

T.C.  
SAKARYA ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ

**EMG VE JİROSKOP VERİLERİ İLE  
ENDÜSTRİYEL ROBOT KOLUNUN  
GERÇEK ZAMANLI KONTROLÜ**

**YÜKSEK LİSANS TEZİ**

**Kenan ERİN**

**Enstitü Anabilim Dalı : MEKATRONİK MÜHENDİSLİĞİ**

**Tez Danışmanı : Yrd. Doç. Dr. Barış BORU**

**Şubat 2018**

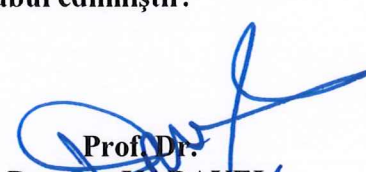
T.C.  
SAKARYA ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ

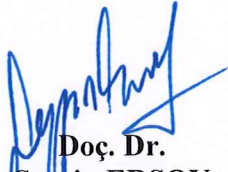
EMG VE JİROSKOP VERİLERİ İLE  
ENDÜSTRİYEL ROBOT KOLUNUN  
GERÇEK ZAMANLI KONTROLÜ

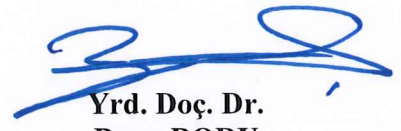
YÜKSEK LİSANS TEZİ  
Kenan ERİN

Enstitü Anabilim Dalı : MEKATRONİK MÜHENDİSLİĞİ

Bu tez 01.02.2018 tarihinde aşağıdaki jüri tarafından oybirliği / ~~oyçokluğu~~ ile kabul edilmiştir.

  
Prof. Dr.  
Durmuş KARAYEL  
Jüri Başkanı

  
Doç. Dr.  
Sezgin ERSOY  
Üye

  
Yrd. Doç. Dr.  
Barış BORU  
Üye

## BEYAN

Tez içindeki tüm verilerin akademik kurallar çerçevesinde tarafımdan elde edildiğini, görsel ve yazılı tüm bilgi ve sonuçların akademik ve etik kurallara uygun şekilde sunulduğunu, kullanılan verilerde herhangi bir tahrifat yapılmadığını, başkalarının eserlerinden yararlanılması durumunda bilimsel normlara uygun olarak atıfta bulunulduğunu, tezde yer alan verilerin bu üniversite veya başka bir üniversitede herhangi bir tez çalışmasında kullanılmadığını beyan ederim.

Kenan ERİN

01.02.2018



## **TEŐEKKÜR**

Yüksek lisans eğitiminin boyunca değerli bilgi ve deneyimlerinden yararlandığım, her konuda bilgi ve desteğini almaktan çekinmediğim, araştırmanın planlanmasından yazılmasına kadar tüm aşamalarında yardımlarını esirgemeyen, teşvik eden, aynı titizlikte beni yönlendiren değerli danışman hocam Yrd. Doç. Dr. Barış BORU'ya teşekkür ederim.

Bu çalışmanın maddi açıdan desteklenmesine olanak sağlayan Sakarya Üniversitesi Bilimsel Araştırma Projeleri (BAP) Komisyon Başkanlığına (Proje No: 2017-50-01-059) teşekkür ederim

Ayrıca hayatım boyunca desteklerinin esirgemeyen ve beni bugünlere getiren annem, babam ve kardeşlerime sonsuz teşekkürlerimi sunarım.

# İÇİNDEKİLER

TEŞEKKÜR .....	i
İÇİNDEKİLER .....	ii
SİMGELER VE KISALTMALAR LİSTESİ .....	iv
ŞEKİLLER LİSTESİ .....	v
TABLOLAR LİSTESİ .....	vi
ÖZET .....	vii
SUMMARY .....	viii
BÖLÜM 1.	
GİRİŞ.....	1
BÖLÜM 2.	
KAYNAK ARAŞTIRMASI .....	3
2.1. Tezin Literatüre Katkısı.....	12
BÖLÜM 3.	
MATERYAL VE YÖNTEM .....	14
3.1. Emg İşareti .....	15
3.2. Myo Armband.....	16
3.3. IRB120 Endüstriyel Robot .....	18
3.4. RAPID Programlama.....	19
3.4.1. Rapid verileri.....	20
3.4.1.1. Değişkenler .....	20
3.4.1.2. Rapid akış kontrol terimleri.....	21
3.4.1.3. Rapid hareket komutları.....	22
3.5. RobotStudio Programı.....	23

3.6. TCP/IP Haberleşme Protokolü.....	25
3.6.1. Rapid programlama TCP/IP haberleşmesi.....	28
3.7. Kuaterniyon Matrisi Ve Euler Açıları.....	32
3.7.1. Kuaterniyon matematiği.....	31
3.7.2. Euler açıları.....	32
BÖLÜM 4.	
DENEY DÜZENEĞİ VE GELİŞTİRİLEN YAZILM .....	34
BÖLÜM 5.	
TARTIŞMA VE SONUÇ .....	40
KAYNAKLAR .....	41
EKLER.....	45
ÖZGEÇMİŞ .....	48

## **SİMGELER VE KISALTMALAR LİSTESİ**

EMG	: Electromyography
EOG	: Electrooculography
EEG	: Electroencephalography
IMU	: Inertial Measurement Unit
HRI	: Human Robot Interface
UDP	: User Datagram Protocol
TCP	: Transmission Control Protocol
IP	: Internet Protocol
PC	: Personal Computer

## ŞEKİLLER LİSTESİ

Şekil 2.1. Pedro ve arkadaşlarının (2010) geliştirdiği HRI sistemi.....	3
Şekil 2.2. Wang ve arkadaşlarının (2012) tasarladıkları mobil robot.....	4
Şekil 2.3. Tanaka ve arkadaşlarının (2005) geliştirdiği elektrikli sandalye.....	5
Şekil 2.4. Yu ve arkadaşlarının (2015) geliştirdiği mobil robot platformu.....	6
Şekil 2.5. Zhao ve arkadaşlarının (2017) geliştirdiği HRI sistemi.....	7
Şekil 2.6. Gürkan ve arkadaşlarının (2017) geliştirdiği tekerlekli sandalye.....	7
Şekil 2.7. Aleotti ve arkadaşlarının (2004) geliştirdiği deney düzeneği.....	8
Şekil 2.8. Liao ve arkadaşlarının (2012) geliştirdiği oyun.....	10
Şekil 3.1. İnsan makina arayüzü sinyal analizi.....	15
Şekil 3.2. Myo Armband.....	16
Şekil 3.3. Thalmic Labs tarafından tanımlanmış hareketler.....	17
Şekil 3.4. IRB120 endüstriyel robotu.....	18
Şekil 3.5. IRB120 robotunun bir noktaya birden fazla mekanik dizilimle gitmesi	24
Şekil 3.6. RobotStudio programı IRB120 robotu.....	25
Şekil 3.7. Sunucu ve istemci arasındaki TCP/IP haberleşmesi.....	27
Şekil 4.1. Deney düzeneğinin görsel şeması.....	34
Şekil 4.2. Myo arm band euler açıları.....	35
Şekil 4.3. Robotu gerçek zamanlı çalıştırmak için yapılan hareket.....	35
Şekil 4.4. Robotu durdurmak için yapılan hareket.....	36
Şekil 4.5. Robotu başlangıç pozisyonuna götürmek için yapılan hareket.....	36
Şekil 4.6. IRB120 robotunun y ekseninde hareket etmesi.....	37
Şekil 4.7. IRB120 robotunun z ekseninde hareket etmesi.....	37
Şekil 4.8. Python sunucu akış diyagramı.....	38
Şekil 4.9. RAPID istemci akış diyagramı.....	39



## TABLolar LİSTESİ

Tablo 3.1. IRB120 robotunun özellikleri.....	18
Tablo 3.2. Rapid deęişkenleri.....	20
Tablo 3.3. Robtarget deęişkeni paket yapısı.....	23
Tablo 3.4. Soket programlama veri tipleri.....	29
Tablo 3.5. Soket programlama istemci komutları.....	29
Tablo 3.6. Soket programlama sunucu komutları.....	30

## ÖZET

Anahtar kelimeler: EMG, Endüstriyel Robot, Myo Armband, Jiroskop, İnsan Robot Arayüzü

Gelişen teknoloji ile giyilebilir cihazlar üzerine akademik geliştirme çalışmaları yapılmakta ve ticari ürünler piyasaya çıkmaya devam etmektedir. Bu çalışmaların önemli bir kısmı insan vücudundaki hareketleri algılayabilen giyilebilir sensörler üzerine odaklanmaktadır.

Bu çalışmada Thalmic Labs tarafından üretilen Myo Armband ürünü ile insan kol hareketlerinin algılanması ve endüstriyel robot kolunu kontrol etmesi ile bir İnsan Robot Arayüzü geliştirilmiştir. Myo Armband bileklik şeklinde olup üzerinde bulunan EMG (Elektromiyografi) ve jiroskop sensörleri ile kolun hareketinin algılanmasında yardımcı olmaktadır. Myo Armband ile bilgisayar sistemi arasında kablosuz bağlantı kurularak hem EMG ve jiroskop verilerinin gerçek zamanlı olarak bilgisayara gönderilmesi mümkündür. Pratik gerçekleştirme için ABB firması tarafından üretilen IRB120 endüstriyel robotu kullanılmıştır. IRB120 endüstriyel robotu kendi kontrolörü dışında ABB firması tarafından verilen yazılım geliştirme kiti kullanılarak ethernet haberleşmesi ile de kontrol edilebilmektedir. Geliştirilen yazılım ile EMG ve jiroskop verilerinden elde edilen veriler hareket ve konum bilgilerine dönüştürülerek ethernet üzerinden gerçek zamanlı gönderilmesi ile endüstriyel robotun insan hareketlerini taklit etmesi sağlanmıştır.

Bu çalışma sonucu olarak gerçekleştirilen yazılım ile yeni HRI (İnsan Robot Arayüzü) sistemi endüstriyel robot kollarının ucuz ve kolay yollardan kontrolünü mümkün kılmaktadır.

# **REAL TIME CONTROL OF INDUSTRIAL ROBOT ARM WITH EMG AND GYROSCOPE DATA**

## **SUMMARY**

Keywords: EMG, Industrial Robot, Myo Armband, Gyroscope, Human Robot Interface

Academic development efforts are rising about wearable devices and commercial products has begun to appear commercial market. This study is mainly focused on wearable sensors that can be used to measure human movements.

In this study, a Human Robot Interface (HRI) was developed with the Myo Armband product by Thalmic Labs, with the perception of human arm movements. Myo Armband is bracelet shaped device and it can be use to detect movements of the arm thanks to EMG (Elektromyography) and gyroscope sensors. By establishing a wireless connection between Myo Armband and the computer system it is possible to send raw EMG and Gyroscope data in real time to the computer. IRB120 industrial robot that produced by ABB Robotics company is used for practical implementation. IRB120 industrial robot can be controlled by third party programmimg languages via ethernet communication besides its own controller. In this study, EMG and gyroscope data has been sent to robot arm over ethernet in order to imitate the human movements.

As a result of this study, a new HRI (Human Robot Interface) system that can be used to control industrial robot by human movements in a cheap and easy way.

## **BÖLÜM 1. GİRİŞ**

Teknolojinin ilerlemesiyle giyilebilir cihazlar hayatımıza girmeye başlamıştır. Geliştirilen giyilebilir cihazların üretiminde ek özellikler olarak ivmeölçer, jiroskop, alan tarayıcı ve kızılötesi sensörler sıklıkla kullanılmaktadır. Bu tip sensörlerin kullanımıyla hareket algılama, ses algılama gibi insan davranışları rahatlıkla tespit edilebilmektedir.

Robot teknolojisi dünyada hızlı gelişme gösteren alanların başında gelmektedir. Robotların gelişmesiyle firmalar robotlarının pazarlamasında farklı özellikler geliştirmeye başlamışlardır. Özellikle akıllı sistemlerin robotlara entegre edilmesiyle robotlar kullanımı daha cazip hale gelerek müşterilerin ilginin çekmeye başlamıştır. Bu nedenle firmalar insanların robotlarla haberleşmesi için insan robot etkileşim ara yüzü (HRI) sistemleri geliştirmişlerdir. Bu tez kapsamında insan robot etkileşimi için elektromiyografi (EMG) ve atalet ölçme birimi (IMU) sensör tabanlı bir yazılım geliştirilmiştir. Geliştirilen bu yazılımda EMG bileklik kullanılmış olup kullanıcının üç farklı el-parmak hareketi tespit edilmiştir ve bu hareketler robotun sürekli olarak çalıştırma, acil stop yaptırma ve başlangıç pozisyonuna gitmesi için tanımlanmıştır. Ayrıca EMG bileklikte bulunan IMU sensörler vasıtasıyla kolun pozisyon bilgisi tespit edilmiş olup robota gerçek zamanlı olarak gönderilmiştir. Bu sayede robotun kullanımı robotun kumandasına bağlı olmadan rahatlıkla yapılabilmektedir.

Tezin ikinci bölümünde problemin tanımı ve bu problemin çözümüne yönelik literatürde yapılmış çalışmalara değinilmiştir.

Üçüncü kısımda kullanılan materyallerin ve yapılan haberleşmesinin yapısından bahsedilmiştir.

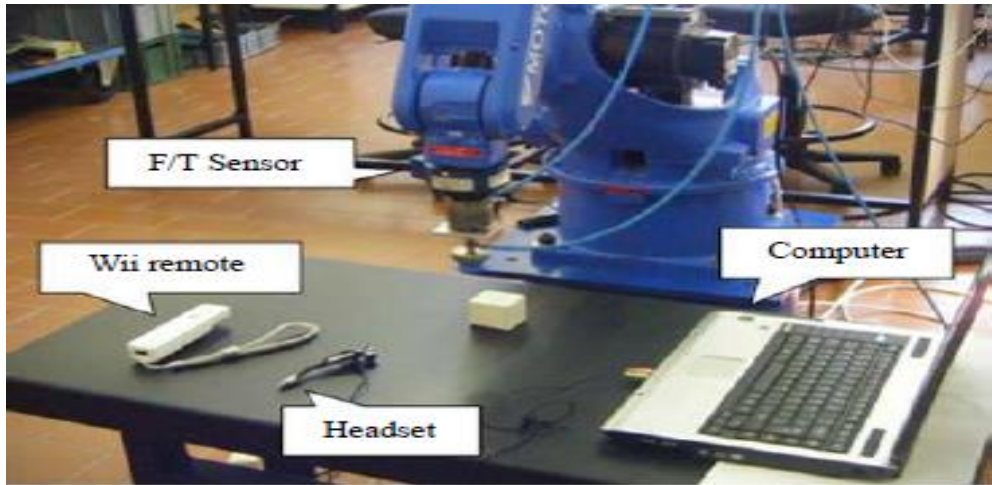
Dördüncü kısımda EMG bileklikten verilerin alımıp işlenmesi ve gerçek zamanlı olarak robota gönderim için yapılan deney düzeneğinden bahsedilmiştir.

Tartışma ve sonuç kısmında ise sistemin avantaj ve dezavantajları ve sistemin ilerleyen zamanlarda nasıl geliştirilebileceğinden bahsedilmiştir.

## BÖLÜM 2. KAYNAK ARAŞTIRMASI

İnsan robot etkileşimi son yıllarda popülerliği hızla artan bir konudur. HRI sistemleri günümüzde cerrahi müdahale, arama kurtarma, insanların girmesinin zor olduğu yerlerde ve özellikle biyomedikal ve robotik alanında yapılan çalışmalarda sıklıkla kullanılmaktadır. Vücudun belirli kaslarına yerleştirilen sensörler ile alınan veriler bilgisayar ortamında işlenip mobil ve endüstriyel robotlarla etkileşim oldukça yaygın hale gelmiştir. EEG (Jonathan ve ark.; 2002; Khuram ve ark., 2012; Yan ve ark., 2016; Ana ve ark.,2016; Fred ve ark., 2016), EOG (Chen ve Wyatt, 2004; Chun ve ark., 2007) EMG (Ranathunga ve Kazuo, 2008; Panagioitis ve Kostas, 2010) ve kamera tabanlı (Roberto ve Nicholas, 1996; Jamie ve ark., 2013 ) hareket çıkarımı literatürde en çok kullanılan yöntemlerdir.

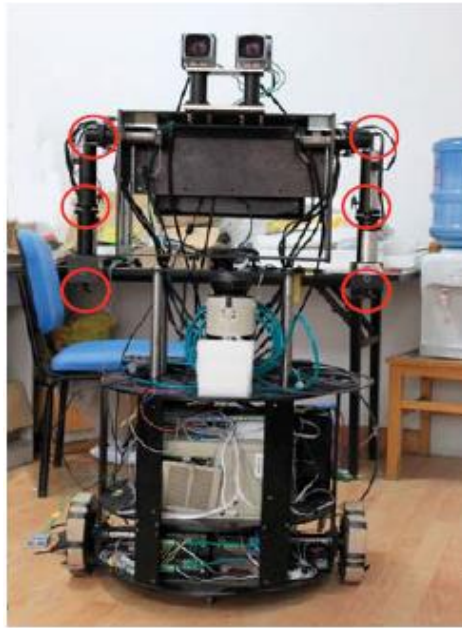
Pedro ve arkadaşları (2010) yaptıkları çalışmada bir HRI sistemi geliştirmişlerdir. Bu çalışmada üzerinde 3 eksenli ivmeölçer bulunan Wii uzaktan kontrol kumandası ile alınan verileri yapay sinir ağları kullanarak sınıflandırmış ve hareket kestiriminde bulunmuşlardır (Şekil 2.1.). Hareket kestiriminden sonra ethernet haberleşmesi ile MOTOMAN endüstriyel robotunun 3 boyutlu uzayda hareketini sağlamışlardır.



Şekil 2.1. Pedro ve arkadaşlarının (2010) geliştirdiği HRI sistemi

Christian ve Thomas (2007) yaptıkları çalışmada kafa üzerine yerleştirilen IMU sensör ile tekerlekli sandalyenin kontrolünü sağlamışlardır. Kullanılan IMU sensör 9 eksenli olup 3 eksenli ivme ölçer, 3 eksenli jiroskop ve 3 eksenli magnemeterden oluşmaktadır. IMU sensör ile alınan verilerden roll, pitch, yaw açıları belirlenmiş olup tekerlekli sandalyenin yön tayini yapılmıştır.

Wang ve arkadaşları (2012) EMG ve IR (kızılötesi) kameraya sahip sensör kullanarak insan benzetimli mobil robotun kontrolünü gerçekleştirmişlerdir (Şekil 2.2.). EMG ve IR kameradan gelen veriler veri toplama kartıyla toplanmıştır. Daha sonra alınan veriler özellik çıkarma ve hareket tanımlama yöntemleriyle anlamlı hale getirilmiş olup motor sürücülerine gerekli bilgiler verilerek mobil robotun hareketi sağlanmıştır. EMG sinyallerini almak için elektrot kullanılmıştır. EMG sinyallerinde oluşan gürültüyü gidermek için 10 Hz yüksek geçiren filtre uygulamışlardır.

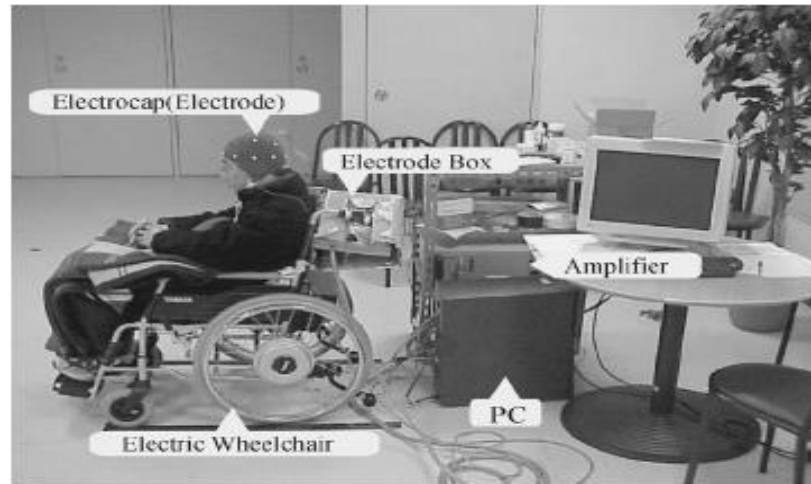


Şekil 2.2. Wang ve arkadaşlarının (2012) geliştirdiği HRI sistemi

Yihsin ve arkadaşları (2014) ivme ölçer sensör ve kameradan alınan verilerden hareket kestirimi yapmışlardır. İvmeölçer sensörden yapılan hareketin koordinatlarını tespit etmek, kameradan alınan görüntülerden ise hareket hızını tespit etmek için yararlanılmıştır. Hangi hareketin yapıldığını bulmak için ise bulanık mantık çıkarımı yöntemini kullanmışlardır.

Umut ve arkadaşları (2015) kola yerleştirilen elektrotlar yardımıyla EMG tabanlı mobil robotun kontrolünü gerçekleştirmişlerdir. Mobil robotun hareketi için 3 hareket tanımlamışlardır. Elektrotlardan alınan sinyallerin genliği düşük olduğundan ilk önce sinyal yükselticiden geçirmişlerdir. Daha sonra bu verileri kullandıkları mikroişlemcinin analog kanallarından okuyup hareketin türüne göre motorlara sinyal göndermişlerdir.

Tanaka ve arkadaşları (2005) EEG sinyallerini kullanarak elektrikli sandalyenin kontrolünü gerçekleştirmişlerdir (Şekil 2.3.). EEG sinyallerini almak için insan kafasına 14 adet elektrot yerleştirmişlerdir. Alınan EEG sinyallerinin frekansı 1024 Hz'dir. Ayrıca EEG sinyallerini etkileyen gürültüden kaçınmak için 0.53-30 Hz arası band geçiren filtre uygulamışlardır. Çalışmanın sonunda elektrikli sandalyenin istenilen koordinata gitme başarı oranını %80 olarak belirlenmiştir.

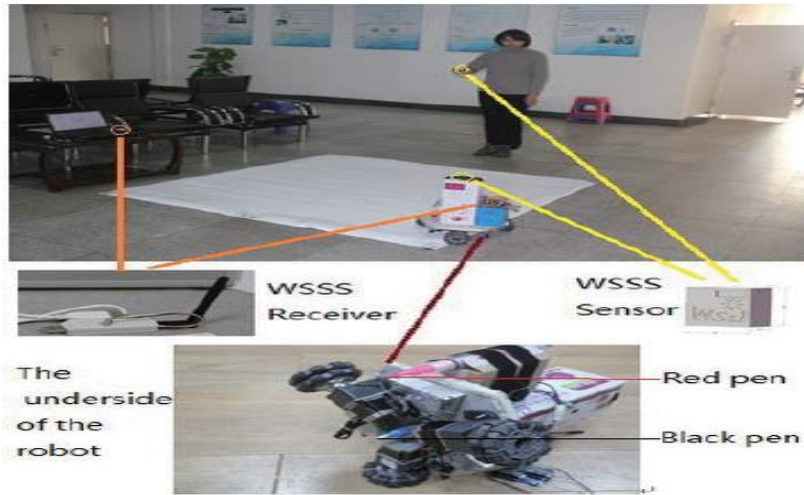


Şekil 2.3. Tanaka ve arkadaşlarının (2005) geliştirdiği elektrikli sandalye

Beyda ve arkadaşları (2014) yüzey elektrotlar yardımıyla alınan sinyaller ile biyonik el simülatörü uygulamasını geliştirmişlerdir. Alınan sinyalleri gürültüden arındırmak için bant geçiren filtreden geçirmiş ve iki temel hareketi (el açma ve kapama) temel bileşen analizi ile sınıflandırmışlardır. Daha sonra sınıflandırılan sinyaller ile MATLAB GUI programında on serbestlik derecesine sahip el simülatörünün kontrolünü sağlamışlardır.

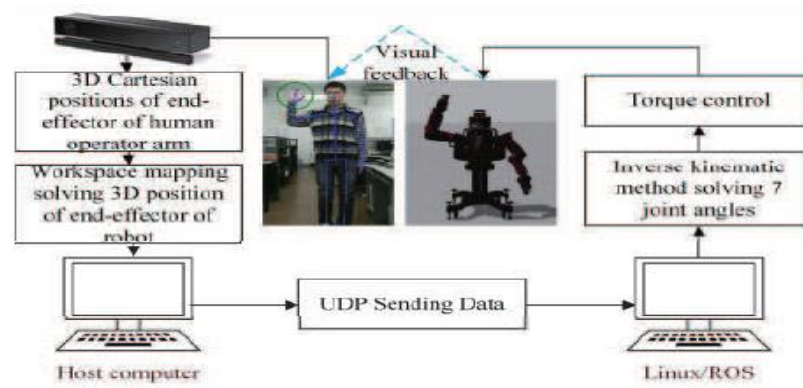


Yu ve arkadaşları (2015) iki IMU sensör ile HRI sistemi gerçekleştirmişlerdir (Şekil 2.4.). Kullanılan iki sensörden biri kolun oryantasyonunu göstermesi için insan koluna diğerini açı hesaplamak için çok yönlü tasarlanmış olan mobil robota entegre etmişlerdir. Kullandıkları IMU sensörler kolun oryantasyonunu gerçek zamanlı olarak vermektedir. Daha sonra itme, çekme ve dönme hareketleriyle mobil robotun kontrolünü sağlamışlardır.



Şekil 2.4. Yu ve arkadaşlarının (2015) geliştirdiği mobil robot platformu

Zhao ve arkadaşları (2016) IR kameraya sahip sensör kullanarak bir HRI sistemi geliştirmişlerdir (Şekil 2.5.). IR kameradan alınan görüntüler ile Baxter robot manipülatörünü kontrol etmişlerdir. İlk önce Baxter robot ve insan kolunun ters kinematik denklemlerini oluşturmuşlardır. Daha sonra kameradan alınan görüntülerden robotun aktuatörlerine sinyal göndermek için kolun açılarını tespit etmişlerdir. Elde edilen açılardan UDP (User Datagram Protocol) haberleşmesi ile insan robot etkileşimini gerçekleştirmişlerdir.



Şekil 2.5. Zhao ve arkadaşlarının (2016) geliştirdiği HRI sistemi

Gürkan ve arkadaşları (2017) ALS (Amiyotrofik Lateral Skleroz) hastaları için akıllı tekerlekli sandalye geliştirmişlerdir (Şekil2.6.). Bu tekerlekli sandalyenin kontrolü için EMG, EEG ve göz hareketleri kullanılarak 3 farklı kontrol düzeneği geliştirmişlerdir. Alınan sinyaller için kablosuz haberleşme (bluetooth) kullanmış ve MATLAB ortamında işlenmiştir. Güvenli navigasyonu sağlamak için ise tekerlekli sandalyeye bir adet IR kameraya sahip sensör yerleştirmişlerdir. Yerleştirilen sensör sandalyenin önünde engel olup olmadığını tespit etmektedir. Yaptıkları çalışmada sistemin başarılı bir şekilde hareket ettiklerini gözlemlemişlerdir.



Şekil 2.6. Gürkan ve arkadaşlarının (2017) geliştirdiği tekerlekli sandalye

Aleotti ve arkadaşları (2004) endüstriyel robot kolunun kola takılan bir sensör vasıtasıyla kontrolü için yeni bir model geliştirmişlerdir (Şekil 2.7.). Yazarlar insan

koluna takılan bir sensörden aldıkları bükme ve eğme değerlerini bilgisayar ortamına almışlardır. Daha sonra bu bükme ve eğme değerleri için MDC (Minimize Deep Coalescence) VE MLFF (Multilayer Feed-Forward) algoritmalarını kurmuşlardır. Algoritma sonucu Pandi-1 isimli robot manipülatörünün açılarını tespit edip robotu kontrol etmişlerdir.



Şekil 2.7. Aleotti ve arkadaşlarının (2004) geliştirdiği deney düzeneği

Mihara ve arkadaşları (2003) kamera ve ses verilerinden aldığı bilgilerle bir HRI sistemi gerçekleştirmişlerdir. Yazarlar bu çalışmada ilk önce kameradan aldığı görüntüden arka planı çıkarmış ve elde ettiği el hareketlerini tanımlamışlardır. Daha sonra tanımladıkları hareketleri pet robot üzerinde denemişlerdir. Pet robotu ya ses ile ya da kameradan aldıkları görüntülerle kontrol etmişlerdir. Pet robota buraya gel dediklerinde veya parmağıyla gel işareti yaptıklarında robotun kullanıcıya doğru gittiğini ve havlama sesi çıkardığını tespit etmişlerdir.

He ve arkadaşları (2016) EEG sinyalleri ile robot kolu kontrol eden bir HRI sistemi tasarlamışlardır . İlk önce kafaya yerleştirilen elektrotlar vasıtasıyla EEG sinyallerini yükselteçten geçirerek bilgisayar ortamına almışlardır. Yazarlar daha sonra alınan EEG sinyallerindeki gürültülerden kurtulmak için Haar Wavelet filtresinden geçirmişlerdir. Gürültülerden arındırılmış sinyallerden LDA (Latent Dirichlet Allocation) algoritmasıyla EEG sinyallerini sınıflandırmışlardır. Sınıflandırılmış EEG sinyalleri robot kontrolörüne gönderilerek robota yukarı aşağı hareketlerini başarı ile yaptırabilmişlerdir.

Costa ve arkadaşları (2000) EEG sinyallerini sınıflandırma algoritmalarını yaşları 23-40 arası olan 5 kadın 5 erkek den oluşan 10 farklı denek üzerinde denemişlerdir. Kullanıcılardan alınan sinyallerden özellik çıkarımı için AGR (Genetic Algorithm) algoritmasını kullanmışlardır. Özellik çıkarımından sonra sınıflandırma işlemi için MLP (Multilayer Perception) algoritması kullanılarak sağa girme ve sola gitme işlemlerini sınıflandırmışlardır. Erkeklerden elde edilen verilerde sınıflandırma başarı oranı %91 kadınlardan elde edilen başarı oranı ise %87 olarak belirlemişlerdir.

Li ve arkadaşları (2014) EMG sinyallerini kullanarak exoskeleton robot kolunun kuvvet kontrolünü sağlamışlardır. Yazarlar bu çalışmada dirsek ve bilek kaslarından kas çifti belirlemişlerdir. Belirledikleri kas çiftinden dört kanallı EMG sinyallerini 1024 Hz ile örnekleyerek toplamışlardır. Topladıkları sinyallerden istenmeyen frekanstaki sinyalleri gidermek için sırasıyla 10 - 500 Hz arasında bant geçiren filtre , dördüncü dereceden Butterworth filtresi ve çentik filtresi uygulamışlardır. Daha sonra filtrelenmiş sinyal üzerinde PI (Proportional – Integral ) denetleyicisi uygulanarak kapalı çevrim kuvvet kontrolünü gerçekleştirmişlerdir

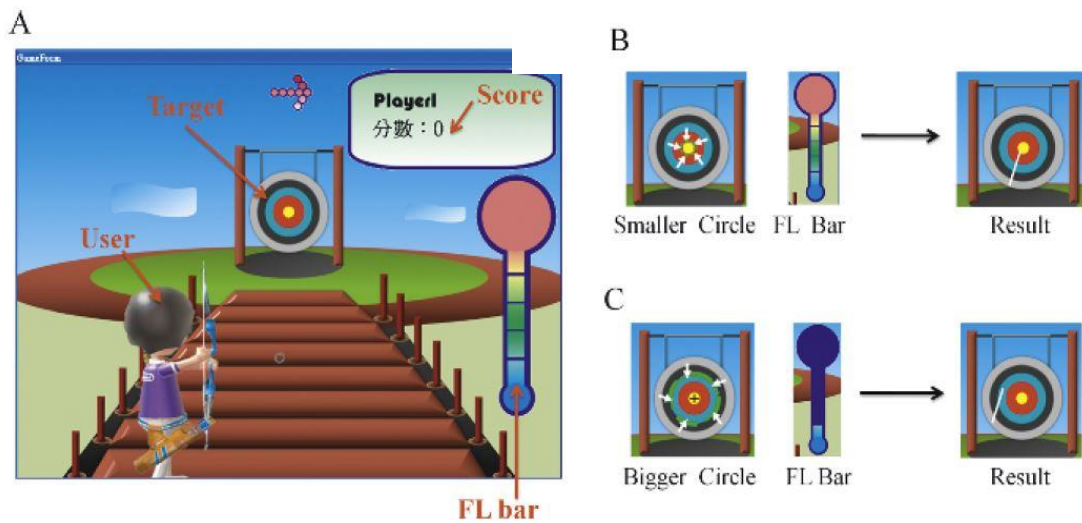
Kennedy ve arkadaşları (2000) özel elektrotlar sayesinde insanların sinir sinyallerini kayıt etmiş ve bilgisayarda sanal klavye uygulamasını kontrol etmeyi başarmışlardır. Yazarlar kayıt ettikleri sinyalleri TTL (Transistor to Transistor Logic) darbelerle çevirmiş ve darbe sinyallerini mouse ve klavye olarak kullanmışlardır. Sanal klavye olarak TalkAssist programını kullanmışlardır. TalkAssist programında TTL darbe sinyalleri ile mouse imlecini kontrol ederek harfler ile yazı yazdırmayı başarmışlardır. Harfi seçtikten sonra enter komutunu gönderince imleç tekrar en başa gelmiştir. Bu uygulamayı bazı hastalar üzerinde deneyerek uzun zaman sonra istenilen yazıları hatasız olarak yazdırabildiklerini gözlemlemişlerdir.

Li ve arkadaşları (2013) EEG ve EMG tabanlı tekerlekli sandalyenin kontrolünü sağlamışlardır. Yazarlar bu çalışmada sağ ve sol el kaldırmayı hayal etmişlerdir. Bu sayede EEG sinyallerini kullanarak tekerlekli sandalyeyi sağa ve sola hareket ettirmişlerdir. Ayrıca yazarlar sağ ve sol dişini gıcırdatarak elde ettikleri EMG sinyallerini ise aracın ileri gitmesi ve durma işlemini yapması için kullanmışlardır.

EEG ve EMG sinyallerinden hareket kestirimi yaptıktan sonra bilgisayardan motor sürücülerini kontrol etmek için CAN-BUS haberleşme protokolünü kullanmışlardır.

Lin ve arkadaşları (2014) araç kullanan insanların uyanıklık düzeyini gösteren bir mobil uygulama geliştirmişlerdir. Yazarlar EEG cihazı ile 15 katılımcı ve 90 dakika süren sürüş sırasındaki beyin sinyallerini toplamışlardır. Topladıkları sinyaller ilk önce 0.5-50Hz'lik bant geçiren filtreden geçirip gürültüleri gidermişlerdir. Daha sonra sinyalleri insan davranışını tespit etmek için SVR (Support Vector Regression) modellemesi yapmışlar ve sinyalleri insanın uyanıklık düzeyine dönüştürmüşlerdir. Ayrıca yazarlar JAVA programlama dilini kullanarak mobil uygulaması geliştirmişlerdir.

Liao ve arkadaşları (2012) EEG sensör tabanlı bir okçuluk oyunu geliştirmişlerdir (Şekil 2.8.). Yazarlar EEG sensör ile beyin sinyallerini okumuşlardır. Alınan sinyalleri ilk önce yükselticiden daha sonra 0.5-50 Hz arası bant geçiren filtreden geçirmişlerdir. Filtrelenen sinyal 12 bit çözünürlüklü AD8609 analog dijital çevirici ile MSP430 mikroişlemcinin içerisine eklemişlerdir. Okçuluk oyununun arayüzünde bir tane bar bulunuyor olup bu bar kişinin hedefe odaklanma seviyesini göstermektedir. Eğer hedef kısa ve odaklanma miktarı yüksek ise kişinin yüksek puan almasını sağlamışlardır.



Şekil 2.8. Liao ve arkadaşlarının (2012) geliştirdiği oyun

Osamah (2000) EMG tabanlı bir HMI sistemi gerçekleştirmişlerdir. Yazar bu çalışmada daha önce hiçbir kasa bağlı rahatsızlığı olmayan 3 sağlıklı erkek bireyden 4 haftalık süre içerisinde veri toplama kartı ile toplamıştır. Daha sonra alınan EMG verileri ilk önce 30Hz'lik yüksek geçiren filtreden sonrasında 250Hz'lik alçak geçiren filtreden ve en son 60Hz'lik bant geçiren filtreden geçirmişlerdir. Sonrasında elde ettikleri sinyalleri sınıflandırmışlar 12 farklı hareketten ve %96'lık başarılı oranı elde etmişlerdir.

Kang ve arkadaşları (2011) EMG ve ivmeölçer verileriyle insansı robotun navigasyonunu sağlamışlardır. 2 kanallı EMG sensöründen alınan verilerini robotun robotu başlatma ve durdurma olarak kullanmışlardır. İvmeölçer veriler ile de robotun navigasyonunu sağlamışlardır. Robotun navigasyonu için ileri, geri, sağ ve sol olmak üzere 4 farklı hareket belirlemişlerdir. Yazarlar sistemin çalışma oranını %99 olarak gözlemlemişlerdir.

James ve Huosheng (2012) EMG ve IMU sensör kullanarak EDUBOT isimli robot kolunu kontrol etmişlerdir. EMG verileriyle robota herhangi bir cismi tutma ve bırakma işlemini yaptırmışlardır. IMU sensöründen gelen verilerden yaw açısıyla robota sağ veya sol, pitch açısıyla da robota yukarı aşağı işlemlerini yaptırmışlardır.

Seong-Og ve arkadaşları (2014) EMG ve IMU sensörleriyle mobil robot platformunu kontrol etmişlerdir. Yazarlar IMU sensörleriyle kolun oryantasyonunu Euler açılarıyla tespit edip mobil robota ileri, geri, sağ ve sola dönme işlemlerini yaptırmışlardır. EMG verilerini Arduino kartıyla toplayıp Matlab programına seri port haberleşmesi ile yollamışlardır. Aldıkları EMG verilerini HMM (Hidden Markov Model) metoduyla sınıflandırıp 2 farklı hareketi tespit etmişlerdir. Bu hareketlerden yumruk hareketini robotu çalıştırma parmakları germe hareketini ise robotu durdurma olarak kullanmışlardır. Yazarlar sistemin başarılı bir şekilde çalıştığını gözlemlemişlerdir.

Guan-Chun ve arkadaşları (2015) giyilebilen bileklik şeklinde olan Myo Armband kullanarak mobil robotun navigasyonunu sağlamışlardır. Yazarlar ilk önce C++

programında oluşturdukları yazılım ile 8 kanallı EMG verisini yüksek başarı ile sınıflandırmışlardır. Yazarlar sınıflandırdıkları hareketlerden yumruk hareketini robotu durdurma, eli serbest bırakma hareketini robotu ileri doğru döndürme, bilek sağ hareketini robotu sağa doğru döndürme ve bilek sol hareketini ise robotu sola doğru döndürme olarak kullanmışlardır. Bu çalışmada yazarlar robotun navigasyonu için 2 farklı method belirlemişlerdir. 1. methodta robotu sadece kas hareketleriyle kontrol etmişler, 2. methodta ise robotu kas ve 9 eksenli IMU sensöründen gelen verilerle kontrol etmişlerdir. Her iki methodta da robotun başarılı bir şekilde hareket ettiği gözlemlenmişler fakat 2. methodta robotun hareket geçişlerinin daha yumuşak bir şekilde yaptığını tespit etmişlerdir.

Zhendong ve arkadaşları (2015) EEG sinyallerini kullanarak akıllı ev sistemini tasarlamışlardır. Yazarlar bu çalışma için EEG sensör içeren başlık kullanmışlardır. Bu EEG başlık 14 adet elektrottan oluşmaktadır. Çalışmada yazarlar ilk önce EEG sinyallerini sınıflandırmışlardır. Daha sonra sınıflandırılan EEG sinyalleri kablosuz haberleşme kullanılarak mikroişlemciye göndermişlerdir. Bu verilerle evde bulunan televizyon, ev ışıkları ve klimayı kontrol etmişlerdir. 5 denek üzerinde yapılan bu çalışmada yüksek başarı oranı elde etmişlerdir.

Mithileysh ve Sharanya (2016) giyilebilen teknoloji olan Myo Armband'ın fizik tedavi çalışmalarında doktoralar tarafından uygun olup olmadığını araştırmışlardır. Bu çalışmada 2 soru üzerinden gitmişlerdir. 1. soru Myo Armband'ın fizik tedavi amaçlı kullanılabilir olduğunu sistem kullanışlılık ölçeği adını verdikleri 24 denek üzerinden yaptıkları sonuçlarla memnuniyet oranını bulmuşlardır. 2. soru Myo Armband'ın sosyal hayatta kullanılıp kullanılmayacağını öğrenmeye çalışmışlardır. Bunların sonucunda sistemin bazı geliştirmeler yapılması sonucunda Myo Armband'ın hastalar üzerinde denenebileceği sonucunu elde etmişlerdir.

## **1.2. Tezin Literatüre Katkısı**

Bu çalışma kapsamında EMG ve IMU sensörleri kullanılarak bir insan-makine etkileşim ara yüzü geliştirilmiştir. Çalışmada kullanılan EMG sensör ve IMU

sensörlerinin beraber bulunduğu giyilebilir teknoloji ürünü olan Myo Armband kullanılmıştır. Kullanılan sensörün üretici firma tarafından ücretsiz olarak verilen yazılım geliştirme kiti kullanılarak farklı projelerde kullanılabilmesi en önemli avantajı olarak karşımıza çıkmaktadır. Ayrıca kullanılan sensörün kablosuz haberleşme ile verilerin toplanabilmesi ekstra avantajı olarak görülmektedir.

Çalışma kapsamında öncelikle EMG verileri ve ham IMU verileri bilgisayar ortamına alınmıştır. Python ortamında geliştirilen yazılım ile robotun hareketi için gerekli dönüşümler uygulanmıştır. Çalışmada robotun güvenli kullanımı amacıyla EMG bileklikten alınan hareketler ile bazı güvenlik önlemleri alınmıştır. Robotun ters bir pozisyona girmesi sonucu doğacak hatalar için çift parmak pozisyonuyla ana pozisyona gitmesi sağlanmış acil stop yapması için ise elin serbest bırakma hareketi kullanılmıştır.

Literatürde kullanılan çalışmaların çoğunda yüksek maliyetli sensörler kullanılmaktadır. Benzer şekilde EMG ve IMU sensörünün beraber kullanıldığı bir endüstriyel robot çalışması bulunmamaktadır. Bu çalışmada kablosuz ve maliyeti fazla olmayan sensörler tercih edilmiştir.

Geliştirilen deney düzeneğinde robotun hareketi sırasında bazı gecikmelerin olduğu gözlemlenmiştir bu gecikmeler tezin dezavantajı olarak görülmektedir. İlerleyen çalışmalarda gecikmelerin giderilmesi üzerine çalışılacaktır.



### **BÖLÜM 3. MATERYAL VE YÖNTEM**

İnsan makine arayüzü ve beyin bilgisayar arayüzü insanların bilgisayarları veya diğer elektronik cihazları, biyolojik sinyalleri kullanarak kontrol etmesini veya iletişim kurmalarını amaçlayan sistemler olarak tanımlanmaktadır. Beyin bilgisayar arayüzü beyin sinyallerini kullanarak sistem kontrol etmek amacı ile kullanırken, insan makine arayüzü genellikle kas hareketlerinden alınan sinyallerini kontrol yöntemi olarak kullanmaktadır.

İnsan makine arayüzü son zamanlarda popülerliği hızla artan uygulamaların başında gelmektedir. Özellikle robotik alanında yapılan kullanıcı dostu uygulamalarda sıklıkla kullanılan sistemler olarak kullanılmaktadır. EMG, EEG, EOG gibi canlı vücutundan alınan sinyaller anlamlı hale getirilerek ve sinyal işleme teknikleri ile sınıflandırılarak anlamlandırılmakta bu sinyaller makinalara aktarılmaktadır.

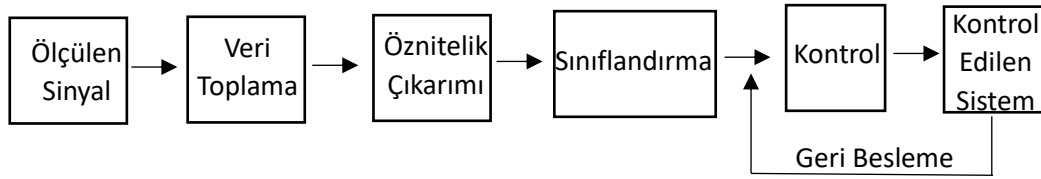
İnsan makine arayüzü veri toplama, öznitelik çıkarımı, sınıflandırma ve kontrol aşamaları olarak dört aşamadan oluşmaktadır.

Veri toplama işlemi bilgisayarla insan vücutundan sensörler vasıtasıyla gerilim, sıcaklık, basınç ve ses gibi elektriksel veya fiziksel niteliklerin ölçülmesi işlemidir. Veri toplama sistemi donanım ve yazılım içeren bir bilgisayardan ya da gömülü sistem donanımından oluşmaktadır.

Öznitelik çıkarımı işlemi boyut azaltma yöntemi olarak bilinmektedir. Buna göre karmaşık olan bir verinin boyutları azaltılarak daha basit bir problem haline indirgenir. Öznitelik çıkarımı genellikle zaman alanı, frekans etki alanı ve zaman ölçeği olmak üzere üç kategoride incelenir.

Sınıflandırma, öznitelik çıkarımı yapılan sinyallerin hangi fiziksel hareket sonucu ortaya çıkarıldığını tespit edilmesi için yapılan işlemdir. Sınıflandırma işlemi sonunda hangi eylemin gerçekleştirildiği kararlaştırılır.

Kontrol işlemi sınıflandırma işlemi sonucunda belirlenen fiziksel hareketin kontrol edilen sistemin aktüatörlerine komutlar göndererek istenilen görevi yerine getirmesi işleminden oluşmaktadır.



Şekil 3.1. İnsan makine arayüzü sinyal analizi

### 3.1. EMG İşareti

Sinir sistemi vücudun hem kontrol hem de haberleşme ünitesidir. Sinir sistemi elektrik sinyaller vasıtasıyla vücudun farklı kısımları arasında iletişim kurmayı sağlayan çok sayıda nörondan oluşmaktadır. Nöronlar vücudun bir bölümünden diğerine darbeler şeklinde elektriksel mesaj ileten özel yapılı hücrelerdir.

Kas hücreleri kasılma ve gevşeme özelliğine sahip hücre demetlerinden oluşmaktadır. Kas hücreleri kasılma ve gevşeme özelliklerinin yanı sıra uyarıcıları alma ve bunlara cevap verme özelliğine de sahiptir (Reaz ve ark.,2006). EMG sinyali ise kasların kasılması ve gevşemesi ile ortaya çıkan elektriksel sinyaller olarak tanımlanmaktadır. Tıp alanında kullanılan EMG sinyalleri kasın durumu veya sinirsel bozukluklarının teşhisinde kullanılırken, mühendislik alanında hareket kontrol için fiziksel sistemlere giriş amacıyla kullanılmaktadır. Günümüzde EMG sinyalleri genellikle robotik kol, el veya robotik engelli sandalyeler yapımında kullanılmaktadır. EMG sinyallerinin kullanılma amacı sinirsel bozukluk yaşayan insanlara yardım edebilecek sistemleri geliştirebilmektir.

EMG sinyalleri kasların kasılmasına dayanan bir sinyal olduğu için diğer biyoelektriksel sinyallere göre yorumlanması daha kolay olmaktadır.

EMG sinyali 10Hz ile 5000Hz arasında değişebilen oldukça yüksek frekanslı bileşenler içermektedir. Bu sebepten EMG sinyali ölçen cihazların bu frekans aralığındaki frekansları ölçebilen cihazlar olması gerekmektedir.

EMG sinyalleri kaslardan yüzey veya iğne tip elektrotlar yardımıyla alınmaktadır ve sinyallerinin ölçülmesi için tüm kaslardan veri alınması mümkün olmadığından belirli bir kas demetinden sinyallerden alınmaktadır. Alınan EMG sinyallerinin genlikleri tepeden tepeye yaklaşık 0mV ile 50mV arasında ölçülmektedir. EMG sinyallerini genlikleri çok küçük olduğundan bu sinyallerin işlenebilmesi için sinyaller ilk önce elektriksel yükseltgeçlerden geçirilmektedir. Daha sonra gürültülerden arındırılması için yazılımsal veya donanımsal filtreleme işlemi yapılmaktadır. Filtreden geçirilen EMG sinyallerine anlam kazandırılması için Şekil 3.1.'de gösterilen sinyali analizi sürecinden geçirilmesi gerekmektedir.

### 3.2. Myo Armband

Bu tez çalışmasında EMG sinyallerini ölçmek için 1 adet EMG bileklik kullanılmıştır. Şekil 3.2.'de kullanılan EMG bileklik gösterilmiştir.



Şekil 3.2. Myo Armband (<https://www.myo.com>, 2017)

Şekil 3.2.'de ki Myo Armband Thalmic Labs firması tarafından üretilen düşük enerjili kablosuz bağlantı (bluetooth) ile çalışmaktadır. Myo Armband üzerinde 8 EMG elektrot, 3 eksenli ivmeölçer, 3 eksenli jiroskop ve 3 eksenli manyetik kuvveti ölçen manyetometre bulunmaktadır. Kola takılan bu bileklik ayrıca kolun 3 boyutlu uzayda oryantasyonunu gösterebilmektedir. Thalmic Labs firması EMG bileklikle beraber proje geliştirebilmek amacıyla ücretsiz olarak yazılım geliştirme kitini kullanıcılara sunmaktadır. Yazılım geliştirme kitinin kütüphanesi Python, C++, C# gibi derleyicilere eklendikten sonra EMG bileklikten veriler alınıp kullanıcılar kendi uygulamalarını geliştirebilmektedirler. Thalmic Labs firması tarafından kullanıma sunulan arayüz ile Şekil 3.3.'de ki 5 farklı el hareketi tanımlanmıştır. Kullanıcılar isterlerse ham EMG verilerini kullanarak sınıflandırma yöntemleriyle farklı hareketlerde tanımlayabilmektedirler.



Şekil 3.3. Thalmic Labs tarafından tanımlanmış hareketler (<https://www.myo.com>, 2017)

Bu çalışmada Myo Arm Band tarafından 3 farklı hareket ve ham oryantasyon verileri yazılım geliştirme kiti kullanılarak Python ortamına aktarılmıştır. Python ortamında alınan jiroskop verileri 20Hz örnekleme frekansıyla örneklendirilmiştir. Python ortamında geliştirilen yazılım ile matematiksel dönüşüm fonksiyonları kullanılarak oryantasyon verileri gerçek zamanlı kontrol işaretlerini üretmek için anlamlı hale getirilmiştir. Myo Armband tarafından verilen 3 farklı hareketten yumruk hareketi robotu gerçek zamanlı çalıştırma, eli serbest bırakma hareketi robotu durdurma ve çift parmak dokunma hareketi robotu daha önce belirlenen ev pozisyonuna göndermek için kullanılmıştır.

### 3.3. IRB120 Endüstriyel Robot

Bu bölümde tez kapsamında kullanılan endüstriyel robot hakkında bilgi verilecektir. Gerçek zamanlı kontrolün gerçekleştirilebilmesi için ABB firmasının ürettiği IRB120 6 eksenli endüstriyel robotu kullanılmıştır. Kullanılan endüstriyel robot Şekil 3.4.'de gösterilmiştir.



Şekil 3.4. IRB120 endüstriyel robotu ([www.new.abb.com/products/robotics/tr](http://www.new.abb.com/products/robotics/tr), 2017)

IRB120 endüstriyel robotun teknik özellikleri Tablo 3.1.'de verilmiştir.

Tablo 3.1. IRB120 robotunun özellikleri (ABB robotics product specification IRB120, 2017)

Parametre	Değeri
IRB120 Ölçüleri	180x180x70mm
Ağırlığı	25kg
Eksen Sayısı	6
Taşıma Kapasitesi	3kg
Besleme Gerilimi	200/600V, 50/60Hz
Güvenlik	Güvenlik çemberi ve Acil Stop Butonu
Kontrolör	IRC5 Compact

IRB120 robotunu kullanmak için ABB firması tarafından geliştirilen Flex Pendant kumandası veya yine ABB firması tarafından geliştirilen Robot Studio bilgisayar geliştirme arayüzü kullanılmaktadır. Endüstriyel robotu programlamak için RAPID programlama dili bilgisi gerekmektedir. RAPID programlama dili ABB firması tarafından hazırlanan ve kullanılan, robotu kontrol etmek için gerekli komutları sağlayan orta seviyeli bir programlama dilidir. Bu çalışmada IRB120 robotu kontrol etme yöntemi olan Flex Pendant ve Robot Studio programları yerine, Python ortamında geliştirilen yazılım ile kontrol edilmiştir. Python ortamında geliştirilen yazılım ile ilk önce Myo Armband'tan kablosuz haberleşme ile alınan EMG ve jiroskop verileri işlenip daha sonra TCP/IP haberleşmesi ile IRB120 robotuna gerçek zamanlı olarak gönderilmiştir.

### **3.4. RAPID Programlama**

Programlama dilleri programcının belli bir algoritmayı donanımlara işletebilmek için standartlaştırılmış komutlardan oluşan notasyondur. Programcı program yazabilmek için çeşitli programlama dillerini kullanabilir. Şimdiye kadar yaklaşık 3000 tane programlama dili geliştirilmiştir. Programlama dilleri kullanma kolaylığına göre 3'e ayrılmaktadır.

Temel seviye programlama dilleri makine koduna yakın programlama dilleridir. Anlaşılması kısmen zor olan bu programlama dilinde kişinin kullanılan işlemci mimarisini iyi bilmesi gerekir (Assembly dili). Orta seviye programlama dilleri temel seviye programlama diline göre daha kolay ve daha kullanışlı olan programlama dilidir. Hem üst hem de alt seviye programlama yapabilirler (C++,Python programlama dili vb.). Yüksek seviye programlama dilleri orta seviyeli programlama dillerine göre daha esnek ve daha kolay anlaşılır programlama dilleridir. Bu programlama dilinde programlar programcıya fonksiyonlar şeklinde verildiğinden kullanıcının adapte olması daha kısa olmaktadır (Visual Basic, Pic Basic programa dilleri vb.).

Temel seviye programlama dillerinin insanlar tarafından anlaşılması görsel bakımdan oldukça zordur. Bilgisayar programcıları yüksek seviyeli programlama dillerini gerçekleştirerek karmaşık olan programlama dillerini basite indirgemişlerdir. RAPID programlama dili ABB Robotics firması tarafından geliştirilip kullanılmakta olan basit İngilizce terimler kullanan yüksek seviyeli bir programlama dilidir.

### 3.4.1. RAPID verileri

#### 3.4.1.1. Değişkenler

RAPID programlama dilinde birden çok veri türü bulunmaktadır. Bunlardan en çok kullanılanları Tablo 3.2.'de verilmiştir.

Tablo 3.2. RAPID değişkenleri (Operating Manual Introduction to RAPID, 2017 )

Veri Türü	Aldığı Değerler
Num	Sayısal İfadeler
String	Metinsel İfadeler
Bool	Mantıksal İfadeler

RAPID programlama dilinde geri kalan data tipleri bu üç temel data tipi üzerine kurulmuştur. Bu üç temel data tipini anlamak diğer data tiplerini anlamayı kolaylaştırmaktadır.

RAPID programlama dilinde değişkenlere değer atamak için VAR kelimesini kullanmaktayız. Aşağıda değişkenlere değer atama ile ilgili örnekler verilmektedir.

VAR num Pozisyon;

VAR string Isim;

VAR bool Onay;

Değişken isimlerini belirledikten sonra RAPID programlama dilinde atama operatörü olarak “:=” atama sembolü kullanılmaktadır. Atama sembolünün anlamı sembolün sağındaki ifadeyi solundaki ifadeye aktarmaktır. Örneğin ;

Pozisyon := 1;

Isim := “ahmet”;

Onay := TRUE;

ifadeleriyle atama işlemi gerçekleştirilmektedir.

#### **3.4.1.2. RAPID akış kontrol terimler**

Program kontrol terimleri her programlama dilinde kullanılan ve koşullara göre programın akışını değiştiren – belirleyen terimlerdir. Bu tez çalışmasında RAPID programlama dilinde kullanılan bazı kontrol akış ifadeleri aşağıda açıklanmıştır.

IF THEN ifadesi mantıksal olarak doğru olduğu zaman THEN ve ENDIF arasındaki programları çalıştıran bir akış kontrol ifadesidir. IF THEN komutunun kullanılmasıyla ilgili örnek aşağıda verilmiştir.

```
IF koşul THEN  
program  
ENDIF
```

WHILE ifade mantıksal olarak doğru olduğu sürece WHILE ve ENDWHILE arasındaki programları çalıştıran akış kontrol ifadesidir. WHILE komutunun kullanılmasıyla ilgili örnek aşağıda verilmiştir.

```
WHILE koşul DO  
program  
ENDWHILE
```



### 3.4.1.3. RAPID hareket komutları

RAPID dili yüksek seviyeli dil olmasının yanı sıra robotları kontrol etmek için özel olarak tasarlanmış olması önemli bir avantajdır. RAPID programlama dilinde üç farklı türde hareket komutu bulunmaktadır.

MoveL hareket komutu robotu o anki pozisyonundan istenilen pozisyona lineer olarak hareket ettirilmesini sağlayan komuttur. Robot kullanılan uç işlemcisini bu iki nokta arasında çizilen doğru boyunca hareket ettirir. MoveL komutunun kullanım şekli aşağıdaki ifadede verilmiştir.

MoveL ToPoint, Speed, Zone, Tool;

1. ToPoint; robotun mevcut pozisyonundan gideceği hedef pozisyondur
2. Speed; robotun hedef pozisyona doğru olan hareket hızıdır.
3. Zone; robotun hedef pozisyona hangi toleransla nasıl yaklaşacağını belirtir.
4. Tool; robotun flanşında iş yapmak amacı ile kullanılan bir eleman varsa bunun bilgisini belirtir.

MoveJ hareket komutu robotu mevcut pozisyonundan istenilen pozisyona ulaşmak için eksenlerini rahat bir yapmasını sağlayan komuttur. Eksenlerin hareketi lineer harekete göre daha rahattır. Bu hareket türünde robotun izleyeceği yol kestirilemez.

MoveC hareket komutu robotu mevcut pozisyonundan belirtilen bir ara noktadan geçerek istenilen noktaya doğru daire çizerek hareket ettirilmesini sağlayan komuttur.

IRB120 robotunun istediğimiz pozisyona gidebilmesi için veri tipinin oluşturulması gerekmektedir. RAPID programlama dilinde bu veri tipi robtarget data olarak verilmektedir. Komutlarda ToPoint şeklinde ifade edilen bilgi bir robtarget veridir. Robtarget data içerisinde dört farklı veri saklayan bir veri paketidir. Robtarget veri tipi Tablo 3.3.'de gösterilmiştir.

Tablo 3.3. Robtarget deęişkeni paket yapısı (Operating Manual Introduction to RAPID, 2017 )

Data Tipi	İsim	Açıklama
Pos	Trans	x,y,z koordinatları
Orient	Rot	Oryantasyon
Confdata	Robconf	Robot Eksen Açılıarı
ExtJoint	Extax	Ek Eksen Takımı

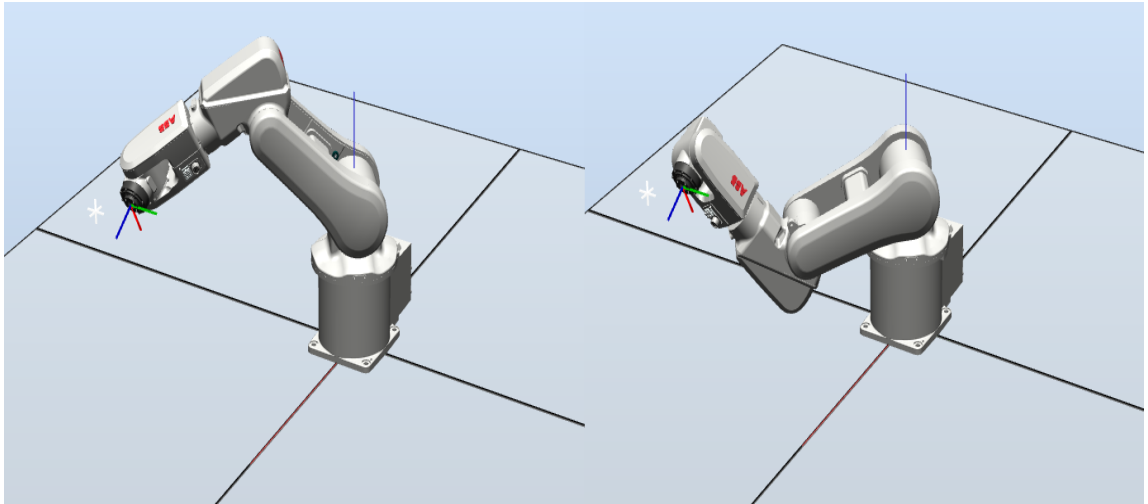
Bu çalışmada RAPID programlama dilini kullanarak IRB120 robotu EMG bileklikten gelen oryantasyon verileriyle hareket ettirilmesi, yeni konuma gönderilmesi amaçlanmıştır. İlk önce RAPID dilinde boş bir robtarget veri tipi oluşturulmuştur. Daha sonra EMG bileklikten 20Hz örnekleme frekansıyla alınan oryantasyon verileri ile sürekli olarak robtarget datasının x,y,z koordinatları güncellenmiştir. Güncellenme işlemi yapıldıktan sonra TCP/IP haberleşme protokolü ile veriler sürekli olarak IRB120 robotuna ethernet haberleşme protokolü vasıtasıyla gönderilmiştir. Koordinatları güncelledikten sonra koordinatlar arası hareket MoveL komutu ile yaparak robotun lineer bir şekilde koordinatlar arasında geçişi sağlanmıştır.

### 3.5. RobotStudio Programı

RobotStudio programı ABB tarafından üretilen tüm endüstriyel robotların programlanmasını ve CAD programlarında tasarlanan fikstürlerin sanal ortamda kullanılmasını sağlayan bir bilgisayar uygulamasıdır. RobotStudio aynı zamanda kullanıcılara çevrimdışı çalışma imkanı sunmaktadır. Bu çevrimdışı çalışma robota bağlandığı zamanda çevrimiçi mod olarak çalışabilmektedir.

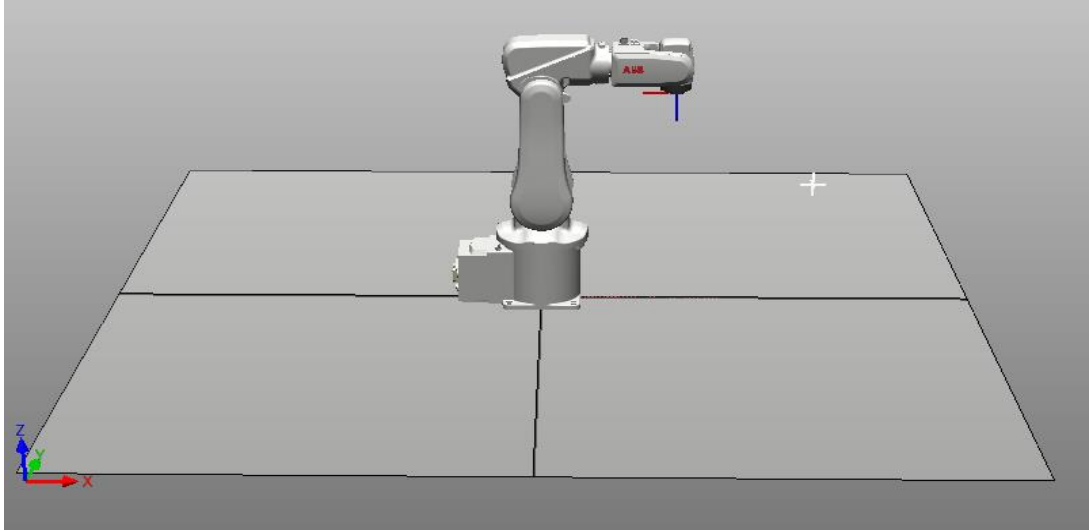
IRC5 ABB tarafından üretilen ve robotların eksenlerini ve IO kartından gelen çevresel sinyalleri kontrol etmek için kullanılan elektronik aksamları ve donanımları kumanda eden temel yazılımları (firmware) barındıran bir kontrol ünitesidir

RobotStudio programında proje yapmadan önce sistemin CAD dosyalarını ve kullanacağımız robotları ABB kütüphanesinden seçip proje dosyasına eklememiz gerekmektedir. RobotStudio ortamında gerçek hayatta yapacağımız projeyi tüm ayrıntılarıyla görsel olarak gerçekleştirip simülasyonlarını gerçekleştirebiliriz. Sistemin görsel olarak tasarımından sonra robotların çalışma alanlarının dışına çıkmadan ve robotun eksenlerinin tekil (singularity) duruma düşmeden robotların hareket planlamalarını yapabiliriz. Robot koordinatları belirlenirken robot gideceği konuma birden fazla şekilde ulaşabileceğinden robot konfigürasyonu sağlanmalıdır. Robot konfigürasyonu robotun gideceği her hareket koordinatı için robotun o koordinata ulaşırken dairesel eksenlerinin mekanik dizilimini belirlemektir. RobotStudio ortamında IRB120 robotunun tek bir noktaya birden fazla mekanik dizilimle nasıl gideceği Şekil 3.5.'de gösterilmiştir.



Şekil 3.5. IRB120 robotunun bir noktaya birden fazla mekanik dizilimle gitmesi

Çalışma ilk önce çevrimdışı program ile RobotStudio ortamında kontrol edilmiştir. Çevrimdışı çalışmada RobotStudio ortamında IRB120 robotuna sanal kontrolör ekleyerek EMG bileklikten alınan oryantasyon verileri robota gönderilerek robotun gerçek zamanlı çalışması gözlemlenmiştir. Gerçek zamanda TCP/IP haberleşmesinin çalışması için robotun yazılımsal olarak PC Interface özelliğinin kendi kontrolöründe bulunması gerekmektedir. PC Interface özelliği ABB firması tarafından haberleşme için sağlanan opsiyonel bir özelliktir. RobotStudio programında IRB120 endüstriyel robotu aşağıdaki şekilde gösterilmiştir.



Şekil 3.6. RobotStudio programı IRB120 robotu

### 3.6. TCP/IP Haberleşme Protokolü

TCP (İletim Kontrol Protokolü) ve IP (İnternet Protokolü) terimlerinin birleştirilmiş hali olarak gösterilmektedir. TCP'nin görevi verilerin eksiksiz biçimde iletilmesini sağlamak IP'nin görevi ise veri adreslerinin doğruluğunu kontrol etmektir (Şirin ve ark, 2015). TCP/IP haberleşmesinde haberleşecek cihazlardan birisi diğerine veriyi paketler halinde göndermektedir. Gönderen bilgisayarın gönderdiği paketlere IP paketler, adresine ise IP adresi denir. Gönderilen paketler içerisinde iletilecek bilgisayarın adresi olduğu için paketler istenilen adrese yönlendiriciler vasıtasıyla gönderilmektedir (Timur, 2005).

TCP/IP haberleşme protokolünde veriler paketler halinde alıcıya yollanır ve alıcı paketleri alarak işlem yapmaktadır. TCP/IP haberleşmesinde 4 katman mevcuttur bu katmanlar; uygulama, taşıma, internet ve ağ arayüzü katmanlarıdır.

Uygulama Katmanı: Bu katmanda veriyi göndermek isteyen uygulama bulunarak gönderilen verinin çeşidine göre farklı protokoller çalıştırılır (HTTP, SMTP, Telnet vb.) ve programlarla taşıma protokollerinin haberleşmesi sağlanır.

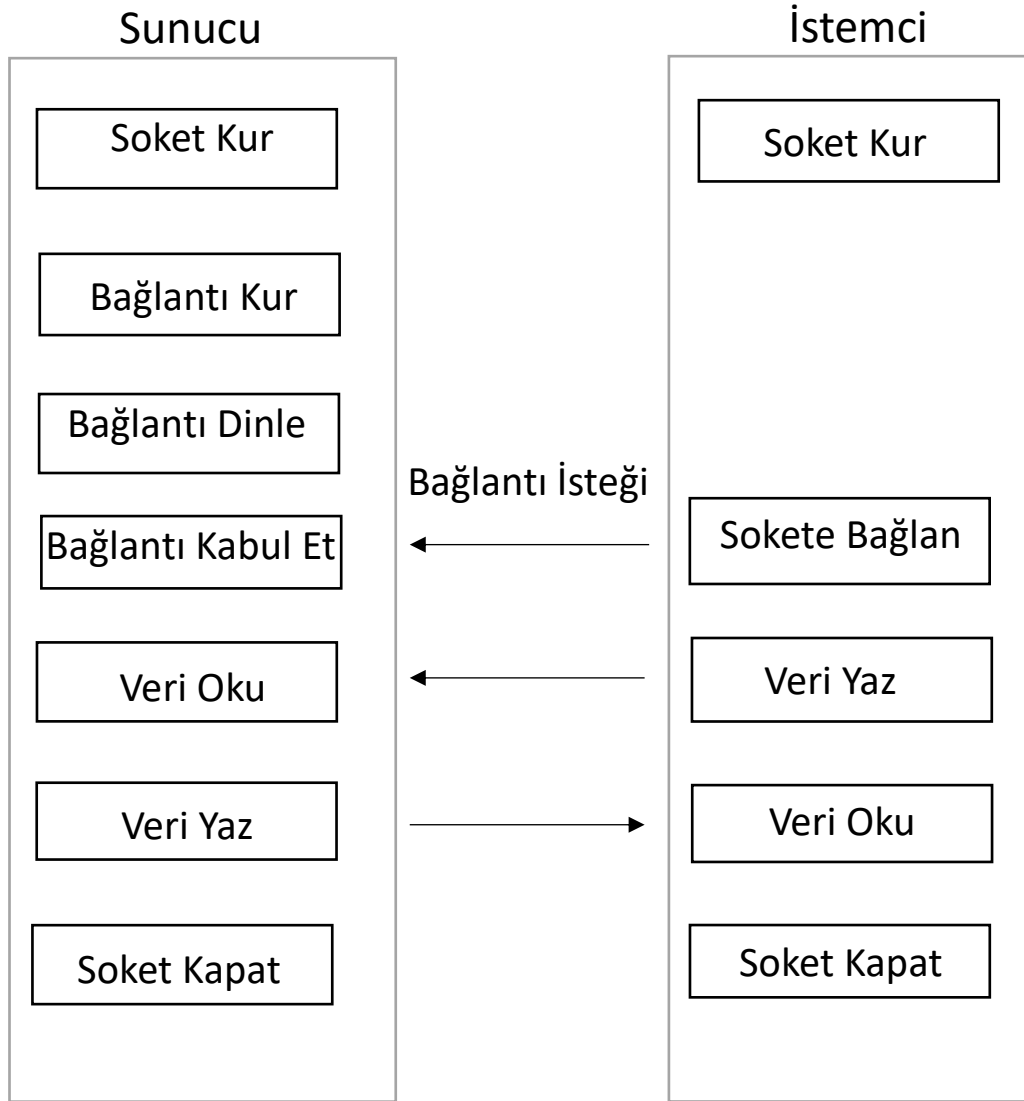
**Taşıma Katmanı:** Bu katmanda verinin nasıl gönderileceğine karar verilir. Alıcı tarafından alınan veriler için bir TCP alındı bilgisi gönderilir ve iletilen paketlerin doğruluğu kontrol edilir. Gönderici kabul mesajı gelmediği sürece paketi tekrar tekrar gönderir ve bu şekilde kayıpsız veri transferi gerçekleştirilmiş olur.

**İnternet Katmanı:** Bu katmanda gönderici ve alıcı IP adresleri veriye ilave edilerek verinin hangi adrese gideceği belirlenir.

**Ağ Arayüzü Katmanı:** Bu katmanda verinin ethernet üzerinden gönderilirken alacağı bir ve sıfır yapısı belirlenir.

TCP/IP haberleşme protokolü sunucu ve istemci arasında metinsel veri alışverişini sağlayan bir haberleşme protokolüdür. Haberleşme sırasında ilk önce sunucu kendisine gelen tüm verileri dinlemektedir. Daha sonra istemci bağlanma isteğini sunucuya iletmektedir. Sunucu istemcinin bağlanma isteğinin kabul ettikten sonra veri alışveriş olayı gerçekleşmektedir.

Çalışmamızda EMG bileklikten alınan veriler Python arayüzünde işlenerek TCP/IP haberleşmesiyle IRB120 robotuna ethernet üzerinden gerçek zamanlı olarak gönderilmiştir. IRB120 ve Python arasında kurulan ethernet haberleşmesinin akış şeması Şekil 3.6.'de gösterilmiştir.



Şekil 3.7. Sunucu ve istemci arasındaki TCP/IP haberleşmesi

### 3.6.1. RAPID programlama TCP/IP haberleşmesi

RAPID programlama dilinde TCP/IP haberleşmesi soket mesajlaşması ile gerçekleşmektedir. Soket mesajlaşmasının amacı 2 cihaz arasındaki veri alışverişini sağlamasıdır. Soket mesajlaşmanın temeli Berkeley Software Distribution Unix firmasına aittir. Berkeley Software firmasının yanı sıra Microsoft Windows firmasında soket mesajlaşmasını desteklemiştir.

IRB120 endüstriyel robotunun soket haberleşmesi ile veri alabilmesi için robotun yazılımsal olarak bu opsiyonu desteklemesi gerekmektedir. Robotun bu opsiyonu olduktan sonra RAPID programlama dili soket haberleşmesi için bazı veri tipleri, komutlar ve fonksiyonları kullanıcıya sunmaktadır.

RAPID programlamada soket haberleşmesi için gerekli aşamalar şu şekilde özetlenebilir;

1. Sunucu ve istemci taraflarında soketin kurulması
2. Sunucu tarafında SocketBind ve SocketListen komutlarıyla bağlantı kurulması
3. Sunucunun gelen isteklere cevap vermesi
4. İstemciden bağlantının talep edilmesi
5. İstemci ve sunucu arasında veri alışverişinin gerçekleşmesi

Soket mesajlaşması yaparken dikkat edilmesi gereken bazı önemli noktalar bulunmaktadır. Bunlardan en önemlisi sunucu ve istemci arasında veri alışverişini yaparken birbiri ardına gelen veri paketleri bazen üst üste çakışabilmektedir. Bu veri çakışmasının önüne geçebilmek için her veri paketinden sonra bir alındı bilgisinin iletilmesi gerekmektedir. Bu çalışmamızda tek yönlü bir veri iletişimi olduğundan sunucu verileri alırken aldığı her veri paketi için istemciye alındı bilgisi göndermektedir. Örneğin istemci birinci veriyi yolladıktan sonra sunucudan alındı bilgisi beklemektedir. Alındı bilgisi sunucudan gönderilince istemci ikinci veriyi göndermektedir.

RAPID programlamada soket haberleşmesi için bazı komut ve fonksiyonlar bulunmaktadır. Soket programlama da kullanılan veri tipleri Tablo 3.4.'de gösterilmektedir

Tablo 3.4. Soket programlama veri tipleri (Operating Manual Introduction to RAPID, 2017 )

<b>Data Tipi</b>	<b>Açıklama</b>
Socketdev	Cihazlarla haberleşme için kullanılan değişken
Socketstatus	Socketdev değişkeninin durumunu sorgular

RAPID programlamada istemci için kullanılan bazı komut ve fonksiyonlar Tablo 3.5.'de gösterilmektedir.

Tablo 3.5. Soket programlama istemci komutları (Operating Manual Introduction to RAPID, 2017 )

<b>Komut</b>	<b>Açıklama</b>
SocketCreate	Socketdev değişkenini kullanarak soket kurmak için kullanılır
SocketConnect	İstemcinin sunucuya bağlanmasını sağlar
SocketSend	Diğer cihaza veri göndermek için kullanılır. Gönderilen verilerin karakter veya metin olması gerekmektedir.
SocketReceive	Diğer cihazdan veri almak için kullanılır. Alınan verilerin karakter veya metin olması gerekmektedir.
SocketClose	Soketi kapatmak için kullanılır

RAPID programlamada sunucu için kullanılan bazı komut ve fonksiyonlar Tablo 3.6.'da gösterilmektedir.



Tablo 3.6. Soket programlama sunucu komutları (Operating Manual Introduction to RAPID, 2017 )

Komutlar	Açıklama
SocketBind	Sunucu üzerinde belirli port numarasına göre soketi bağlamak için kullanılır
SocketListen	SocketBind ile belirtilen bağlantı noktalarını dinlemek için kullanılır
SocketAccept	Sunucu tarafından istemcinin bağlanma talebini kabul etmek için kullanılır

Bu çalışmada Python ile RAPID dili arasında soket mesajlaşması yapılmıştır. Soket mesajlaşması için veri alışverişinin metin veya karakter olarak yapılması gerekmektedir. EMG bileklikten alınan veriler Python ortamına tam sayı olarak gelmektedir. Python ortamına gelen tam sayılar metin veya karakter olarak dönüştürülüp soket haberleşmesi ile IRB120 endüstriyel robotuna gönderilmektedir. Örneğin robotun  $x= 100$ ,  $y= 150$  ve  $z= 200$  noktasına gitmesini istiyorsak gönderilmesi gereken soket mesajı  $[100,150,200]$  olması gerekmektedir.

IRB120 robotuna gelen bu metin verisini robotun algılaması için tekrar tamsayıya çevrilmesi gerekmektedir. RAPID programlama dilinde kullanılan StrToVal komutu metin veya karakter dizisi tamsayıya çevirmektedir. Örneğin;

```
VAR string pozisyon := "[100,150,200]";
VAR bool onay ;
VAR pos yenipozisyon;
onay := StrToVal (pozisyon,yenipozisyon)
```

Şeklide bir kod parçasında metin olarak verilen pozisyon değişkeni tam sayıya çevrilerek yeni pozisyon değişkenine gönderilmektedir.

### 3.7. Kuaterniyon Matrisi Ve Euler Açılımları

Kuaterniyon matrisi 4 boyutlu bir matris olup cismin 3 boyutlu uzayda oryantasyonunu belirtmek için kullanılır. Kuaterniyon matrisi ilk olarak William Roman Hamilton tarafından 1843 yılında keşfedilmiştir. Kuaterniyon matrisinin diğer oryantasyon gösterimlerinden en önemli farkı tekillik noktalarının olmaması ve hesaplama yükünün daha az olması gösterilmektedir. Kuaterniyon matrisi bu özelliklerinden kaynaklı olarak robotik sistemlerde ters kinematik ve uçuş kontrol probleminin dinamikleri hesaplanırken efektif olarak kullanılmaktadır. Kuaterniyon matrisinin genel gösterimi aşağıdaki denklem 3.1’de gösterilmiştir.

$$q = [ q_0 \ q_1 \ q_2 \ q_3 ] \quad (3.1)$$

Denklem 3.1’de gösterilen kuaterniyon matrisi Myo Armband tarafından verilen ve insan kolunun 3 boyutlu uzaydaki konumunu gösteren matristir.

#### 3.7.1. Kuaterniyon matematiği

Kuaterniyon sayı sistemi geniş ve karmaşık bir matematik sistemi içermektedir. Bu kısımda kuaterniyon matematiğinin kısaca bahsedilecektir.

Kuaterniyon matrisinin formu denklem 3.2’de gösterilmiştir.

$$|q| = \sqrt{q_s^2 + q_x^2 + q_y^2 + q_z^2} \quad (3.2)$$

Kuaterniyon normu  $|q| = 1$  olan sayılara birim kuaterniyon denilmektedir. Bir kuaterniyonun normalizasyonu kendisinin normuna bölünmesiyle elde edilmektedir.

Normalizasyon denklemi denklem 3.3’de gösterilmiştir.

$$\|q\| = \frac{q}{|q_s|} \quad (3.3)$$

Bir Kuaterniyon matrisinin eşleniği denklem 3.4' de gösterilmiştir.

$$q^* = [q_s \ -q_x \ -q_y \ -q_z] \quad (3.4)$$

Bir kuaterniyon matrisinin tersi aşağıdaki denklemde gösterilmiştir.

$$q^{-1} = \frac{q^*}{|q|} \quad (3.5)$$

$q_1$  ve  $q_2$  isimli iki kuaterniyon matrisinin çarpımı aşağıdaki denklemde gösterilmiştir.

$$q_1 * q_2 = q = \begin{bmatrix} s \\ \vec{v} \end{bmatrix} = \begin{bmatrix} s_1 \cdot s_2 - \vec{v}_1 \times \vec{v}_2 \\ s_1 \cdot \vec{v}_2 + s_2 \cdot \vec{v}_1 + \vec{v}_1 \times \vec{v}_2 \end{bmatrix} \quad (3.6)$$

### 3.7.2. Euler açıları

Rijit bir cismin yönelimini bir koordinat sistemine göre sabit bir eksen etrafında ardışık üç parametre ile açıklayan açılara Euler açıları denilmektedir. Euler açıları kuaterniyon matrislerine göre yorumlanması daha kolay olmakla beraber robot hesaplamalarında robotun tekil noktalarından kaynaklı olarak kullanılması dezavantaj olarak gösterilebilmektedir. Euler açıları roll, pitch ve yaw olmak üzere 3 parametre ile ifade edilmektedir.

Çalışmamızda EMG bileklikten EMG verileri dışında kolun 3 boyutlu uzayda yönelim vektöründe gerçek zamanlı olarak elde edilebilmekteyiz. EMG bileklikten alınan yönelim vektörü bize 4 elemanlı kuaterniyon matrisi olarak verilmektedir. Kuaterniyon matrisini robotu hareket ettirmek için euler açılarına dönüştürülmesi gerekmektedir. EMG bileklikten alınan veriler 20hz ile Python ortamına alındıktan sonra matematiksel dönüşüm fonksiyonları ile Euler açılarına dönüştürülmüştür. Kuaterniyon matrisinden Euler açılarına dönüşüm denklemleri denklem 3.7, 3.8 ve 3.9'da gösterilmiştir.

$$\text{roll} = \tan^{-1} \left( \frac{2 \cdot (q_1 \cdot q_2 + q_0 \cdot q_3)}{q_3^2 + q_2^2 - q_1^2 - q_0^2} \right) \quad (3.7)$$

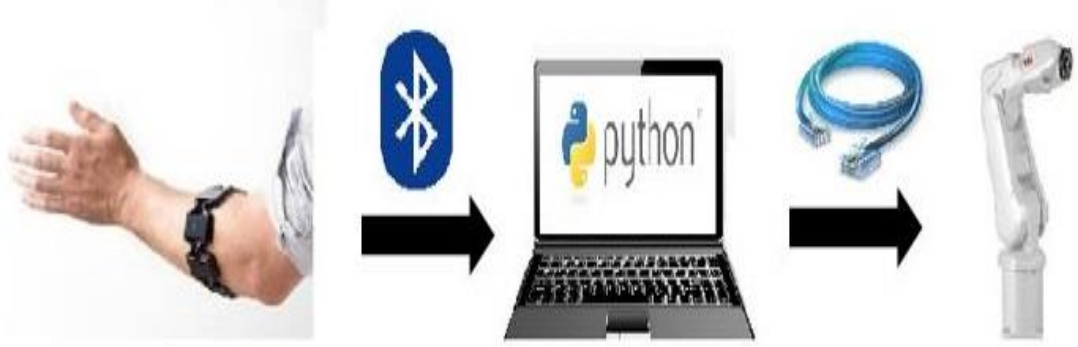
$$\text{pitch} = \sin^{-1}(-2 \cdot (q_0 \cdot q_2 - q_1 \cdot q_3)) \quad (3.8)$$

$$\text{yaw} = \tan^{-1} \left( \frac{2 \cdot (q_0 \cdot q_1 + q_3 \cdot q_2)}{q_3^2 - q_2^2 - q_1^2 + q_0^2} \right) \quad (3.9)$$

Dönüşüm denklemleri elde edildikten sonra robotu 3 ekseninde gerçek zamanlı kontrol etmek için roll, pitch ve yaw açılarından robotun pozisyon denklemlerine dönüştürülmesi gerekmektedir.

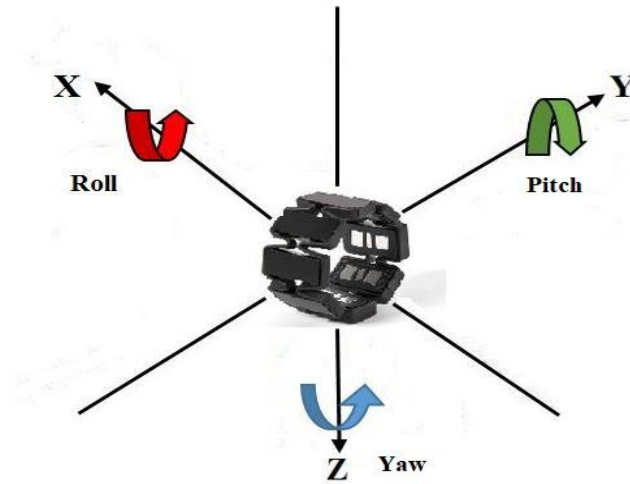
## BÖLÜM 4. DENEY DÜZENEĞİ VE GELİŞTİRİLEN YAZILIM

Bu bölümde yapılan çalışmanın deney düzeneğinden bahsedilecektir. Endüstriyel robotu kontrol etmek için ilk önce EMG bileklikten veriler alınmış olup bunlar Python ortamında işlendikten sonra robota TCP/IP haberleşmesi ile ethernet vasıtasıyla gönderilmiştir. Bu çalışma için deney düzeneğinin görsel şeması Şekil 4.1.'de gösterilmiştir.



Şekil 4.1. Deney düzeneğinin görsel şeması

Bu çalışmada EMG bileklik ile kolun hareketleri 3 boyutlu ortamda incelenmiştir. EMG sensörde mevcut olan IMU sensör ile kol hareketleri izlenmiştir ve robotun insan kolunu takip etmesi sağlanmıştır. IMU sensörün roll açısı robotu x ekseninde, yaw açısı z ekseninde, pitch açısı y ekseninde kontrol etmek için kullanılmıştır. EMG bileklikten elde edilen Euler açılarından robotun hareket ettirilmesi Şekil 4.2.'de gösterilmiştir.



Şekil 4.2. Myo arm band euler açıları

Myo Armband hareket ettirildiğinde euler açıları  $-90$  ile  $+90$  arasında değişim göstermektedir. Bu açı değişiminin pozisyon değişimine dönüştürülmesi gerekmektedir. Yapılan çalışmada robotun üç ekseninde gidebileceği maksimum koordinatlar tespit edilmiş olup dönüşüm fonksiyonu ile açı bilgisinden pozisyon bilgisine dönüşüm sağlanmıştır. Bu sayede robot X ekseninde  $1.6\text{mm/derece}$ , Y ekseninde  $2.7\text{mm/derece}$ , Z ekseninde  $2.2\text{mm/derece}$  oranlarında hareket etmektedir.

EMG sinyallerinden elde edilen bazı hareketler ile robotun kullanımı kolaylaştırılmıştır. Myo Armband tarafından beş farklı hareketten yumruk hareketi robotu gerçek zamanlı çalıştırma, eli serbest bırakma hareketi robotu durdurma ve çift parmak dokunma hareketi ise robotu başlangıç pozisyonu için verilen bir koordinata göndermek için kullanılmıştır. Robotu kontrol etmek için kullanılan hareketler Şekil 4.3. Şekil 4.4. ve Şekil 4.5.'de gösterilmiştir.



Şekil 4.3. Robotu gerçek zamanlı çalıştırmak için yapılan hareket



Şekil 4.4. Robotu durdurmak için yapılan hareket



Şekil 4.5. Robotu başlangıç pozisyonuna götürmek için yapılan hareket

Robotu kontrol edecek hareketler belirlendikten sonra robotu kontrolü gerçek zamanlı olarak sağlanmıştır. Sistemde bazı gecikmeler olmasına rağmen başarılı bir şekilde çalıştığı gözlemlenmiştir. Sistemde kullanılan deney düzeneği ve sistemin gerçek zamanlı çalışması Şekil 4.6. ve Şekil 4.7.'de gösterilmiştir.



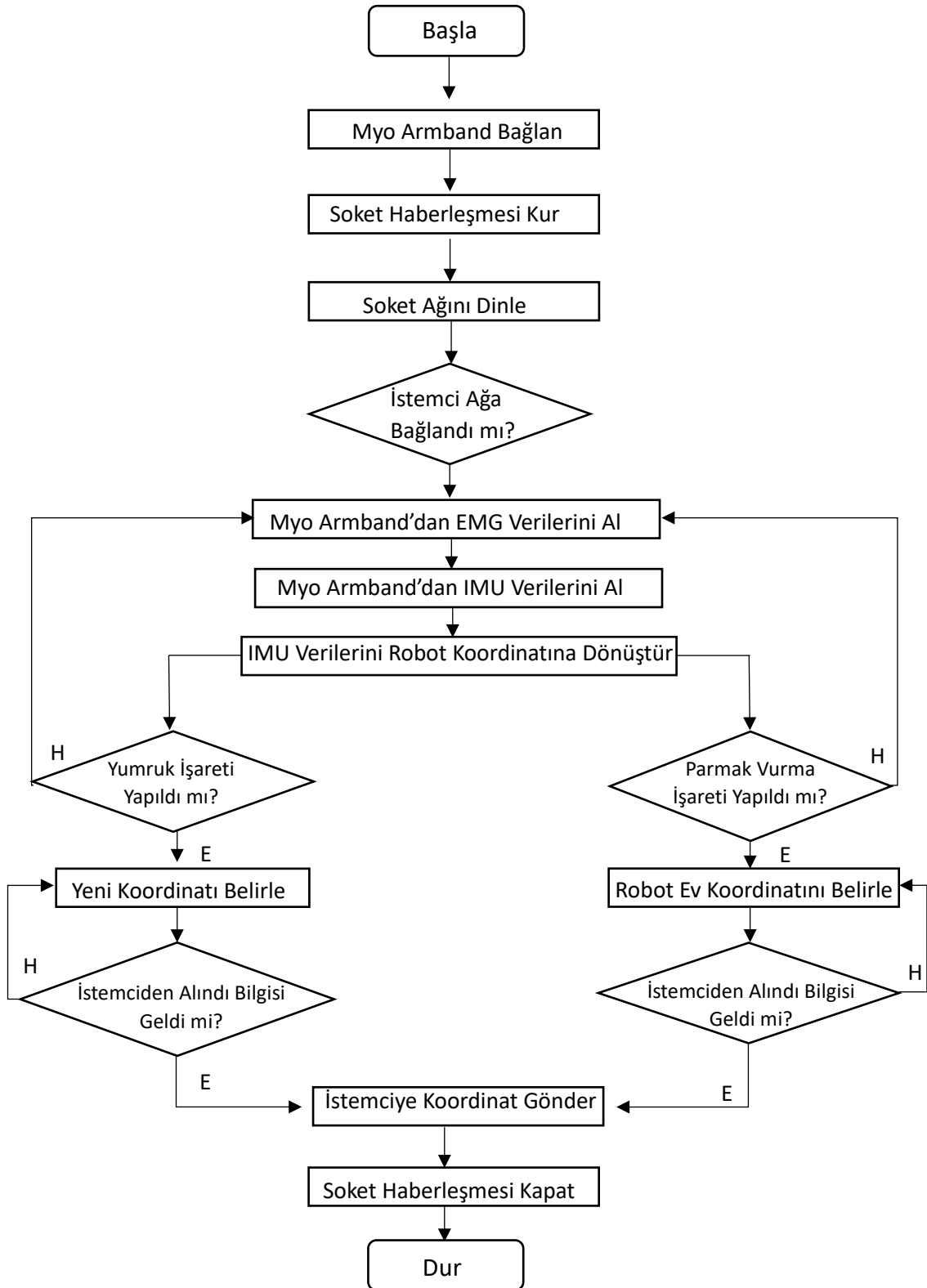
Şekil 4.6. IRB120 robotunun y ekseninde hareket etmesi



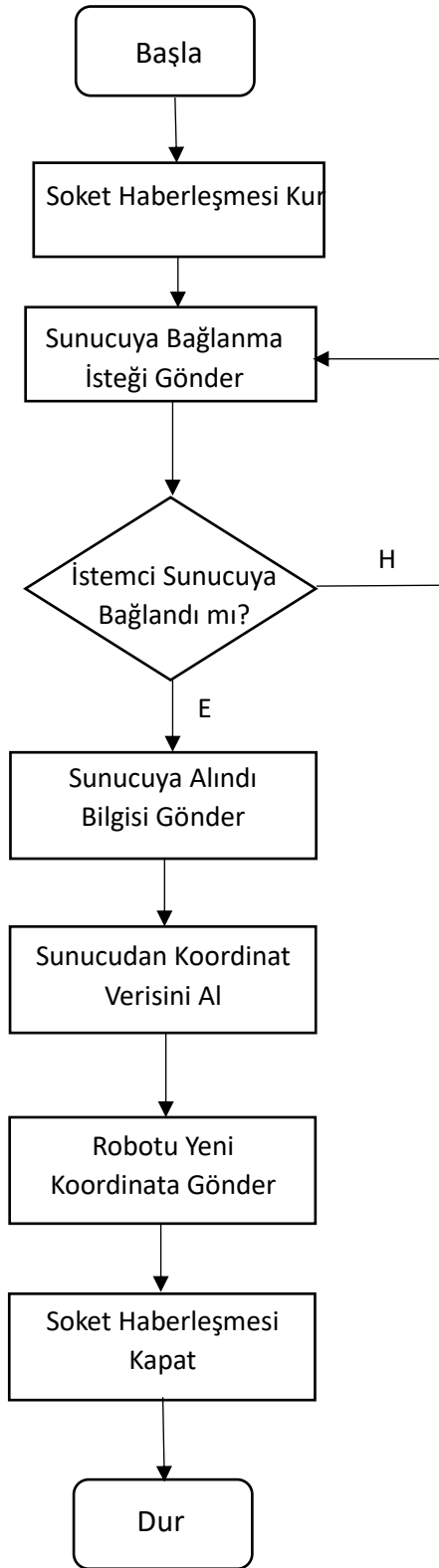
Şekil 4.7. IRB120 robotunun z ekseninde hareket etmesi

Bu çalışma kapsamında iki farklı yazılım geliştirilmiştir. Birinci yazılım Python derleyicisinde sunucu olarak yazılmış olup, ikinci yazılım RAPID programlama diliyle istemci olarak yazılmıştır. Birinci yazılımın amacı Myo Armband'dan ham EMG ve IMU verilerini alıp işlemek, haberleşmeyi başlatmak; ikinci yazılımın amacı ise haberleşme ile alınan robot koordinat verileriyle robotu o koordinata götürmektir. Kullanılan sunucu ve istemci yazılımlarının akış şemaları Şekil 4.8. ve Şekil 4.9.'da gösterilmiştir.





Şekil 4.8. Python sunucu akış diyagramı



Şekil 4.9. RAPID istemci akış diyagramı

## **BÖLÜM 5. TARTIŞMA VE SONUÇ**

Bu çalışmada giyilebilir ürün olan EMG bileklikle endüstriyel robotun kontrolü üzerinde çalışılmıştır. Çalışmada operatöre kullanım kolaylığı sağlayan bir İnsan Robot Arayüzü oluşturulmuştur. Çalışma ilk önce RobotStudio ortamında denenmiş ve başarılı bir şekilde çalışmıştır. RobotStudio simülasyon ortamında güvenli sınırlar, değişik hızlar kontrol edildikten sonra, gerçek IRB120 robotu üzerinde çalışma gerçekleştirilmiştir. IRB120 robotunu gerçekte yüksek hızlarda hareket ettirmek ani hareketlerde robotun üzerinde oluşan atalet momenti nedeniyle mekanik sorunlar oluşturabilmektedir. Bu sebeple gerçek robot üzerinde çalıştırmada düşük hızlarda hareket ettirilmiş ve sistemin başarılı bir şekilde çalıştığı gözlemlenmiştir. EMG bileklikten alınan veriler Euler açılarına dönüştürdükten sonra robota gerçek zamanlı olarak gönderilmiştir. Gönderilen verilerle robot x,y,z eksenlerinde lineer hareket yaparak kol bandının oluşturduğu yörüngeyi takip etmektedir. Takip etme işlemi devam ederken TCP/IP haberleşmesinin verdiği gecikmelerden kaynaklı robotun biraz geç hareket ettiği gözlemlenmiştir.

Yapılan çalışmada robot kolunun 0.2 mm hassasiyet ile çalıştırılabildiği gözlemlenmiştir. Hız ve çalışma sınırları kriterlerine göre değerlendirildiğinde sistemde iyileştirmelere ihtiyaç bulunmaktadır.

İlerleyen çalışmalarda EMG bileklikte bulunan 8 adet EMG elektrot ile alınan veriler yapay sinir ağları ile sınıflandırarak yeni hareket kestiriminde bulunulacaktır. Bu hareketler ile endüstriyel robota farklı işlemler yaptırılacaktır. Ayrıca IRB120 robotunda bulunan Externally Guided Motion özelliği kullanılacaktır. Externally Guided Motion özelliği IRB120 robotu ethernet haberleşmesini çok hızlı bir şekilde yapmaktadır. Bu sayede robotumuzun gecikmesi azaltılmış olacaktır.

## KAYNAKLAR

- ABB ROBOTICS, Operating Manual Intoduction to RAPID , Document ID: 3HAC0966-50 Revision: 1.
- ABB ROBOTICS, Product Specification IRB120, Document ID: 3HAC05960-001 Revision: Q.
- Ana, C. Christoph, M. M., Pieter, V. M., Serge, V., Gijss, P. 2016. Directed Functional Brain Connectivity Based on EEG Source Imaging Methodology and Application to Temporal Lobe Epilepsy, Transactions on Biomedical Engineering, 63(12), 2619-2628.s.
- Baocheng, W., Chenguang, Y., Qing, X., 2012. Human-machine Interfaces bases on Yihsin, H., Eri, S. S., Kazuyoshi, W., Toru, Y., Norio, T., 2014. A Study for Assistant Robotic System using Motion Capture Method for Adapting to Human Robot Interface, 758-762.
- Beyda, T., Turgay, K., Arif, G., 2014. EMG Tabanlı El Hareketinin Analizi Aracılığı ile Robot El Simülatörünün Kontrolü, Signal Processing and Communications Applications Conference, 389-392.
- Bir, T., 2005. Endüstriyel Veri İletişiminde TCP/IP Uygulaması, İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Elektrik Mühendisliği Bölümü, Yüksek Lisans Tezi.
- Chin-Teng, L., Chun-Hsiang, C., Chih-Sheng, H., Shu-Fang, T., Shao-Wei, L., Yen-Hsuan, C., Li-Wei, K., 2014. Wireless and Wearable EEG System for Evaluating Driver Vigilance, 8(2), 165-176.
- Chun, S. L. T., Pei, J., Jonh, Q. G., Huosheng, H., Kui, Y. 2007, EMG Based Hands-Free Wheelchair Control with EOG Attention Shift Detection with EOG Attention Shift Detection, 1266-1271.
- Christian, M., Thomas, R., Udo, F., 2007, Applying a 3DOF Orientation Tracker as a Human-Robot Interface for Autonomous Wheelchairs, Rehabilitation Robotics(ICORR), 52-59.
- Ernane, J. X. C., Euvaldo, F., Cabral, J., 2000. EEG-Based discrimination between imagination of left and right hanf movements using adaptive gaussian representation, Medical engineering and physics, 22(5), 345-348.
- Fred, A., Jhon, M., Christian, P., Francisco, C. 2016. Hybrid BCI System to Operate and Electric Wheelchair and a Robotic Arm for Navigation and Manipulation Tasks, Advanced Robotics and its Social Impacts(ARSO), 249-254.

- Guan-Chun, L., Heng-An, L., Yi-Hsiang, M., Chien-Jung, Y., 2015. Intuitive Muscle Gesture Based Robot Navigation Control Using Wearable Gesture Armband, International Conference on Machine Learning and Cybernetics, 389-395.
- Gurkan, K., Hasan, O., Suat, K., Omer, S., 2017. Desing and Implementation of a Multi Sensor Based Brain Computer Interface for a Robotic Wheelchair, Journal of Intelligent and Robotic Systems, 1(17),247-263.
- Isao, M., Yasunobu, Y., Miwako, D., 2003. A Real-Time Vision-Based Interface Using Motion Processor and Application to Robotics, Systems and Computer in Japan, 10-19.
- Jacopo, A., Alexander, S., Tom, D., 2004. Position Teaching of a Robot Arm by Demonstration with a Wearable Input Device, International Conference on Intelligent Manipulation and Grasping.
- James, C., Huosheng, H., 2012. A Multi Sensor Armband based on Muscle and Motion Measurements, International Conference on Robotics and Biomimetics, 1098-1103.
- Jamie, S., Toby, S., Alex, K., Andrew, F., Mark, F., Andrew, B., Mat, C., Richard, M., 2013. Real-Time Human Pose Recognition in Parts from Single Depth Images, Communications of the ACM, 56(1), 116-124.
- Jonathan, R.W., Niels, B., Dennis, J. M., Gert, P., Theresa, M. V. 2002. Brain-computer interface for communication and control,111(6): 767-791.
- Kang, H., Kiwon, R., Kyoung-Jin, Y., Hyun-chool, K., 2011. Intuitive Robot Navigation Using Wireless EMG and Acceleration Sensors on Human Arm, International Symposium on Intelligent Signal Processing and Communication Systems, 1-4.
- Kazuo, T., Kazuyuki, M., Hua, O. W., 2005. Electroencephalogram-Based Control of an Electric Wheelchair, Transactions on Robotics, 762-766.
- Kennedy, P. R., Bakay, R. A. E., Moore, M. M., Adams, K., Goldwaihte, J., 2000. Direct Control of a Computer from the Human Central Nervous System, Transactions on Rehabilitation Engineering, 8(2), 198-202.
- Khuram, N., Jamshed, I., Habib, U. R. 2012. Brain Controllerd Human Robot Interface, Robotics and Aritifical Intelligence(ICAI), 2012 International Conference, 55-60.
- Lijun, Z., Yihuan, L., Ke, W., Peidong, L., Ruifeng, L., 2016. An Intuitive Human Robot Interface for Tele-operation, International Conference on Real-Time Computing and Robotics, 454-459.
- Lun-De, L., Chi-Yu, C., I-Jan, W., Sheng-Fu, C., Shih-Yu, L., Boi-Wei, C., Jhy-Yeong, C., Chin-Teng, L., 2012. Gaming Control Using a Wearable and Wireless EEG-based brain-computer interface device with novel dry foam-based sensors, 9(1).
- Mithileys, S., Sharanya, R., 2016. MYO Armband for Physiotherapy Healtcare: A Case Study Using Gesture Recognition Application, International Conference on Communication Systems and Networks, 1-6.

- Osamah, A.A., 2000. EMG-based Human Machine Interface System, International Conference on Multimedia and Expo, 2, 925-928.
- Panagioitis, K. A., Kostas, J. K., 2010. EMG-Based Control of a Robot Arm Using Low-Dimensional Embeddings, Transactions on Robotics, 26(2), 393-398.
- Pedro, N. J., Norberto, P. A., Paulo, M., 2010. High-Level programming and control for industrial robotics : using a hand-held accelerometer-based input device for gesture and posture recognition, Industrial Robot: An International Journal, 37(3), 137-147.
- Quingwen, Y., Zhiguo, L., Chong, L., 2015, Human-robot Interface Based on WSSS IMU Sensors, International Conference on Cyber Technology in Automation, Control and Intelligent Systems, 846-849.
- Ranathunga, A. R. C. G., Kazuo, K., 2008. EMG-Based Control of an Exoskeleton Robot for Human Forearm and Wrist Motion Assist, International Conference on Robotics and Automation, 731-736.
- Reaz, M. B. I., Hussain, M.S., Mohd-Yasin, F., 2006. Tecniques of EMG signal analysis: detection, processing, classification and application, Biological Procedures Online, 8(1), 11-35.
- Roberto, C., Nicholas, J. H., 1996. Human-Robot interface by pointing with uncalibrated stereo vision, Image and Vision Computing , 14(3),171-178.
- Seong-Og, S. Donghan, K., Yong-Ho, S., 2014, Controlling Mobile Robot Using IMU and EMG Sensor-based Gesture Recognition, International Conference on Broadband and Wireless Computing, Communication and Application, 554-557.
- Sirin, A., Onur, A., Ali, F. E., 2015, Endüstriyel Sistemler için FPGA Tabanlı Modbus Ağ Geçidi Tasarımı ve Uygulaması, Otomatik Kontrol Ulusal Toplantısı, 445-451.
- Umut, M., Serdar, K., Ömer, Ş., 2015, EMG Kontrollü Mobil Robot Uygulaması, Otomatik Kontrol Ulusal Toplantısı, 1-4.
- Wei, H., Yue, Z., Haoyue, T., Changyin, S., Wei, F., 2016, A Wireless BCI and BMI System for Wearable Robots, IEEE Transactions On Systems, Man, and Cybernetics, 46(7), 936-946.
- [www.myo.com](http://www.myo.com), Erişim Tarihi: 02.10.2017.
- [www.new.abb.com/products/robotics/tr](http://www.new.abb.com/products/robotics/tr), Erişim Tarihi: 11.10.2017.
- Yan, B. Xuanfang, W. Li, Z., Shihang, C. 2016, Advanced Information Management Communicates, Electronics and Automation Control Conference(IMCEC), 232-236.
- Yingxi, C., Wyatt, S. N. 2004. A-Human-Robot Interface Based on Electrooculography, International Conference on Robotics and Automation, 243-248.
- Yihsin, H., Eri Sato, S., Kazuyoshi, W., Toru, Y., Norio, T., 2014, A Study for Assistant Robotic System using Motion Capture Method for Adapting to Human – robot Interface, 758-762.

- Zhendong, L., Shun, H., Feng, D., 2015. The Development of a Smart House Based on Brain-Computer Interface, Conference on Robotics and Biommetics, 1012-1017.
- Zhijun, L., Baocheng, W., Fuchun, S., Chenguang, Y., Qing, X., Weidong, Z., 2014. sEMG – Based Joint Force Control for an Upper-Limb Power –Assist Exoskeleton Robot, Journal of Biomedical and Health Informatics, 18(3), 1043-1050.
- Zhijun, L., Shuangshuang, L., Chun-Yi, S., Guanglin, L., 2013. Hybrid Brain/Muscle-Actuated, Control of an Intelligent Wheelchair, International Conference on Robotics and Biomimetics, 19-25.

## EKLER

### EK 1 : Python sunucu program parçaları

```
myo = feed.wait_for_single_device(timeout=2.0)
soket = socket.socket(socket.AF_INET,socket.SOCK_STREAM)
HOST = "169.254.215.95"
PORT = 5993
soket.bind((HOST,PORT))
print "%s:%d server başlatıldı." %(HOST,PORT)
print "Kullanıcı bekleniyor."
soket.listen(1)
baglanti,adres = soket.accept()
print "Bir bağlantı kabul edildi.", adres
while hub.running and myo.connected:
    pose=myo.pose
    if pose=="double_tap":
        x=220
        y=0
        z=400
        rob_gonder()
        time.sleep(2)
    while pose=="fist":
        gonder()
        pose=myo.pose
        rob_gonder()
def gonder():
    global cnt,x,y,z
    quat = myo.orientation
    roll = atan2(quat.y * quat.z + quat.w * quat.x,
```



```
float(0.5) -(quat.x * quat.x + quat.y * quat.y))
pitch = asin(float(-2) * (quat.x * quat.z - quat.w * quat.y))
yaw = atan2((quat.x * quat.y + quat.w * quat.z),
float(0.5)-(quat.y * quat.y + quat.z * quat.z))
roll_w = int((roll / (pi*float(2)))*180)
pitch_w = int((pitch / (pi*float(2)))*180)
yaw_w = int((yaw / (pi*float(2)))*180)
def rob_gonder():
    alindi= baglanti.recv(20)
    if alindi=="data":
        data = "[" +str(x) +"," +str(y)+"," +str(z)+" "]"
        print x,z,y
        baglanti.send(data)
```

**EK 2 : RAPID istemci program parçaları**

```
PROC Initialize()
```

```
    currentTool := [TRUE ,[[0,0,0],[1,0,0,0]], [0.001 , [0,0,0.001],[1,0,0,0],0,0,0]];
```

```
    currentWobj := [FALSE,TRUE, "" ,[[ 0,0,0],[1,0,0,0]],[[0,0,0],[1,0,0,0]]];
```

```
    currentSpeed := [100,50,0,0];
```

```
    currentZone := [FALSE,0.3,0.3,0.3,0.003,0.3,0.003];
```

```
ENDPROC
```

```
PROC Main()
```

```
    Initialize;
```

```
    SocketCreate socket1;
```

```
    SocketConnect socket1 , "169.254.215.95",5993;
```

```
    WHILE TRUE DO
```

```
        SocketSend socket1 \Str:= "data";
```

```
        received_string := "";
```

```
        SocketReceive socket1 \Str:= received_string ;
```

```
        ok := StrToVal(received_string,x);
```

```
        TPWrite received_string;
```

```
        cartesianTarget := [x,[0,1,0,0],[0,0,-3,0],
```

```
[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];
```

```
        MoveL cartesianTarget,v1000,fine,tool0 \WObj:=wobj0;
```

```
    ENDWHILE
```

```
    SocketClose socket1;
```

```
ENDPROC
```

## ÖZGEÇMİŞ

Kenan Erin, 15.09.1991'de Mersin'de doğru. İlkokul eğitimini Mersin'de ortaokul eğitimini Ankara'da ve lise eğitimini Mersin'de tamamladı. 2009 yılında Dumlupınar Lisesi'nden mezun oldu. 2010 yılında başladığı Kocaeli Üniversitesi Mekatronik Mühendisliği Bölümü'nü 2015 yılında bitirdi. 2016 Şubat ayında Sakarya Üniversitesi Fen Bilimleri Enstitüsü, Mekatronik Mühendisliği Anabilim Dalı'nda Yüksek Lisans eğitimine başladı. 2016 Şubat ayında Mersin Üniversitesi Tarsus Teknoloji Fakültesi Mekatronik Mühendisliği Bölümü'nde ÖYP Araştırma Görevlisi olarak göreve başladı ve görevlendirme ile Sakarya Üniversitesi Mekatronik Mühendisliği Bölümü'nde Araştırma Görevlisi olarak görevine devam etmektedir.