

BURSA TEKNİK ÜNİVERSİTESİ ❖ FEN BİLİMLERİ ENSTİTÜSÜ

HİBRİT MONTAJ SİSTEMLERİNDE İNSAN-ROBOT ETKİLEŞİMİ



YÜKSEK LİSANS TEZİ

Mustafa ÇOBAN

Mekatronik Mühendisliği Anabilim Dalı

HAZİRAN 2018

BURSA TEKNİK ÜNİVERSİTESİ ❖ FEN BİLİMLERİ ENSTİTÜSÜ

HİBRİT MONTAJ SİSTEMLERİNDE İNSAN-ROBOT ETKİLEŞİMİ

YÜKSEK LİSANS TEZİ

**Mustafa ÇOBAN
(161081505)**

Mekatronik Mühendisliği Anabilim Dalı

Tez Danışmanı: Doç. Dr. Gökhan GELEN

HAZİRAN 2018

BTÜ, Fen Bilimleri Enstitüsü'nün 161081505 numaralı Yüksek Lisans Öğrencisi Mustafa ÇOBAN, ilgili yönetmeliklerin belirlediği gerekli tüm şartları yerine getirdikten sonra hazırladığı "HİBRİT MONTAJ SİSTEMLERİNDE İNSAN-ROBOT ETKİLEŞİMİ" başlıklı tezini aşağıda imzaları olan jüri önünde başarı ile sunmuştur.

Tez Danışmanı : **Doç. Dr. Gökhan GELEN**

Bursa Teknik Üniversitesi

Jüri Üyeleri : **Dr. Öğr. Üyesi Murat PEKER**

Niğde Ömer Halisdemir Üniversitesi

Dr. Öğr. Üyesi Ekrem DÜVEN

Bursa Teknik Üniversitesi

Savunma Tarihi : 21 Haziran 2018

FBE Müdürü : **Doç. Dr. Murat ERTAŞ**

Bursa Teknik Üniversitesi

...../...../.....

İNTİHAL BEYANI

Bu tezde görsel, işitsel ve yazılı biçimde sunulan tüm bilgi ve sonuçların akademik ve etik kurallara uyularak tarafımdan elde edildiğini, tez içinde yer alan ancak bu çalışmaya özgü olmayan tüm sonuç ve bilgileri tezde kaynak göstererek belgelediğimi, aksinin ortaya çıkması durumunda her türlü yasal sonucu kabul ettiğimi beyan ederim.

Öğrencinin Adı Soyadı: Mustafa ÇOBAN

İmzası :

X X X X X



Aileme ve arkadaşlarıma,

ÖNSÖZ

Yüksek lisans çalışmam boyunca, her türlü konuda bilgi, tecrübe ve desteğini benden esirgemeyen, fikirleriyle çalışmama yön veren değerli danışmanım Sayın Doç. Dr. Gökhan GELEN'e, bölüm hocalarıma, arkadaşlarıma ve hayatım boyunca bana destek olan aileme sonsuz teşekkürlerimi sunarım.

Çalışma kapsamında kullandığım endüstriyel robotu bölümümüze hibe eden Mitsubishi Electric'e ve çalışmamı 171N07 proje numarası altında destekleyen Bursa Teknik Üniversitesi BAP Koordinatörlüğü'ne teşekkür ederim.

Son olarak, 2210/A Genel Yurt İçi Yüksek Lisans Burs Programı kapsamında çalışmamı destekleyen Türkiye Bilimsel ve Teknolojik Araştırma Kurumu (TÜBİTAK) Başkanlığı'na teşekkür ederim.

Haziran 2018

Mustafa ÇOBAN

İÇİNDEKİLER

Sayfa

ÖNSÖZ	v
İÇİNDEKİLER	vi
KISALTMALAR	vii
SEMBOLLER	viii
ÇİZELGE LİSTESİ.....	ix
ŞEKİL LİSTESİ.....	x
ÖZET	xi
SUMMARY	xii
1. GİRİŞ	1
1.1 Literatür Araştırması	1
1.2 Tezin Amacı	5
2. HİBRİT MONTAJ SİSTEMLERİ VE DENEY DÜZENEGİ	7
2.1 Oluşturulan Deneysel Hibrit Montaj Sistemi	9
2.2 Sistemde Kullanılan Cihazlar	11
2.2.1 Myo kol bandı	11
2.2.2 Raspberry Pi 3	12
2.2.3 Robot montaj modülü.....	13
2.2.4 Güvenlik ışık bariyeri.....	13
2.2.5 Güvenlik rölesi	14
2.2.6 Endüstriyel robot.....	15
3. İNSAN KOL HAREKETLERİNİN ALGILANMASI VE ROBOT ETKİLEŞİMİ	16
3.1 Kol Hareketlerinin Algılanması	16
3.1.1 Myo kol bandı ile algılanabilen kol hareketleri	16
3.1.2 Myo kol bandı ile Raspberry Pi 3 arasındaki veri alışverişinin sağlanması	17
3.2 Kol Hareket Verilerinin Robot Denetleyicisine Aktarımı.....	17
3.3 Robot Kontrol Algoritmasının Gerçekleştirilmesi	18
3.3.1 Melfa Basic-V robot programlama dili	18
3.3.2 Robot ile Raspberry Pi 3 arasındaki veri alışverişinin sağlanması	20
3.3.3 Kol hareket verileriyle gerçekleştirilen temel robot uygulaması	21
4. İNSAN-ROBOT ETKİLEŞİMİ İLE MONTAJ İŞLEMLERİNİN GERÇEKLEŞTİRİLMESİ	24
4.1 Performans Testleri	27
4.2 Montaj Sürelerinin Karşılaştırılması	31
5. SONUÇ VE ÖNERİLER.....	33
KAYNAKLAR	34
EKLER	37
ÖZGEÇMİŞ.....	47

KISALTMALAR

cm	: Santimetre
EMG	: Elektromiyografi
FAT	: Function Approximation Technique
GB	: Gigabayt
GHz	: Gigahertz
Kg	: Kilogram
LQR	: Linear Quadratic Regulator
mm	: Milimetre
sn	: Saniye
TCP/IP	: Transmission Control Protocol/Internet Protocol

SEMBOLLER

t : Zaman



ÇİZELGE LİSTESİ

	<u>Sayfa</u>
Çizelge 4.1 : Robotun tek başına yaptığı montaj esnasında geçen zaman.....	28
Çizelge 4.2 : İnsan-robot etkileşimi ile yapılan montaj esnasında geçen zaman.....	30
Çizelge 4.3 : Görevlerin ortalama tamamlanma süresi.....	32



ŞEKİL LİSTESİ

	<u>Sayfa</u>
Şekil 2.1 : Hibrit montaj sistemi	7
Şekil 2.2 : Parça taşımayı sağlayan hibrit sistem.....	7
Şekil 2.3 : Hibrit montaj sistemlerindeki insan-robot etkileşim düzeyleri	8
Şekil 2.4 : Tasarlanan deneysel hibrit montaj sistemi	9
Şekil 2.5 : Montajı yapılan iş parçaları	9
Şekil 2.6 : Deneysel hibrit montaj sistemi	10
Şekil 2.7 : Profil plakadaki parçaların yerleşim düzeni	11
Şekil 2.8 : Cihazlar arası bağlantı şeması	11
Şekil 2.9 : Myo kol bandı	12
Şekil 2.10 : Raspberry Pi 3	12
Şekil 2.11 : Robot montaj modülü	13
Şekil 2.12 : Güvenlik ışık bariyeri	14
Şekil 2.13 : Güvenlik rölesi	14
Şekil 2.14 : Endüstriyel robot	15
Şekil 3.1 : Myo kol bandı ile algılanabilen kol hareketleri	16
Şekil 3.2 : Terminal ekranına aktarılan kol hareketleri	17
Şekil 3.3 : Raspberry Pi 3 için kullanılan istemci algoritması.....	18
Şekil 3.4 : Robot için kullanılan sunucu algoritması.....	21
Şekil 3.5 : Al-bırak uygulaması için oluşturulan algoritma.....	21
Şekil 3.6 : Al-bırak uygulaması ile elde edilen durum değişim grafiği.....	22
Şekil 4.1 : İnsanın ve robotun gerçekleştirdiği görev paylaşım şeması.....	25
Şekil 4.2 : Robot tarafından gerçekleştirilen görev algoritması	26
Şekil 4.3 : Robotun tek başına yaptığı montaj işlemleri.....	27
Şekil 4.4 : Robotun tek başına gerçekleştirdiği görevlerin ortalama tamamlanma süreleri.....	29
Şekil 4.5 : İnsan-robot etkileşimi ile yapılan montaj işlemleri.....	29
Şekil 4.6 : İnsan-robot etkileşimi ile gerçekleştirilen görevlerin ortalama tamamlanma süreleri	30
Şekil 4.7 : Montajı tamamlanan ürün sayısına göre geçen zaman.....	31
Şekil 4.8 : Görevlerin tamamlanması esnasında geçen süreler.....	32

HİBRİT MONTAJ SİSTEMLERİNDE İNSAN-ROBOT ETKİLEŞİMİ

ÖZET

Robotlar ve endüstriyel ürün teknolojilerindeki gelişmelerle birlikte, insanlarla robotların etkileşimde bulunarak çalışmaları konusunda yapılan araştırma sayısı artış göstermiştir. Bu araştırmaların neticesinde, işbirlikçi robot olarak adlandırılan yeni nesil endüstriyel robotlar ortaya çıkmıştır. İnsanlar ve işbirlikçi robotlar, endüstriyel ortamlarda üretim, montaj, taşıma gibi görevleri güvenli biçimde yerine getirebilmektedir. Geleneksel endüstriyel robotlar, insanlardan uzak olacak şekilde çitlerle çevrili alanlarda çalışmaktadır. İşbirlikçi robotlar ise, insanlar ile aynı çalışma ortamında güvenli biçimde işlemlerini gerçekleştirirler. Bu şekilde zaman ve mekan tasarrufu sağlayarak verimliliği artırırlar. Özellikle küçük ve orta ölçekli üretim işlemlerinde, işbirlikçi robotlar ile ürün kalitesinin artırılması sağlanabilmektedir.

Tez kapsamında, altı serbestlik dereceli endüstriyel bir robotun, insanla etkileşimli olarak örnek endüstriyel ürünlerin montajını yapması sağlanmıştır. İnsan ve robot etkileşimi ile yapılan montaj işlemleri için ilk olarak, içerisinde kas sinyallerini algılayan sensörler bulunduran bir kol bandı ile insan kol hareketleri algılanmıştır. Algılanan kol hareketleri, Ethernet üzerinden robot denetleyicisine aktarılmıştır. İnsan ve robotun yapacağı görevler, robot özellikleri ve insan kabiliyeti dikkate alınarak paylaştırılmıştır. Bu görev paylaşımı, kol hareketleri ve montaj senaryosu dikkate alınarak robot kontrol algoritması oluşturulmuş ve kodlanmıştır. İnsanın ve robotun ortak çalışma alanındaki güvenliğinin sağlanması amacıyla, robot denetleyicisine güvenlik rölesi üzerinden güvenlik ışık bariyeri bağlanmıştır. Kullanılan ışık bariyeri, çalışma alanına girilip girilmediğinin algılanmasını sağlamaktadır. Yapılan uygulamalar sonucu, insan-robot işbirliğiyle endüstriyel ürünlerin başarılı olarak montajlandığı gözlenmiştir. Çalışmanın verimliliğini ölçmek için, insan ve robotun beraber yaptığı montaj işlemleri, robotun tek başına yaptığı montaj işlemleriyle karşılaştırılmış ve insan-robot etkileşimi ile yapılan işlemlerin daha hızlı ve daha verimli gerçekleştiği belirlenmiştir.

Anahtar kelimeler: İşbirlikçi robot, İnsan-robot etkileşimi, Hibrit montaj sistemleri, Robot programlama.

REALIZATION OF HUMAN-ROBOT COLLABORATION IN HYBRID ASSEMBLY SYSTEMS

SUMMARY

Together with improvements in robotics and industrial product technologies, the number of research conducted on interaction between humans and robots has increased. As a result of these research, a new generation of industrial robots, called collaborative robots, have emerged. Humans and collaborative robots can safely perform tasks such as production, assembly, transportation in industrial environments. Conventional industrial robots work in fenced areas, away from human beings. Collaborative robots operate safely in the same working area with humans. They increase efficiency by saving time and space in this way. Particularly in small and medium scale production processes, it is possible to increase the product quality with collaborative robots.

Within the scope of the thesis, a six degree-of-freedom industrial robot has been provided for the assembly of sample industrial products interacting with human beings. For the assembly process with human and robot interaction, firstly, human arm movements was perceived by using an armband containing sensors for detecting muscle signals. Detected arm movements were transferred to robot controller via ethernet. Tasks to be done by human beings and robots were shared by taking into consideration robot characteristics and human ability. By considering this task sharing, arm movements and assembly scenario, robot control algorithm was created and coded. A safety light curtain was connected to the robot controller via safety relay to ensure the safety of human and robot in the joint work area. The light curtain that was used detects whether or not the work area is entered. As a result of applications, it has been observed that industrial products have been successfully assembled by human-robot cooperation. In order to measure the efficiency of the work, the assembly operations performed by human-robot collaboration were compared with the assembly operations performed by robot alone. It has been determined that the operations performed by the human-robot interaction performed faster and more efficiently.

Keywords: Collaborative robot, Human-robot interaction, Hybrid assembly systems, Robot programming.

1. GİRİŞ

Robotik alanında geçmişten günümüze kadar önemli arařtırmalar yapılmıř ve büyük geliřmeler kaydedilmiřtir. Robotların ve uygulama alanlarının sayısının artması ile, insan-robot etkileřimi ve insan-robot iřbirlięi konuları üzerine yapılan çalıřmalar büyük önem kazanmıřtır. Bu çalıřmaların sonucunda iřbirlikçi robotlar ortaya çıkmıřtır. İřbirlikçi robotlar, insanlarla etkileřimli olarak çeřitli görevleri yerine getiren ve insanın iř yükünü hafifleten robotlardır [1].

Robotięin geliřiminde, robotların dięer makinelerle etkileřimli olarak çalıřması, önemli bir arařtırma konusu olmuřtur. Robotlar uzun bir süre boyunca üretim bantlarında, insanlardan uzak olacak řekilde çalıřtırılmıřtır. Robotlar, insanlarla etkileřim halinde olmadıklarından dolayı, olası kazalarda insanlara zarar vermemeleri için çitlerle çevrili ortamlarda operasyonlarını gerçekteřirmişlerdir. Bu tarz sorunların önüne geçmek ve robotların daha verimli çalıřmalarını saęlamak için insan-robot etkileřimi konusunda birçok çalıřma yapılmıřtır.

1.1 Literatür Arařtırması

İřbirlikçi robotlar konusundaki ilk çalıřmalar 1996 yılında Northwestern Üniversitesi'nde gerçekteřirilmiřtir [2]. İřbirlikçi robot ile ilgili ilk patentler ise 1997 yılında alınmıřtır.

İřbirlikçi robotların geliřimiyle birlikte, insanların ve robotların aynı çalıřma alanında güvenli biçimde hibrit imalat ve montaj yapmaları mümkün hale gelmiřtir. Hibrit montaj sistemleri, son yıllarda oldukça önem kazanan bir akademik çalıřma konusu olmuřtur.

İnsan ve robotların etkileřimini başarılı bir řekilde saęlamak için, robotların ve denetlenecek sistemin özelliklerinin dikkate alınması gerekmektedir. Endüstride kullanılan iřbirlikçi robotların, kontrol ve otomasyon sistemlerinde esneklik, maliyet, geliřtirilebilirlik ve güvenlik gibi unsurları karşılayabilmeleri beklenmektedir.

İnsansı yapıdaki Baxter [3] ve Sawyer [4] adlı robotlar, gelişmiş işbirlikçi robot örnekleri olarak verilebilir.

İşbirlikçi robotların endüstride kullanımında en önemli parametrelerden biri güvenlidir. Endüstriyel bir robotun, insanla işbirliği içerisinde güvenli olarak nesnelere taşıma işlemleri, sisteme entegre edilen kameralar yardımıyla izlenebilmektedir [5]. Robotun taşıyacağı nesneye olan uzaklığına göre yeni güvenlik bölgeleri belirlenebilmektedir. Robot bu sayede gerekli güvenlik denetimini sağlayabilmektedir.

Endüstriyel üretim bantlarındaki insan-makine işbirliği uygulamaları [6]'da detaylı şekilde sunulmuştur. Bu çalışmada; hibrit montaj sistemleri, insan ile makinenin ortak çalışma alanları ve şekilleri, farklı tip makineler ve robotlarla insanların beraber çalışmaları konularında bilgiler verilmiştir.

Cherubini ve arkadaşları tarafından yapılan çalışmada, insan-robot etkileşimi ile homokinetik mafsal montajı gerçekleştirilmiştir [7]. Çalışmada kullanılan robot, insanla etkileşimli olarak mafsal montajını yapmakta ve çalışma esnasında insanı tehlikeye sokacak durumları, sisteme entegre edilen kamera yardımıyla denetlemektedir.

İnsan ve robot işbirliği ile yapılacak işlemlerde önemli parametrelerden birisi de insan ve robotun yapacağı görevlerin paylaşımıdır. Hibrit montaj işlemleri için insanın ve robotun yapacağı görevler belirlenerek uygulamalar gerçekleştirilebilmektedir [8].

İnsan-robot etkileşimiyle uygulamaların yapılması için sonlu durum otomatları kullanılarak yapılan çalışmanın detayları [9]'da açıklanmıştır. Çalışmadaki robotik sistem, sonlu durum otomatları ile modellenmiş ve çift kollu ABB robot ile uygulamalar gerçekleştirilmiştir.

İnsan ve robot işbirliği ile güvenli biçimde al-bırak (pick and place) uygulamaları gerçekleştirilebilmektedir [10]. Sistem üzerine eklenen kameralar, görüntü işleme tekniğini kullanarak uygulama alanını denetlemektedir. Çarpışma olacağı tespit edilirse, robot uygun şekilde yönelimini değiştirerek sistem güvenliğini sağlayabilmektedir.

Robotik sistemlerdeki gelişmeler, otomotiv endüstrisinde de insan-robot işbirliği uygulamalarının gerçekleştirilmesini sağlamıştır [11,12]. Otomobilin gövde, güç

aktarma organları, genel montajı ve üretim aşamalarında çeşitli düzeyde insan-robot işbirliği uygulamaları bulunmaktadır.

Üretim hatlarındaki güvenlik kısıtları ve unsurlarını dikkate alarak, insan-robot etkileşimiyle verimliliği arttırmayı sağlayan yöntemin detayları [13]'te sunulmuştur. Bu çalışmada, güvenlik standartlarına göre robotun hareket planlamasını optimize edecek algoritmalar geliştirilmiştir. Çalışma alanının denetimi Microsoft Kinect Kamera ile yapılırken, ABB endüstriyel robot kullanılarak uygulama gerçekleştirilmiştir.

Hibrit montaj sistemlerinde insan ve robot arasındaki güvenliği sağlamak için farklı yöntemler kullanılabilir. Shen ve Reinhart tarafından yapılan çalışmada, robotun tutucusuna bir düzlemsel lazer sensör yerleştirilmiştir [14]. Bu sensör yardımıyla, gripper ve iş parçası arasındaki mesafe ölçülebilmektedir. LabVIEW yazılımı kullanılarak bir algoritma geliştirilmiş ve altı eksenli bir robot üzerinde uygulama gerçekleştirilmiştir. Endüstriyel bir robotun güvenli şekilde çalışmasını sağlamak için mesafe sensörünün kullanıldığı yöntem [15]'te açıklanmıştır. Bu çalışmada, robotun bir parçasına yerleştirilen mesafe sensörleri yardımıyla çevre denetimi yapılmaktadır. Sensörlerden gelen verilere göre oluşturulan kontrol algoritmaları, ABB endüstriyel robotu üzerinde uygulamaya dönüştürülmüştür.

İnsan-robot işbirliğine dayalı çalışma ortamlarında güvenlik ve üretim gereksinimlerini karşılamak için kinematik kontrol stratejileri geliştirilmiştir [16]. Sistem güvenliğini sağlamak için, robot hızında uygulanabilir kısıtlamalar türetilmiştir. Robot hızı, bu kısıtlamalar yardımıyla insanın uzaklığına bağlı olarak sınırlanabilmekte ve güvenli uygulamalar yapılabilmektedir.

İnsan-robot işbirliği uygulamalarında, fiziksel insan-robot etkileşimin yanı sıra sosyal insan-robot etkileşimini de içeren denetleyiciler geliştirilmiştir [17]. İnsan-robot işbirliği sisteminin kinematiği tamamlanarak, robotun optimal hızını bulmak için yeni bir kontrol yöntemi önerilmiştir. Önerilen yöntem yardımıyla, insanın iş yükü azaltılarak insan-robot işbirliği uygulamalarının performansı artırılabilir.

Liu ve arkadaşları tarafından [18]'de yapılan çalışmada, insan-robot işbirliğindeki verimi arttırmak için sonlu durum makineleri yardımıyla insanın yönelimini hesaplayan bir kontrol yöntemi geliştirilmiştir. Önerilen yöntemle birlikte,

operatörün davranışları ve yönelimleri tahmin edilerek, insan-robot işbirliği ile etkili operasyonlar gerçekleştirilebilmektedir.

Etkili ve kararlı bir insan-robot etkileşimi sağlamak amacıyla, iki adet Myo kol bandı kullanılarak geliştirilen yöntemin detayları [19]'da sunulmuştur. Myo kol bantlarından gerçek zamanlı olarak alınan veriler, insanın kolunun kartezyen empedans parametreleri olarak kullanılmıştır. İnsanın kolu ile robotun arasındaki empedans modelini eşleştirmek için Linear Quadratic Regulator (LQR) adlı kontrol metodu kullanılmıştır. Empedans modeli yardımıyla robotun pozisyon kontrolünü sağlamak için Function Approximation Technique (FAT) adı verilen adaptif kontrol metodu tasarlanmış ve bu metod simülasyonlarla doğrulanmıştır.

Robotlar, insanla yakın olarak çalıştığı sistemlerde, çevresindeki hareketler ile değişimlerin farkında olmalı ve buna göre güvenli biçimde işlemlerini yerine getirmelidir. İnsan ve robot etkileşimi ile güvenli operasyonları gerçekleştirmek için geliştirilen stokastik optimizasyon tabanlı yöntem [20]'de açıklanmıştır. İnsanın hareketlerini tahmin ederek robotun yörüngesini değiştirmeyi sağlayan bu yöntem yardımıyla, olası çarpışma durumlarından kaçınılmakta ve işlemler güvenli biçimde gerçekleştirilmektedir.

Hibrit montaj sistemlerinde, insan ve robot arasındaki görev paylaşım planını oluşturmayı sağlayan yöntemin detayları [21]'de anlatılmıştır. Bu çalışmada, gerçekleştirilecek uygulamalar için insanlar ve robotların beraber çalıştığı sistemlerin senaryoları oluşturulmuş, bu senaryoların uygulanabilme olasılığı göz önünde bulunarak toplam üretim maliyetleri hesaplanmıştır. Üretim maliyetlerinin en düşük olduğu senaryo ile, insan ve robot arasındaki görevler belirlenmiş ve yöntemin etkinliği basit bir örnekle doğrulanmıştır.

İnsanlar ve robotların etkileşimli olarak çalışması için, insan ve robot arasındaki görev dağılımını sağlamak en önemli konulardan birisidir. Bunun için, tasarlanacak sistemdeki uygulamaların iyi biçimde tanımlanması ve görev dağılımının buna göre yapılması gerekmektedir [22]. Görev dağılımını sağlamak için yapılan prosesin süresi, ilave yatırımlar, proses kalitesi gibi unsurların yanı sıra, insanın ve robotun yapabileceği işlemler de göz önünde bulundurulmalıdır.

Endüstrideki yüksek kalite gereksinimleri, ergonomik olmayan işlemler ve nitelikli işçilerin eksikliği gibi unsurlar, otomasyon teknolojisinin kullanımını

gerektirmektedir. Ancak özellikle büyük bileşenlerin üretiminde yapılan montaj işlemleri, hâlâ insanlar tarafından elle gerçekleştirilmektedir. İnsan-robot işbirliğinin verdiği olanaklar sayesinde, insanın ve robotun becerilerine dayalı bir iş bölümü yöntemiyle bu zorluklar karşılanabilir. İnsanın ve robotun becerilerine dayanarak önerilen yöntemin detayları [23]'te verilmiştir. Önerilen yöntemle, insanın ve robotun becerileri detaylı şekilde analiz edilmiş, bu analize göre görev atamaları yapılmış ve montaj uygulamaları gerçekleştirilmiştir.

Üretim endüstrisi, insanlarla birlikte çalışan güvenli robotlara dayalı olarak yürütülen yeni montaj oluşumlarına doğru yol almaktadır. Wang ve arkadaşları tarafından [24]'te yapılan derlemede, montaj işlemleri dikkate alınarak, insan-robot işbirliği konusundaki araştırma ve sınıflandırma çalışmaları sunulmuştur. İnsan-robot işbirliği konusundaki anahtar terimlerin ve ilişkilerin tanımlanmasının yanı sıra, güvenilir işlemlerin gerçekleşmesi için çözüm yöntemleri de önerilmiştir. Yapılacak uygulama tipine göre hangi algılama, hesaplama, kontrol yapılarının kullanılması gerektiği ile ilgili bilgiler verilmiştir.

1.2 Tezin Amacı

Bu tez kapsamında, bölümümüzün Robotik ve Otomasyon Laboratuvarı'nda bulunan Mitsubishi RV-7FL-D endüstriyel robot kullanılarak, insan ve robot etkileşimi ile örnek endüstriyel ürünlerin montajlarının gerçekleştirilmesi amaçlanmıştır. Bu doğrultuda ilk olarak, insan ve robotun örnek bir endüstriyel ürünü, gerekli görevleri paylaşarak birleştirebileceği deney düzeneği oluşturulmuştur. Bu deney düzeneğinde, insanın kol hareketleri, içerisinde elektromiyografi (EMG) sensörleri bulunduran Myo isimli kol bandı ile algılanmıştır. Algılanan kol hareketleri, Raspberry Pi 3 mini bilgisayarı aracılığıyla metin formatında veriye dönüştürülmüş ve bu veriler, Ethernet haberleşmesiyle endüstriyel robota gönderilerek insan-robot etkileşimi sağlanmıştır. Myo kol bandından gelen kol hareket verileri ve robot montaj modülü üzerinde bulunan sensör bilgilerinin doğrultusunda, parça birleştirme işlemi gerçekleştirecek olan robot programlaması yapılmıştır. Güvenlik ışık bariyeri ve güvenlik rölesi kullanılarak çalışma alanının denetimi sağlanmış ve montaj uygulamaları gerçekleştirilmiştir.

Bu tez beş bölümden oluşmaktadır. İlk bölümde, tezin amacını ve literatür araştırmasını bulunduran giriş kısmı bulunmaktadır. İkinci bölümde, tez kapsamında

kullanılan cihazlar ve oluşturulan deneysel hibrit montaj sistemi tanıtılmaktadır. Üçüncü bölümde, kol hareketlerinin algılanması ve insan-robot etkileşiminin gerçekleşmesini sağlayan yöntemler anlatılmıştır. Dördüncü bölümde, insan-robot etkileşimiyle gerçekleştirilen montaj uygulamalarına yer verilmiştir. Beşinci bölümde ise çalışmadan elde edilen sonuçlar tartışılmaktadır.



2. HİBRİT MONTAJ SİSTEMLERİ VE DENEY DÜZENİĞİ

İnsan ve robotun etkileşimli olarak aynı çalışma alanında, endüstriyel ürünlerin montajını gerçekleştirebildiği sistemlere hibrit montaj sistemleri adı verilir. Hibrit montaj sistemleri, üretimde kaliteyi ve verimliliği arttırmak amacıyla ortaya çıkmıştır. Şekil 2.1’de örnek bir hibrit montaj sistemi görülmektedir [23].



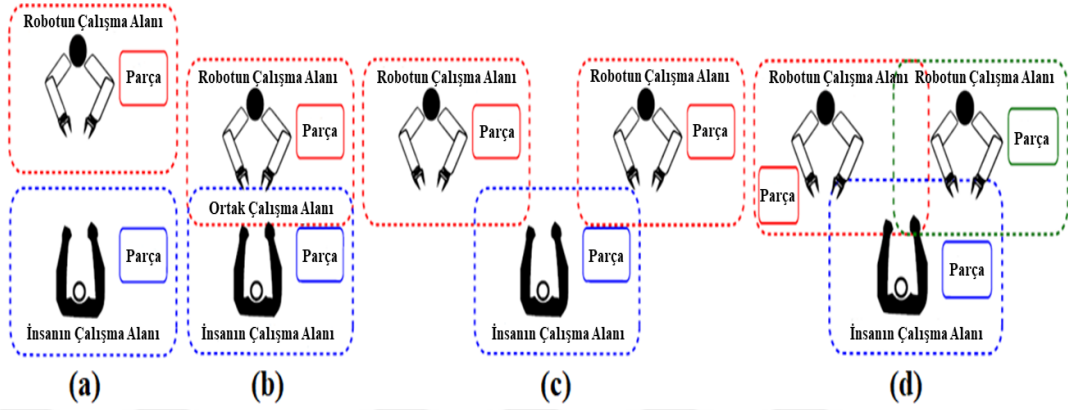
Şekil 2.1 : Hibrit montaj sistemi [23]

Hibrit sistemlerde, montaj işlemlerinin yanı sıra, parça taşıma ve yükleme işlemleri de gerçekleştirilmekte. Şekil 2.2’deki sistemde bulunan robot, parçayı kaldıracak olan operatöre destek olarak, insanın iş yükünü azaltmayı sağlamaktadır [10].



Şekil 2.2 : Parça taşımayı sağlayan hibrit sistem [10]

Hibrit montaj sistemlerindeki insan-robot etkileşiminin, farklı şekilleri ve düzeyleri bulunmaktadır. Tek robot-düşük etkileşim, tek robot-yüksek etkileşim, çoklu robot-orta etkileşim ve çoklu-robot yüksek etkileşim gibi düzeyleri bulunan hibrit montaj sistemleri Şekil 2.3'te görülmektedir [17].



Şekil 2.3 : Hibrit montaj sistemlerindeki insan-robot etkileşim düzeyleri: (a) Tek robot-düşük etkileşim (b) Tek robot-yüksek etkileşim (c) Çoklu robot-orta etkileşim (d) Çoklu robot-yüksek etkileşim [17]

Şekil 2.3(a)'da görülen tek robot-düşük etkileşim düzeyinde, robot direkt olarak insanla etkileşim halinde değildir. İşlem yapılacak parçalar insan tarafından robotun çalışacağı alana getirilirken, robot hareketsiz ve motorları durmuş biçimde beklemektedir. İnsan işlemini bitirdiğinde, çalışma alanından çıkar ve robotu çalıştırarak operasyonun gerçekleşmesini sağlar. Robotun operasyonunu gerçekleştirmesi sırasında, insan çalışma alanına girmemektedir.

Şekil 2.3(b)'de görülen tek robot-yüksek etkileşim düzeyinde, insan ve robot aynı çalışma alanında işlemlerini gerçekleştirmektedir. İnsan işlemini gerçekleştirirken, robotun motorları çalışır durumda bulunmaktadır. İnsanın ve robotun yaptığı işlemler eş zamanlı olarak gerçekleşebilir.

Şekil 2.3(c)'de görülen çoklu robot-orta etkileşim düzeyinde, insan işlemini gerçekleştirirken, robotlar otomatik moda ve motorları durmuş biçimde bulunmaktadır. İnsan, her iki robot için de parça yükleme işlemlerini gerçekleştirir ve işlemini bitirdiğinde robotların çalışma alanından çıkar. Çalışma alanından çıktığında, robotlar operasyona başlar ve kendi çalışma alanlarında işlemlerini gerçekleştirir.

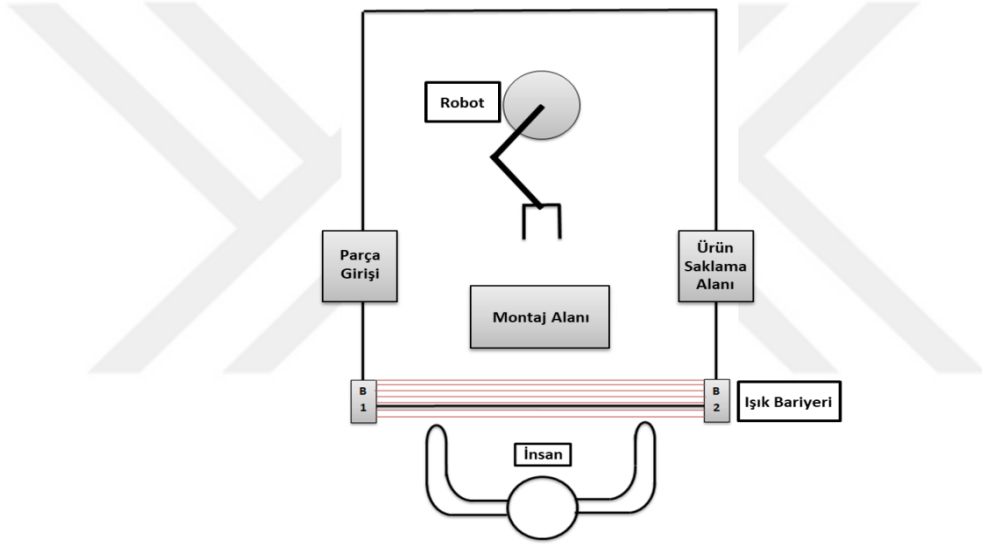
Şekil 2.3(d)'de görülen çoklu robot-yüksek etkileşim düzeyinde, insan işlemini gerçekleştirirken, robotlar otomatik moda ve motorları çalışır durumda

bulunmaktadır. İnsan tarafından işlemler gerçekleşirken, robotlar da hareket edebilir ve operasyonlarını aynı çalışma alanında gerçekleştirebilir.

Tez kapsamında, Şekil 2.3(b)'de görülen tek robot-yüksek etkileşim düzeyinde montaj işlemleri gerçekleştirilmiştir.

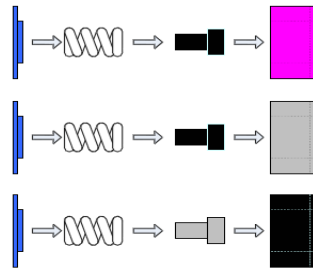
2.1 Oluşturulan Deneysel Hibrit Montaj Sistemi

Tez kapsamında gerçekleştirilecek çalışmalar için tasarlanan hibrit montaj sistemi Şekil 2.4'te görülmektedir. Bu sisteme göre, robot, karşısında bulunan insanla birlikte montaj işlemlerini gerçekleştirmekte ve montajlanan ürünleri saklama alanına bırakmaktadır.



Şekil 2.4 : Tasarlanan deneysel hibrit montaj sistemi

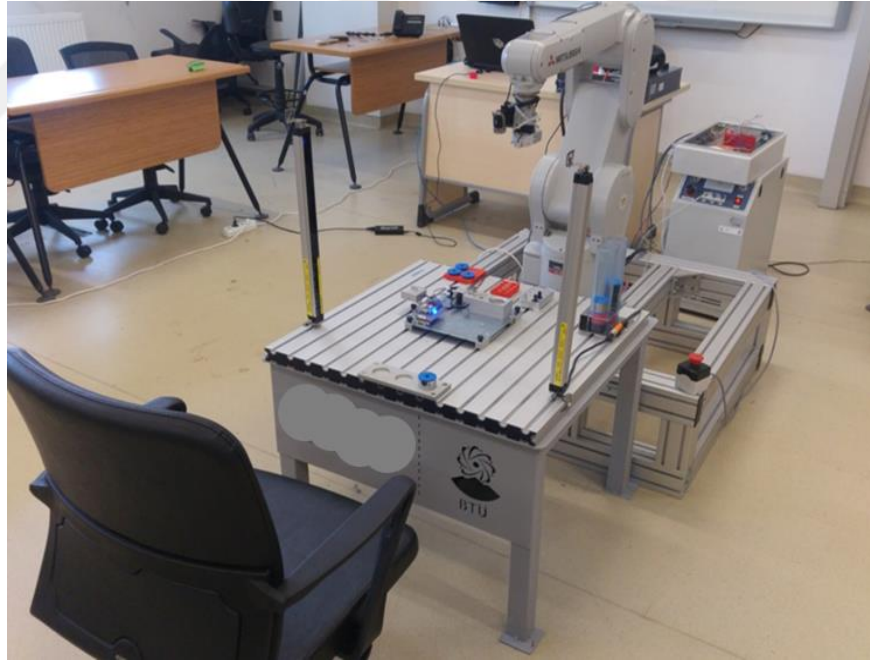
İnsan-robot işbirliği ile yapılan montaj işlemlerinde kullanılan iş parçaları Şekil 2.5'te görülmektedir. İş parçaları; kırmızı, siyah ve gri renkli gövdeler, siyah ve gri renkli pistonlar, yaylar ve mavi renkli kapaklardan oluşmaktadır. Kırmızı ve gri renkli gövdelerin içerisine siyah renkli piston, siyah gövdenin içerisine ise gri renkli piston montajlanmaktadır.



Şekil 2.5 : Montajı yapılan iş parçaları

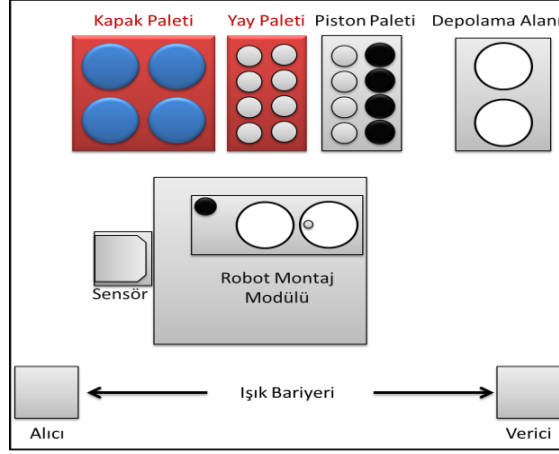
Şekil 2.5'teki iş parçalarının montajı için, ilk olarak gövdenin oryantasyon testi yapılır ve testi tamamlanan gövde montaj alanına getirilir. Montaj alanına getirilen gövde siyah renkli ise içerisine gri renkli piston, gövde kırmızı veya gri renkli ise içerisine siyah renkli piston bırakılır. Gövde içerisine pistonun ardından yay bırakılır. Montajı tamamlamak için, kapağın oryantasyon testi yapılır ve kapak, gövde üzerinde çevrilerek montaj tamamlanır. Gövde içerisine piston ve yay bırakma işlemleri insan tarafından gerçekleştirilirken, diğer işlemler robot tarafından gerçekleştirilmektedir.

İnsan ve robot işbirliğiyle montaj işlemlerini gerçekleştirmek için öncelikle profil plaka üzerine ürünlerin birleştirilmesini sağlayan robot montaj modülü eklenmiştir. Robot montaj modülünde bulunan sensörler, robot denetleyicisine bağlanmıştır. Montaj alanına girilip girilmediğinin algılanması için, profil plakanın iki kenarına ışık bariyerinin alıcı ve verici cihazları yerleştirilmiştir. Işık bariyeri ile robot denetleyicisi arasındaki bağlantı, güvenlik rölesi kullanılarak sağlanmıştır. Oluşturulan deneysel hibrit montaj sistemi Şekil 2.6'da görülmektedir.



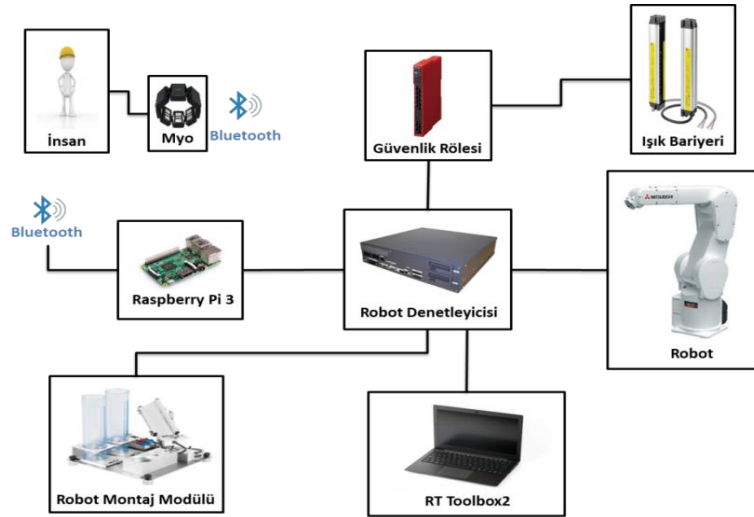
Şekil 2.6 : Deneysel hibrit montaj sistemi

Deneysel hibrit montaj sistemindeki profil plakanın üzerinde bulunan parçaların yerleşim düzeni Şekil 2.7'de görülmektedir. Kapak paleti dört adet parça bulundururken, yay paleti ve piston paleti sekiz adet parça bulundurabilmektedir. Piston paletinde dört adet siyah, dört adet gri renkli piston bulunmaktadır. Montajı yapılan parçalar, depolama alanına bırakılmaktadır.



Şekil 2.7 : Profil plakadaki parçaların yerleşim düzeni

Deneysel hibrit montaj sisteminde bulunan cihazlar arasındaki bağlantı ve haberleşme şekillerini gösteren şema, Şekil 2.8’de görülmektedir. İnsanın kolunda bulunan Myo kol bandı, Raspberry Pi 3 mini bilgisayarı ile Bluetooth üzerinden haberleşmektedir. Raspberry Pi 3 ve robotun programlanması için kullanılan dizüstü bilgisayar, Ethernet kablosu yardımıyla robot denetleyicisine bağlanmıştır. Robot denetleyicisine, robot montaj modülünde bulunan sensörler ve güvenlik ışık bariyeri bağlanmıştır. Işık bariyerinin robot denetleyicisine bağlanması, güvenlik rölesi kullanılarak gerçekleştirilmiştir.



Şekil 2.8 : Cihazlar arası bağlantı şeması

2.2 Sistemde Kullanılan Cihazlar

2.2.1 Myo kol bandı

Myo kol bandı, Thalmic Labs tarafından geliştirilen, kol kaslarındaki EMG sinyallerini ve kol hareketlerini algılayarak teknolojik cihazların kablosuz olarak

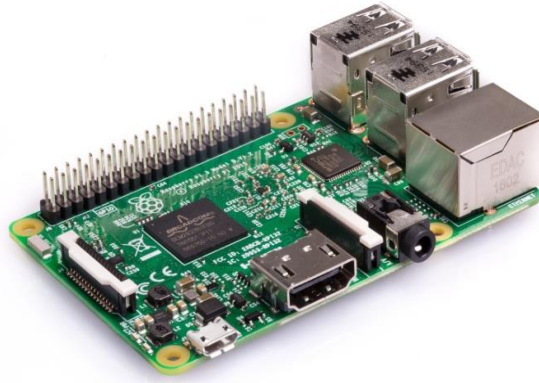
kontrol edilmesini sađlayan bir cihazdır. İerisinde bulunan jiroskop, ivmeöler, manyetometre ve EMG sensörleri aracılıđıyla kol ve kas hareketlerini algılamaktadır. Myo kol bandı, Bluetooth 4.0 teknolojisi ile haberleşmekte ve Windows, Linux, Mac OS, iOS ve Android işletim sistemli cihazlarla çalışabilmektedir. Kullanılan kol bandı Şekil 2.9’da görülmektedir [25].



Şekil 2.9 : Myo kol bandı [25]

2.2.2 Raspberry Pi 3

Raspberry Pi, kredi kartı büyüklüğünde bir mini bilgisayardır. Masaüstü veya dizüstü bilgisayarların yapabileceđi çođu işlemleri yapabilmektedir. Raspberry Pi’nin çeşitli modelleri bulunmaktadır. Tez kapsamında Şekil 2.10’da görülen Raspberry Pi 3 modeli kullanılmıştır [26]. Raspberry Pi 3 üzerindeki ARMv8 64-bit mimarisine sahip 4 çekirdekli işlemci, 1.2 Gigahertz (GHz) frekansında çalışmaktadır. Üzerinde dahili Wi-Fi ve Bluetooth gibi önemli özellikler bulunduran bu model, 1 Gigabayt (GB) belleđe sahiptir. Kullanılan Raspberry Pi 3 için işletim sistemi olarak, Raspberry Pi için özel olarak hazırlanmış bir Debian sürümü olan Raspbian kullanılmıştır.



Şekil 2.10 : Raspberry Pi 3 [26]

2.2.3 Robot montaj modülü

Robot montaj modülü, Festo tarafından geliştirilen ve özel olarak üretilmiş olan silindirik parçaların montajlarının yapılmasını sağlayan bir sistemdir. İçerisinde bir adet parça geldiğini algılayan, bir adet parçaların konumlandırma testinin yapılmasını sağlayan toplam iki adet optoelektronik sensör bulunmaktadır. Robot montaj modülünde montajı yapılabilen kırmızı, gri ve siyah renkte olmak üzere üç farklı silindir gövdesi bulunmaktadır. Kırmızı ve gri renkli silindirlerin montajı için siyah renkli pistonlar, siyah renkli silindirlerin montajı için ise gri renkli pistonlar kullanılmaktadır. Pistonun üzerine yerleştirilmek için yaylar ve montajın tamamlanmasını sağlayan silindir kapakları modül içerisinde bulunan diğer elemanlardır. Montajı yapılan parçaların depolanması için iki adet saklama alanı içeren robot montaj modülü Şekil 2.11’de görülmektedir [27].



Şekil 2.11 : Robot montaj modülü [27]

2.2.4 Güvenlik ışık bariyeri

Güvenlik ışık bariyerleri, endüstriyel ortamlarda tehlikeli bölgelerin güvenliğinin sağlanması için kullanılan optoelektronik cihazlardır. Işık bariyeri, bir adet alıcı ve bir adet verici cihazdan oluşmaktadır. Tez kapsamında kullanılan, Omron firmasına ait F3SJ serisi güvenlik ışık bariyeri Şekil 2.12’de görülmektedir [28]. Bu ışık bariyeri, insan-robot etkileşimiyle gerçekleştirilecek montaj işlemlerinde, çalışma alanına girilip girilmediğinin algılanması için kullanılmıştır.



Şekil 2.12 : Güvenlik ışık bariyeri [28]

2.2.5 Güvenlik rölesi

Güvenlik röleleri, endüstriyel ortamlarda emniyet fonksiyonlarını yerine getiren cihazlardır. Tehlikeli durumların oluştuğu zamanlarda gerekli müdahalelerde bulunarak sistem güvenliğini sağlayabilmektedirler. Güvenlik rölelerinin kontrollü şekilde hareket durdurma, hareketli korumaların pozisyonlarını izleme, acil durdurma/kapatma gibi işlevleri bulunmaktadır. Güvenlik röleleri; ışık bariyerleri, emniyet paspasları, iki el kontrol cihazları, acil durdurma butonları gibi cihazlarla beraber kullanılabilirler. Tez kapsamında kullanılan, Omron firmasına ait G9SE serisi güvenlik rölesi Şekil 2.13'te görülmektedir [28]. Bu güvenlik rölesi, ışık bariyeri ile robot denetleyicisi arasındaki bağlantıyı sağlamak üzere kullanılmıştır.



Şekil 2.13 : Güvenlik rölesi [28]

2.2.6 Endüstriyel robot

Mitsubishi Electric tarafından üretilen ve montaj işlemlerinin yapılmasında insanla beraber çalışan endüstriyel robot Şekil 2.14'te görülmektedir [29]. Mitsubishi Melfa RV-7FL-D olarak adlandırılan ve altı serbestlik dereceli olan bu robot, 7 kilogram (kg) taşıma kapasitesi ve 908 milimetre (mm) erişim mesafesine sahiptir. Ayrıca CAN-BUS, RS-232, RS-485 ve Ethernet haberleşmelerini de gerçekleştirebilmeyi sağlayan bir denetleyiciye sahiptir. Robot, kendisine ait öğretme kumandası (teach pendant) veya Mitsubishi'ye ait RT Toolbox2 adlı yazılım aracılığıyla programlanmaktadır. Mitsubishi tarafından geliştirilen Melfa Basic-IV veya Melfa Basic-V programlama dili kullanılarak robotun programlanması yapılabilmektedir. Tez kapsamında Melfa Basic-V programlama dili kullanılmıştır.



Şekil 2.14 : Endüstriyel robot [29]

3. İNSAN KOL HAREKETLERİNİN ALGILANMASI VE ROBOT ETKİLEŞİMİ

3.1 Kol Hareketlerinin Algılanması

Bu tez kapsamında, insan kol hareketlerinin algılanması için Myo isimli kol bandı kullanılmıştır. Kol hareketlerinden elde edilen verilerin anlamlandırılması işlemi ise Raspberry Pi 3 mini bilgisayar aracılığıyla yapılmıştır.

3.1.1 Myo kol bandı ile algılanabilen kol hareketleri

Myo kol bandı, içerisinde bulun sensörler yardımıyla, kol kaslarındaki EMG sinyallerini algılabilen ve bu sinyaller yardımıyla kol hareketlerini tanımlamayı sağlayan bir cihazdır. Myo kol bandı ile, önceden tanımlanmış beş adet kol hareketi algılanabilmektedir. Myo ile algılanabilen kol hareketleri Şekil 3.1’de görülmektedir [30].



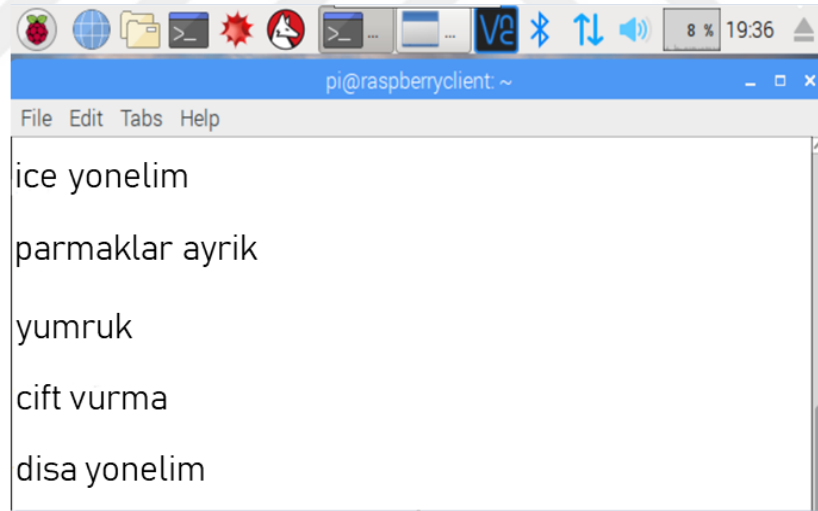
Şekil 3.1 : Myo kol bandı ile algılanabilen kol hareketleri

Şekil 3.1’de, “Çift Vurma” hareketi, orta parmak veya yüzük parmağını, baş parmağa iki kez vurarak gerçekleştirilir. Parmakları bütün halinde bilek içine doğru döndürerek “İçe Yönelim”, bilek dışına döndürerek ise “Dışa Yönelim” hareketi

gerçekleştirilir. Parmaklar yumruk yapıldığında “Yumruk” hareketi, parmaklar birbirinden ayrıldığında ise “Parmaklar Ayırık” hareketi gerçekleştirilir.

3.1.2 Myo kol bandı ile Raspberry Pi 3 arasındaki veri alışverişinin sağlanması

Myo kol bandı ile robot arasındaki iletişimi sağlamak üzere Raspberry Pi 3 mini bilgisayar kullanılmıştır. Myo kol bandı, diğer cihazlarla Bluetooth üzerinden haberleşmektedir. Raspberry Pi 3 ve Myo kol bandı arasında haberleşmeyi ve veri alışverişini sağlamak için PyoConnect isimli program kullanılmıştır [31]. PyoConnect, Linux işletim sistemine sahip bilgisayarlar için Myo kol bandı ile haberleşmeyi sağlayan, Python programlama dili ile yazılmış bir programdır. PyoConnect programı kullanılarak Myo kol bandı ve Raspberry Pi 3 arasındaki haberleşme sağlanmıştır. PyoConnect programı içinde, Myo kol bandında tanımlı hareketlerin algılanması için özel fonksiyonlar bulunmaktadır. Bu fonksiyonlar kullanılarak Python dilinde programlama yapılmış ve kol hareket verileri Raspberry Pi 3 terminal ekranına aktarılmıştır. Terminal ekranına aktarılan kol hareket verileri Şekil 3.2’de görülmektedir.

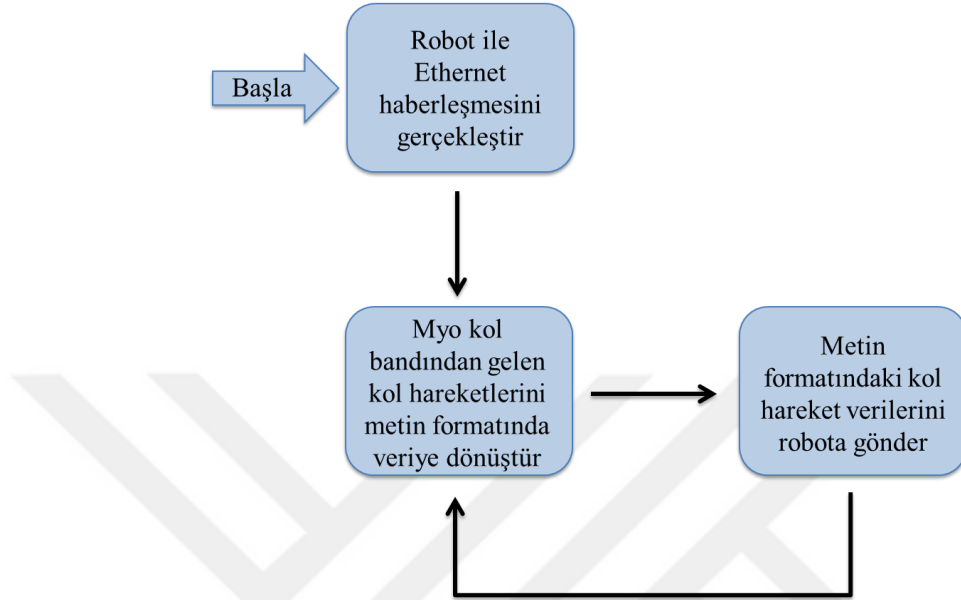


Şekil 3.2 : Terminal ekranına aktarılan kol hareketleri

3.2 Kol Hareket Verilerinin Robot Denetleyicisine Aktarımı

Robot denetleyicisi dış dünya ile Ethernet ve Transmission Control Protocol/Internet Protocol (TCP/IP) ile haberleşmektedir. Bu nedenle, Myo kol bandı ile alınan verilerin Raspberry Pi 3 mini bilgisayar tarafından robot denetleyicisine gönderilmesinde Ethernet haberleşme yöntemi kullanılmıştır. Verilerin aktarımı için robota ve Raspberry Pi’ye TCP/IP protokolü kullanılarak haberleşme programı

yazılmıştır. Ethernet haberleşme programlarında robot sunucu (server) olarak, Raspberry Pi 3 ise istemci (client) olarak seçilmiştir. Raspberry Pi 3 için istemci programı Python dilinde yazılmıştır. Kullanılan istemci algoritması Şekil 3.3'te görülmektedir.



Şekil 3.3 : Raspberry Pi 3 için kullanılan istemci algoritması

Şekil 3.3'teki algoritmaya göre, Raspberry Pi 3 mini bilgisayarını, ilk olarak Ethernet üzerinden robot denetleyicisiyle haberleşmektedir. Haberleşmenin gerçekleşmesinin ardından, Myo kol bandı tarafından algılanan kol hareket verileri, metin (string) formatında veriye dönüştürülerek robot denetleyicisine gönderilmektedir.

3.3 Robot Kontrol Algoritmasının Gerçekleştirilmesi

İnsan kol hareketlerinin algılanması ve robot denetleyicisine gönderilmesinin ardından, montaj işlemlerini gerçekleştirmek için robot kontrol algoritması oluşturulmuş ve Melfa Basic-V dilinde robot programlaması yapılmıştır. Tezin bu kısmında, ilgili algoritmalar ve uygulamalar açıklanmaktadır.

3.3.1 Melfa Basic-V robot programlama dili

Melfa Basic-V, Mitsubishi Electric tarafından geliştirilen ve Mitsubishi robotların kontrolü için kullanılan bir robot programlama dilidir. İçerisinde bulunan hareket komutları, programlama komutları, haberleşme komutları, tanımlama komutları çoklu iş (multitasking) komutları yardımıyla robota istenilen hareketlerin yaptırılmasını sağlar.

Tez kapsamında robotun programlanması için Mitsubishi Electric tarafından geliştirilen RT Toolbox2 yazılımı kullanılmıştır. Bu yazılım aracılığıyla robota hareket komutlarının gönderilmesi, giriş-çıkışların kontrolü, robotun servo motorlarının durumlarının takibi, robotların programlanması gibi birçok işlem yapılabilmektedir.

Robotun programlanmasında ve hareketlerinde kullanılan bazı komutların kısa tanımı aşağıda yapılmıştır.

MOV: Robotun, mevcut pozisyonundan hedef pozisyona en kısa sürede gitmesi için kullanılan komuttur. Eklem enterpolasyon olarak da adlandırılır.

MVS: Robotun, mevcut pozisyonundan hedef pozisyona lineer yol izleyerek gitmesi için kullanılan komuttur. Lineer enterpolasyon olarak da adlandırılır.

MVR, MVR2, MVR3, MVC: Robotun, öğretilen iki noktadan dairesel bir yörünge oluşturarak geçmesini sağlayan komutlardır. Dairesel enterpolasyon olarak da adlandırılırlar.

CNT: Robotun, birden fazla pozisyondan duraksamadan geçmesi için kullanılan komuttur.

OVRD: Tüm program hareketlerinin maksimum hıza göre yüzde cinsinden belirtilen hızda yapılmasını sağlayan komuttur.

ACCEL: Kalkış, duruş rampasını yüzde cinsinden ayarlayan komuttur.

SPD: Lineer ve dairesel enterpolasyon için robotun hızını mm/s cinsinden ayarlayan komuttur.

OADL: Optimum kalkış durum rampalarının aktif edilmesi veya edilmemesini sağlayan komuttur.

HOPEN, HCLOSE: Robotun tutucusunun açık veya kapalı pozisyonlara geçmesini sağlayan komutlardır.

WAIT: Bir sinyalin veya değişkenin, program satırındaki değerine gelene kadar programın beklemesini sağlayan komuttur.

DLY: Kendisinden sonra yazılan saniye değeri kadar programın beklemesini sağlar. Ayrıca çıkış sinyallerinin belirli bir süre aktif olması için de kullanılabilen komuttur.

GOTO: Programın satır numarasından bağımsız olarak yanında belirtilmiş etiketteki satıra atlamasını sağlayan komuttur.

IF-ELSE-THEN-ENDIF: Şartların belirtilmesi ve bu şartlara göre işlemler yaptırmak için kullanılan komutlardır.

WHILE-WEND: WHILE komutu önündeki şart sağlandığı sürece WHILE-WEND arasındaki programı çalıştırmayı sağlayan komuttur.

GOSUB-RETURN: Alt program çağırmak için kullanılan, program içinde GOSUB komutunun yanında belirtilmiş etiketteki kısma programı yönlendirmeyi sağlayan komuttur. RETURN komutu ile çağırılan yere geri gönderilir.

FOR-NEXT: Programın bir bölümünü belirli bir sayıda döngüye almak için kullanılan komuttur.

ACT: Programın herhangi bir satırından gelen sinyal, alarm, servolardaki yüklenmeler gibi kontrol değişkenlerini kullanarak, mevcut programı alt programa yönlendirmek için kullanılan komuttur. Kesme (interrupt) olarak da adlandırılır. RETURN 0 komutu ile çağırılan yere geri gönderilir.

DIM: Programa bir dizi yapısı tanımlamak için kullanılan komuttur.

P_CURR, J_CURR: Anlık pozisyon bilgisini X, Y, Z cinsinden veya açı cinsinden vermeyi sağlayan komuttur.

OPEN: Bir dosya ya da haberleşme portu açmayı sağlayan komuttur.

CLOSE: Bir dosyayı veya haberleşme portunu kapatmayı sağlayan komuttur.

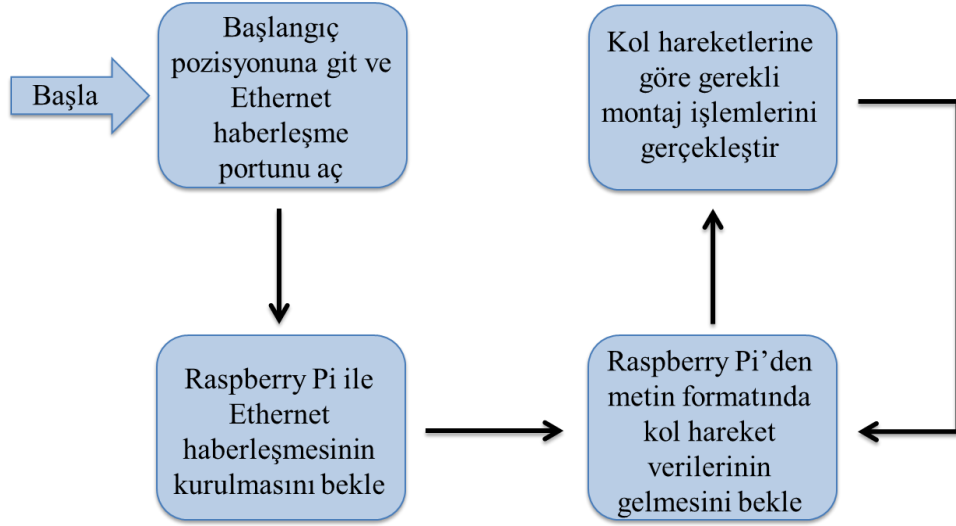
INPUT: Port üzerinden veri almayı sağlayan komuttur.

PRINT: Port üzerinden karşıdaki cihaza veri göndermeyi sağlayan komuttur.

END: Programın sonlanmasını sağlayan komuttur.

3.3.2 Robot ile Raspberry Pi 3 arasındaki veri alışverişinin sağlanması

Raspberry Pi tarafından gönderilen verilerin robot tarafından alınması için, robota TCP/IP kullanılarak Ethernet haberleşmesini sağlayacak program yazılmıştır. Bu programda robot sunucu (server) olarak görev yapmaktadır. Robota ait haberleşme programı, Melfa Basic-V programlama dili kullanılarak yazılmıştır. Robota ait sunucu algoritması Şekil 3.4'te görülmektedir.

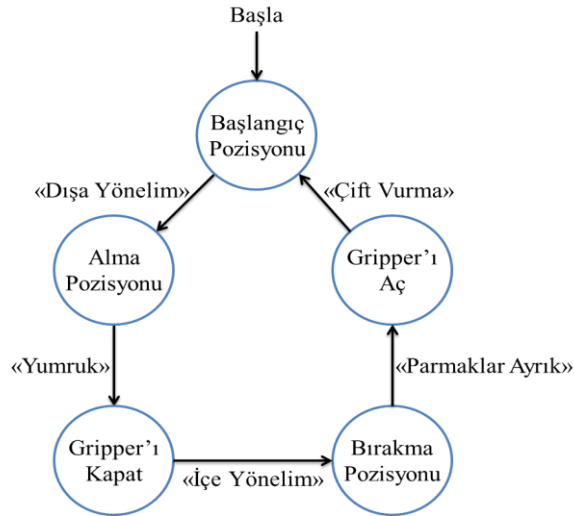


Şekil 3.4 : Robot için kullanılan sunucu algoritması

Şekil 3.4'teki algoritmaya göre, robot ilk olarak başlangıç pozisyonuna gider ve Ethernet haberleşme portunu açar. Burada, Raspberry Pi 3 mini bilgisayar ile haberleşmenin gerçekleşmesini beklemektedir. Robot, Ethernet haberleşmesinin sağlanmasının ardından, Raspberry Pi 3 mini bilgisayarından metin formatındaki verileri alarak, bu verilere göre gerekli montaj işlemlerini gerçekleştirmektedir.

3.3.3 Kol hareket verileriyle gerçekleştirilen temel robot uygulaması

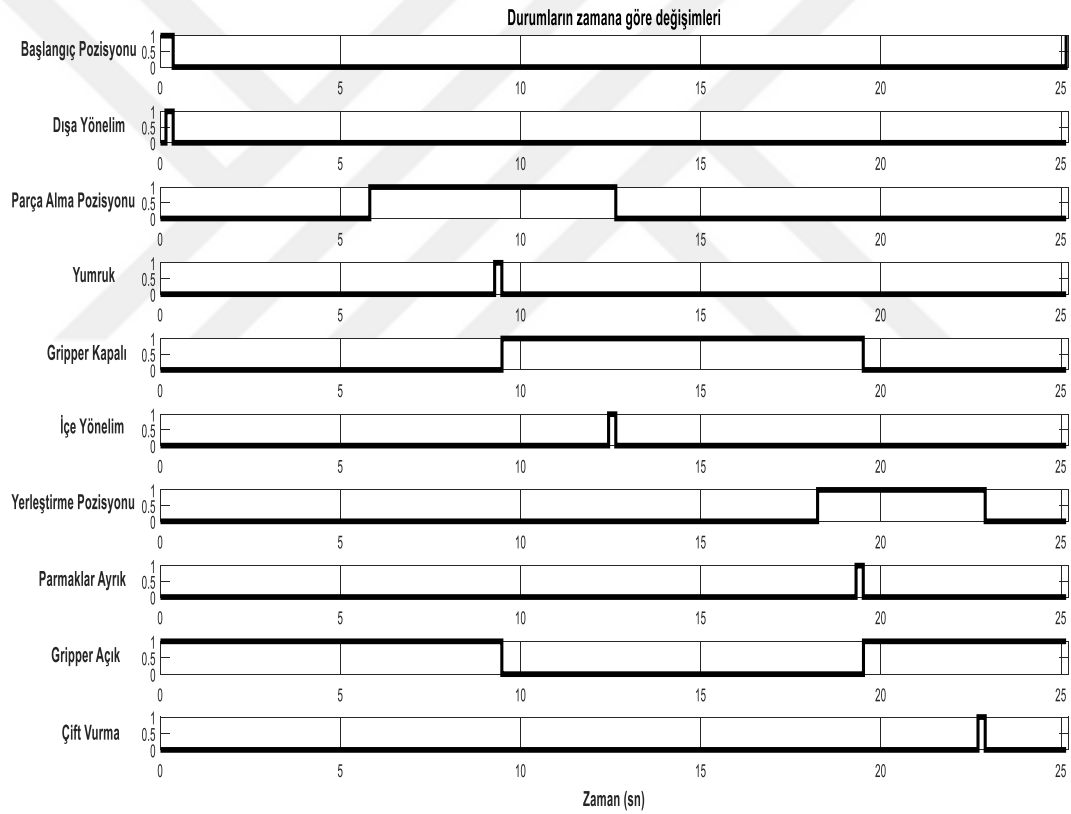
Myo kol bandından alınan verilerinin çalışıp çalışmadığının denetimi için, al-bırak (pick and place) uygulamasını gerçekleştirecek robot programlaması yapılmış ve uygulama gerçekleştirilmiştir. Uygulamayı gerçekleştirmek için Myo tarafından algılanabilen beş adet kol hareket verisi kullanılmıştır. Bu uygulama için oluşturulan algoritma Şekil 3.5'te görülmektedir.



Şekil 3.5 : Al-bırak uygulaması için oluşturulan algoritma

Şekil 3.5'teki algoritmaya göre, robot ilk olarak başlangıç pozisyonunda bulunmaktadır. İnsandan gelen “Dışa Yönelim” kol hareketiyle, robot parçayı alma pozisyonuna gider ve “Yumruk” hareketini bekler. İnsandan “Yumruk” hareketi geldiğinde, robot gripper’ı kapatır ve parçayı alır. Robot, “İçe Yönelim” kol hareketiyle bırakma pozisyonuna gider. Bırakma pozisyonundayken “Parmaklar Ayrık” hareketi geldiğinde gripper’ı açar ve parçayı bırakır. İnsandan “Çift Vurma” kol hareketi geldiğinde, robot başlangıç pozisyonuna gider ve operasyonunu tamamlar.

Bu algoritma çerçevesinde oluşturulan kod, robota aktarılarak çalışması incelenmiştir. İlgili görevlerin bir çevrim tamamlanması sonucunda robottan alınan verilere göre elde edilen durum değişim grafiği Şekil 3.6’da sunulmuştur.



Şekil 3.6 : Al-bırak uygulaması ile elde edilen durum değişim grafiği

Şekil 3.6'daki durum değişim grafiğine göre, robot ilk olarak gripper’ı açık olarak başlangıç pozisyonunda bulunmaktadır. “Dışa Yönelim” hareketi yapıldığında, parçayı alma pozisyonuna gitmek üzere harekete başlar. Grafiğin yaklaşık olarak 6. saniyesinde, ilgili durumun set olmasından dolayı robotun parça alma pozisyonuna ulaştığı anlaşılmaktadır. Robot parça alma pozisyonundayken, yaklaşık olarak 9. saniyede “Yumruk” hareketi yapıldığı görülmektedir. Bu hareketle beraber, robot

gripper'ı kapatarak parçayı alır. Robot parçayı aldıktan sonra, 13. saniye civarında “İçe Yönelim” hareketi yapıldığında parça alma pozisyonundan bırakma pozisyonuna gitmek üzere tekrar hareket geçer. Bu sırada alma pozisyonu reset olur. Robot yaklaşık olarak 18. saniyede parça bırakma pozisyonuna gelir. Bu pozisyondayken 19. saniye civarında “Parmaklar Ayrık” hareketi geldiğinde, robot gripper'ı açarak parçayı bırakır. Parçayı bırakan robot, yaklaşık 23. saniyede “Çift Vurma” hareketi geldiğinde başlangıç pozisyonuna geri dönmek üzere harekete geçer ve 25. saniye civarında başlangıç pozisyonuna gelerek uygulamasını bu şekilde tamamlar.



4. İNSAN-ROBOT ETKİLEŞİMİ İLE MONTAJ İŞLEMLERİNİN GERÇEKLEŞTİRİLMESİ

Bu bölümde, insan-robot etkileşimi ile montaj işlemlerini gerçekleştirecek robot programlama algoritmaları ve montajın sadece robot tarafından yapılması ile insan-robot işbirliğiyle yapılması arasındaki performans karşılaştırma sonuçları sunulmuştur.

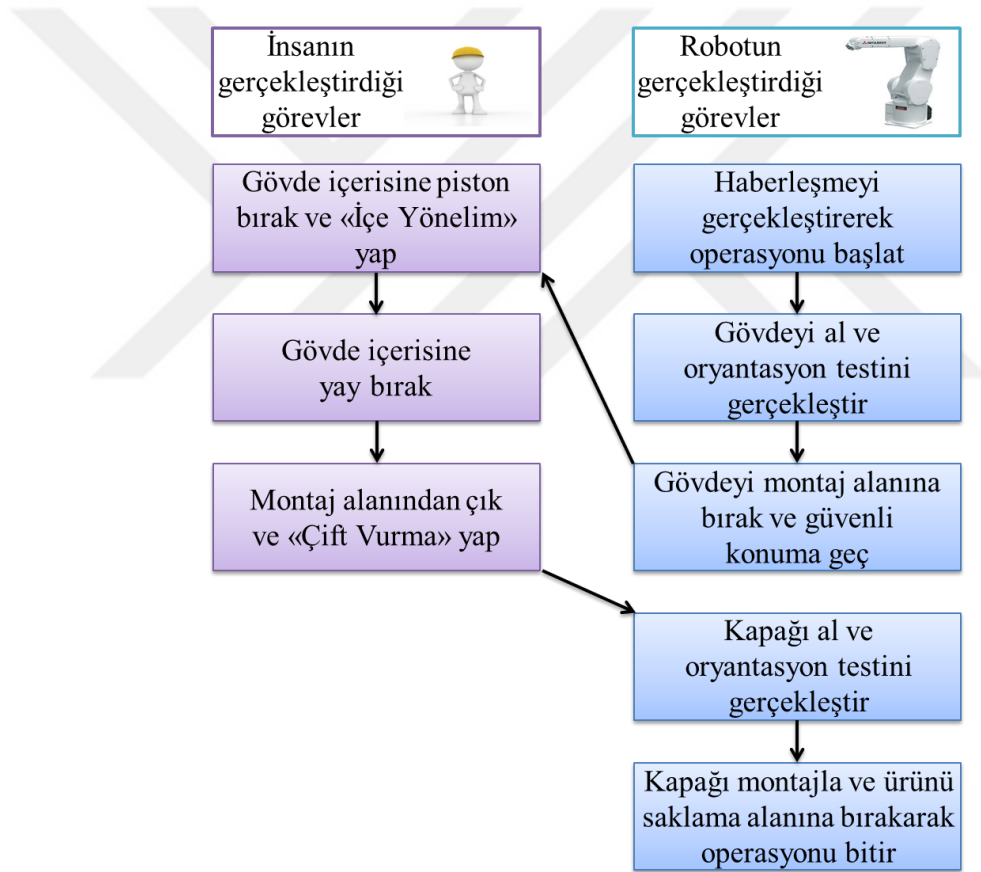
Tezin ikinci bölümünde, Şekil 2.5'te sunulan iş parçalarının, insan-robot etkileşimiyle montajı için oluşturulan yöntem aşağıdaki şekilde belirlenmiştir;

- Robot başlangıçta ana pozisyona gider. Burada Myo kol bandından veri alan Raspberry Pi 3 mini bilgisayar ile haberleşmenin sağlanmasını bekler.
- Raspberry Pi ile haberleşmeyi sağlayan robot, operasyona başlamak için giriş kısmına gövdenin bırakılmasını bekler.
- Giriş kısmına montajı yapılacak gövde bırakıldığında robot operasyona başlar. Robot, gövdeyi alarak oryantasyon testinin yapılacağı pozisyona getirir ve burada gövdenin testini gerçekleştirir. Oryantasyon testinin sonunda gövdeyi doğru konumda montaj alanına bırakır.
- Gövdeyi montaj alanına bırakan robot, kendisine önceden tanımlanmış olan güvenli pozisyona geçer ve insanın gövde içerisine piston ile yayı yerleştirmesini bekler.
- İnsan, gövde içerisine önce piston yerleştirir ve ilgili kol hareket verisini robota gönderir. Robot pistonun yerleştirilme bilgisini aldığı anda yay için bekler. İnsan, yay yerleştirdikten sonra ilgili kol hareketini robota göndererek operasyona devam edebileceğini bildirir.
- Robot, piston ve yay atılma bilgisini aldıktan sonra kapak paletinden bir adet kapak alır. Kapağı, oryantasyon testinin yapılacağı pozisyona götürür ve kapağın testini gerçekleştirir. Oryantasyon testinin sonunda kapağı doğru

konumda montaj alanına getirir ve gövde üzerinde çevirerek montajı tamamlar.

- Robot, montajı tamamlanan parçayı saklama alanına bırakır. Parçayı bıraktıktan sonra ana pozisyona geçerek yeni operasyona başlamak için gövdenin gelmesini bekler.
- Operasyon esnasında montaj alanına girilmesi durumunda, robot hızını en düşük seviyeye çeker ve işlemine ara verir. Montaj alanından çıkıldıktan sonra operasyona kaldığı yerden devam eder.

Bu montaj yöntemine göre belirlenen görev paylaşım şeması Şekil 4.1'deki görülmektedir.



Şekil 4.1 : İnsanın ve robotun gerçekleştirdiği görev paylaşım şeması

Montaj işlemlerinde görev paylaşımı oluşturulurken, robotun gerçekleştirmede zorluk yaşayabileceği görevlerin insan tarafından yapılması kararlaştırılmıştır. Montaj için kullanılan yay parçaları, esnek yapıya sahip olduğu için robot tarafından alınmasında güçlükler ortaya çıkmaktadır. Ayrıca pistonların ve yayların robot tarafından alınması için, özel bir gripper tasarımı gerekmektedir. Gövdeler ve

kapaklar, geniş çaplı parçalar olduğunda, robot tarafından rahatlıkla alınabilmektedir. Bu nedenle, gövde ve kapakların robot tarafından, pistonlar ve yayların ise insan tarafından işleme alınması planlanmıştır.

Görev paylaşımının yapılmasının ardından, robot tarafından gerçekleştirilecek işlemler için Şekil 4.2’de sunulan algoritma oluşturulmuştur. Bu algoritmaya uygun olarak, Melfa Basic-V dilinde robot programlaması yapılmıştır.



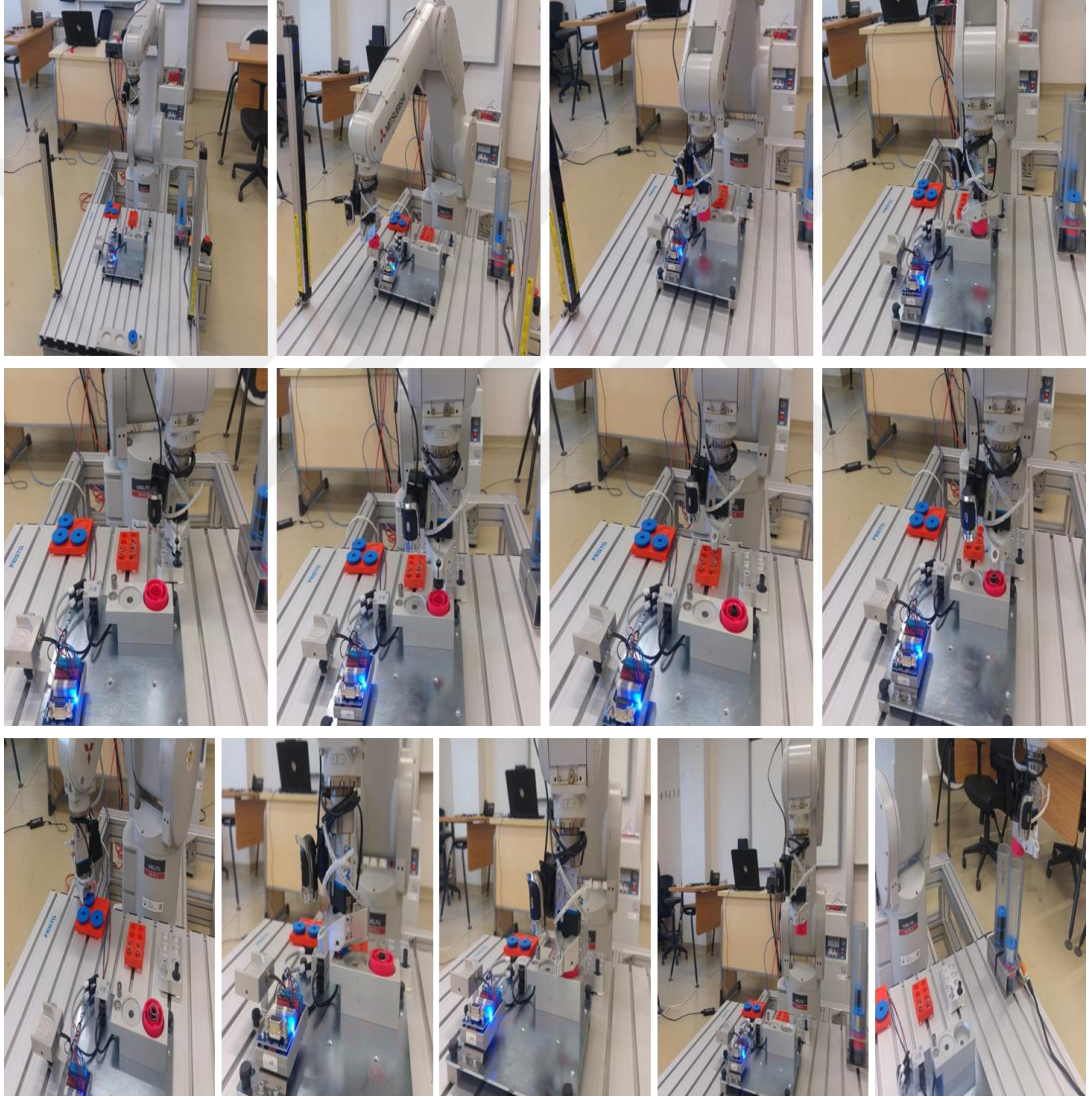
Şekil 4.2 : Robot tarafından gerçekleştirilen görev algoritması

Robotun insanla etkileşim halinde montaj yapması nedeniyle, robot programında bir güvenlik önlemi alınmıştır. Bu önlemi almak için, sistemde kullanılan güvenlik ışık bariyeri yardımıyla, robot programında bir güvenlik alt programı tanımlanmıştır. Robotun operasyonu esnasında montaj alanına girilip girilmediğinin denetimi, güvenlik alt programı tarafından yapılmıştır. Operasyon esnasında montaj alanına girilirse, güvenlik alt programı çağırılır. Güvenlik alt programında, robotun hızı düşürülür ve çalışmaya ara vermesi sağlanır. Montaj alanından çıkıldığında, robot operasyonuna kaldığı yerden devam eder.

4.1 Performans Testleri

İnsan-robot etkileşimiyle hibrit montaj işleminin performansını karşılaştırmak üzere montaj işlemlerinin süreleri incelenmiştir. Tezin bu kısmında, gerçekleştirilen karşılaştırma çalışmaları ve elde edilen sonuçlar sunulmuştur.

Performans testlerinin karşılaştırılmasında ilk olarak, ürün montajları sadece robot tarafından gerçekleştirilmektedir. İş parçalarının robot tarafından montajlanması esnasındaki görüntüler Şekil 4.3'te verilmiştir.



Şekil 4.3 : Robotun tek başına yaptığı montaj işlemleri

Robotun tek başına gerçekleştirdiği yirmi beş adet ürünün montajı esnasında geçen süreler, denetleyici içerisinde veri olarak alınmıştır. Alınan verilere göre ürün montajı esnasında geçen zaman, Çizelge 4.1'de sunulmuştur.

Çizelge 4.1 : Robotun tek başına yaptığı montaj esnasında geçen zaman.

Montajlanan Ürün Sayısı	Geçen Zaman (sn)	Montajlanan Ürün Sayısı	Geçen Zaman (sn)
1	149,553	14	2065,8
2	301,02	15	2208,9
3	446,037	16	2348,2
4	591,665	17	2491,1
5	734,499	18	2630,2
6	873,607	19	2768,6
7	1015,2	20	2904,6
8	1150,8	21	3046,2
9	1304,5	22	3184,6
10	1445,1	23	3324,2
11	1599,6	24	3462,3
12	1767,9	25	3628,1
13	1918,5		

Karşılaştırmanın daha sağlıklı yapılabilmesi amacı ile montaj işleminin tümü yerine, ilgili işlemler alt görevler olarak tanımlanmıştır. Bu kapsamda on adet alt görev tanımlanması uygun olacaktır. Bu görevler şu şekildedir:

Görev 1: Girişten gövdenin alınması,

Görev 2: Gövdenin montaj alanındaki ara bölgeye bırakılması,

Görev 3: Gövdenin oryantasyon testinin gerçekleştirilmesi,

Görev 4: Gövdenin montaj alanına bırakılması,

Görev 5: Gövde içerisine piston yerleştirilmesi,

Görev 6: Gövde içerisine yay yerleştirilmesi,

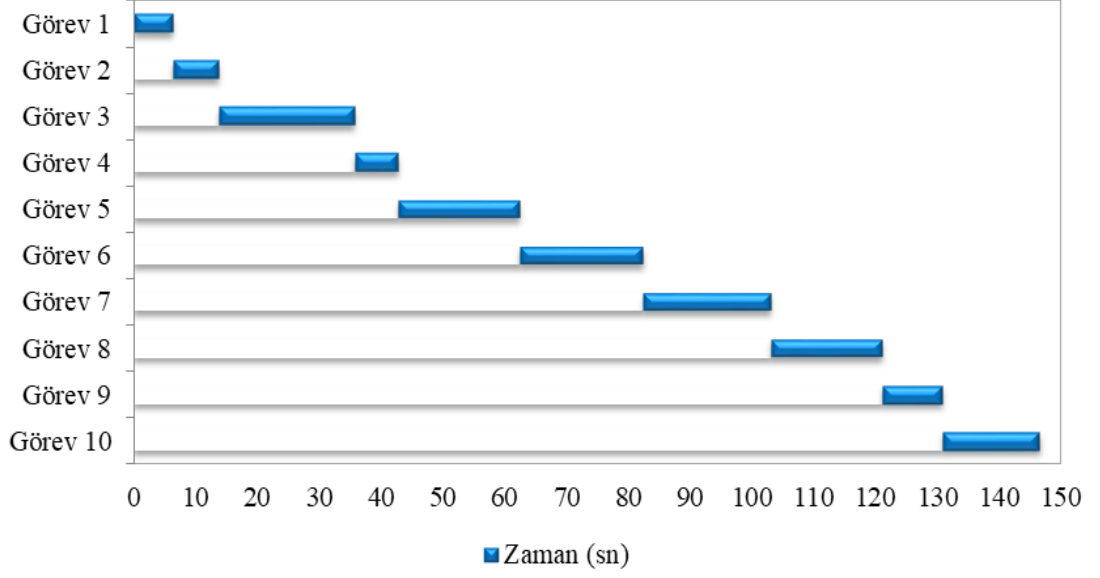
Görev 7: Kapağın test alanına getirilmesi,

Görev 8: Kapağın oryantasyon testinin gerçekleştirilmesi,

Görev 9: Kapağın gövdeye montajının sağlanması,

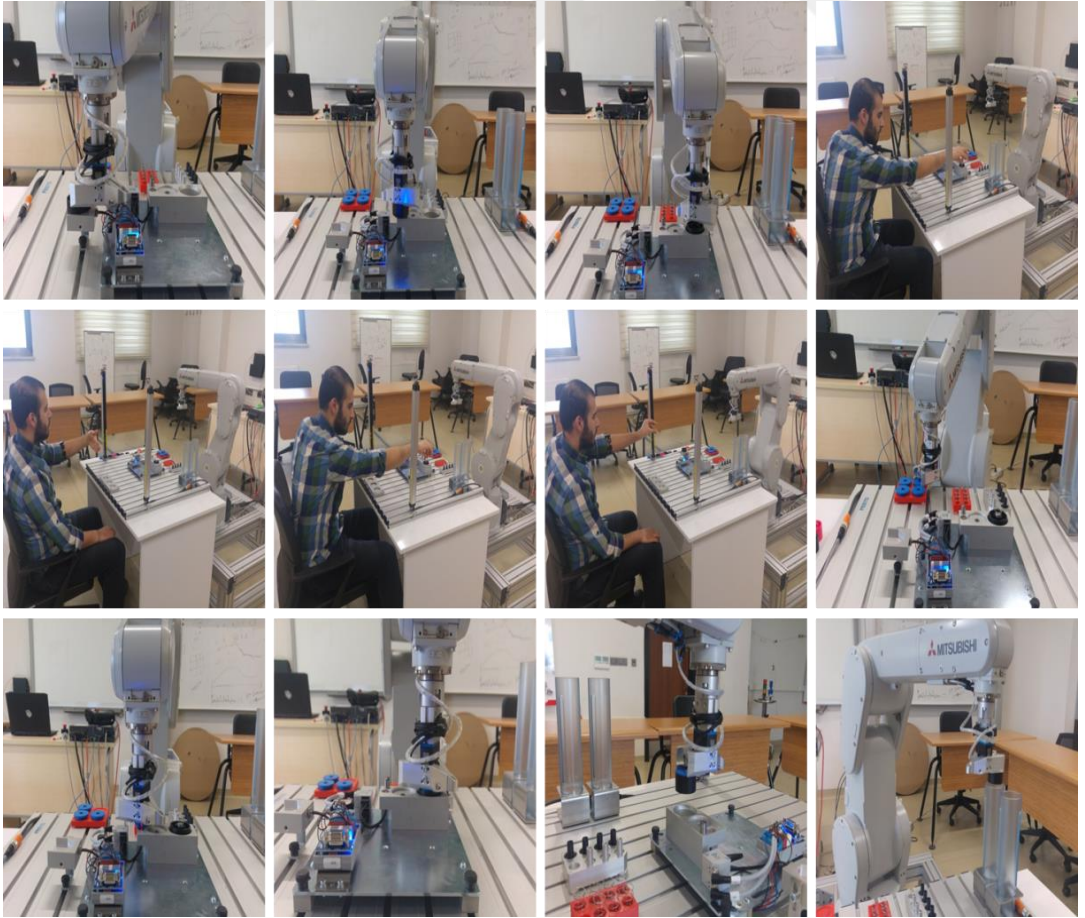
Görev 10: Montajlanan ürünün saklama alanına bırakılması.

Bu görevlere uygun olarak gerçekleştirilen yirmi beş adet ürünün montajı sonucunda, görevlerin ortalama gerçekleşme süreleri Şekil 4.4'te grafik olarak sunulmuştur.



Şekil 4.4 : Robotun tek başına gerçekleştirdiği görevlerin ortalama tamamlanma süreleri

Robotun tek başına yaptığı montaj işlemlerinden sonra, insan-robot etkileşimiyle ürün montajları gerçekleştirilmiştir. İnsan-robot etkileşimi ile yapılan montaj işlemleri esnasındaki görüntüler Şekil 4.5'te görülmektedir.



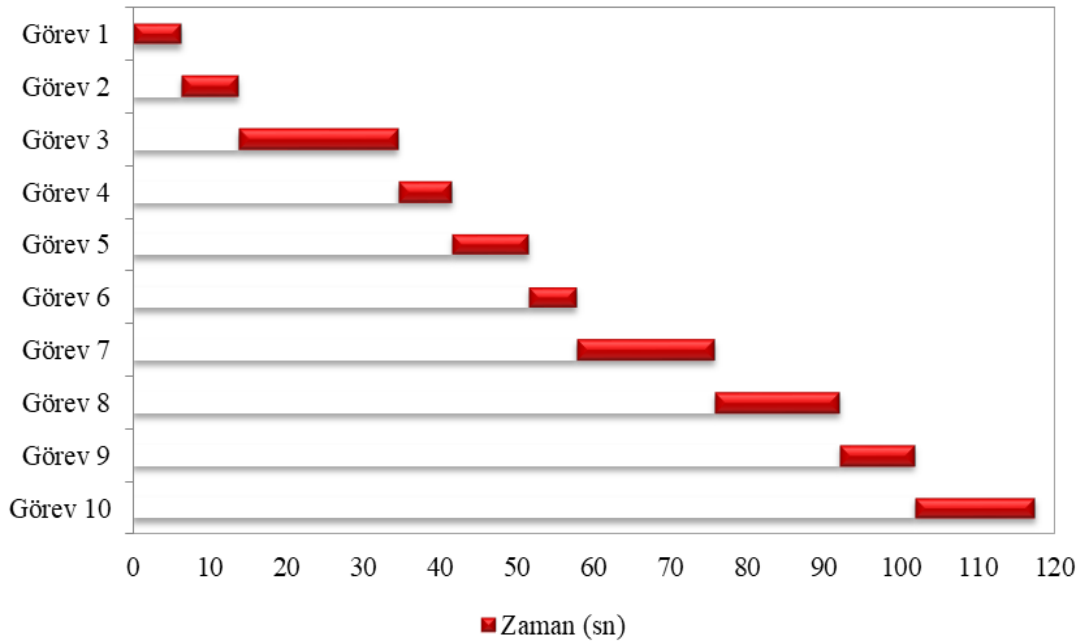
Şekil 4.5 : İnsan-robot etkileşimi ile yapılan montaj işlemleri

İnsan-robot etkileşimi ile yirmi beş adet ürünün montajı tamamlanarak, her bir ürünün montajı esnasında geçen süreler, denetleyici içerisinden veri olarak alınmıştır. Alınan verilere göre, ürün montajı esnasında geçen zaman, Çizelge 4.2’de sunulmuştur.

Çizelge 4.2 : İnsan-robot etkileşimi ile yapılan montaj esnasında geçen zaman.

Montajlanan Ürün Sayısı	Geçen Zaman (sn)	Montajlanan Ürün Sayısı	Geçen zaman (sn)
1	111,396	14	1690,2
2	239,31	15	1805,7
3	363,065	16	1910,7
4	484,217	17	2027,4
5	623,353	18	2144,4
6	761,351	19	2255,7
7	876,075	20	2379,5
8	994,055	21	2488,7
9	1116,3	22	2597,1
10	1239,9	23	2704,5
11	1352,5	24	2817,4
12	1468,1	25	2936,3
13	1577,8		

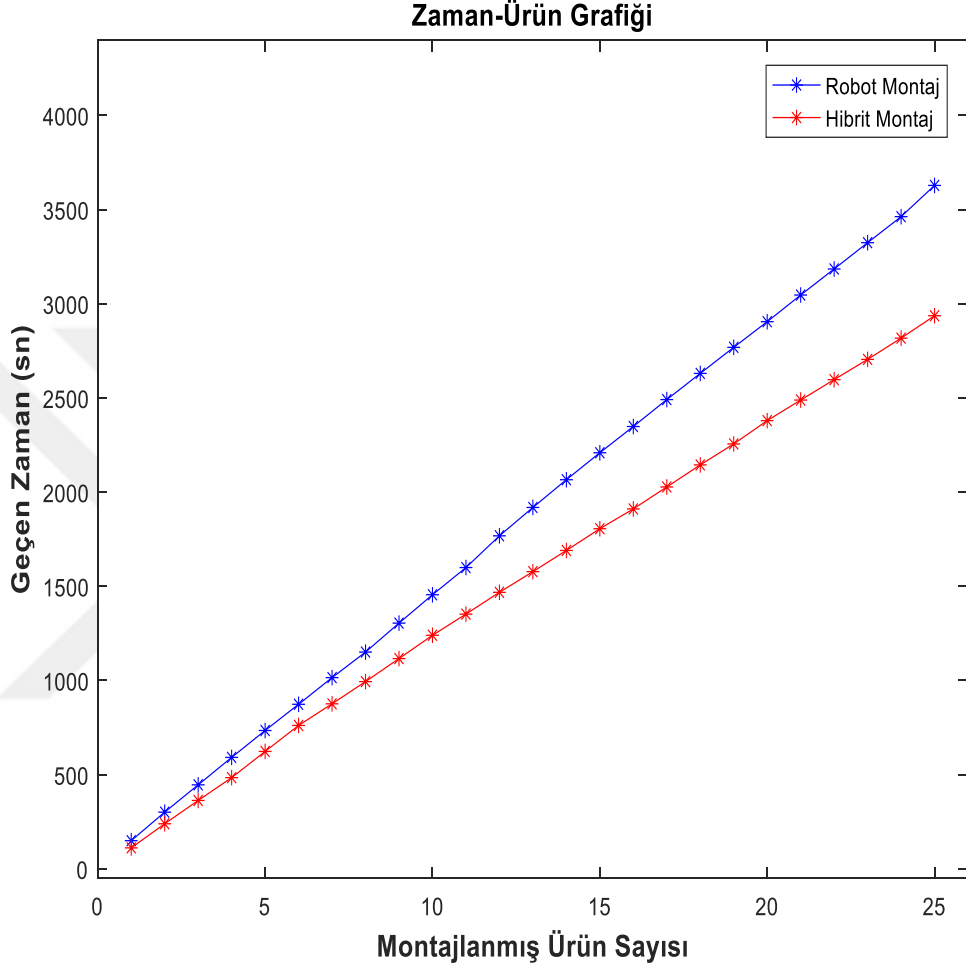
Tanımlanan görevlere göre gerçekleştirilen yirmi beş adet ürünün montajının sonucunda, görevlerin ortalama gerçekleşme süreleri Şekil 4.6’da grafik olarak sunulmuştur. Buradaki 5. ve 6. görevler, insan tarafından gerçekleştirilmiştir.



Şekil 4.6 : İnsan-robot etkileşimi ile gerçekleştirilen görevlerin ortalama tamamlanma süreleri

4.2 Montaj Sürelerinin Karşılaştırılması

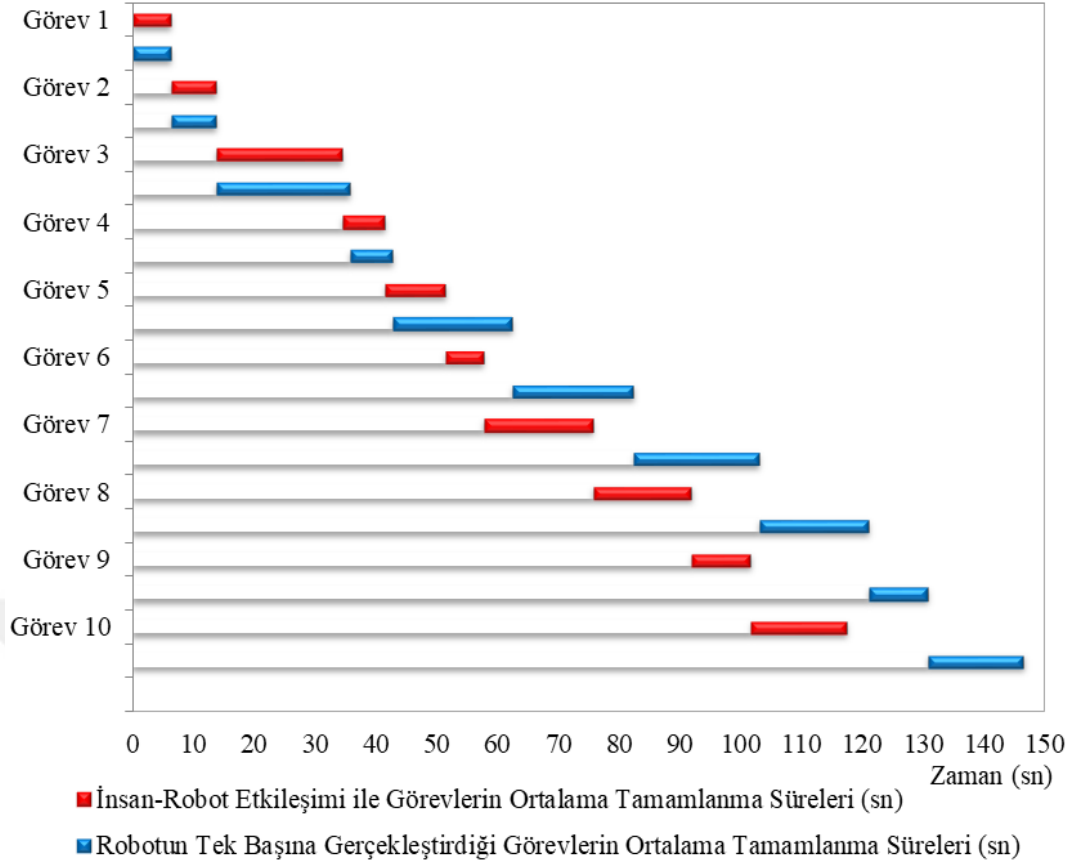
Uygulamalar sonucunda, yirmi beş adet ürünün montajlarının tamamlanma süreleri ve görevlerin ortalama tamamlanma süreleri karşılaştırılmıştır. Buna göre montajı tamamlanan ürün sayısına göre geçen zamanın grafiği Şekil 4.7’de görülmektedir.



Şekil 4.7 : Montajı tamamlanan ürün sayısına göre geçen zaman

Şekil 4.7’de verilen grafikte görüldüğü üzere, insan-robot etkileşimi ile yapılan yirmi beş adet ürünün montajı yaklaşık 3000 saniye, robotun tek başına yaptığı ürünün montajı ise yaklaşık 3600 saniye sürmektedir. Sonuç olarak, insan-robot etkileşimi ile yapılan montajlar, robotun tek başına yaptığı ürün montajlarından daha hızlı gerçekleşmektedir. Montajlanan ürün sayısı arttıkça, iki montaj biçimi arasındaki zaman farkı giderek açılmaktadır.

Atanan görevlerin ortalama gerçekleşme zamanlarını gösteren grafik Şekil 4.8’de sunulmuştur.



Şekil 4.8 : Görevlerin tamamlanması esnasında geçen süreler

Şekil 4.8'deki grafikte görüldüğü üzere, insan-robot etkileşimi ile gerçekleştirilen montajdaki görevler ortalama 120 saniyede tamamlanırken, robotun tek başına gerçekleştirdiği montajdaki görevler ortalama 150 saniyede tamamlanmaktadır. Montajdaki beşinci ve altıncı görevin insan tarafından gerçekleştirilmesi, ürün montajının tamamlanmasında iyi bir zaman kazandırmaktadır. Her bir görevin ortalama tamamlanma süresi, Çizelge 4.3'te karşılaştırmalı olarak sunulmuştur.

Çizelge 4.3 : Görevlerin ortalama tamamlanma süresi.

Görevler	Robot Montaj (sn)	Hibrit Montaj (sn)
1	5,9815	5,9803
2	7,0618	7,0615
3	20,7757	19,577
4	6,6796	6,6827
5	18,5522	9,4487
6	18,9392	5,8368
7	19,4739	17,0421
8	17,0551	15,384
9	9,2011	9,2004
10	14,9405	14,8572

5. SONUÇ VE ÖNERİLER

Tez kapsamında, endüstriyel ürün montajının insan ve robot etkileşimi ile hibrit olarak yapılması amaçlanmıştır. Bu doğrultuda, deneysel bir hibrit montaj sistemi oluşturulmuştur. İnsan ve robotun etkileşimi, üzerinde EMG sensörleri bulunan kol bandı aracılığıyla sağlanmıştır. Bu bant, kaslardan aldığı sinyalleri anlamlandırmakta ve kablosuz olarak diğer cihazlarla iletişim kurabilmektedir.

Hibrit montaj sisteminde robotun ve insanın yerine getireceği görevler, montajlanacak ürün özellikleri ve robot kabiliyetleri dikkate alınarak insan ile robot arasında paylaştırılmıştır. Robot ile kol bandı arasında haberleşme sağlanmış ve montajı gerçekleştirecek algoritma robota programlanmıştır.

Performansın karşılaştırılması açısından, montajın sadece robot tarafından yapılması ve insan-robot etkileşimi ile hibrit olarak yapılması esnasında geçen süreler, toplam süre ve ortalama görev süresi olarak incelenmiştir. Yapılan uygulamalar sonucunda, hibrit montaj senaryosunun, hem toplam süre ölçüğünde, hem de görev bazında daha hızlı gerçekleştiği gözlemlenmiştir.

Sanayi ölçüğünde hibrit montaj senaryoları için görev paylaşımının, robot özellikleri, çevresel kısıtlar, tutucu özellikleri ve parametreleri, operasyonel kısıtlar ve benzeri birçok kısıtın dikkate alınarak optimize edilmesi gerekmektedir. İleriki çalışmalarda bu konular üzerine araştırmalar gerçekleştirilecektir.

KAYNAKLAR

- [1] Peshkin, M. A., Colgate, J. E., Wannasuphprasit, W., Moore, C. A., Gillespie, R. B., & Akella, P. (2001). Cobot Architecture, *IEEE Transactions on Robotics and Automation*, 17(4), 377-390.
- [2] Colgate, J. E., Peshkin, M. A., & Wannasuphprasit, W. (1996). Nonholonomic Haptic Display, *Proceedings of the IEEE International Conference on Robotics and Automation*, (Vol. 1, pp. 539-544). Minneapolis, April 22-28.
- [3] Guizzo, E., & Ackerman, E. (2012). How Rethink Robotics Built Its New Baxter Robot Worker, *IEEE spectrum*, 18.
- [4] Guizzo, E. (2015). Sawyer: Rethink Robotics Unveils New Robot, *Spectrum, IEEE*.
- [5] Vogel, C., Walter, C., & Elkmann, N. (2015). A Projection-based Sensor System for Ensuring Safety While Grasping and Transporting Objects by an Industrial Robot, *2015 IEEE International Symposium on Robotics and Intelligent Sensors (IEEE IRIS2015)*, (pp. 271-277). Langkawi, October 18-20.
- [6] Krüger, J., Lien, T. K., & Verl, A. (2009). Cooperation of Human and Machines in Assembly Lines, *CIRP Annals-Manufacturing Technology*, 58(2), 628-646.
- [7] Cherubini, A., Passama, R., Crosnier, A., Lasnier, A., & Fraisse, P. (2016). Collaborative Manufacturing with Physical Human-Robot Interaction, *Robotics and Computer-Integrated Manufacturing*, 40, 1-13.
- [8] Johannsmeier, L., & Haddadin, S. (2017). A Hierarchical Human-Robot Interaction-planning Framework for Task Allocation in Collaborative Industrial Assembly Processes, *IEEE Robotics and Automation Letters*, 2(1), 41-48.
- [9] Ding, H., Schipper, M., & Matthias, B. (2013). Collaborative Behavior Design of Industrial Robots for Multiple Human-Robot Collaboration, *2013 44th International Symposium on Robotics (ISR)*, (pp. 1-6). Seoul, October 24-26.
- [10] Gecks, T., & Henrich, D. (2005). Human-Robot Cooperation: Safe Pick-and-Place Operations, *2005 IEEE International Workshop on Robots and Human Interactive Communication (ROMAN 2005)*, (pp. 549-554). Nashville, August 13-15.
- [11] Masinga, P., Campbell, H., & Trimble, J. A. (2015). A Framework for Human Collaborative Robots, Operations in South African Automotive Industry, *2015 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*, (pp. 1494-1497). Singapore, December 6-9.
- [12] Shi, J., Jimmerson, G., Pearson, T., & Menassa, R. (2012). Levels of Human and Robot Collaboration for Automotive Manufacturing, *Proceedings of the 27*

Workshop on Performance Metrics for Intelligent Systems, (pp. 95-100). College Park, March 20-22.

[13] **Zanchettin, A. M. & Rocco, P.** (2013). Path-consistent Safety in Mixed Human-Robot Collaborative Manufacturing Environments, *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, (pp. 1131-1136). Tokyo, November 3-7.

[14] **Shen, Y. & Reinhart, G.** (2013). Safe Assembly Motion – A Novel Approach for Applying Human-Robot Co-operation in Hybrid Assembly Systems, *2013 IEEE International Conference on Mechatronics and Automation (ICMA)*, (pp. 7-12), Takamatsu, August 4-7.

[15] **Avanzini, G. B., Ceriani, N. M., Zanchettin, A. M., Rocco, P., & Bascetta, L.** (2014). Safety Control of Industrial Robots Based on a Distributed Distance Sensor, *IEEE Transactions on Control Systems Technology*, 22(6), 2127-2140.

[16] **Zanchettin, A. M., Ceriani, N. M., Rocco, P., Ding, H., & Matthias, B.** (2016). Safety in Human-Robot Collaborative Manufacturing Environments: Metrics and Control, *IEEE Transactions on Automation Science and Engineering*, 13(2), 882-893.

[17] **Sadrifaridpour, B., Saeidi, H., & Wang, Y.** (2016). An Integrated Framework for Human-Robot Collaborative Assembly in Hybrid Manufacturing Cells, *2016 IEEE International Conference on Automation Science and Engineering (CASE)*, (pp. 462-467). Fort Worth, August 21-25.

[18] **Liu, T., Wang, J., & Meng, M. Q. H.** (2014). Human Robot Cooperation Based on Human Intention Inference, *2014 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, (pp. 350-355). Bali, December 5-10.

[19] **Chen, X., Yang, C., Fang, C., & Li, Z.** (2017). Impedance Matching Strategy for Physical Human Robot Interaction Control, *2017 13th IEEE Conference on Automation Science and Engineering (CASE)*, (pp. 138-144). Xi'an, August 20-23.

[20] **Oguz, O. S., Sari, O. C., Dinh, K. H., & Wolherr, D.** (2017). Progressive Stochastic Motion Planning for Human-Robot Interaction, *2017 26th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, (pp. 1194-1201). Lisbon, August 28-September 1.

[21] **Takata, S. & Hirano, T.** (2011). Human and Robot Allocation Method for Hybrid Assembly Systems, *CIRP Annals-Manufacturing Technology*, 60 (1), 9-12.

[22] **Ranz, F., Hummel, V., & Sihn, W.** (2017). Capability-based Task Allocation in Human-Robot Collaboration, *Procedia Manufacturing*, 9, 182-189.

[23] **Müller, R., Vette, M., & Mailahn, O.** (2016). Process-oriented Task Assignment for Assembly Processes with Human-Robot Interaction, *Procedia CIRP*, 44, 210-215.

[24] **Wang, X. V., Kemény, Z., Váncza, J., & Wang, L.** (2017). Human-Robot Collaborative Assembly in Cyber-physical Production: Classification Framework and Implementation, *CIRP Annals*, 66 (1), 5-8.

[25] **Myo Gesture Control Armband | Wearable Technology by Thalmic Labs.** (t.y.). Erişim: 5 Şubat 2018, <https://www.myo.com/>

[26] **Raspberry Pi – Teach, Learn, and Make with Raspberry Pi.** (t.y.). Eriřim: 5 Őubat 2018, <https://www.raspberrypi.org/>

[27] **Ana Sayfa | Festo Turkey.** (t.y.). Eriřim: 5 Őubat 2018, https://www.festo.com/cms/tr_tr/index.htm

[28] **Omron Trkiye.** (t.y.). Eriřim: 5 Őubat 2018, <https://omron.com.tr/tr/home>

[29] **MITSUBISHI ELECTRIC TRKİYE.** (t.y.). Eriřim: 5 Őubat 2018, <http://tr.mitsubishielectric.com/tr/>

[30] **PyoConnect – Myo Market.** (t.y.). Eriřim: 5 Őubat 2018, <https://market.myo.com/app/559adc2ee4b0f2c8982c9138/pyoconnect>

[31] **PyoConnect – MyoConnect for Linux.** (t.y.). Eriřim: 5 Őubat 2018, <http://www.fernandocsentino.net/pyoconnect/>



EKLER

EK A: Melfa Basic-V Robot Programlama Kodu

```
"" ----- Robot Collaborative Montaj Algoritması ----- ""
"" ----- Pozisyonların Bilgilendirilmesi ----- ""
"" --- Phome: Home pozisyon
"" --- Ppickcover(8): Kapakların bulunduğu 8 adet pozisyon. 1-5, 2-6, 3-7, 4-8
pozisyonları aynı olacak
"" --- Pdrop(8): Parçaların bırakılacağı 8 adet pozisyon. 1-3-5-7, 2-4-6-8 pozisyonları
aynı olacak
"" --- Ppickbody_back: Gövdeyi gripperın arka kısmıyla almak için öğretilen
pozisyon
"" --- Pplacebody_back: Gövdeyi gripperın arka kısmıyla bırakmak için öğretilen
pozisyon
"" --- Ppickbody_mid: Gövdeyi gripperın orta kısmıyla almak için öğretilen pozisyon
"" --- Ptestbody: Gövdeyi test etmek için öğretilen pozisyon
"" --- Passemble: Gövdenin montaj alanına bırakılması için öğretilen pozisyon
"" --- Passemble2: Kapağın montaj alanına bırakılması için öğretilen pozisyon
"" --- Pplacecover_back: Kapağın, gripperın arka kısmıyla test alanına bırakılması
için öğretilen pozisyon
"" --- Pplacecover_mid: Kapağın, gripperın orta kısmıyla tutulması için öğretilen
pozisyon
"" --- Ppickpart: Gripperın arka kısmıyla montajlanmış parçayı almak için öğretilen
pozisyon
"" --- Psecure: Robotu güvenli bölgede beklemesi için öğretilen pozisyon
Dim Ppickcover(8)
Dim Pplacecover_back(8)
Dim Mrobot(10,11)
Dim Probot(10,11)
Dim Pdrop(8)
Def Pos Passemble
Def Pos Passemble2
Def Pos Pdrop_aux
Def Pos Phome
Def Pos Ppickbody_back
Def Pos Ppickbody_mid
Def Pos Ppickpart
Def Pos Pplacebody_back
Def Pos Pplacecover_mid
Def Pos Psecure
Def Pos Ptestbody
Def Jnt Jrotate
Def Jnt Jrotate46
Def Jnt Jrotate50
```

Def Jnt Jrotate65
 Passemble=(+678.99,-44.78,+296.20,-177.08,-0.41,-99.62)(7,0)
 Passemble2=(+679.77,-45.65,+284.24,-177.07,-0.41,-99.62)(7,0)
 Pdrop_aux=(+672.32,+11.22,+498.84,-177.19,-0.35,-101.27)(7,0)
 Phome=(+425.27,-93.43,+607.43,-177.08,-0.41,-99.62)(7,0)
 Ppickbody_back=(+694.27,-156.61,+283.82,+179.99,-2.85,-9.90)(7,15728640)
 Ppickbody_mid=(+675.41,-92.55,+296.15,-177.08,-0.41,-99.62)(7,0)
 Ppickpart=(+717.16,-51.59,+284.66,-177.18,-0.35,-101.27)(7,0)
 Pplacebody_back=(+714.91,-96.02,+295.43,-177.08,-0.41,-99.62)(7,0)
 Pplacecover_mid=(+657.51,-123.32,+284.84,-177.07,-0.41,-99.62)(7,0)
 Psecure=(+111.89,-391.93,+657.10,-178.82,-0.96,-104.37)(7,0)
 Ptestbody=(+687.50,-124.51,+302.93,-177.09,-0.41,-99.62)(7,0)
 Pdrop(1)=(+570.95,+204.68,+461.55,-177.18,-0.35,-101.27)(7,0)
 Pdrop(2)=(+640.88,+198.26,+460.02,-177.18,-0.35,-101.26)(7,0)
 Pdrop(3)=(+570.95,+204.68,+461.55,-177.18,-0.35,-101.27)(7,0)
 Pdrop(4)=(+640.88,+198.26,+460.02,-177.18,-0.35,-101.26)(7,0)
 Pdrop(5)=(+570.95,+204.68,+461.55,-177.18,-0.35,-101.27)(7,0)
 Pdrop(6)=(+640.88,+198.26,+460.02,-177.18,-0.35,-101.26)(7,0)
 Pdrop(7)=(+570.95,+204.68,+461.55,-177.18,-0.35,-101.27)(7,0)
 Pdrop(8)=(+640.88,+198.26,+460.02,-177.18,-0.35,-101.26)(7,0)
 Ppickcover(1)=(+356.25,-103.69,+228.25,+177.77,-1.94,+31.46)(7,15728640)
 Ppickcover(2)=(+283.34,-271.31,+226.66,+178.07,+2.25,+121.78)(7,0)
 Ppickcover(3)=(+514.34,-158.31,+234.14,-178.33,-2.44,-52.07)(7,0)
 Ppickcover(4)=(+459.68,-328.55,+231.08,-177.53,+1.63,-141.08)(7,0)
 Ppickcover(5)=(+356.25,-103.69,+228.25,+177.77,-1.94,+31.46)(7,15728640)
 Ppickcover(6)=(+283.34,-271.31,+226.66,+178.07,+2.25,+121.78)(7,0)
 Ppickcover(7)=(+514.34,-158.31,+234.14,-178.33,-2.44,-52.07)(7,0)
 Ppickcover(8)=(+459.68,-328.55,+231.08,-177.53,+1.63,-141.08)(7,0)
 Pplacecover_back(1)=(+696.43,-126.05,+292.70,-177.07,-0.43,-99.24)(7,0)
 Pplacecover_back(2)=(+697.85,-126.21,+294.13,-177.07,-0.43,-99.24)(7,0)
 Pplacecover_back(3)=(+696.14,-125.64,+293.79,-177.07,-0.43,-99.24)(7,0)
 Pplacecover_back(4)=(+696.08,-125.43,+291.89,-177.07,-0.43,-99.24)(7,0)
 Pplacecover_back(5)=(+696.43,-126.05,+292.70,-177.07,-0.43,-99.24)(7,0)
 Pplacecover_back(6)=(+697.85,-126.21,+294.13,-177.07,-0.43,-99.24)(7,0)
 Pplacecover_back(7)=(+696.14,-125.64,+293.79,-177.07,-0.43,-99.24)(7,0)
 Pplacecover_back(8)=(+696.08,-125.43,+291.89,-177.07,-0.43,-99.24)(7,0)
 Jrotate=(+0.00,+0.00,+0.00,+0.00,+0.00,+1.00,+0.00,+0.00)
 Jrotate46=(+0.00,+0.00,+0.00,+0.00,+0.00,+48.50,+0.00,+0.00)
 Jrotate50=(+0.00,+0.00,+0.00,+0.00,+0.00,+50.00,+0.00,+0.00)
 Jrotate65=(+0.00,+0.00,+0.00,+0.00,+0.00,+65.00,+0.00,+0.00)
 OAdl Off
 Ovrđ 10
 Mov Phome
 Dly 1
 Open "COM2:" As #1
 Wait M_Open(1)=1 ' Baęlantının kurulmasını bekle
 ' Baęlantı kuruldu
 *Operasyon
 HOpen 1
 Dly 0.3

```

HClose 1
Dly 0.2
HOpen 1
Dly 0.5
Ovrd 10
Mov Phome
Dly 2
M_Timer(2)=0 ' Timer2'yi sıfırla
M_01#=5 ' Yardımcı değişken
M1=1 ' Yardımcı değişken
Open "cobot_veriler.txt" For Append As #2
*Basla ' Operasyona başlangıç
Def Act 1, M_In(3)=0 GoSub *Guvencik ' Güvenlik alt program tanımlaması
"" Gövde Alma Programı ""
If M_01#=5 And M_In(3)=1 Then ' Gövde almaya git
*GovdeAl ' Gövde alma alt programı
Act 1=1
Wait M_In(1)=1
Act 1=0
Mrobot(M1,1)=M_Timer(2) ' Gövdenin girişe bırakıldığı zaman
Probot(M1,1)=P_Curr
P1=P_Curr
Mcobotx1=P1.X
Mcoboty1=P1.Y
Mcobotz1=P1.Z
Print #2, "Tcobot(",M1,",1)=",Mrobot(M1,1),"/1000",";"
Print #2, "Pcobot_x(",M1,",1)=",Mcobotx1,";"
Print #2, "Pcobot_y(",M1,",1)=",Mcoboty1,";"
Print #2, "Pcobot_z(",M1,",1)=",Mcobotz1,";"
Act 1=1
HOpen 1
Ovrd 30
" Gripperın arka kısmıyla gövdeyi al
Mov Ppickbody_back, -50
Ovrd 3
Mvs Ppickbody_back
Dly 0.5
HClose 1
Mrobot(M1,2)=M_Timer(2) ' Gövdenin girişten alındığı zaman
Probot(M1,2)=P_Curr
P2=P_Curr
Mcobotx2=P2.X
Mcoboty2=P2.Y
Mcobotz2=P2.Z
Print #2, "Tcobot(",M1,",2)=",Mrobot(M1,2),"/1000",";"
Print #2, "Pcobot_x(",M1,",2)=",Mcobotx2,";"
Print #2, "Pcobot_y(",M1,",2)=",Mcoboty2,";"
Print #2, "Pcobot_z(",M1,",2)=",Mcobotz2,";"
Dly 0.5
Mvs Ppickbody_back, -50

```

```

Ovrđ 50
" Gripperın arka kısmındaki gövdeyi ortadan almak için uygun yere bırak
Mov Pplacebody_back, -50
Ovrđ 5
Mvs Pplacebody_back
Dly 0.5
HOpen 1
Mrobot(M1,3)=M_Timer(2) ' Gövdenin buffera bırakıldığı zaman
Probot(M1,3)=P_Curr
P3=P_Curr
Mcobotx3=P3.X
Mcoboty3=P3.Y
Mcobotz3=P3.Z
Print #2, "Tcobot(",M1,",3)=",Mrobot(M1,3),"/1000",";"
Print #2, "Pcobot_x(",M1,",3)=",Mcobotx3,";"
Print #2, "Pcobot_y(",M1,",3)=",Mcoboty3,";"
Print #2, "Pcobot_z(",M1,",3)=",Mcobotz3,";"
Dly 0.5
Mvs Pplacebody_back, -50
" Gripperın orta kısmıyla gövdeyi tekrar al
Ovrđ 50
Mov Ppickbody_mid, -50
Ovrđ 3
Mvs Ppickbody_mid
Dly 0.5
HClose 1
Dly 0.5
Mvs Ppickbody_mid, -50
Act 1=0
M_01#=1
EndIf
"" Gövde Konumlandırma Testi Programı ""
If M_01#=1 And M_In(3)=1 Then ' Gövde alma programı bittikten sonra
konumlandırma testi yap
*GovdeTestEt ' Gövde test etme alt programı
Ovrđ 50
Act 1=1
Mov Ptestbody, -50
Ovrđ 3
Mvs Ptestbody
Dly 0.5
" Gövdeyi test et
If M_In(2)=0 Then GoTo *RotateBody1 Else GoTo *RotateBody2
*RotateBody1
Ovrđ 50
Jrotatebody1_aux=J_Curr
Jrotatebody1=Jrotatebody1_aux-Jrotate50
Mov Jrotatebody1
Dly 0.3
GoTo *RotateBody2

```

```

*RotateBody2
While M_In(2)=1
OvrD 80
Jrotatebody2_aux=J_Curr
Jrotatebody2=Jrotatebody2_aux+Jrotate
Mov Jrotatebody2
WEnd
Dly 0.3
OvrD 50
Jorientbody_aux=J_Curr
Jorientbody=Jorientbody_aux+Jrotate50
Mov Jorientbody
GoTo *BodyTestFinish
*BodyTestFinish
OvrD 3
Porientbody=P_Curr
Mrobot(M1,4)=M_Timer(2) ' Robotun gövde oryantasyonunu tamamladığı zaman
Probot(M1,4)=P_Curr
P4=P_Curr
Mcobotx4=P4.X
Mcoboty4=P4.Y
Mcobotz4=P4.Z
Print #2, "Tcobot(",M1,",4)=",Mrobot(M1,4)," /1000",";"
Print #2, "Pcobot_x(",M1,",4)=",Mcobotx4,";"
Print #2, "Pcobot_y(",M1,",4)=",Mcoboty4,";"
Print #2, "Pcobot_z(",M1,",4)=",Mcobotz4,";"
Dly 0.2
Mvs Porientbody, -50
Act 1=0
M_01#=2
EndIf
"" Gövdeyi Yerleştirme Programı ""
If M_01#=2 And M_In(3)=1 Then ' Gövde konumlandırma testi bittikten sonra
gövdeyi yerleştir
*GovdeYerlestir ' Gövde yerleştirme alt programı
Pplacebody=Porientbody+Passemble-Ptestbody
Act 1=1
OvrD 50
Mov Pplacebody, -50
OvrD 3
Mvs Pplacebody
Dly 0.5
HOpen 1
Mrobot(M1,5)=M_Timer(2) ' Robotun gövdeyi montaj alanına yerleştirdiği zaman
Probot(M1,5)=P_Curr
P5=P_Curr
Mcobotx5=P5.X
Mcoboty5=P5.Y
Mcobotz5=P5.Z
Print #2, "Tcobot(",M1,",5)=",Mrobot(M1,5)," /1000",";"

```

```

Print #2, "Pcobot_x(",M1,",5)=",Mcobotx5,";"
Print #2, "Pcobot_y(",M1,",5)=",Mcoboty5,";"
Print #2, "Pcobot_z(",M1,",5)=",Mcobotz5,";"
Dly 0.5
Mvs Pplacebody, -50
Ovrd 50
Mov Psecure
Act 1=0
M_01#=3
EndIf
Input #1, C1$ ' Client'dan string verilerini al
If M_01#=3 And C1$="waveIn" And M_In(3)=1 Then ' Pistonun atıldığı zaman
Act 1=1
Mrobot(M1,6)=M_Timer(2) ' Pistonun atıldığı zaman
Probot(M1,6)=P_Curr
P6=P_Curr
Mcobotx6=P6.X
Mcoboty6=P6.Y
Mcobotz6=P6.Z
Print #2, "Tcobot(",M1,",6)=",Mrobot(M1,6),"/1000",";"
Print #2, "Pcobot_x(",M1,",6)=",Mcobotx6,";"
Print #2, "Pcobot_y(",M1,",6)=",Mcoboty6,";"
Print #2, "Pcobot_z(",M1,",6)=",Mcobotz6,";"
Act 1=0
M_01#=4
EndIf
"" Kapak Test Etme ve Yerleştirme Program ""
If M_01#=4 And C1$="doubleTap" And M_In(3)=1 Then ' waveIn gelirse kapak test
et
*KapakTestEt
Mrobot(M1,7)=M_Timer(2) ' Yayın atıldığı zaman
Probot(M1,7)=P_Curr
P7=P_Curr
Mcobotx7=P7.X
Mcoboty7=P7.Y
Mcobotz7=P7.Z
Print #2, "Tcobot(",M1,",7)=",Mrobot(M1,7),"/1000",";"
Print #2, "Pcobot_x(",M1,",7)=",Mcobotx7,";"
Print #2, "Pcobot_y(",M1,",7)=",Mcoboty7,";"
Print #2, "Pcobot_z(",M1,",7)=",Mcobotz7,";"
" Gripperın arka kısmıyla kapak almaya git
Ovrd 40
Act 1=1
Mov Ppickcover(M1), -50
Ovrd 3
Mvs Ppickcover(M1)
Dly 0.5
HClose 1
Dly 0.5
Mvs Ppickcover(M1), -50

```



```

" Kapağı test alanına bırak
Ovrđ 40
Mov Pplacecover_back(M1), -50
Ovrđ 3
Mvs Pplacecover_back(M1)
Dly 0.5
HOpen 1
Mrobot(M1,8)=M_Timer(2) ' Kapağın test alanına bırakıldığı zaman
Probot(M1,8)=P_Curr
P8=P_Curr
Mcobotx8=P8.X
Mcoboty8=P8.Y
Mcobotz8=P8.Z
Print #2, "Tcobot(",M1,",8)=",Mrobot(M1,8),"/1000",";"
Print #2, "Pcobot_x(",M1,",8)=",Mcobotx8,";"
Print #2, "Pcobot_y(",M1,",8)=",Mcoboty8,";"
Print #2, "Pcobot_z(",M1,",8)=",Mcobotz8,";"
Dly 0.5
Mvs Pplacecover_back(M1), -50
" Gripperın orta kısmıyla kapağı tut
Ovrđ 40
Mov Pplacecover_mid, -50
Ovrđ 3
Mvs Pplacecover_mid
Dly 0.5
HClose 1
Dly 0.5
" Kapağı test et
If M_In(2)=1 Then GoTo *RotateCover1 Else GoTo *RotateCover2
*RotateCover1
Ovrđ 50
Jrotatecover1_a=J_Curr
Jrotatecover1=Jrotatecover1_a+Jrotate50
Mov Jrotatecover1
Dly 0.3
GoTo *RotateCover2
*RotateCover2
While M_In(2)=0
Ovrđ 80
Jrotatecover2_a=J_Curr
Jrotatecover2=Jrotatecover2_a+Jrotate
Mov Jrotatecover2
WEnd
Dly 0.3
Ovrđ 50
Jorientcover_aux=J_Curr
Jorientcover=Jorientcover_aux-Jrotate65
Mov Jorientcover
GoTo *CoverTestFinish
*CoverTestFinish

```

```

Ovrđ 3
Porientcover=P_Curr
Mrobot(M1,9)=M_Timer(2) ' Kapađın oryantasyon testinin yapıldıđı zaman
Probot(M1,9)=P_Curr
P9=P_Curr
Mcobotx9=P9.X
Mcoboty9=P9.Y
Mcobotz9=P9.Z
Print #2, "Tcobot(",M1,",9)=",Mrobot(M1,9),"/1000",";"
Print #2, "Pcobot_x(",M1,",9)=",Mcobotx9,";"
Print #2, "Pcobot_y(",M1,",9)=",Mcoboty9,";"
Print #2, "Pcobot_z(",M1,",9)=",Mcobotz9,";"
Ovrđ 3
Dly 0.2
Mvs Porientcover, -50
*AssembleCover
Pplacecover=Porientcover+Passemble2-Pplacecover_mid
Ovrđ 40
Mov Pplacecover, -50
Ovrđ 3
Mvs Pplacecover
" Kapađı çevirerek montajla
Dly 0.5
Ovrđ 20
Jplacecover=J_Curr
Jassemblecover=Jplacecover+Jrotate46
Mov Jassemblecover
Passemblecover=P_Curr
Dly 0.5
HOpen 1
Mrobot(M1,10)=M_Timer(2) ' Kapađın montajının tamamlandıđı zaman
Probot(M1,10)=P_Curr
P10=P_Curr
Mcobotx10=P10.X
Mcoboty10=P10.Y
Mcobotz10=P10.Z
Print #2, "Tcobot(",M1,",10)=",Mrobot(M1,10),"/1000",";"
Print #2, "Pcobot_x(",M1,",10)=",Mcobotx10,";"
Print #2, "Pcobot_y(",M1,",10)=",Mcoboty10,";"
Print #2, "Pcobot_z(",M1,",10)=",Mcobotz10,";"
Dly 0.5
Ovrđ 5
Mov Passemblecover, -50
GoTo *PickPart
" Gripperın arka kısmıyla parçayı al
*PickPart
Ovrđ 50
Mov Ppickpart, -50
Ovrđ 3
Mvs Ppickpart

```

```

Dly 0.5
HClose 1
Dly 0.5
Mvs Ppickpart, -50
*DropPart
" Parçayı saklama alanına bırak
Ovrd 50
Cnt 1
Mov Pdrop_aux
Cnt 0
Mov Pdrop(M1), -50
Ovrd 3
Mvs Pdrop(M1)
Dly 0.5
HOpen 1
Mrobot(M1,11)=M_Timer(2) ' Montajı yapılan ürünün saklama alanına bırakıldığı
zaman
Probot(M1,11)=P_Curr
P11=P_Curr
Mcobotx11=P11.X
Mcoboty11=P11.Y
Mcobotz11=P11.Z
Print #2, "Tcobot(",M1,",11)=",Mrobot(M1,11),"/1000",";"
Print #2, "Pcobot_x(",M1,",11)=",Mcobotx11,";"
Print #2, "Pcobot_y(",M1,",11)=",Mcoboty11,";"
Print #2, "Pcobot_z(",M1,",11)=",Mcobotz11,";"
Dly 0.5
Mvs Pdrop(M1), -50
Ovrd 20
Mov Phome
Dly 1
M1=M1+1
Act 1=0
M_01#=5
EndIf
GoTo *Basla
End
*Güvenlik ' Güvenlik alt programı
OAdl Off
Ovrd 3
Wait M_In(3)=1
Dly 0.5
Return 0

```

EK B: Raspberry Pi 3 Python Kodu

```

#!/usr/bin/python
from PyoConnect import *
myo = Myo(sys.argv[1] if len(sys.argv) >= 2 else None)

```

```

data1 = "veri1"
data2 = "veri2"
data3 = "veri3"
data4 = "veri4"
data5 = "veri5"
import socket
host = "192.168.0.20"
port = 10004
s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
s.connect((host,port))
global data1
global data2
global data3
global data4
global data5
def onUnlock():
    myo.rotSetCenter()
    myo.unlock("hold")
def onPoseEdge(pose, edge):
    if (pose == "waveOut") and (edge == "on"):
        data1 = str("waveOut")
        print pose
        s.send(data1)
    elif (pose == "fist") and (edge == "on"):
        data2 = str("fist")
        print pose
        s.send(data2)
    elif (pose == "fingersSpread") and (edge == "on"):
        data3 = str("fingersSpread")
        print pose
        s.send(data3)
    elif (pose == "doubleTap") and (edge == "on"):
        data4 = str("doubleTap")
        print pose
        s.send(data4)
    elif (pose == "waveIn") and (edge == "on"):
        data5 = str("waveIn")
        print pose
        s.send(data5)
    else:
        print ""
myo.onPoseEdge = onPoseEdge
myo.onUnlock = onUnlock
myo.connect()
while True:
    myo.run()
    myo.tick()

```

ÖZGEÇMİŞ



Ad-Soyad : Mustafa ÇOBAN
Doğum Tarihi ve Yeri : 08.10.1993 / ANTALYA
E-posta : m.coban@btu.edu.tr

ÖĞRENİM DURUMU:

- **Lisans** : 2015, Yıldız Teknik Üniversitesi, Makine Fakültesi, Mekatronik Mühendisliği Bölümü

MESLEKİ DENEYİM VE ÖDÜLLER:

- Araştırma Görevlisi – Bursa Teknik Üniversitesi (Şubat 2016 – devam ediyor)
- Stajyer – TSN Mühendislik (Ağustos 2014 – Eylül 2014)
- Stajyer – ASELSAN A. Ş. (Temmuz 2014 – Ağustos 2014)
- Stajyer – Buga Otis Asansör San. Ve Tic. A. Ş. (Temmuz 2013 – Ağustos 2013)
- Bölüm Birincisi – Yıldız Teknik Üniversitesi Mekatronik Mühendisliği (2015)

TEZDEN TÜRETİLEN ESERLER, SUNUMLAR VE PATENTLER:

- **Çoban, M. & Gelen, G.** (2018). Endüstriyel Robotlar için Ayrık Olay Sistemi Yaklaşımı ile Formel Kod Elde Edilmesi, *4. Türkiye Robotbilim Konferansı (ToRK 2018)*, (ss. 13-18). İstanbul : Boğaziçi Üniversitesi, Nisan 12-14.

DİĞER ESERLER, SUNUMLAR VE PATENTLER:

- **Özcan, S., Çoban, M. & Gelen, G.** (2017). An Easy to Use Joystick Controller Design for Industrial Robots, *5th International Conference on Advanced Technology and Science (ICAT'17)*, (ss. 795-798). İstanbul : Bahçeşehir Üniversitesi, Mayıs 09-12.