bu çalışmada iki eksenli, küresel yörünge çizen ve özel uç işlevcili robotun tasarımı, imalatı ve deneysel çalışmaları gerçekleştirildi. Pnömatik uç işlevci küresel ve kulaklı cisimleri kavramak için tasarlandı. Ayrıca uç işlevcinin zeminle paralellliğini bozmayacak şekilde uç işlevci ile ikinci eksen arasında kasnak sistemi tasarlandı. Bu sayede cisim bulunduğu yerdeki duruş pozisyonunun koruyarak istenilen noktaya taşınması sağlandı. Eklenen kasnak sistemi bir nevi üçüncü eksen görevi gördü ve ikinci eksendeki servo motoru eyleyici olarak kullanarak eksen ve motor tasarrufu sağladı.

Robot kolunda kullanılan iki adet AC servo motorun PWM ile sürülmesi ve konum kontrolü Arduino UNO mikroişlemci ile sağlanmıştır. Kullanılan mikroişlemci güncel olması, ucuz olması ve açık kaynaklara sahip olması nedeniyle tercih edilmiştir. Arduino programı üzerinden alınan enkoder verileri Matlab programı ile haberleştirilmiştir. Bu sayede Matlab programı üzerinden kontrol ekranı ve grafikler elde edilmiştir. Her bir servo motor, sınır şartlarında eksen hareketlerini elde edecek şekilde dönüş miktarı ve yönü ayarlanabilir şekilde çalıştırılmıştır.

Robot kolunun titreşimsiz, ani hareketlerden kaçınarak ve rezonansa girmeden hareket etmesi için yörünge planlaması denklemleri oluşturulmuştur. Robot kolunun Solidworks programında kütleli özellikleri dikkate alınarak ve Lagrange Euler metodu kullanılarak dinamik denklemleri oluşturulmuştur. Elde edilen yörünge planlaması ve Lagrange Euler denklemleri Matlab programı üzerinden örnek açı değeri verilerek benzetim oluşturulmuş ve grafikler elde edilmiştir.

Robot kolunda eksenlere açı değerleri verilmesi durumunda hangi koordinata gideceği veya uç işlevcinin koordinatı verildiğinde hangi açıyla değerleri ile bu koordinata ulaşacağı sırasıyla düz ve ters kinematik hesaplamalar sayesinde bulunmuştur. Hesaplamalar örnek açı değerleri verilerek doğrulanmıştır. Doğrulanmış bu hesaplamalar Matlab arayüzüne gömülmüş ve komut ekranı oluşturulmuştur. Sisteme örnek açı değerleri verilerek başarıyla çalıştığı gözlemlenmiştir.

Deney aşamasında verilen örnek açı değerleri ve tekrarlamalı hareketlerin sapma ve hata değerlerine bakıldığında yapılacak işlem veya prosesin çok hassas olacağı kanaati oluşmamıştır. Bu sapma değerlerine göre robot kolunun bu haliyle daha ziyade orta hassasiyette işlemlerde kullanılması tavsiye edilir. Ancak yüzey toleransları, yatak toleransları ve imalat hataları giderildiği zaman bu hataların ve sapmaların büyük oranda

ortadan kalkacağı ifade edilebilir. Bununla birlikte yazılımsal olarak uygun seçilecek PID değerleri ile motor karakteristikleri düzenlenerek yörüngedeki sapmalar optimize edilebilir. Bu haliyle mümkün olan en uygun PID değerleri Ziegler Nichols yöntemi ile belirlenmeye çalışılmış fakat bulunan bu değerler ile elde edilen sistemin cevabı uygun olmamıştır. Bu sebeple bulunan PID değerlerine  $\pm$  ekleme ve çıkarma yapılarak yaklaşık  $k_p$ ,  $k_i$  ve  $k_d$  değerleri bulunmuştur.

Yapılan bu çalışmada robot kolu ve uç işlevcinin tasarımı yapılmış ve elde edilen sonuçlara göre endüstriyel uygulamalarda kullanılabileceği ortaya konulmuştur. Ancak birkaç değişiklik yapılması ve geliştirmenin yapılması uygulamada kolaylık sağlamanın yanında karşılaşılabilecek bazı sorunların çözümünü de gerçekleştirebilir.

Robot koluna uygun kontrol arayüzü ve uygun kontrol yöntemi seçilerek yüksek toleranslar içinde hareket edebilen hassas bir tasarım yapılabilir.

Bu robot kolu günümüzdeki fabrika ortamlarında bir işçinin bir mesai süreci içerisinde yapabileceği işi daha kısa bir zamanda ve daha iyi bir standartta yapması amacıyla kullanılabilir. Aynı zamanda iş güvenliği gerektiren tehlikeli ortamlarda kullanılabilme özelliğine sahiptir. Fabrika ortamı incelenerek kullanılacak ortamın ihtiyacına göre robot kolunun boyutsal değişimi ve farklı uç işlevci uygulamaları ile ortama uygun hale getirilebilir.

Tekrarlı hareketler için 10 ve 30 tekrar sayılarının standart sapmalarının değişimi %7,4 ile %17,4 aralığında olduğu için robot kolunun imalatı esnasında ve boyutlandırılmasında düzeltmelere ihtiyaç duyulduğu görülmektedir. Bu sapma değerlerinin hassas işlere uygun olabilmesi için imalat ve montaj hatalarının giderilmesi gerekir. Benzer çalışmalarda bu sapma değerlerinin %0,1 civarında olduğu görülmüştür.

Robot kolu ve uç işlevci belirli bir malzemeye odaklanarak tasarım gerçekleştirilmiştir. Ancak malzeme optimizasyonu veya farklı malzemeler kullanılarak aynı işin yapılması mümkün olabilir. Bu da bize daha küçük motor ve redüktör sistemi kullanılarak daha düşük maliyetli ve daha uygun tasarım yapılarak bu robot kolunun optimize edilmesi imkânı sağlanabilir. Bu tezde BAP tarafından araştırma projesi ile sağlanan motorlar ve redüktörler kullanıldığı için çalışmanın içerisinde boyutsal optimizasyon konusu ele alınamamıştır. Belirli bir çalışma alanı tespit edilerek bu boyutlara uygun robot kol ve uç işlevci tasarımı yapılmış ve herhangi bir mukavemet hesaplaması yapılmasına ihtiyaç duyulmayacak malzeme seçilmiştir. Bu optimizasyon ve

uygun malzeme seçimi yapıldığı takdirde robot kolunun hareketi esnasında ivme zaman grafiğinde oluşan pikler görülmeyecektir.



## KAYNAKLAR

- Adalı, Ş. (2001). *3-boyutlu Robot Simülasyonu*. (Yüksek Lisans Tezi).  
<https://polen.itu.edu.tr/handle/11527/10668> adresinden erişildi.
- Adar, N. G., Ören, H. ve Kozan, R. (2013). 5 Serbestlik Dereceli Robot Kolunun Modellenmesi ve Kontrolü. *Sakarya Üniversitesi Fen Bilimleri Enstitüsü Dergisi*, 17(1), 155-160. doi:10.16984/saufbed.22465
- Akı, M. O. (t.y.). Robot Bilimi Ders Notları.  
<http://www.ozanaki.com/home/uploads/teaching/secbst426-robotik-bilim/robot-bilimi-05-robot-aktuatorler.pdf> adresinden erişildi.
- Al Jazari-The Elephant Clock. (t.y.). <https://www.atlasobscura.com/articles/the-robot-clocks-of-12thcentury-turkey> adresinden erişildi.
- Awareness In The Fields of Robotics. (2015). *Self Study History*. 16 Şubat 2020 tarihinde <https://selfstudyhistory.com/2015/10/16/gs-paper-3-awareness-in-the-fields-of-robotics/> adresinden erişildi.
- Bejczy, A. K. ve Paul, R. P. (1981). Simplified Robot Arm Dynamics for Control. *1981 20th IEEE Conference on Decision and Control including the Symposium on Adaptive Processes* içinde (ss. 261-262). 1981 20th IEEE Conference on Decision and Control including the Symposium on Adaptive Processes, sunulmuş bildiri. doi:10.1109/CDC.1981.269524
- Ben-Ari, M. ve Mondada, F. (2018). *Elements of Robotics*. Cham: Springer International Publishing. doi:10.1007/978-3-319-62533-1
- Berardinis. (2000, 1 Mart). Motion Design 101: Gears and belts. *Machine Design*. 17 Şubat 2020 tarihinde <https://www.machinedesign.com/automation-iiot/article/21832368/motion-design-101-gears-and-belts> adresinden erişildi.

- Bernier, C. (t.y.). How Pneumatic End Effectors Work? *Robotiq*. 17 Şubat 2020 tarihinde <https://blog.robotiq.com/bid/65604/How-Pneumatic-End-Effectors-Work> adresinden erişildi.
- Bingül, Z. ve Küçük, S. (2017a). *Robot Kinematiği* (3.). Umuttepe Yayınları.
- Bingül, Z. ve Küçük, S. (2017b). *Robot Dinamiği ve Kontrolü* (2.). Umuttepe Yayınları.
- Boggus, M. (t.y.). Character Animation Forward and Inverse Kinematics (CSE 3541). *SlidePlayer*. 23 Şubat 2020 tarihinde <https://slideplayer.com/slide/12902351/> adresinden erişildi.
- Briot, S. ve Khalil, W. (2015). *Dynamics Modeling of Parallel Robots*. (S. Briot ve W. Khalil, Ed.)Mechanisms and Machine Science. Cham: Springer International Publishing. doi:10.1007/978-3-319-19788-3\_8
- Cao, Y., Lu, K., Li, X. ve Zang, Y. (2011). Accurate Numerical Methods for Computing 2D and 3D Robot Workspace. *International Journal of Advanced Robotic Systems*, 8(6), 1-13. doi:10.5772/45686
- Controlling a KUKA Industrial Robot Using a SIMATIC S7-1500. (t.y.). 22 Şubat 2020 tarihinde <https://docplayer.net/47895646-Controlling-a-kuka-industrial-robot-using-a-simatic-s7-1500.html> adresinden erişildi.
- Craig, J. J. (2004). *Introduction to Robotics: Mechanics and Control* (3.). Upper Saddle River, N.J: Pearson.
- Denavit–Hartenberg Parameters. (2019, 7 Aralık). *Wikipedia*. [https://en.wikipedia.org/w/index.php?title=Denavit%20%80%93Hartenberg\\_parameters&oldid=929693604](https://en.wikipedia.org/w/index.php?title=Denavit%20%80%93Hartenberg_parameters&oldid=929693604) adresinden erişildi.
- Devol, G. C. (1961, 13 Haziran). Programmed Article Transfer. <https://patents.google.com/patent/US2988237/en> adresinden erişildi.
- Dişlitaş, S. (2015). *Endüstriyel Robot Programlama* (1.). Çorum.

- Duran, M. A. ve Ankaralı, A. (2010). Üç Serbestlik Dereceli Puma Tipi Bir Manipülâtörün PID Kontrolü. *Selçuk-Teknik Dergisi*, 9(1), 79-98.
- E-Learning Courses (Turkish)-Servos. (t.y.). 17 Şubat 2020 tarihinde <https://www.mitsubishielectric.com/fa/assist/e-learning/tur.html> adresinden erişildi.
- Eren, O. ve Kaftanoğlu, B. (2001). Altı Serbetlik Dereceli Bir Endüstriyel Robotun Tasarımı İmalatı ve Çalıştırılması. *Makina Tasarım ve İmalat Dergisi*, 4(2), 103-111.
- Fu, K. S., Gonzalez, R. C. ve Lee, C. S. G. (1987). *Robotics: Control, Sensing, Vision, and Intelligence*. McGraw-Hill Book Company.
- Gasparetto, A. ve Scalera, L. (2019). From the Unimate to the Delta Robot: The Early Decades of Industrial Robotics. doi:10.1007/978-3-030-03538-9\_23
- George Devol: A Life Devoted to Invention, and Robots - IEEE Spectrum. (t.y.).*IEEE Spectrum: Technology, Engineering, and Science News*. 18 Şubat 2020 tarihinde <https://spectrum.ieee.org/automaton/robotics/industrial-robots/george-devol-a-life-devoted-to-invention-and-robots> adresinden erişildi.
- Grasp, Sight, and Picking Evolve In Autonomous Robots. (2019, 5 Şubat).*The Robot Report*. 17 Şubat 2020 tarihinde <https://www.therobotreport.com/grasp-sight-picking-evolve-robots/> adresinden erişildi.
- Grippers for Robots. (t.y.).*RobotWorx*. 17 Şubat 2020 tarihinde </articles/grippers-for-robots> adresinden erişildi.
- Hartenberg, R. S. ve Denavit, J. (1955). A Kinematic Notation for Lower-Pair Mechanisms Based on Matrices, 215-221.
- Hasırcı, U. (t.y.). Düzce Üniversitesi Teknoloji Fakültesi Elektrik/Elektronik Mühendisliği Bölümü/EET107 Elektrik-Elektronik Mühendisliğine Giriş (Ders

Notları).

<http://akademik.duzce.edu.tr/Content/Dokumanlar/ugurhasirci/DersNotlari/4651c2ec-12a6-4b8f-9ade-862d2ee84d2c.pdf> adresinden erişildi.

Industrial Robots. (t.y.).*Distrelec*. 17 Şubat 2020 tarihinde

<https://www.distrelec.de/en/industrial-robots/cms/knowhow-industrial-robots> adresinden erişildi.

International Federation of Robotics. (t.y.).

<https://ifr.org/downloads/press2018/IFR%20World%20Robotics%20Presentation%20-%2018%20Sept%202019.pdf> adresinden erişildi.

ISO 8373:2012(en), Robots and robotic devices—Vocabulary. (t.y.). 16 Şubat 2020 tarihinde <https://www.iso.org/obp/ui/#iso:std:iso:8373:ed-2:v1:en> adresinden erişildi.

Joubair, A. (2014). What are Accuracy and Repeatability in Industrial Robots? 17 Şubat 2020 tarihinde <https://blog.robotiq.com/bid/72766/What-are-Accuracy-and-Repeatability-in-Industrial-Robots> adresinden erişildi.

Kahya, E. ve Arın, S. (2015). Kivi Meyvesi Hasadında Robot Kullanımı Üzerine Bir Araştırma. *Süleyman Demirel Üniversitesi Fen Bilimleri Enstitüsü Dergisi*, 18. doi:10.19113/sdufbed.26274

Karanjule, D. B., Bhamare, S. S. ve Rao, T. H. (2017). Effect of Cold Drawing Pass Schedule on Mechanical Properties and Microstructure of ST 52 during Cold Drawing of Seamless Tubes and its Influence on Springback. *Advances in Science, Technology and Engineering Systems Journal*, 2(3), 1202-1210. doi:10.25046/aj0203152



- Kayışlı, K. ve Uğur, M. (2017). 3 Serbestlik Dereceli Bir Robot Kolun Bulanık Mantık ve PID ile Kontrolü. *Gazi Üniversitesi Fen Bilimleri Dergisi Part C: Tasarım ve Teknoloji*, 223-234. doi:10.29109/http-gujsc-gazi-edu-tr.339907
- Koç, S. ve Doğan, C. (2017). Beş Eksenli Bilyalı Vidalı Mil Tahrikli Robot Kolun Tasarımı ve Mach3 Kartıyla Kontrolü. *Batman Üniversitesi Yaşam Bilimleri Dergisi*, 7(1/2), 37-47.
- KUKA smartPAD-2. (t.y.).*KUKA AG*. 17 Şubat 2020 tarihinde <https://www.kuka.com/tr-tr/basın/news/2018/03/smartpad-2-if-award> adresinden erişildi.
- Kumar, V. (2010). 50 Years of Robotics [From the Guest Editors]. *IEEE Robotics Automation Magazine*, 17(3), 8-8. doi:10.1109/MRA.2010.938493
- Micieta, B., Zavodska, L., Rakyta, M. ve Binasova, V. (2015). Sustainable Concept for Green Logistics and Energy Efficiency in Manufacturing. *DAAAM International Scientific Book 2015* içinde (ss. 391-400). doi:10.2507/daaam.scibook.2015.33
- Mitsubishi PA10-7CE robot. (t.y.).*ResearchGate*. 17 Şubat 2020 tarihinde [https://www.researchgate.net/figure/Mitsubishi-PA10-7CE-robot\\_fig1\\_221908922](https://www.researchgate.net/figure/Mitsubishi-PA10-7CE-robot_fig1_221908922) adresinden erişildi.
- Mustafa, A. M. ve Al-Saif, A. (2014). Modeling, Simulation and Control of 2-R Robot. *Global Journals Inc. (USA)*, 14(1), 49-54.
- Öztürk, M. (2014, 6 Haziran). *Antropomorfik Robotların Dinamiği ve Adaptif Kontrol Uygulamaları: Matlab/simulink Modelleme*. (Yüksek Lisans Tezi). <https://polen.itu.edu.tr/handle/11527/14494> adresinden erişildi.
- Pneumatic Cylinders. (t.y.).*Tameson*. 17 Şubat 2020 tarihinde [https://tameson.com/pneumatic-cylinders.html?id\\_country=211](https://tameson.com/pneumatic-cylinders.html?id_country=211) adresinden erişildi.

Robot Teknolojisi Ege Üniversitesi Ege MYO Mekatronik Programı Ders Notları. (t.y.).

[http://tec.ege.edu.tr/dersler/b2\\_robot\\_mimarisi.pdf](http://tec.ege.edu.tr/dersler/b2_robot_mimarisi.pdf) adresinden erişildi.

Samadikhoshkho, Z., Zareinia, K. ve Janabi-Sharifi, F. (2019). A Brief Review on Robotic Grippers Classifications. *2019 IEEE Canadian Conference of Electrical and Computer Engineering (CCECE)*, 1-4. doi:10.1109/CCECE.2019.8861780

Saygılı, Ç. (2006, 1 Aralık). *Scara Tipi Bir Robotun Tasarımı ve Animasyonu*. (Yüksek Lisans Tezi).

<http://acikerisim.selcuk.edu.tr:8080/xmlui/handle/123456789/7333> adresinden erişildi.

Schilling, R. J. (2003). *Fundamentals of Robotics: Analysis and Control* (5.). Prentice Hall of India.

Servo Motor Sürücü Bağlantısı, Entegresi Yapma ve Kontrolü. (t.y.). *Kontrol Kalemi*. 17

Şubat 2020 tarihinde <https://www.kontrolkalemi.com/forum/konu/servo-motor-s%C3%BCr%C3%BCc%C3%BC-ba%C4%9Flant%C4%B1s%C4%B1-entegresi-yapma-ve-kontrol%C3%BC.125909/> adresinden erişildi.

Servo Motor Technology. (t.y.). *Oriental Motor*. 17 Şubat 2020 tarihinde

<https://www.orientalmotor.com/servo-motors/technology/servo-motor-overview.html#> adresinden erişildi.

Siciliano, B., Sciavicco, L., Villani, L. ve Oriolo, G. (2009). *Robotics: Modelling, planning and control*. London: Springer.

Spong, M. W., Hutchinson, S. ve Vidyasagar, M. (2004). *Robot Dynamics and Control* (2.).

Teach Pendants Information. (t.y.). *Engineering360*. 17 Şubat 2020 tarihinde

[https://www.globalspec.com/learnmore/motion\\_controls/operator\\_interfaces/teach\\_pendants](https://www.globalspec.com/learnmore/motion_controls/operator_interfaces/teach_pendants) adresinden erişildi.

- Thomas, P. P. (2016). The Analysis of Coriolis Effect on a Robot Manipulator (ss. 370-375). *International Journal of Innovations in Engineering and Technology (IJJET)*, sunulmuş bildiri.
- Tonbul, T. ve Sarıtaş, M. (2013). Beş Eksenli Bir Edubot Robot Kolunda Ters Kinematik Hesaplamalar ve Yörünge Planlaması. *Gazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi*, 18(1), 145-167.
- Uzuner, S. (2012). *5 Eksenli Manipülâtörün Hareket Planlanması ve Kontrol Programının Oluşturulması*. (Yayımlanmamış yüksek lisans tezi). Marmara Üniversitesi Fen Bilimleri Enstitüsü.
- What is a Servo Motor and How it Works? (t.y.).*RealPars*. 17 Şubat 2020 tarihinde <https://realpars.com/servo-motor/> adresinden erişildi.
- What is an Encoder? (t.y.).*RealPars*. 17 Şubat 2020 tarihinde <https://realpars.com/encoder/> adresinden erişildi.
- Yavuz, E., Alıcı, M. ve Uyar, E. (2015). 3 Serbestlik Dereceli (3R) Bir Robot Manipülâtörünün Kontrolü ve Görüntü İşlemeye Dayalı Nesne Taşınması Control of 3 DOF (3R) Robot Manipulator and Moving Objects Based on Image Processing (ss. 654-659). Otomatik Kontrol Ulusal Toplantısı, sunulmuş bildiri.

## EKLER

### EK-1 Robot kolu sürmede kullanılan Matlab ve Arduino kodları

```
clc;
clear;
clear all;
global motor1;
motor1=serial('COM9','BAUD',9600); % Make sure the baud rate and COM port is
% same as in Arduino IDE
fopen(motor1);

global motor2;
motor2=serial('COM10','BAUD',9600); % Make sure the baud rate and COM port is
% same as in Arduino IDE
fopen(motor2);
L1=523;
L2=565.2;
L3=129;

encoder1rad{:} = [];
encoder2rad{:} = [];
encoder1deg{:} = [];
encoder2deg{:} = [];
eksen1_hiz{:} = [];
eksen2_hiz{:} = [];
eksen1_ivme{:} = [];
eksen2_ivme{:} = [];
Pkonum{:} = [];

Px{:} = [];
Py{:} = [];
Pz{:} = [];
Px_hiz{:} = [];
Py_hiz{:} = [];
Pz_hiz{:} = [];
Px_ivme{:} = [];
Py_ivme{:} = [];
Pz_ivme{:} = [];

gec1='';
gec2='';
gec1_son=0;
gec2_son=0;
i=1;
val=1;
tekrar=1;
tekrar_sayac=0;
tekrar_gec=0;
val1=110100000;
val2=110100000;
bitti1=0;
bitti2=0;
while(val==1)
    if(tekrar==1)
        if(tekrar_sayac==0)
            kin= input('Düz kin mi ters kin mi (1/0)\n');
            end
        end
```

Şekil EK-1.1. Robot kolu sürmede kullanılan Matlab kodları-1

```

if (kin==1)
    if(tekrar_sayac==0)
        mot= input('Hangi eksen veya eksenler hareket edicek
(11/10/01/00)\n');
        end

        if (mot==10)
            %bitti2=1;
            val1= input('1. eksen Hareket mesajını giriniz\n');
            fprintf(motor1,'%ld\n',val1);
            fprintf(motor2,'%ld\n',val2);
            fscanf(motor1,'%s\n')
            hiz_S1=(mod(abs(val1),1000000)-mod(abs(val1),1000))/1000000;
        end
        if(mot==01)
            % bitti1=1;
            val2= input('2. eksen Hareket mesajını giriniz\n');
            fprintf(motor2,'%ld\n',val2);
            fprintf(motor1,'%ld\n',val1);
            fscanf(motor2,'%s\n')
            hiz_S2=(mod(abs(val2),1000000)-mod(abs(val2),1000))/1000000;
        end
        if(mot==11)

            if(tekrar_sayac==0)
                tekrar= input('tekrar modu acilsin mi (1/0)\n');
                end
            if(tekrar==0)
                val1= input('1. eksen Hareket mesajını giriniz\n');
                val2= input('2. eksen Hareket mesajını giriniz örnekleme
hizi 1.eksen ile aynı olmalı \n');
                hiz_S1=(mod(abs(val1),1000000)-mod(abs(val1),1000))/1000000;
                hiz_S2=hiz_S1;
                fprintf(motor1,'%ld\n',val1);
                fscanf(motor1,'%s\n');
                fprintf(motor2,'%ld\n',val2);
                fscanf(motor2,'%s\n');
            end

            if(tekrar==1)
                if(tekrar_sayac==0)
                    tekrar_sayac= input('tekrar sayisini giriniz\n');
                    tekrar_sayac=tekrar_sayac*2;
                    tekrar_gec=tekrar_sayac;
                    val1= input('1. eksen Hareket mesajını giriniz\n');
                    val2= input('2. eksen Hareket mesajını giriniz örnekleme
hizi 1.eksen ile aynı olmalı \n');
                    val1_g=val1;
                    val2_g=val2;
                    hiz_S1=(mod(abs(val1),1000000)-
mod(abs(val1),1000))/1000000;
                    hiz_S2=hiz_S1;
                end
            end
        end
    end
end

```

Şekil EK-1.2. Robot kolunu sürmede kullanılan Matlab kodları-2

```

if (mod(tekrar_gec-tekrar_sayac,2)==0)
    fprintf(motor1,'%ld\n',val1);
    fscanf(motor1,'%s\n');
    fprintf(motor2,'%ld\n',val2);
    fscanf(motor2,'%s\n');
else
    fprintf(motor1,'%ld\n',110050000);
    fscanf(motor1,'%s\n');
    fprintf(motor2,'%ld\n',110050000);
    fscanf(motor2,'%s\n');

end
if(mot==00)
    %gripper açma kapama
end
end
while (or(~bitti1,~bitti2))
    if (bitti1==0)
        gec1=fscanf(motor1,'%s\n');
        if(strcmp(gec1,'bitti')==1)
            bitti1=1;
        else
            gec1_son=gec1;
        end
        % bitti1=strcmp(gec1,'bitti');
    end
    if (bitti2==0)
        gec2=fscanf(motor2,'%s\n');
        if(strcmp(gec2,'bitti')==1)
            bitti2=1;
        else
            gec2_son=gec2;
        end
        %bitti2=strcmp(gec2,'bitti');
    end
    if(strcmp(gec1,'bitti')==1)
        encoder1rad{i}=((str2double(gec1_son)/30000)*pi*2);
        encoder1deg{i}=((str2double(gec1_son)/30000)*360);
    else
        encoder1rad{i}=((str2double(gec1)/30000)*pi*2);
        encoder1deg{i}=((str2double(gec1)/30000)*360);
    end
    if(strcmp(gec2,'bitti')==1)
        encoder2rad{i}=((str2double(gec2_son)/30000)*pi*2);
        encoder2deg{i}=((str2double(gec2_son)/30000)*360);
    else
        encoder2rad{i}=((str2double(gec2)/30000)*pi*2);
        encoder2deg{i}=((str2double(gec2)/30000)*360);
    end
end

```

Şekil EK-1.3. Robot kolunu sürmede kullanılan Matlab kodları-3

```

        i=i+1;
        if (mot==11)
            pause(hiz_S1);
        end
        if (mot==01)
            pause(hiz_S2);
        end
        if (mot==10)
            pause(hiz_S1);
        end
    end
end
if (kin==0)
    Pix= input('Px degerini giriniz\n');
    Piy= input('Py degerini giriniz\n');
    Piz= input('Pz degerini giriniz\n');
    t11=round(-log(-((Pix^2 + Piy^2 - 16641)^(1/2) + 129i)/(Pix -
Piy*1i))*1i,4)
    t12=round(-log(((Pix^2 + Piy^2 - 16641)^(1/2) - 129i)/(Pix -
Piy*1i))*1i,4)
    a=(((Pix^2)+(Piy^2)-16641)/(3.1945104*10^5))^(1/2);
    b=(((1-a^2)^(1/2)));
    t21=atan(b/a)*(180/pi)
    t22=-atan(b/a)*(180/pi)
end

if (tekrar_sayac~=0)
    tekrar_sayac=tekrar_sayac-1;
    pause(1);
else
    tekrar=0;
end
if (tekrar==0)
    val= input('devam etmek istiyor musunuz (1/0) \n');
    if (val==1)
        tekrar=1;
    end
else
    val=1;
end
end
end
bitti1=0;bitti2=0;
end
val= input('sıfır konumuna gidilsin mi (1/0) \n');
if (val==1)
    fprintf(motor1,'%ld\n',100000000);
    fprintf(motor2,'%ld\n',100000000);
end

```

Şekil EK-1.4. Robot kolunu sürmede kullanılan Matlab kodları-4

```

if (kin==1)
    encoder1rad=cell2mat(encoder1rad);
    encoder2rad=cell2mat(encoder2rad);
    encoder1deg=cell2mat(encoder1deg);
    encoder2deg=cell2mat(encoder2deg);
    % Px=-
    sin(encoder1rad(length(encoder1rad)))*L3+cos(encoder1rad(length(encoder1rad)))*c
    os(encoder2rad(length(encoder2rad)))*L2;
    %
    Py=cos(encoder1rad(length(encoder1rad)))*L3+sin(encoder1rad(length(encoder1rad))
    )*cos(encoder2rad(length(encoder2rad)))*L2;
    % Pz=-sin(encoder2rad(length(encoder2rad)))*L2+L1;

    t1 = 0 : hiz_S1 : ((length(encoder1deg)-1)*hiz_S1);
    t2 = 0 : hiz_S2 : ((length(encoder2deg)-1)*hiz_S2);
    t3 = hiz_S1 : hiz_S1 : ((length(encoder1deg)-1)*hiz_S1);
    t4 = hiz_S2 : hiz_S2 : ((length(encoder2deg)-1)*hiz_S2);
    t5 = hiz_S1 : hiz_S1 : ((length(encoder1deg)-2)*hiz_S1);
    t6 = hiz_S2 : hiz_S2 : ((length(encoder2deg)-2)*hiz_S2);
    %Açısal konum
    fig= input('grafik istiyor musunuz (1/0) \n');
    if(fig==1)
        figure
        plot(t1,encoder1deg,'--b',t2,encoder2deg,'-.r')
        grid on
        xlabel ('Zaman (s)','fontsize',14)
        ylabel('eksen 1(t) ve eksen 2(t)','fontsize',14)
        title('eksen 1(t) ve eksen 2(t)
Konum(Aç1)/zaman(s) grafiği','fontsize',14)
        legend('eksen1','eksen2');
        hold off;

    end
    %/figure
    %plot(t1,encoder1rad,'--b',t2,encoder2rad,'-.r')
    %grid on
    %xlabel ('Zaman (s)','fontsize',14)
    %ylabel('eksen 1(t) ve eksen 2(t)','fontsize',14)
    %title('eksen 1(t) ve eksen 2(t) Radyan(pi)/zaman(s) grafiği','fontsize',14)
    %legend('eksen1','eksen2');
    %hold off;
    j=i-1;
    while(i-1)

        Px{i-1}=-sin(encoder1rad(i-1))*L3+cos(encoder1rad(i-
1))*cos(encoder2rad(i-1))*L2;
        Py{i-1}=cos(encoder1rad(i-1))*L3+sin(encoder1rad(i-
1))*cos(encoder2rad(i-1))*L2;
        Pz{i-1}=-sin(encoder2rad(i-1))*L2+L1;

        i=i-1;
    end
end

```

Şekil EK-1.5. Robot kolunu sürmede kullanılan Matlab kodları-5



```

Px=cell2mat(Px);
Py=cell2mat(Py);
Pz=cell2mat(Pz);
i=j;
while(i-1)
Pkonum{i-1}=(Px(i-1)^2+Py(i-1)^2+Pz(i-1)^2)^(1/2);
    i=i-1;
end
Pkonum=cell2mat(Pkonum);
i=j;
while(i-1)
    if(i~=1)

        eksen1_hiz{i-1}=(encoder1deg(i)-encoder1deg(i-1))/hiz_S1;
        eksen2_hiz{i-1}=(encoder2deg(i)-encoder2deg(i-1))/hiz_S2;
        Px_hiz{i-1}=(Px(i)-Px(i-1))/hiz_S1;
        Py_hiz{i-1}=(Py(i)-Py(i-1))/hiz_S1;
        Pz_hiz{i-1}=(Pz(i)-Pz(i-1))/hiz_S1;

    end
    i=i-1;
end
i=j;

Px_hiz=cell2mat(Px_hiz);
Py_hiz=cell2mat(Py_hiz);
Pz_hiz=cell2mat(Pz_hiz);
eksen1_hiz=cell2mat(eksen1_hiz);
eksen2_hiz=cell2mat(eksen2_hiz);
while(i-1)
    if(i~=2)

        eksen1_ivme{i-2}=(eksen1_hiz(i-1)-eksen1_hiz(i-2))/(hiz_S1);
        eksen2_ivme{i-2}=(eksen2_hiz(i-1)-eksen2_hiz(i-2))/(hiz_S1);

        Px_ivme{i-2}=(Px_hiz(i-1)-Px_hiz(i-2))/hiz_S1;
        Py_ivme{i-2}=(Py_hiz(i-1)-Py_hiz(i-2))/hiz_S1;
        Pz_ivme{i-2}=(Pz_hiz(i-1)-Pz_hiz(i-2))/hiz_S1;
    end
    i=i-1;
end
i=j;

Px_ivme=cell2mat(Px_ivme);
Py_ivme=cell2mat(Py_ivme);
Pz_ivme=cell2mat(Pz_ivme);

eksen1_ivme=cell2mat(eksen1_ivme);
eksen2_ivme=cell2mat(eksen2_ivme);

```

Şekil EK-1.6. Robot kolunu sürmede kullanılan Matlab kodları-6

```

if(fig==1)
    %Pkonom
    figure
    plot3(Px,Py,Pz)
    grid on
    xlabel ('Px','fontsize',14)
    ylabel('Py','fontsize',14)
    zlabel('Pz','fontsize',14)
    title('Gripper Konum(mm)/zaman(s) grafiđi','fontsize',14)
    legend('PKonum');
    hold off;
    %Açısal hiz
    figure
    plot(t3,eksen1_hiz,'--b',t4,eksen2_hiz,'-.r')
    grid on
    xlabel ('Zaman (s)','fontsize',14)
    ylabel('Hiz(derece/s)','fontsize',14)
    title('eksen 1(t) ve eksen 2(t)
Hiz(Derece/s)/zaman(s) grafiđi','fontsize',14)
    legend('eksen1 hizi','eksen2 hizi');
    hold off;
    %Açısal ivme
    figure
    plot(t5,eksen1_ivme,'--b',t6,eksen2_ivme,'-.r')
    grid on
    xlabel ('Zaman (s)','fontsize',14)
    ylabel('İvme(derece/s^2)','fontsize',14)
    title('eksen 1(t) ve eksen 2(t)
İvme(Derece/s^2)/zaman(s) grafiđi','fontsize',14)
    legend('eksen1 ivme','eksen2 ivme');
    hold off;
    %kinematik konum
    figure
    plot(t1,Px,'--b')
    grid on
    hold on
    plot(t2,Py,'-.r')
    plot(t2,Pz,'.g')
    hold off
    xlabel ('Zaman (s)','fontsize',14)
    ylabel('Konum','fontsize',14)
    title('Px Py ve Pz Konum(mm)/zaman(s) grafiđi','fontsize',14)
    legend('Px','Py','Pz');
    %kinematik hiz
    figure
    plot(t3,Px_hiz,'--b')
    grid on
    hold on

```

Şekil EK-1.7. Robot kolunu sürmede kullanılan Matlab kodları-7

```

        plot(t3,Py_hiz,'-.r')
        plot(t3,Pz_hiz,'.g')
        xlabel ('Zaman (s)','fontsize',14)
        ylabel('Hiz','fontsize',14)
        title('Px Py ve Pz Hiz(mm/s)/zaman(s)grafiği','fontsize',14)
        hold off
        legend('Px','Py','Pz');
        %kinematik ivme
        figure
        plot(t5,Px_ivme,'--b')
        grid on
        hold on
        plot(t5,Py_ivme,'-.r')
        plot(t5,Pz_ivme,'.g')
        xlabel ('Zaman (s)','fontsize',14)
        ylabel('İvme','fontsize',14)
        title('Px Py ve Pz İvme(mm/s^2)/zaman(s)grafiği','fontsize',14)
        hold off
        legend('Px','Py','Pz');
    end
end
fprintf('P(x,y,z)=Deg(eksen1,eksen2) \n');
fprintf('P(%f,%f,%f)=Deg(%f,%f)
\n',Px(i),Py(i),Pz(i),encoder1deg(i),encoder2deg(i));

fclose(motor1);
delete(motor1);
fclose(motor2);
delete(motor2);
%          newobjs = instrfind;delete(newobjs)

```

Şekil EK-1.8. Robot kolunu sürmede kullanılan Matlab kodları-8

```

#include <PID_v1.h>
#define MotEnable 6 // Motor Enable pin Runs on PWM signal
#define MotFwd 5 // Motor Forward pin normal 4
#define MotRev 7 // Motor Reverse pin
String readString; //This while store the user input data
long User_Input0 = 0; // This while convert input string into integer
long User_Inputabs = 0;
int User_Input = 0; // This while convert input string into integer
int User_InputG = 0;
int seri_hizMS = 500;
int serial_basla = 0; // This while convert input string into integer
uint32_t startMS = 0;
uint32_t currentMS = 0;
uint32_t lastMS = 0;
uint32_t difMS = 0;
int encoderPin1 = 2; // Encoder Output 'A' must connected with intreput pin of
arduino.
int encoderPin2 = 3; // Encoder Otput 'B' must connected with intreput pin of
arduino.
volatile int lastEncoded = 0; // Here updated value of encoder store.
volatile long encoderValue = 0; // Raw encoder value
int PPR = 30000; // Encoder Pulse per revolution.
int angle = 360; // Maximum degree of motion.
int REV = 0; // Set point REQUIRED ENCODER VALUE
int lastMSB = 0;
int lastLSB = 0;
int gec = 1024;
int gec1 = 0;
double kp = 0.0473, ki = 0.00011 , kd = 0.0001; // tam deđerler
// double kp = 0.0473,ki = 0.00011 , kd = 0.0001; // modify for optimal
performance
double input = 0, output = 0, setpoint = 0;
PID myPID(&input, &output, &setpoint, kp, ki, kd, DIRECT);
void setup() {
  pinMode(MotEnable, OUTPUT);
  pinMode(MotFwd, OUTPUT);
  pinMode(MotRev, OUTPUT);
  Serial.begin(9600); //initialize serial communication
  pinMode(encoderPin1, INPUT_PULLUP);
  pinMode(encoderPin2, INPUT_PULLUP);
  digitalWrite(encoderPin1, HIGH); // turn pullup resistor on
  digitalWrite(encoderPin2, HIGH); // turn pullup resistor on
  //call updateEncoder() when any high/low changed seen
  //on interrupt 0 (pin 2), or interrupt 1 (pin 3)
  attachInterrupt(0, updateEncoder, CHANGE);
  attachInterrupt(1, updateEncoder, CHANGE);
}

```

Şekil EK-1.9. Robot kolunu sürmede kullanılan 1. Motora ait Arduino PID kodları-1

```

TCCR1B = TCCR1B & 0b1111000 | 1; // set 62KHz PWM to prevent motor noise
myPID.SetMode(AUTOMATIC); //set PID in Auto mode
myPID.SetSampleTime(1); // refresh rate of PID controller
myPID.SetOutputLimits(-125, 125); // this is the MAX PWM value to move motor,
here change in value reflect change in speed of motor.
}
void loop() {
  currentMS = millis();
  while (Serial.available()) { // Check if the serial data is available.
    delay(3); // a small delay
    char c = Serial.read(); // storing input data
    readString += c; // accumulate each of the characters in readString
  }
  if (readString.length() > 0) { // Verify that the variable contains
information
    Serial.println(readString.toInt()); // printing the input data in integer
form
    User_Input0 = readString.toInt(); // here input data is store in integer
form
    User_Inputabs = abs(User_Input0); gec1 = 1; serial_basla = 1;
    startMS = millis();
  }
  //110500360//11000

  if (User_Inputabs / 100000000 == 1) {
    User_Input = User_Input0 % 1000;
    if (User_Inputabs / 10000000 == 11) {
      seri_hizMS = (User_Inputabs / 1000) % 10000;
    }
  }
  if (serial_basla == 1) {
    if (currentMS % seri_hizMS == 0 & seri_hizMS != 0) { //seri iletişim
gönderi hızı
      if (gec != currentMS / seri_hizMS) {
        gec = currentMS / seri_hizMS;
        // Serial.print("value\t"); //encoder
        Serial.println(encoderValue);
        // Serial.print("Rev\t"); //Rev
        // Serial.println(REV); // printing REV value
        // Serial.println(User_Input);
        // Serial.println(currentMS);
        lastMS = millis();
      }
    }
  }
}
}
}

```

Şekil EK-1.10. Robot kolunu sürmede kullanılan 1. Motora ait Arduino PID kodları-2

```

else if ((lastMS + (2 * (seri_hizMS)) == currentMS) & gec1 == 1 ) {
  if (currentMS % seri_hizMS == 0 & seri_hizMS != 0) {
    if (gec != currentMS / seri_hizMS) {
      gec = currentMS / seri_hizMS;
      Serial.println(encoderValue);
    }
  }
}
else if (gec1 == 1 )
{ delay(200);
  Serial.println(encoderValue);
  Serial.println("bitti");
  gec1 = 0;
}
REV = map (User_Input, 0, 360, 0, 30000); // mapping degree into pulse
setpoint = REV; //PID while work to achive this value consider as SET value
input = encoderValue ; // data from encoder consider as a Process value
myPID.Compute(); // calculate new output
pwmOut(output);
}
void pwmOut(int out) {
  if (out > 0) { // if REV > encoderValue motor move in forward direction.
    analogWrite(MotEnable, out); // Enabling motor enable pin to reach the
    desire angle
    forward(); // calling motor to move forward
  }
  else if (out < 0) {
    analogWrite(MotEnable, abs(out)); // if REV < encoderValue motor move in
    forward direction.
    reverse(); // calling motor to move reverse
  }
  else if (out == 0) {
    analogWrite(MotEnable, abs(out)); // if REV < encoderValue motor move in
    forward direction.
    reverse(); // calling motor to move reverse
    serial_basla = 0;
  }
}

```

Şekil EK-1.11. Robot kolunu sürmede kullanılan 1. Motora ait Arduino PID kodları-3

```

readString = ""; // Cleaning User input, ready for new Input
}
void updateEncoder() {
  int MSB = digitalRead(encoderPin1); // MSB = most significant bit
  int LSB = digitalRead(encoderPin2); // LSB = least significant bit
  int encoded = (MSB << 1) | LSB; // converting the 2 pin value to single number
  int sum = (lastEncoded << 2) | encoded; //adding it to the previous encoded
  value
  if (sum == 0b1101 || sum == 0b0100 || sum == 0b0010 || sum == 0b1011)
  encoderValue ++;
  if (sum == 0b1110 || sum == 0b0111 || sum == 0b0001 || sum == 0b1000)
  encoderValue --;
  lastEncoded = encoded; // store this value for next time
}
void forward () {
  digitalWrite(MotFwd, HIGH);
  digitalWrite(MotRev, LOW);
}
void reverse () {
  digitalWrite(MotFwd, HIGH);
  digitalWrite(MotRev, HIGH);
}
void finish () {
}
}

```

Şekil EK-1.12. Robot kolunu sürmede kullanılan 1. Motora ait Arduino PID kodları-4

```

#include <PID_v1.h>
#define MotEnable 6 // Motor Enable pin Runs on PWM signal
#define MotFwd 5 // Motor Forward pin normal 4
#define MotRev 7 // Motor Reverse pin
String readString; // This while store the user input data
long User_Input0 = 0; // This while convert input string into integer
long User_Inputabs = 0;
int User_Input = 0; // This while convert input string into integer
int seri_hizMS = 500;
int serial_basla = 0; // This while convert input string into integer
uint32_t startMS = 0;
uint32_t currentMS = 0;
uint32_t lastMS = 0;
uint32_t difMS = 0;
int encoderPin1 = 2; // Encoder Output 'A' must connected with intreput pin of
arduino.
int encoderPin2 = 3; // Encoder Otput 'B' must connected with intreput pin of
arduino.
volatile int lastEncoded = 0; // Here updated value of encoder store.
volatile long encoderValue = 667; // Raw encoder value
int PPR = 30000; // Encoder Pulse per revolution.
int angle = 360; // Maximum degree of motion.
int REV = 0; // Set point REQUIRED ENCODER VALUE
int lastMSB = 0;
int lastLSB = 0;
int gec = 1024;
int gec1 = 0;
double kp = 0.0253, ki = 0.0001 , kd = 0.00001; // modify for optimal
performance
double input = 0, output = 0, setpoint = 0;
PID myPID(&input, &output, &setpoint, kp, ki, kd, DIRECT);
void setup() {
  pinMode(MotEnable, OUTPUT);
  pinMode(MotFwd, OUTPUT);
  pinMode(MotRev, OUTPUT);
  Serial.begin(9600); // initialize serial comunication
  pinMode(encoderPin1, INPUT_PULLUP);
  pinMode(encoderPin2, INPUT_PULLUP);
  digitalWrite(encoderPin1, HIGH); // turn pullup resistor on
  digitalWrite(encoderPin2, HIGH); // turn pullup resistor on
  //call updateEncoder() when any high/low changed seen
  //on interrupt 0 (pin 2), or interrupt 1 (pin 3)
  attachInterrupt(0, updateEncoder, CHANGE);
  attachInterrupt(1, updateEncoder, CHANGE);
}

```

Şekil EK-1.13. Robot kolunu sürmede kullanılan 2. Motora ait Arduino PID kodları-1



```

TCCR1B = TCCR1B & 0b1111000 | 1; // set 62KHz PWM to prevent motor noise
myPID.SetMode(AUTOMATIC); // set PID in Auto mode
myPID.SetSampleTime(1); // refresh rate of PID controller
myPID.SetOutputlimits(-125, 125); // this is the MAX PWM value to move motor,
here change in value reflect change in speed of motor.
}
void loop() {
  currentMS = millis();
  while (Serial.available()) { // Check if the serial data is available.
    delay(3); // a small delay
    char c = Serial.read(); // storing input data
    readString += c; // accumulate each of the characters in readString
  }
  if (readString.length() > 0) { // Verify that the variable contains
information
    Serial.println(readString.toInt()); // printing the input data in integer
form
    User_Input0 = readString.toInt(); // here input data is store in integer
form
    User_Inputabs = abs(User_Input0);gec1=1;serial_basla = 1;
  }
  //110500360//11000
  if (User_Inputabs / 100000000 == 1) {
    User_Input = User_Input0 % 1000;
    if (User_Inputabs / 100000000 == 11) {
      seri_hizMS = (User_Inputabs / 1000) % 10000;
    }
  }
  REV = map (User_Input, 0, 360, 0, 30000); // mapping degree into pulse

  setpoint = REV; // PID while work to achive this value
consider as SET value
  input = encoderValue ; // data from encoder consider as a Process
value
  myPID.Compute(); // calculate new output
  pwmOut(output);
}

```

**Şekil EK-1.14.** Robot kolunu sürmede kullanılan 2. Motora ait Arduino PID kodları-2

```

if (serial_basla == 1) {
  if (currentMS % seri_hizMS == 0 & seri_hizMS != 0) { // seri iletişim
gönderi hızı
    if (gec != currentMS / seri_hizMS) {
      gec = currentMS / seri_hizMS;
      // Serial.print("value\t"); //encoder
      Serial.println(encoderValue);
      // Serial.print("Rev\t"); // Rev
      // Serial.println(REV); // printing REV value
      // Serial.println(User_Input);
      // Serial.println(currentMS);
      lastMS = millis();
    }
  }
}
else if ((lastMS + (2 * (seri_hizMS)) == currentMS) & gec1 == 1 ) {
  if (currentMS % seri_hizMS == 0 & seri_hizMS != 0) {
    if (gec != currentMS / seri_hizMS) {
      gec = currentMS / seri_hizMS;
      Serial.println(encoderValue);
      //Serial.println("bitti");
    }
  }
}
//pid ayarında kapalı
else if (gec1 == 1 )
{ delay(200);
  Serial.println(encoderValue);
  Serial.println("bitti");
  gec1 = 0;
}
}
void pwmOut(int out) {
  if (out > 0) { // if REV > encoderValue motor move in forward direction.
    analogWrite(MotEnable, out); // Enabling motor enable pin to reach the
desire angle
    forward(); // calling motor to move forward
  }
  else if (out < 0) {
    analogWrite(MotEnable, abs(out)); // if REV < encoderValue motor move in
forward direction.
    reverse(); // calling motor to move reverse
  }
  else if (out == 0) {
    analogWrite(MotEnable, abs(out)); // if REV < encoderValue motor move in
forward direction.
  }
}

```

Şekil EK-1.15. Robot kolunu sürmede kullanılan 2. Motora ait Arduino PID kodları-3

```

reverse(); // calling motor to move reverse
  serial_basla = 0; // pid ayarında kapalı
}
readString = ""; // Cleaning User input, ready for new Input
}
void updateEncoder() {
  int MSB = digitalRead(encoderPin1); // MSB = most significant bit
  int LSB = digitalRead(encoderPin2); // LSB = least significant bit
  int encoded = (MSB << 1) | LSB; //converting the 2 pin value to single number
  int sum = (lastEncoded << 2) | encoded; //adding it to the previous encoded
  value
  if (sum == 0b1101 || sum == 0b0100 || sum == 0b0010 || sum == 0b1011)
  encoderValue ++;
  if (sum == 0b1110 || sum == 0b0111 || sum == 0b0001 || sum == 0b1000)
  encoderValue --;
  lastEncoded = encoded; //store this value for next time
}
void forward () {
  digitalWrite(MotFwd, HIGH);
  digitalWrite(MotRev, LOW);
  // digitalWrite(MotFwd, LOW);
}
void reverse () {
  digitalWrite(MotFwd, HIGH);
  digitalWrite(MotRev, HIGH);
  // digitalWrite(MotFwd, LOW);
}
void finish () {
  // digitalWrite(MotFwd, LOW);
  //digitalWrite(MotRev, LOW);
}
}

```

Şekil EK-1.16. Robot kolunu sürmede kullanılan 2. Motora ait Arduino PID kodları-4

## EK-2 Robot kolu dinamiğinde oluşturulan Lagrange Euler denklemleri

### İlk eklem için değişkenler

$$Deltal := \begin{bmatrix} -0.10994 \\ -101.92403 \\ -12.72789 \\ 1 \end{bmatrix}$$

$$hl := \begin{bmatrix} -0.10994 \cos(\theta_1) + 101.92403 \sin(\theta_1) \\ -0.10994 \sin(\theta_1) - 101.92403 \cos(\theta_1) \\ 510.27211 \\ 1 \end{bmatrix}$$

$$bl := \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

$$I\_I := \begin{bmatrix} \left[ -27625.48148 \sin(\theta_1) \cos(\theta_1) + 1.906044694 \cdot 10^9 \cos(\theta_1)^2 + 3.191640261 \cdot 10^8, \right. \\ \left. 1.906044694 \cdot 10^9 \sin(\theta_1) \cos(\theta_1) + 27625.48148 \cos(\theta_1)^2 - 13812.74074, \right. \\ \left. -3.230172558 \cdot 10^5 \cos(\theta_1) + 3.584281234 \cdot 10^8 \sin(\theta_1) \right], \\ \left[ 1.906044694 \cdot 10^9 \sin(\theta_1) \cos(\theta_1) + 27625.48148 \cos(\theta_1)^2 - 13812.74074, \right. \\ \left. 27625.48148 \sin(\theta_1) \cos(\theta_1) - 1.906044694 \cdot 10^9 \cos(\theta_1)^2 + 2.225208720 \cdot 10^9, \right. \\ \left. -3.230172558 \cdot 10^5 \sin(\theta_1) - 3.584281234 \cdot 10^8 \cos(\theta_1) \right], \\ \left[ -3.230172558 \cdot 10^5 \cos(\theta_1) + 3.584281234 \cdot 10^8 \sin(\theta_1), -3.230172558 \cdot 10^5 \sin(\theta_1) \right. \\ \left. - 3.584281234 \cdot 10^8 \cos(\theta_1), 2.050572752 \cdot 10^9 \right] \end{bmatrix}$$

$$J_1 := \begin{bmatrix} 0.10994 \sin(\theta_1) + 101.92403 \cos(\theta_1) & 0 & 0 \\ -0.10994 \cos(\theta_1) + 101.92403 \sin(\theta_1) & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

$$A_1 := \begin{bmatrix} 0.10994 \sin(\theta_1) + 101.92403 \cos(\theta_1) & 0 & 0 \\ -0.10994 \cos(\theta_1) + 101.92403 \sin(\theta_1) & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$B_1 := \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

**İkinci eklem için değişkenler**

$$Deltah2 := \begin{bmatrix} -231.86858 \\ 0 \\ 0.24344 \\ 1 \end{bmatrix}$$

$$h2 := \begin{bmatrix} 333.33142 \cos(\theta_1) \cos(\theta_2) - 0.24344 \sin(\theta_1) \\ 333.33142 \sin(\theta_1) \cos(\theta_2) + 0.24344 \cos(\theta_1) \\ -333.33142 \sin(\theta_2) + 523. \\ 1. \end{bmatrix}$$

$$b2 := \begin{bmatrix} -\sin(\theta_1) \\ \cos(\theta_1) \\ 0 \end{bmatrix}$$

$$\begin{aligned}
I_2 := & \left[ \left[ -4.623490926 \cdot 10^8 \cos(\theta_1)^2 \cos(\theta_2)^2 - 8.697208930 \cdot 10^5 \sin(\theta_1) \cos(\theta_2) \cos(\theta_1) \right. \right. \\
& + 1.907610 \cdot 10^5 \cos(\theta_1)^2 + 4.646513249 \cdot 10^8, \\
& -4.623490926 \cdot 10^8 \sin(\theta_1) \cos(\theta_2)^2 \cos(\theta_1) + 8.697208930 \cdot 10^5 \cos(\theta_1)^2 \cos(\theta_2) \\
& - 4.348604465 \cdot 10^5 \cos(\theta_2) + 1.907610 \cdot 10^5 \sin(\theta_1) \cos(\theta_1), \\
& \left. \sin(\theta_2) \left( 4.623490926 \cdot 10^8 \cos(\theta_1) \cos(\theta_2) + 4.348604465 \cdot 10^5 \sin(\theta_1) \right) \right], \\
& \left[ -4.623490926 \cdot 10^8 \sin(\theta_1) \cos(\theta_2)^2 \cos(\theta_1) + 8.697208930 \cdot 10^5 \cos(\theta_1)^2 \cos(\theta_2) \right. \\
& - 4.348604465 \cdot 10^5 \cos(\theta_2) + 1.907610 \cdot 10^5 \sin(\theta_1) \cos(\theta_1), \\
& 4.623490926 \cdot 10^8 \cos(\theta_1)^2 \cos(\theta_2)^2 - 4.623490926 \cdot 10^8 \cos(\theta_2)^2 \\
& + 8.697208930 \cdot 10^5 \sin(\theta_1) \cos(\theta_2) \cos(\theta_1) - 1.907610 \cdot 10^5 \cos(\theta_1)^2 \\
& + 4.648420859 \cdot 10^8, \sin(\theta_2) \left( 4.623490926 \cdot 10^8 \sin(\theta_1) \cos(\theta_2) \right. \\
& \left. \left. - 4.348604465 \cdot 10^5 \cos(\theta_1) \right) \right], \\
& \left[ \sin(\theta_2) \left( 4.623490926 \cdot 10^8 \cos(\theta_1) \cos(\theta_2) + 4.348604465 \cdot 10^5 \sin(\theta_1) \right), \right. \\
& \left. \sin(\theta_2) \left( 4.623490926 \cdot 10^8 \sin(\theta_1) \cos(\theta_2) - 4.348604465 \cdot 10^5 \cos(\theta_1) \right), \right. \\
& \left. 4.623490926 \cdot 10^8 \cos(\theta_2)^2 + 2.492993299 \cdot 10^6 \right]
\end{aligned}$$

$$\begin{aligned}
J_2 := & \left[ \left[ -333.33142 \sin(\theta_1) \cos(\theta_2) - 0.24344 \cos(\theta_1), -333.33142 \cos(\theta_1) \sin(\theta_2), \right. \right. \\
& \left. \left. -333.33142 \cos(\theta_1) \sin(\theta_2) \right], \right. \\
& \left[ 333.33142 \cos(\theta_1) \cos(\theta_2) - 0.24344 \sin(\theta_1), -333.33142 \sin(\theta_1) \sin(\theta_2), \right. \\
& \left. -333.33142 \sin(\theta_1) \sin(\theta_2) \right], \\
& \left[ 0, -333.33142 \cos(\theta_2), -333.33142 \cos(\theta_2) \right], \\
& \left[ 0, -\sin(\theta_1), 0 \right], \\
& \left[ 0, \cos(\theta_1), 0 \right], \\
& \left[ 1, 0, 0 \right]
\end{aligned}$$

$$\begin{aligned}
A_2 := & \left[ \left[ -333.33142 \sin(\theta_1) \cos(\theta_2) - 0.24344 \cos(\theta_1), -333.33142 \cos(\theta_1) \sin(\theta_2), \right. \right. \\
& \left. \left. -333.33142 \cos(\theta_1) \sin(\theta_2) \right], \right. \\
& \left[ 333.33142 \cos(\theta_1) \cos(\theta_2) - 0.24344 \sin(\theta_1), -333.33142 \sin(\theta_1) \sin(\theta_2), \right. \\
& \left. -333.33142 \sin(\theta_1) \sin(\theta_2) \right], \\
& \left[ 0, -333.33142 \cos(\theta_2), -333.33142 \cos(\theta_2) \right]
\end{aligned}$$

$$B_2 := \begin{bmatrix} 0 & -\sin(\theta_1) & 0 \\ 0 & \cos(\theta_1) & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

Üçüncü eklem için değişkenler

$$Delta h3 := \begin{bmatrix} 4.14070 \\ 58.96571 \\ -48.81630 \\ 1 \end{bmatrix}$$

$$b3 := \begin{bmatrix} -\sin(\theta_1) \\ \cos(\theta_1) \\ 0 \end{bmatrix}$$

$$h3 := \begin{bmatrix} [4.14070 \cos(\theta_1) \cos(\theta_2)^2 + 4.14070 \cos(\theta_1) \sin(\theta_2)^2 - 80.18370 \sin(\theta_1) \\ + 565.2 \cos(\theta_1) \cos(\theta_2)] \\ [4.14070 \sin(\theta_1) \cos(\theta_2)^2 + 4.14070 \sin(\theta_1) \sin(\theta_2)^2 + 80.18370 \cos(\theta_1) \\ + 565.2 \sin(\theta_1) \cos(\theta_2)] \\ [523. - 58.96571 \sin(\theta_2)^2 - 58.96571 \cos(\theta_2)^2 - 565.2 \sin(\theta_2)] \\ [1.] \end{bmatrix}$$

$$I_3 := \begin{bmatrix} [3.548462040 \cdot 10^7 + 6.72145295 \cdot 10^6 \cos(\theta_1)^2 - 7.829005576 \cdot 10^5 \sin(\theta_1) \cos(\theta_1), \\ -3.914502788 \cdot 10^5 + 6.72145295 \cdot 10^6 \sin(\theta_1) \cos(\theta_1) \\ + 7.829005576 \cdot 10^5 \cos(\theta_1)^2, -0.0004 \cos(\theta_1) \cos(\theta_2)^4 \\ - 7.995437187 \cdot 10^5 \cos(\theta_1) + 1.221598627 \cdot 10^7 \sin(\theta_1)], \\ [-3.914502788 \cdot 10^5 + 6.72145295 \cdot 10^6 \sin(\theta_1) \cos(\theta_1) \\ + 7.829005576 \cdot 10^5 \cos(\theta_1)^2, 4.220607335 \cdot 10^7 - 6.72145295 \cdot 10^6 \cos(\theta_1)^2 \\ + 7.829005576 \cdot 10^5 \sin(\theta_1) \cos(\theta_1), -0.0004 \sin(\theta_1) \cos(\theta_2)^4 \\ - 7.995437187 \cdot 10^5 \sin(\theta_1) - 1.221598627 \cdot 10^7 \cos(\theta_1)], \\ [-0.0004 \cos(\theta_1) \cos(\theta_2)^4 - 7.995437187 \cdot 10^5 \cos(\theta_1) \\ + 1.221598627 \cdot 10^7 \sin(\theta_1), -0.0004 \sin(\theta_1) \cos(\theta_2)^4 - 7.995437187 \cdot 10^5 \sin(\theta_1) \\ - 1.221598627 \cdot 10^7 \cos(\theta_1), 3.243868237 \cdot 10^7] \end{bmatrix}$$

$$J_3 := \begin{bmatrix} [-4.14070 \sin(\theta_1) - 80.18370 \cos(\theta_1) - 565.2 \sin(\theta_1) \cos(\theta_2), \\ -565.2 \cos(\theta_1) \sin(\theta_2), -565.2 \cos(\theta_1) \sin(\theta_2)], \\ [4.14070 \cos(\theta_1) - 80.18370 \sin(\theta_1) + 565.2 \cos(\theta_1) \cos(\theta_2), \\ -565.2 \sin(\theta_1) \sin(\theta_2), -565.2 \sin(\theta_1) \sin(\theta_2)], \\ [0, -565.2 \cos(\theta_2), -565.2 \cos(\theta_2)], \\ [0, -\sin(\theta_1), -\sin(\theta_1)], \\ [0, \cos(\theta_1), \cos(\theta_1)], \\ [1, 0, -\sin(\theta_1)] \end{bmatrix}$$

$$A_3 := \begin{bmatrix} [-4.14070 \sin(\theta_1) \cos(\theta_2)^2 - 4.14070 \sin(\theta_1) \sin(\theta_2)^2 - 80.18370 \cos(\theta_1) \\ -565.2 \sin(\theta_1) \cos(\theta_2), -565.2 \cos(\theta_1) \sin(\theta_2), -565.2 \cos(\theta_1) \sin(\theta_2)], \\ [4.14070 \cos(\theta_1) \cos(\theta_2)^2 + 4.14070 \cos(\theta_1) \sin(\theta_2)^2 - 80.18370 \sin(\theta_1) \\ + 565.2 \cos(\theta_1) \cos(\theta_2), -565.2 \sin(\theta_1) \sin(\theta_2), -565.2 \sin(\theta_1) \sin(\theta_2)], \\ [0, -565.2 \cos(\theta_2), -565.2 \cos(\theta_2)] \end{bmatrix}$$

$$B_3 := \begin{bmatrix} 0 & -\sin(\theta_1) & -\sin(\theta_1) \\ 0 & \cos(\theta_1) & \cos(\theta_1) \\ 1 & 0 & -\sin(\theta_1) \end{bmatrix}$$

### Robot kolunun toplam kütle matrisi

$$Dq := \begin{bmatrix} [2.569057201 \cdot 10^9 + 1.818509043 \cdot 10^9 \cos(\theta_2)^2 + 1.472508526 \cdot 10^7 \cos(\theta_2), \\ -1.221598627 \cdot 10^7 + 1.423955558 \cdot 10^8 \sin(\theta_2), -1.221598627 \cdot 10^7 \\ -3.243868237 \cdot 10^7 \sin(\theta_1) + 1.428304163 \cdot 10^8 \sin(\theta_2)], \\ [-1.221598627 \cdot 10^7 + 1.423955559 \cdot 10^8 \sin(\theta_2), 1.856295895 \cdot 10^9, 1.391644570 \cdot 10^9 \\ + 1.221598627 \cdot 10^7 \sin(\theta_1)], \\ [-1.221598627 \cdot 10^7 - 3.243868237 \cdot 10^7 \sin(\theta_1) + 1.428304164 \cdot 10^8 \sin(\theta_2), \\ 1.391644570 \cdot 10^9 + 1.221598627 \cdot 10^7 \sin(\theta_1), 2.443197254 \cdot 10^7 \sin(\theta_1) \\ + 1.424083253 \cdot 10^9 - 3.2438682 \cdot 10^7 \cos(\theta_1)^2] \end{bmatrix}$$



## Coriolis ve Merkezkaç Kuvvet vektörü

$$\begin{aligned} CM_1 := & -0.04 \theta_{n_1} \theta_{n_2} \cos(\theta_1) \sin(\theta_2) \sin(\theta_1) \\ & + 0.0001 \cos(\theta_1)^4 \cos(\theta_2) \theta_{n_2}^2 - 0.0002 \cos(\theta_1)^4 \theta_{n_2} \theta_{n_3} \\ & + 0.0003 \theta_{n_3}^2 \sin(\theta_1) \cos(\theta_1)^2 - 0.01 \theta_{n_2} \theta_{n_3} \cos(\theta_1)^3 \\ & + 0.0002 \theta_{n_2} \theta_{n_3} \cos(\theta_1)^2 \\ & - 0.0006 \cos(\theta_1)^2 \cos(\theta_2)^2 \sin(\theta_1) \theta_{n_3}^2 \\ & + 0.0003 \theta_{n_2} \theta_{n_3} \sin(\theta_1) \cos(\theta_1)^2 \\ & - 0.02 \theta_{n_3}^2 \cos(\theta_1) \cos(\theta_2)^2 \sin(\theta_1) - 0.01 \theta_{n_3}^2 \cos(\theta_1)^3 \\ & + 0.0001 \theta_{n_3}^2 \cos(\theta_1)^2 + 0.0001 \theta_{n_2}^2 \cos(\theta_1)^2 \\ & - 0.0001 \cos(\theta_1)^4 \theta_{n_3}^2 - 0.0001 \cos(\theta_1)^4 \theta_{n_2}^2 \\ & - 0.0006 \cos(\theta_1)^2 \cos(\theta_2)^2 \sin(\theta_1) \theta_{n_2} \theta_{n_3} \\ & + 0.02 \theta_{n_3}^2 \cos(\theta_1) \cos(\theta_2)^4 \sin(\theta_1) \\ & + 0.0006 \theta_{n_3}^2 \sin(\theta_1) \cos(\theta_1)^2 \cos(\theta_2)^4 \\ & - 3.637018086 \cdot 10^9 \theta_{n_1} \theta_{n_2} \sin(\theta_2) \cos(\theta_2) \\ & + 0.0006 \theta_{n_2} \theta_{n_3} \sin(\theta_1) \cos(\theta_1)^2 \cos(\theta_2)^4 \\ & + 1.428304163 \cdot 10^8 \theta_{n_2} \theta_{n_3} \cos(\theta_2) + 0.05 \theta_{n_1}^2 \cos(\theta_2) \\ & + 1.423955558 \cdot 10^8 \theta_{n_2}^2 \cos(\theta_2) - 1.221598626 \cdot 10^7 \theta_{n_3}^2 \cos(\theta_1) \\ & - 1.221598626 \cdot 10^7 \theta_{n_2} \theta_{n_3} \cos(\theta_1) \\ & - 3.243868236 \cdot 10^7 \theta_{n_3}^2 \cos(\theta_1) \sin(\theta_1) \\ & - 1.472508526 \cdot 10^7 \theta_{n_1} \theta_{n_2} \sin(\theta_2) \\ & - 0.10 \theta_{n_1}^2 \cos(\theta_1)^2 \cos(\theta_2) \end{aligned}$$

$$\begin{aligned}
CM_2 := & -0.001 \theta_n I^2 \cos(\theta_1)^2 \sin(\theta_2) + 1.221598627 \cdot 10^7 \cos(\theta_1) \theta_n \_1 \theta_n \_3 \\
& + 7.362542629 \cdot 10^6 \theta_n I^2 \sin(\theta_2) \\
& - 1.428304163 \cdot 10^8 \theta_n \_1 \theta_n \_3 \cos(\theta_2) \\
& + 0.01 \theta_n \_1 \theta_n \_2 \cos(\theta_2) - \theta_n I^2 \cos(\theta_1)^2 \sin(\theta_2) \cos(\theta_2) \\
& - 0.01 \theta_n I^2 \cos(\theta_1) \sin(\theta_2) \sin(\theta_1) + 0.0002 \cos(\theta_1)^4 \theta_n \_1 \theta_n \_2 \\
& + 0.0002 \cos(\theta_1)^4 \theta_n \_1 \theta_n \_3 \\
& + 1.818509044 \cdot 10^9 \theta_n I^2 \sin(\theta_2) \cos(\theta_2) \\
& - 0.0002 \theta_n \_1 \theta_n \_2 \cos(\theta_1)^2 - 0.0002 \theta_n \_1 \theta_n \_3 \cos(\theta_1)^2
\end{aligned}$$

$$\begin{aligned}
CM_3 := & 2.443197254 \cdot 10^7 \cos(\theta_1) \theta_n \_1 \theta_n \_3 \\
& + 1.428304164 \cdot 10^8 \theta_n \_1 \theta_n \_2 \cos(\theta_2) \\
& + 0.0002 \cos(\theta_1)^4 \theta_n \_1 \theta_n \_2 + 0.0002 \cos(\theta_1)^4 \theta_n \_1 \theta_n \_3 \\
& + 6.487736474 \cdot 10^7 \theta_n \_1 \theta_n \_3 \cos(\theta_1) \sin(\theta_1) \\
& - 0.0002 \theta_n \_1 \theta_n \_2 \cos(\theta_1)^2 - 0.0002 \theta_n \_1 \theta_n \_3 \cos(\theta_1)^2 \\
& + 1.221598627 \cdot 10^7 \theta_n \_1 \theta_n \_2 \cos(\theta_1) \\
& - 3.243868237 \cdot 10^7 \theta_n I^2 \cos(\theta_1)
\end{aligned}$$

$$\begin{aligned}
C := & \left[ \left[ -0.04 \theta_{n_1} \theta_{n_2} \cos(\theta_1) \sin(\theta_2) \sin(\theta_1) \right. \right. \\
& + 0.0001 \cos(\theta_1)^4 \cos(\theta_2) \theta_{n_2}^2 - 0.0002 \cos(\theta_1)^4 \theta_{n_2} \theta_{n_3} \\
& + 0.0003 \theta_{n_3}^2 \sin(\theta_1) \cos(\theta_1)^2 - 0.01 \theta_{n_2} \theta_{n_3} \cos(\theta_1)^3 \\
& + 0.0002 \theta_{n_2} \theta_{n_3} \cos(\theta_1)^2 \\
& - 0.0006 \cos(\theta_1)^2 \cos(\theta_2)^2 \sin(\theta_1) \theta_{n_3}^2 \\
& + 0.0003 \theta_{n_2} \theta_{n_3} \sin(\theta_1) \cos(\theta_1)^2 \\
& - 0.02 \theta_{n_3}^2 \cos(\theta_1) \cos(\theta_2)^2 \sin(\theta_1) - 0.01 \theta_{n_3}^2 \cos(\theta_1)^3 \\
& + 0.0001 \theta_{n_3}^2 \cos(\theta_1)^2 + 0.0001 \theta_{n_2}^2 \cos(\theta_1)^2 \\
& - 0.0001 \cos(\theta_1)^4 \theta_{n_3}^2 - 0.0001 \cos(\theta_1)^4 \theta_{n_2}^2 \\
& - 0.0006 \cos(\theta_1)^2 \cos(\theta_2)^2 \sin(\theta_1) \theta_{n_2} \theta_{n_3} \\
& + 0.02 \theta_{n_3}^2 \cos(\theta_1) \cos(\theta_2)^4 \sin(\theta_1) \\
& + 0.0006 \theta_{n_3}^2 \sin(\theta_1) \cos(\theta_1)^2 \cos(\theta_2)^4 \\
& - 3.637018086 \cdot 10^9 \theta_{n_1} \theta_{n_2} \sin(\theta_2) \cos(\theta_2) \\
& + 0.0006 \theta_{n_2} \theta_{n_3} \sin(\theta_1) \cos(\theta_1)^2 \cos(\theta_2)^4 \\
& + 1.428304163 \cdot 10^8 \theta_{n_2} \theta_{n_3} \cos(\theta_2) + 0.05 \theta_{n_1}^2 \cos(\theta_2) \\
& + 1.423955558 \cdot 10^8 \theta_{n_1}^2 \cos(\theta_2) - 1.221598626 \cdot 10^7 \theta_{n_3}^2 \cos(\theta_1) \\
& - 1.221598626 \cdot 10^7 \theta_{n_2} \theta_{n_3} \cos(\theta_1) \\
& - 3.243868236 \cdot 10^7 \theta_{n_3}^2 \cos(\theta_1) \sin(\theta_1) \\
& - 1.472508526 \cdot 10^7 \theta_{n_1} \theta_{n_2} \sin(\theta_2) \\
& \left. - 0.10 \theta_{n_1}^2 \cos(\theta_1)^2 \cos(\theta_2) \right], \\
& \left[ -0.001 \theta_{n_1}^2 \cos(\theta_1)^2 \sin(\theta_2) \right. \\
& + 1.221598627 \cdot 10^7 \cos(\theta_1) \theta_{n_1} \theta_{n_3} \\
& + 7.362542629 \cdot 10^6 \theta_{n_1}^2 \sin(\theta_2) \\
& - 1.428304163 \cdot 10^8 \theta_{n_1} \theta_{n_3} \cos(\theta_2) \\
& + 0.01 \theta_{n_1} \theta_{n_2} \cos(\theta_2) - \theta_{n_1}^2 \cos(\theta_1)^2 \sin(\theta_2) \cos(\theta_2) \\
& - 0.01 \theta_{n_1}^2 \cos(\theta_1) \sin(\theta_2) \sin(\theta_1) + 0.0002 \cos(\theta_1)^4 \theta_{n_1} \theta_{n_2} \\
& + 0.0002 \cos(\theta_1)^4 \theta_{n_1} \theta_{n_3} \\
& + 1.818509044 \cdot 10^9 \theta_{n_1}^2 \sin(\theta_2) \cos(\theta_2) \\
& \left. - 0.0002 \theta_{n_1} \theta_{n_2} \cos(\theta_1)^2 - 0.0002 \theta_{n_1} \theta_{n_3} \cos(\theta_1)^2 \right], \\
& \left[ 2.443197254 \cdot 10^7 \cos(\theta_1) \theta_{n_1} \theta_{n_3} \right. \\
& + 1.428304164 \cdot 10^8 \theta_{n_1} \theta_{n_2} \cos(\theta_2) \\
& + 0.0002 \cos(\theta_1)^4 \theta_{n_1} \theta_{n_2} + 0.0002 \cos(\theta_1)^4 \theta_{n_1} \theta_{n_3} \\
& + 6.487736474 \cdot 10^7 \theta_{n_1} \theta_{n_3} \cos(\theta_1) \sin(\theta_1) \\
& - 0.0002 \theta_{n_1} \theta_{n_2} \cos(\theta_1)^2 - 0.0002 \theta_{n_1} \theta_{n_3} \cos(\theta_1)^2 \\
& + 1.221598627 \cdot 10^7 \theta_{n_1} \theta_{n_2} \cos(\theta_1) \\
& \left. - 3.243868237 \cdot 10^7 \theta_{n_1}^2 \cos(\theta_1) \right] \Big]
\end{aligned}$$

### Yerçekimi vektörü

$$Z_1 := 0.$$

$$Z_2 := -2.777844329 \cdot 10^7 \cos(\theta_2)$$

$$Z_3 := -1.744307561 \cdot 10^7 \cos(\theta_2)$$

$$G := \begin{bmatrix} 0. \\ -2.777844329 \cdot 10^7 \cos(\theta_2) \\ -1.744307561 \cdot 10^7 \cos(\theta_2) \end{bmatrix}$$

### Robot kolunun her bir eklemine etki eden tork

$$\begin{aligned}
 \tau_1 := & -0.04 \theta_{n_1} \theta_{n_2} \cos(\theta_1) \sin(\theta_2) \sin(\theta_1) \\
 & + 0.0001 \cos(\theta_1)^4 \cos(\theta_2) \theta_{n_2}^2 - 0.0002 \cos(\theta_1)^4 \theta_{n_2} \theta_{n_3} \\
 & + 0.0003 \theta_{n_3}^2 \sin(\theta_1) \cos(\theta_1)^2 - 0.01 \theta_{n_2} \theta_{n_3} \cos(\theta_1)^3 \\
 & + 0.0002 \theta_{n_2} \theta_{n_3} \cos(\theta_1)^2 \\
 & - 0.0006 \cos(\theta_1)^2 \cos(\theta_2)^2 \sin(\theta_1) \theta_{n_3}^2 \\
 & + 0.0003 \theta_{n_2} \theta_{n_3} \sin(\theta_1) \cos(\theta_1)^2 \\
 & - 0.02 \theta_{n_3}^2 \cos(\theta_1) \cos(\theta_2)^2 \sin(\theta_1) - 0.01 \theta_{n_3}^2 \cos(\theta_1)^3 \\
 & + 0.0001 \theta_{n_3}^2 \cos(\theta_1)^2 + 0.0001 \theta_{n_2}^2 \cos(\theta_1)^2 \\
 & - 0.0001 \cos(\theta_1)^4 \theta_{n_3}^2 - 0.0001 \cos(\theta_1)^4 \theta_{n_2}^2 \\
 & - 0.0006 \cos(\theta_1)^2 \cos(\theta_2)^2 \sin(\theta_1) \theta_{n_2} \theta_{n_3} \\
 & + 0.02 \theta_{n_3}^2 \cos(\theta_1) \cos(\theta_2)^4 \sin(\theta_1) \\
 & + 0.0006 \theta_{n_3}^2 \sin(\theta_1) \cos(\theta_1)^2 \cos(\theta_2)^4 \\
 & - 3.637018086 \cdot 10^9 \theta_{n_1} \theta_{n_2} \sin(\theta_2) \cos(\theta_2) \\
 & + 0.0006 \theta_{n_2} \theta_{n_3} \sin(\theta_1) \cos(\theta_1)^2 \cos(\theta_2)^4 \\
 & + 1.428304163 \cdot 10^8 \theta_{n_2} \theta_{n_3} \cos(\theta_2) + 0.05 \theta_{n_1}^2 \cos(\theta_2) \\
 & + 1.423955558 \cdot 10^8 \theta_{n_2}^2 \cos(\theta_2) - 1.221598626 \cdot 10^7 \theta_{n_3}^2 \cos(\theta_1) \\
 & - 1.221598626 \cdot 10^7 \theta_{n_2} \theta_{n_3} \cos(\theta_1) \\
 & - 3.243868236 \cdot 10^7 \theta_{n_3}^2 \cos(\theta_1) \sin(\theta_1) \\
 & - 1.472508526 \cdot 10^7 \theta_{n_1} \theta_{n_2} \sin(\theta_2) \\
 & - 0.10 \theta_{n_1}^2 \cos(\theta_1)^2 \cos(\theta_2) + 2.569057201 \cdot 10^9 \theta_{nn_1} \\
 & - 1.221598627 \cdot 10^7 \theta_{nn_2} - 1.221598627 \cdot 10^7 \theta_{nn_3} \\
 & + 1.472508526 \cdot 10^7 \cos(\theta_2) \theta_{nn_1} + 1.428304163 \cdot 10^8 \theta_{nn_3} \sin(\theta_2) \\
 & + 1.423955558 \cdot 10^8 \theta_{nn_2} \sin(\theta_2) + 0.00007 \theta_{nn_1} \cos(\theta_1)^2 \cos(\theta_2)^4 \\
 & + 1.818509043 \cdot 10^9 \theta_{nn_1} \cos(\theta_2)^2 - 3.243868237 \cdot 10^7 \theta_{nn_3} \sin(\theta_1)
 \end{aligned}$$

$$\begin{aligned}
\tau_2 := & -2.777844329 \cdot 10^7 \cos(\theta_2) - 0.001 \theta_{n\_1}^2 \cos(\theta_1)^2 \sin(\theta_2) \\
& + 1.221598627 \cdot 10^7 \cos(\theta_1) \theta_{n\_1} \theta_{n\_3} \\
& + 7.362542629 \cdot 10^6 \theta_{n\_1}^2 \sin(\theta_2) \\
& - 1.428304163 \cdot 10^8 \theta_{n\_1} \theta_{n\_3} \cos(\theta_2) \\
& + 0.01 \theta_{n\_1} \theta_{n\_2} \cos(\theta_2) - \theta_{n\_1}^2 \cos(\theta_1)^2 \sin(\theta_2) \cos(\theta_2) \\
& - 0.01 \theta_{n\_1}^2 \cos(\theta_1) \sin(\theta_2) \sin(\theta_1) + 0.0002 \cos(\theta_1)^4 \theta_{n\_1} \theta_{n\_2} \\
& + 0.0002 \cos(\theta_1)^4 \theta_{n\_1} \theta_{n\_3} \\
& + 1.818509044 \cdot 10^9 \theta_{n\_1}^2 \sin(\theta_2) \cos(\theta_2) \\
& - 0.0002 \theta_{n\_1} \theta_{n\_2} \cos(\theta_1)^2 - 0.0002 \theta_{n\_1} \theta_{n\_3} \cos(\theta_1)^2 \\
& - 1.221598627 \cdot 10^7 \theta_{nn\_1} + 1.856295895 \cdot 10^9 \theta_{nn\_2} \\
& + 1.391644570 \cdot 10^9 \theta_{nn\_3} - 0.20 \theta_{nn\_3} \cos(\theta_1)^2 \\
& + 1.221598627 \cdot 10^7 \theta_{nn\_3} \sin(\theta_1) + 1.423955559 \cdot 10^8 \theta_{nn\_1} \sin(\theta_2) \\
\tau_3 := & -1.744307561 \cdot 10^7 \cos(\theta_2) + 2.443197254 \cdot 10^7 \cos(\theta_1) \theta_{n\_1} \theta_{n\_3} \\
& + 1.428304164 \cdot 10^8 \theta_{n\_1} \theta_{n\_2} \cos(\theta_2) \\
& + 0.0002 \cos(\theta_1)^4 \theta_{n\_1} \theta_{n\_2} + 0.0002 \cos(\theta_1)^4 \theta_{n\_1} \theta_{n\_3} \\
& + 6.487736474 \cdot 10^7 \theta_{n\_1} \theta_{n\_3} \cos(\theta_1) \sin(\theta_1) \\
& - 0.0002 \theta_{n\_1} \theta_{n\_2} \cos(\theta_1)^2 - 0.0002 \theta_{n\_1} \theta_{n\_3} \cos(\theta_1)^2 \\
& + 1.221598627 \cdot 10^7 \theta_{n\_1} \theta_{n\_2} \cos(\theta_1) \\
& - 3.243868237 \cdot 10^7 \theta_{n\_1}^2 \cos(\theta_1) - 1.221598627 \cdot 10^7 \theta_{nn\_1} \\
& + 1.391644570 \cdot 10^9 \theta_{nn\_2} + 1.424083252 \cdot 10^9 \theta_{nn\_3} \\
& + 0.26 \theta_{nn\_3} \cos(\theta_2)^2 - 3.243868220 \cdot 10^7 \theta_{nn\_3} \cos(\theta_1)^2 \\
& + 2.443197254 \cdot 10^7 \theta_{nn\_3} \sin(\theta_1) - 3.243868237 \cdot 10^7 \theta_{nn\_1} \sin(\theta_1) \\
& + 1.221598627 \cdot 10^7 \theta_{nn\_2} \sin(\theta_1) + 1.428304164 \cdot 10^8 \theta_{nn\_1} \sin(\theta_2)
\end{aligned}$$

## ÖZGEÇMİŞ

### KİŞİSEL BİLGİLER

**Adı Soyadı** : Muhammet Mevlüt KARACA  
**Uyruğu** : Türkiye  
**Doğum Yeri ve Tarihi** : Konya-1994  
**Telefon** : +905069830783  
**Faks** : -  
**e-mail** : mmevlutkaraca@hotmail.com

### EĞİTİM

Derece	Adı, İlçe, İl	Bitirme Yılı
Lise	: Büyükkoyuncu Anadolu Lisesi, Selçuklu, Konya	2012
Üniversite	: KTO Karatay Üniversitesi, Karatay, Konya	2017
Yüksek Lisans	: Necmettin Erbakan Üniversitesi, Meram, Konya	
Doktora	: -	

### İŞ DENEYİMLERİ

Yıl	Kurum	Görevi
2017	FMC Hidrolik-Dövme Otomotiv Mak. San. Tic. Ltd. Şti.	Dövme Kalıphane- Tasarım Bölümü
2018	Aydınlar Yedek Parça San. Tic. A.Ş.	Arge Elemanı

### YABANCI DİLLER

İngilizce

### YAYINLAR

Karaca M. M., Özçelik Z., 2019, Fabrika Otomasyonu için Robot Kol ve Gripperin Modellenmesi ve Kontrolü IATENS'19 (The International Aluminium-Themed Engineering and Natural Sciences Conference), 1144-1144