



T.C.
NECMETTİN ERBAKAN ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ



**NESNELERİN İNTERNETİ YÖNTEMİ İLE
KİNETİK YÜRÜME ANALİZ SİSTEMİNİN
GELİŞTİRİLMESİ**

Muhammet ÇALIŞKAN

YÜKSEK LİSANS TEZİ

Endüstri Mühendisliği Anabilim Dalı

Ekim-2018
KONYA
Her Hakkı Saklıdır

TEZ KABUL VE ONAYI

Muhammet ÇALIŞKAN tarafından hazırlanan “Nesnelerin İnterneti Yöntemi İle Kinetik Yürüme Analiz Sisteminin Geliştirilmesi” adlı tez çalışması .../.../... Tarihinde aşağıdaki jüri tarafından oy birliği / oy çokluğu ile Necmettin Erbakan Üniversitesi Fen Bilimleri Enstitüsü Endüstri Mühendisliği Anabilim Dalı’nda YÜKSEK LİSANS TEZİ olarak kabul edilmiştir.

Jüri Üyeleri

İmza

Başkan

Prof. Dr. Şakir TAŞDEMİR

.....

Danışman

Dr. Öğr. Üyesi Abdullah Erdal TÜMER

.....

Üye

Prof. Dr. Sabri KOÇER

.....

Yukarıdaki sonucu onaylarım.

Prof. Dr. Ahmet AVCI
FBE Müdürü

Bu tez çalışması Necmettin Erbakan Üniversitesi Bilimsel Araştırma projeleri Koordinatörlüğü tarafından 161319004 nolu proje ile desteklenmiştir.

TEZ BİLDİRİMİ

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

DECLARATION PAGE

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

Muhammet ÇALIŞKAN

Tarih:

ÖZET

YÜKSEK LİSANS TEZİ

NESNELERİN İNTERNETİ YÖNTEMİ İLE KİNETİK YÜRÜME ANALİZ SİSTEMİNİN GELİŞTİRİLMESİ

Muhammet ÇALIŞKAN

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

Danışman: Dr. Öğr. Üyesi Abdullah Erdal TÜMER

2018, 138 Sayfa

Jüri

**Dr. Öğr. Üyesi Abdullah Erdal TÜMER
Prof. Dr. Sabri KOÇER
Prof. Dr. Şakir TAŞDEMİR**

Nesnelerin İnterneti 1991 yılında Cambridge üniversitesinde bir grup akademisyenin; kahve makinesinden 3dk. da bir alınan görüntüyü akademisyenlerin bilgisayarlarına gönderen bir sistem geliştirmeleriyle ortaya çıkmış ve günümüze kadar bir çok çalışmada kullanılmıştır. Bu çalışmalar; sağlık, akıllı evler, akıllı çevre, akıllı su, akıllı tarım, akıllı hayvancılık, enerji ve şehircilik gibi birçok farklı alanda kullanılmıştır. Nesnelerin İnterneti; nesnelerin insan müdahalesi olmadan birbirleriyle haberleşmesi esasına dayanır. E-Sağlık; nesnelerin İnternetinin kullanımı için önemli alanlardan bir tanesidir. Sağlık alanı; hastadaki değişimlerin anlık algılanıp, anlık müdahale edilmesi gereken bir alandır. Hastayı sürekli gözlem altında tutsanız bile insan vücudundaki birçok değişimi gözlemle anlama şansınız yoktur. Nesnelerin internetinde kullanılan sensörlerin bu değişimleri anlık algılayıp, beklenenin dışındaki değişimlerde uzmanları uyarması ile bu sorun ortadan kalkmaktadır.

Bu tezde Nesnelerin İnterneti' nin Yürüme Analizinde kullanımı ile ilgili bir çalışma gerçekleştirilmiştir. Yürüme insan hareketlerinin en temel şekillerinden birisidir. Deneyimli kişiler için bile, yürümeyi gözle değerlendirmek oldukça güçtür. İnsan gözü, yürüme esnasında mili saniyeler içerisinde olan hareketleri tam olarak algılayamaz. Ayrıca yürüme yalnızca eklem hareketlerinden ibaret olmayıp gözle anlaşılmayacak kas ve kuvvet aktivitelerini içerir. Doğru teşhis ve başarılı tedavi için normal yürüme özellikleri bilinmeli ve patolojik olan yürümeden ayırt edilebilmelidir. Bunun için yürümenin tüm argümanlarını kaydedecek, bu bilgileri sayısal veriye dönüştürecek, karşılaştırma ve tekrar incelemeye olanak sağlayacak sistemler gereklidir. Bu sistemler yürüme analiz sistemleridir.

Yürüme Analizi; gözleme dayalı, kinematik, EMG, kinetik analiz gibi çeşitli analiz türlerinden oluşmaktadır. Sağlıklı bir teşhis için bu analiz türleri bir arada kullanılmalıdır. Kinetik analiz hareketin oluşmasına neden olan kuvvetlerin (yer tepkimesi kuvvetleri, eklem momentleri, eklem güçleri) incelenmesidir. Ayrıca sağlıklı bir insanda iki ayağa uygulanan kuvvetin aynı olması gerekmektedir.

Çalışmada yürüme kinetik analizi için giyilebilir sensör tabanlı bir sistem geliştirilmiştir. Sistem Nesnelerin İnterneti yöntemi ile çalışmaktadır. Sistemde Arduino Mega 2560, HC-05 Bluetooth Modülü ve kuvvet sensörleri kullanılmıştır. Yürüme fazlarında ayak tabanında oluşan kuvvet dağılımlarını algılayan sensörler yerleştirilmiş ayakkabılar, bu ayakkabılardan verileri bilgisayara bluetooth teknolojisi ile kablolu bir şekilde gönderebilen geliştirme kartları ve gelen verileri yorumlayan yazılım sistemi çalışmanın genel yapısını oluşturmaktadır. Sistem 14 erkek ve 4 kadından oluşan katılımcı grubu üzerinde denenmiştir. Bu tezle yürüme analizinin; maliyet, uzman gerekliliği ve doğal yürüyüşü ölçmenin zorluğu gibi kısıtlardan kurtulması; yürüme analizinin yaygınlaşması hedeflenmektedir. Çalışmanın sağladığı diğer bir katkı ise ayak tabanındaki yüksek basınç noktalarının tespiti ile hastaya uygun tabanlık üretilerek ayaklarda oluşacak doku kaybının engellenmesidir. Çalışma sonunda 18 katılımcıdan 6

tanesinde sađlıklı yürüme tespit edilmiştir. Çalışmada ölçülen en yüksek ağırlık 105 kg dır. Sistem 2 katılımcıda hata vermiş; bunlardan birisi 110 kg erkek, diğeri 99 kg kadındır. Kadın katılımcıda hata verme nedeni ayaklarında platin olduđu için elektronik devrenin ve kablosuz haberleşme ünitesinin etkilenmesi olarak değerlendirilmiştir.

Anahtar Kelimeler: Bluetooth, Kinetik Analiz, Nesnelerin İnterneti, Yürüme Analiz Sistemi



ABSTRACT

MS THESIS

THE DEVELOPMENT OF THE KINETIC GAIT ANALYSIS BASED ON THE INTERNET OF THINGS

Muhammet ÇALIŞKAN

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

Advisor: Asst. Prof. Dr. Abdullah Erdal TÜMER

2018, 138 Pages

Jury

Asst. Prof. Dr. Abdullah Erdal TÜMER

Prof. Dr. Sabri KOÇER

Prof. Dr. Şakir TAŞDEMİR

The Internet of Things; came out with a system development that sent the picture taken from the coffee machine every 3 minutes to the academicians' computers, and it was used in as much work as everyday by a group of academics from Cambridge University in 1991. These studies are ; E-Health, Home Automation, Smart Environment, Intelligent Water, Intelligent Agriculture, Intelligent Livestock, Intelligent Energy, Intelligent Cities. The Internet of Things; is based on the principle that objects communicate with each other without human intervention. E-health; It is one of the important areas for the use of IOT. Health area; a field in which changes in the patient should be perceived and intervened momentarily. Even if you keep the patient under constant observation, you do not have a chance to observe many changes in the human body. This problem does not arise from the fact that the sensors used on the internet of things instantly perceive these changes and warn the experts in the changes that are not expected.

In this thesis, a study on the use of the Internet of Things in the Walking Analysis was carried out. Walking is one of the most basic forms of human movements. Even for experienced people, it is difficult to visually evaluate walking. The human eye can not fully perceive the movements within the milli-seconds during walking. In addition, walking involves not only the joint movements but also the invisible force and muscle activities. For accurate diagnosis and successful treatment, normal gait characteristics should be known and distinguishable from pathologic. For this it is necessary to record all the components of the walk, to convert this information into a numerical data base, to enable comparison and review. These systems are walking analysis systems.

Gait Analysis; Observation Based, Kinematic, EMG, Kinetic Analysis. For healthy diagnosis, these types of analyzes should be used together. Kinetic analysis is the study of forces (ground reaction forces, joint moments, joint forces) that cause movement to occur. In addition, the force applied to the two legs must be the same in a healthy person.

A wearable sensor-based system was developed for walking kinetic analysis in the study. The system works with the Internet method of objects and Arduino Mega 2560, HC-05 Bluetooth Module, Force sensors are used in the system. Sensors that sense the force distributions on the soles of the feet during walking phases are the general structure of the shoe layouts, the development cards that can send data wirelessly from these shoes to the computer via Bluetooth technology, and the software system that interprets the incoming data. The system was tested on 14 male and 4 female participant groups. With this thesis we aimed to analyze the walking; cost, expert necessity and the difficulty of measuring natural gait, and making gait analysis more cost effective. Another contribution of the study is to detect the high pressure points on the soles of the feet and to prevent the loss of tissue due to the patient by producing

appropriate insoles. At the end of the study, healthy walking was detected in 6 of 18 participants. The highest weight measured in the study is 105 kg. The system has made an error in 2 participants; one of them is 110 kg male, the other is 99 kg female. The mistake in the female participant is the influence of the electronic gadget and the wireless communication unit because it is platinum in the foot.

Keywords: Bluetooth, Gait Analysis System, Internet of Things, Kinetic Analysis



ÖNSÖZ

Bu tez çalışmasında, bana her türlü maddi ve manevi destek veren anneme, babama ve eşime teşekkürü borç bilirim. Tezin gerçekleşmesinde bana kılavuzluk eden, tavsiyeleri ile daha doğruya erişmemi sağlayan, çalışmamda beni yüreklendiren ve aydınlatan, yardımlarını hiçbir zaman esirgemeyen tez danışmanım Sayın Dr. Öğr. Üyesi Abdullah Erdal Tümer' e teşekkür ederim.

Tezimi 161319004 nolu proje ile destekleyen Necmettin Erbakan Üniversitesi BAP (Bilimsel Araştırma Projeleri) koordinatörlüğüne teşekkür ederim.

Muhammet ÇALIŞKAN
KONYA-2018



İÇİNDEKİLER

ÖZET	v
ABSTRACT.....	vii
ÖNSÖZ	ix
İÇİNDEKİLER	x
SİMGELER VE KISALTMALAR.....	xiii
1. GİRİŞ.....	1
2. KAYNAK ARAŞTIRMASI	4
2.1. Literatür Analizi.....	4
2.1.1. Nesnelerin interneti teknolojisinin biyomedikal alandaki uygulamaları	4
2.1.2. Yürüme analizinin geçmişi	6
2.1.3. Yürüme kinetiği ile ilgili yapılan çalışmalar	7
2.2. Nesnelerin İnterneti.....	12
2.2.1. İnternetin aşamaları.....	13
2.2.2. Uygulama alanları.....	14
2.2.3. Nesnelerin interneti genel ölçütleri.....	15
2.2.3.1. Güvenilirlik.....	16
2.2.3.2. Kullanılabilirlik (Availability).....	16
2.2.3.3. Taşınabilirlik (Mobility)	16
2.2.3.4. Ölçeklenebilirlik (Scalability).....	16
2.2.3.5. Heterojenlik (Heterogeneity)	17
2.2.3.6. Birlikte çalışabilirlik (Interoperability).....	17
2.2.3.7. Güvenlik/Gizlilik (Security/Privacy).....	17
2.2.3.8. Performans (Performance).....	17
2.2.4. Nesnelerin internetinde kullanılan mimari modeller	17
2.2.4.1. Üç Katmanlı Mimari model.....	18
2.2.4.2. Servis Odaklı Mimari (SOA).....	18
2.2.4.3. Beş Katmanlı Mimari.....	20
2.2.4.4. Dört Katmanlı Mimari model	21
2.2.4.5. Altı Katmanlı Mimari model	21
2.2.5. Haberleşme protokolleri	21
2.2.6. Nesnelerin İnterneti'nde kullanılan haberleşme teknolojileri.....	22
2.2.6.1. Zigbee	22
2.2.6.2. Bluetooth.....	22
2.2.6.3. 6LoWpan	23
2.2.6.4. Wifi	23
2.2.6.5. Hücresel	23
2.2.7. Kablosuz Vücut Alan Ağları (KVAA)	24
2.3. Yürüme	25
2.3.1. Yürüme döngüsü.....	25

2.3.1.1. Basma fazı ve evreleri.....	26
2.3.1.2. Salınım Fazı ve Evreleri	31
2.3.1.3. Çift Destek Fazı	33
2.3.2. Yürümenin incelenmesinde kullanılan terimler.....	34
2.3.3. Anatomik düzlemler	35
2.3.4. Patolojik yürüme.....	36
2.4. Yürüme Analizi.....	36
2.4.1. Yürüme analizinin önemi.....	37
2.4.2. Yürüme analizinde kullanılan donanımlar.....	38
2.4.3. Yürüme analizinde kullanılan yöntemler.....	38
2.4.3.1. Gözleme dayalı analiz.....	38
2.4.3.2. Video analiz	39
2.4.3.3. Kinematik analiz	39
2.4.3.4. Dinamik Elektromiyografi (EMG)	40
2.4.3.5. Kinetik analiz.....	41
2.4.4. Yürüme Analizinin kısıtları	43
3. MATERYAL VE YÖNTEM.....	44
3.1. Kullanılan Donanımlar	44
3.1.1. Arduino	44
3.1.1.1. Arduino kartlarının genel özellikleri ve aralarındaki farklar	44
3.1.1.2. Arduino Mega 2560	45
3.1.2. Pcb kartı	47
3.1.2.1. Pcb kartın özellikleri.....	48
3.1.2.2. Pcb kartın son hali.....	48
3.1.3. Kuvvet Sensörü.....	49
3.1.3.1. Analog veri	50
3.1.4. HC-05 Bluetooth Modülü	52
3.1.4.1. Çalışma mantığı	52
3.1.4.2 HC – 05 Bluetooth Modülü özellikleri	53
3.1.4.3. AT komutları.....	53
3.1.4.5. HC – 05 Modülünün bilgisayara tanıtımı	56
3.1.5. 3D yazıcı	57
3.2. Kullanılan Yazılımlar	58
3.2.1. Microsoft Visual studio	59
3.2.2. C#.....	59
3.2.2.1. Kullanılan formlar.....	61
3.2.3. Arduino IDE	66
3.2.3.1. Arduino ve Arduino IDE arasında bağlantı kurma	67
3.2.4. Fritzing.....	68
3.2.5. Autodesk 3Ds Max	69
3.2.6. Cura.....	71
3.2.7. Microsoft Office Access	72
3.2.7.1. Veritabanı nesneler	72
3.3. Nesnelerin İnterneti (IoT)	73
3.3.1. Seçilen mimari ve sistemdeki işleyişi.....	73
3.3.2. IoT platform örneği.....	73
3.3.3. Sistemin blok diyagramı	74
3.3.4. Sistemde kullanılan yazılımların akış diyagramları.....	75

3.3.5. Önerilen sistemin genel yapısı	76
3.4. Sistemin Birleşimi ve Haberleşmesi	77
3.5. Sistemin Çalışması ve Katılımcılara Uygulanması	80
3.5.1. Hazırlık	81
3.5.2. Sistemin kullanımı	82
4. ARAŞTIRMA SONUÇLARI VE TARTIŞMA.....	89
5. SONUÇLAR VE ÖNERİLER	93
5.1. Sonuçlar	93
5.2. Öneriler	94
KAYNAKLAR	95
EKLER	104
ÖZGEÇMİŞ	Hata! Yer işareti tanımlanmamış.

SİMGELER VE KISALTMALAR

Simgeler

V: Volt

N: Newton

KM: Kilo Metre

P: Basınç

F: Kuvvet

S: Yüzey

MS: Mili Saniye

MA: Mili Amper

KOHM: Kilo Ohm

SN: Saniye

Kısaltmalar

YAS: Yürüme Analiz Sistemi

EMG: Elektromiyografi

VAM: Vücut Ağırlık Merkezi

YTK: Yer Tepkime Kuvveti

YTKV: Yer Tepkime Kuvveti Vektörü

YEMG: Yüzeysel Elektromiyografi

GRF: Ground Reaction Force (Zemin Reaksiyon Kuvveti)

FSR: Force Sensitive Resistor

IOT: Internet Of Things

RFID: Radio Frequency Identification

1. GİRİŞ

Nesnelerin İnterneti (IoT) fikri Cambridge Üniversitesinde 15 akademisyenin 1991 yılında bir kahve makinesini birlikte kullanmalarıyla ortaya çıkmıştır. Kahve almaya gittiklerinde genelde makinedeki kahvenin tükendiğini gören akademisyenler bu sorunu ortadan kaldıracak bir sistem geliştirdiler. Bu sistem, kahve makinesinin 3 dk. da bir görüntüsünü alıp akademisyenlerin bilgisayarlarına gönderiyordu. Böylece Nesnelerin İnterneti kavram olarak doğmasa da fikir olarak doğmuştur (Kutup, 2011). Nesnelerin İnterneti kavramından ilk kez 1999 yılında Kevin Ashton tarafından Procter & Gamble şirketi için hazırlanan bir tanıtımda bahsedilmiştir (Kutup, 2011; López-de-Armentia vd., 2012).

IoT; Sağlık, Enerji, Şehircilik, Çevrecilik, Tarım, Hayvancılık, Alışveriş, Lojistik, Ölçüm sistemleri, Endüstriyel Kontrol, Güvenlik ve Acil Durumlar gibi birçok alanda kullanılmaktadır (Gökrem ve Bozuklu 2016). Sağlık hizmetleri ani gelişen bir durumda en hızlı şekilde müdahale edilmesi gereken bir alandır. Hızlı müdahale veya erken teşhis için ya hasta sürekli gözetimde tutulmalı ki bunun da kısıtları mevcuttur. Bu kısıtlar, uzmanın gözlemlerle anlayamayacağı bazı değer değişimleridir. Nesnelerin İnternetinde ise hastadan sürekli değerler alınabilir ve olumsuz bir değişiklik olduğunda anında uzmana bildirilebilir ve hızlı müdahale gerçekleştirilebilir.

Bu proje Nesnelerin İnterneti kavramının sağlık alanında kullanımıyla ilgili olup bir Yürüme Analizi sistemi gerçekleştirilmiştir. Nesnelerin interneti son dönemlerde oldukça popüler olan bir yöntemdir. Kısaca nesnelerin kablosuz haberleşmesi ve verilerin insan müdahalesi olmadan yazılımlar aracılığıyla yorumlandığı, depolandığı yöntemle nesnelerin interneti (Internet of Things - IoT) denilmektedir. Tezimizde sensörlerin Bluetooth aracılığı ile kablosuz olarak sistemle haberleşmesi ve gelen verilerin sistem tarafından yorumlanması sağlanmıştır.

Bir noktadan bir noktaya yer değiştirmek maksadıyla, herhangi bir tanesi her zaman yere temas etmek şartıyla, ilerlemek ve destek için iki bacağın beraber kullanılmasına yürüme denir (Kanatlı vd., 2006). Bir insan ömrünün büyük bir kısmını ayakları üzerinde geçirir ki buda ayakların insan sağlığı için ne kadar önemli olduğunu gösterir.

Yürüyüş örüntüsü; ayak açısı, adım uzunluğu, yürüme hızı, adım genişliği, ayağın tabanında ki basınç dağılımı, ayakların birbirine göre uyumu gibi ölçülebilir birçok sayısal veri içermektedir (Bakbak ve Kayacan, 2014).

Ayak, insan vücudunun tüm ağırlığını taşıdığından fiziksel olarak en fazla zorlanan organdır. Ayak tabanına uygulanan basınç değerlerinin ölçümleri ile ilgili çalışmalar 1980'li yıllardan itibaren başlamış olup; ortopedik cerrahi, ayak biyomekaniği, diyabetik ayak ve ortez ayakkabı modifikasyonuna yönelik çalışmalar yapılmıştır. Ayak taban basıncına dış (eksternal) yüklerin etkisi vardır (Yılmaz vd., 2017).

Yürüme analizi; yürümenin sayısal verilere dönüştürülmesi, bu sayısal verilerin yorumlanıp değerlendirilmesidir. Yürüme bozukluğuna sebep olan birçok sorun tecrübe sahibi hekimler tarafından gözle yapılan kontrollerle anlaşılabilse de problemi sayısal olarak yorumlayıp daha sonra yeniden değerlendirmek için kaydetmek ve yapılan tedavinin ne kadar etkili olduğunu nesnel biçimde ortaya koymak adına yürüme analiz sistemleri zorunlu hale gelmiştir (Milli Eğitim Bakanlığı 2, 2011).

19. Yüzyılın sonlarından itibaren uygulanmaya başlayan yürüme analizleri spor (Gouwanda vd., 2008; Di Stasi vd., 2013), güvenlik sistemleri için insanları tanıma (Derawi vd., 2010) ve tıp (Sutherland, 2005) gibi çeşitli alanlarda uygulama alanı bulmuştur. Tıbbi olarak yürüme kişilerin yaşam kaliteleri hakkında önemli bilgiler içeren sayısal verilerden oluşur. Yürümede parkinson gibi nörolojik hastalar, kalp hastaları, herhangi bir kaza sonucu ya da yaşlılık ve inme gibi nedenlerle yürüyemeyen hastalar hakkında hastalığın gelişimlerini tespit etmek için sağlıklı ve sayısal bilgilere ihtiyaç vardır. Her ne kadar yürüme analiz sistemleri tedavi amaçlı kullanılan sistemler olmasa da teşhis ve tedavi yönteminin belirlenmesinde yardımcı bir araç olmakla beraber, doğru ve güvenilir bir yürüme analizi; hastalıkların teşhis edilebilmesi için en uygun tedavi yönteminin bulunmasına önemli ölçüde yardımcı olmaktadır (Kanatlı vd., 2006).

Yürüme analiz laboratuvarlarında, gözleme dayalı analiz, video analiz, kinematik analiz, dinamik elektromiyografi ve kinetik analiz gibideğerlendirme yöntemleri vardır. Bu yöntemler 2.4.3 nolu bölümde ayrıntılı olarak anlatılacaktır.

Bu yöntemlerden kinetik analizde kullanılan kuvvet platformlarının çok çeşitli kısıtları vardır. Bunlar; laboratuvarında kısa mesafede yapılan yürümeyle doğal yürüyüşü ölçmenin zorluğu, yüksek maliyet ve uzman ekip gerekliliğidir (Yavuzer, 2009, 2014; Maeda vd., 2018).

Tezimizde bu kısıtlar dikkate alınarak kinetik analiz yöntemi ile ayak tabanında oluşan kuvvetlerin incelenmesi için bir ayakkabı geliştirilmiş ve ayak tabanında oluşan

kuvvetlerin incelenmesi yapılmıştır. Normal bir yürüyüşte, yürüme boyunca her iki ayak tabanında ki basınç dağılımlarının eşit olması gerekmektedir. (Angunsri vd., 2011). Ayrıca ayak tabanlarında meydana gelen yüksek basınç uygulanan noktaların bulunması ve bu yüksek basıncın ayak tabanına homojen dağılımını sağlayacak tabanlıkların üretilmesine yardımcı olacaktır. Özellikle diyabet hastalarında bu yüksek basınç noktaları müdahale edilmediği takdirde doku kaybına hatta ayağın yok olmasına neden olabilmektedir (Gülçimen ve Ülkü, 2008).

Bu tez çalışması 5 ana bölümden oluşmaktadır. Bu bölümler şu şekilde tasarlanmıştır: 1. bölüm giriş bölümüdür. 2. bölümde tez çalışmamız ile ilgili geçmişten bugüne yapılmış çalışmalardan, tezimizin temelini oluşturan nesnelerin internetinden, yürüme ve yürüme analizine ilişkin kaynakçaya yer verilmiştir. 3. bölümde çalışmamızda kullanılan materyaller, metotlar ve çalışmamızın nesnelerin interneti ile olan ilişkisinden bahsedilmiştir. 4. bölüm de test sonucu elde edilen bulgular ve bunların değerlendirilmesinden oluşmaktadır. 5. bölüm sonuç ve önerilerin verildiği bölümdür.

2. KAYNAK ARAŞTIRMASI

2.1. Literatür Analizi

Yürüme analizi günümüze kadar birçok araştırmanın konusu olmuştur. Yapılan literatür taramalarında elde edilen sonuçlar şu şekildedir: YÖK veri tabanında “Gait Analysis” başlığı ile 32 sonuç listelenmiştir. Aynı veri tabanında “Yürüme Analizi” başlığı altında 29 sonuç, “Nesnelerin İnterneti” başlığı altında 22 sonuç listelenmiştir (Ulusal Tez Merkezi, 2018). Ayrıca Avrupa’dan birçok üniversitenin kayıtlı olduğu DART Europe tez veri tabanında “Gait Analysis” başlığı ile 276, “İnternet of Things” başlığı ile 323 sonuç listelenmiştir (Dart Europe, 2018). Bu durum yürüme analizi ve nesnelerin interneti ile ilgili yeterli sayıda tez çalışmasının yapılmadığını göstermektedir. Bu çalışmanın literatüre de katkı yapması beklenmektedir.

2.1.1. Nesnelerin interneti teknolojisinin biyomedikal alandaki uygulamaları

Nesnelerin İnterneti tabanlı sistemler farklı sağlık alanlarında kullanılabilmektedir. Teşhis, tedavi, koruyucu sistemler, hasta takip sistemleri ve uzaktan hasta bakım hizmetleri bu alanlarda bazılarıdır. Nesnelerin İnterneti’nin önemli iki bileşeni: Kablosuz Vücut Alan Ağları (KVAA) ve Radyo Frekanslı Tanımlama Sistemleridir (RFID) (Aktaş vd., 2016). Bu bölümde, Nesnelerin İnterneti’nin sağlık alanına yönelik yapılmış çalışmalara yer verilmiştir.

Harvard Üniversitesinde Malan ve arkadaşları tarafından 2004 yılında CodeBlue isimli yapılan çalışmada MICA düğümleri kullanılmış ve kablosuz alan ağlarının (KAA) tıbbi uygulamalarda kullanım olanakları araştırılmıştır. Çalışma birçok hastaya ait SPO2 ve EKG sinyallerinin takibi ve kayıt edilmesi amacıyla yapılmıştır (Malan vd., 2004).

Baker ve arkadaşları 2007 yılında büyük bir çalışma ekibiyle Intel ve Ford gibi çok büyük firmaların katkılarıyla uzaktan hasta takibi ile sağlık hizmetlerinin maliyetlerini azaltmaya yönelik çeşitli sistemler tasarlamışlardır (Baker vd., 2007).

Lee ve arkadaşları 2007 yılında yaptıkları çalışmada yaşlı hastaların evde bakım hizmeti için EKG sinyallerinin takibi ve bu sinyallerin değerlendirilmesi üzerine bir sistem geliştirmişlerdir. Geliştirilen sistem ile eş zamanlı olarak alınan sinyaller analiz edilip, uzmana gelişmiş bir teşhis olanağı sağlanmıştır (Lee vd., 2007).

Yang ve arkadaşları 2008 yılında yaptıkları çalışmada gerçek zamanlı hasta izlemelerini gerçekleştirmek maksadıyla RFID uyumlu bir algılayıcı sensör düğümü tasarlamışlardır. Çalışmada mikro denetleyicili kablosuz algılayıcı sistemi geliştirilmiştir (Yang vd., 2008).

Luo ve arkadaşları 2009 yılında yaptıkları çalışmada nesnelerin interneti altyapısını kullanmaktadır. Hastalarda bulunan algılayıcılardan gelen verilerin veri tabanına kaydedilmesi ve gelen verilerin analiz edildikten sonra bir yazılım ile tanı koyulması sağlanmıştır (Luo vd., 2009).

Chunming ve Xuemei 2010 yılında yaptıkları çalışmada geliştirilen hasta izleme sisteminde Zigbee teknolojisini kullanmışlardır. Bu çalışmada iki yönlendirme algoritması karşılaştırılmış ve analiz edilmiştir (Chunming ve Xuemei, 2010).

Atzori ve arkadaşları 2010 yılında yaptıkları çalışmada biyomedikal uygulamalarda hasta izleme sistemlerinin zorunluluğundan bahsetmişler ve nesnelerin interneti tabanlı biyomedikal uygulamalara geniş yer vermişlerdir (Atzori vd., 2010).

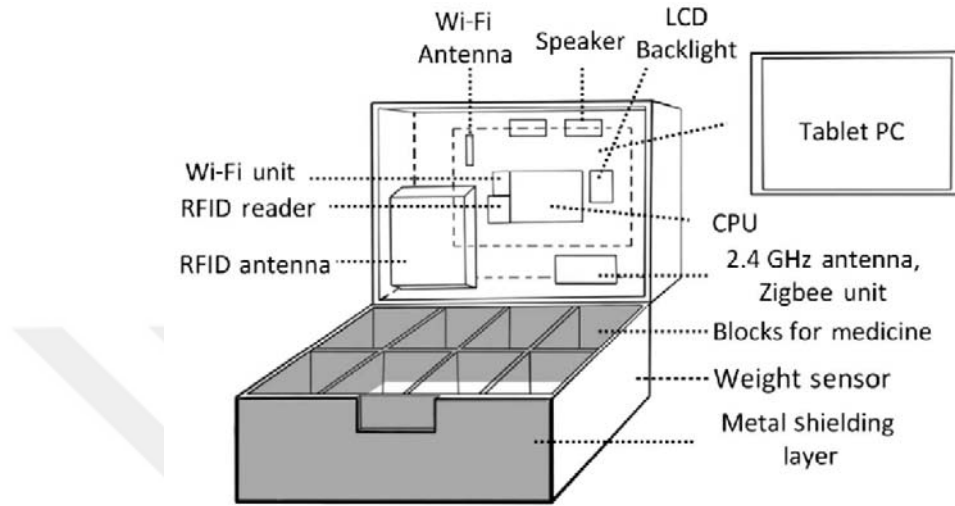
Jara ve arkadaşları 2010 yılında yaptıkları çalışmada ilaçların olumsuz yan etkilerini bulup, ilaçların bu yan etkilerini önlemek amacıyla Nesnelerin İnterneti tabanlı bir platform önermişlerdir. Platformda hasta ve ilaç bilgileri sisteme tanımlanmaktadır. Gerçekleşebilecek problemleri önlemek adına hastanın alerji bilgileri, sağlık bilgileri ile eşleştirilmektedir. Olası bir olumsuz durumda hasta ve doktorlara uyarı verilmektedir (Jara vd., 2010).

Rohokale ve arkadaşları 2011 yılında yaptıkları çalışmada köyde, kasabada yaşayıp hastane vb. sağlık merkezlerine uzak olan insanların sağlık durumlarının, Nesnelerin İnterneti tabanlı bir sistemle uzaktan sürekli izlenmesi amacıyla bir sistem gerçekleştirmişlerdir (Rohokale vd., 2011).

Zhang ve Hu 2013 yılında Open-Zigbee modelini kullanarak yaptıkları çalışmada bu modeli medikal uygulamalara uyumlu hale getirmişlerdir. Çalışmada OPNET programındaki açık kaynak kodlu Zigbee modeli modifiye edilmiştir. Gerçekleştirilen senaryoya göre hastalardan elde edilen fizyolojik veriler erişim noktası üzerinden sistem merkezine iletilmiştir. Sistem erişim noktasından gelen bu verileri analiz ederek sonuçlar üretmektedir (Zhang ve Hu, 2013).

Yang ve arkadaşları 2014 yılında yaptıkları çalışmada nesnelerin internetini temel alarak akıllı sağlık hizmeti sistemi platformu geliştirmişlerdir. Çalışmada IMEDBOX ve IMEDPACK adında iki platform geliştirilmiştir. Sistemler ilaç kutularına yerleştirilen ağırlık sensörleri ile çalışmaktadır. Sürekli kutu ağırlığı

karşılaştırılıp ilgili saatte ilaç alınıp alınmadığını ana sisteme bildirmekte, hastayı uyarmakta ve doktoru bilgilendirmektedir. RFID ile hastalara ait reçete ve ilaç bilgileri sisteme veri tabanı üzerinden yüklenebilmektedir. Şekil 2.1. de Imedbox' ın çalışması ve kullanılan teknolojilere ait bilgiler yer almaktadır (Yang vd., 2014).



Şekil 2.1. Imedbox akıllı ilaç kutusu (Wi-Fi unit= Wifi ünitesi, RFID reader= Radyo Frekans okuyucu, RFID antenna= Radyo Frekans Anteni, Wi-Fi Antenna= Wifi Anteni, LCD Backlight= Arka Işığ, CPU= İşlemci, Blocks for medicine= İlaç için bloklar, Weight sensor= Ağırlık sensörü, Metal Shielding layer= Metal koruyucu tabaka)

Aktaş ve arkadaşları 2016 yılında kablosuz vücut alan ağırları (KVAA) kullanılarak insanlardan alınan sinyallerin bilgisayar sistemlerine gönderilmesine dair bir çalışma gerçekleştirmişlerdir. Hastada elde edilen EKG sinyallerinin kablosuz ortamda aktarılması sistemin önemli bir bileşenidir. Sistemin benzetimi OPNET programında gerçekleştirilmiştir (Aktaş vd., 2016).

2.1.2. Yürüme analizinin geçmişi

Hareket analizinin başlangıcı M.Ö. 350'li yıllarda Aristo'ya dayanmaktadır. Eklem hareketlerinin kasların kasılması sonucu gerçekleştiğini Aristo, kasların kasılmasını sinir sisteminin yönettiğini ise Galen bulmuştur. Galileo'nun öğrencisi Borelli, 19. Yüzyılda da Marey yürüyüş ile ilgili çalışmalar yapmışlardır (Özaras, Yalçın, 2001). Birçok kaynak Giovanni Alfonso Borelli'nin "De Motu Animalium" (16. y.y.) çalışmasını biyomekaniğin başlangıcı olarak tanımlamaktadır (İmrenk, 2011). Weber kardeşlerin 1830'larda modern sayılabilecek yürüme analizi çalışmaları bulunmaktadır. Fotoğraf çekebilen makinelerin hayatımıza girmeye başlaması ile

birlikte yürüyen veya hareket eden insan fotoğraflarının çekilmesi ve değerlendirilmesi dönemi başlamıştır (Özaras, Yalçın, 2001; Yavuzer, 2014).

Yürüme analizinin gelişmesinde önem teşkil eden bir diğer aşama Braune ve Fisher'in 1895'te görüntüleri sayısal verilere dönüştürmesidir. Çalışmalarında deneğin eklemlerine ampul bağlayıp hareketi seri fotoğraflar halinde çekip ampullerin konumlarını grafiksel veriye dönüştürmeyi başarmışlardır. 1950'li yıllarda başladıkları çalışmalarıyla gerçek anlamda yürüme analizini klinik kullanıma sokan bilim insanları VerneInman ve Jacquelin Perry'dir. Bilişim teknolojilerinin giderek gelişmesi ve sağlık alanında kullanımının yaygınlaşması ile klinik kullanıma dair yürüme analiz platformları yaygınlaşmış ve birçok ülkede kullanılmaya başlanmıştır. Laboratuvarlarda uygulanan yürüme analizi sistemleri, yerini hareketin doğal halinde ve doğal ortamda ölçülmesine bırakmaktadır (Yavuzer, 2014).

2.1.3. Yürüme kinetiği ile ilgili yapılan çalışmalar

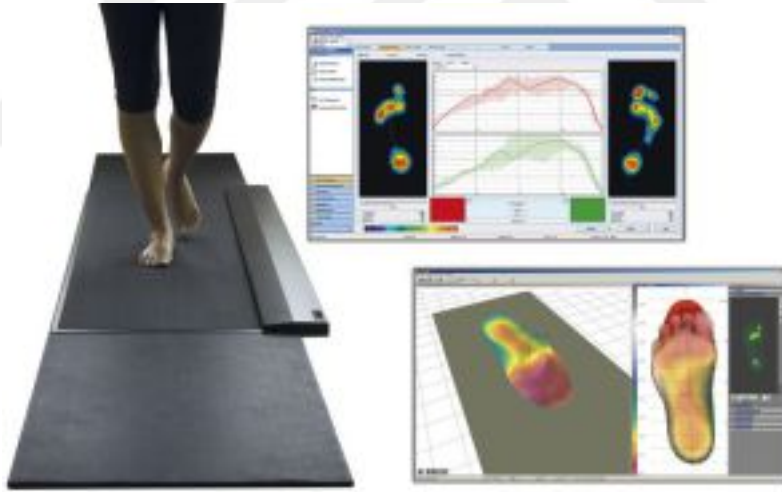
Yürüme kinetiği insan vücudunun hareketlerinin sonucunda neler olacağını açıklar. Kinetik analiz çoğunlukla kuvvet sensörleri ile donatılmış zeminler üzerine odaklanmıştır. Kinetik analizin en temel ve en önemli görevi; yürüyüş sırasında alt ekstremité basınç değerlerinin doğru ve verimli bir şekilde hesaplanmasıdır. Vücut hareketlerinin klinik teşhisinde yürüme kinetiği önemli olmuştur (Bozer, 2007).

Kinetik analizde kullanılan yaklaşımlar çoğunlukla kuvvet platformları analiz yöntemidir. Bu platformları yürüme parkuruna yerleştirilmiş basınç sensörleri ile ayak tabanında oluşan basıncı algılayarak bilgisayar sistemlerine aktarır. Ayak tabanında oluşan kuvvet, maksimum kuvvet, kuvvet dağılımı, basınç merkezi gibi birçok sayısal veriye ulaşmak mümkündür. Sadece durma değil zıplama, koşma, yürüme vb. birçok değişken sayısal verilere de kuvvet platformlarıyla ulaşılabilir. Kuvvet platformlarıyla elde edilen yer tepkime kuvveti değerlerini yürüme analizi çalışmalarında diğer analiz verileri ile birlikte kullanarak yürüme vb. birçok fonksiyonu yapabilmemiz için gereken eklemler moment değerlerini hesaplamak olağandır (Gülçimen ve Ülkü, 2008). Kuvvet platformları ile ayak tabanında oluşan kuvvet değerlerinin ölçüldüğü birçok çalışmaya literatürde rastlanmaktadır. Şekil 2.2. de örnek bir Yürüme Analiz çalışmasından görüntü verilmiştir.



Şekil 2.2. Yürüme Analiz çalışması (Alsan Ortopedi, 2018).

Şekil 2.3. de örnek bir Yürüme Analiz yazılımı ve sistemin kullanımı gösterilmektedir.



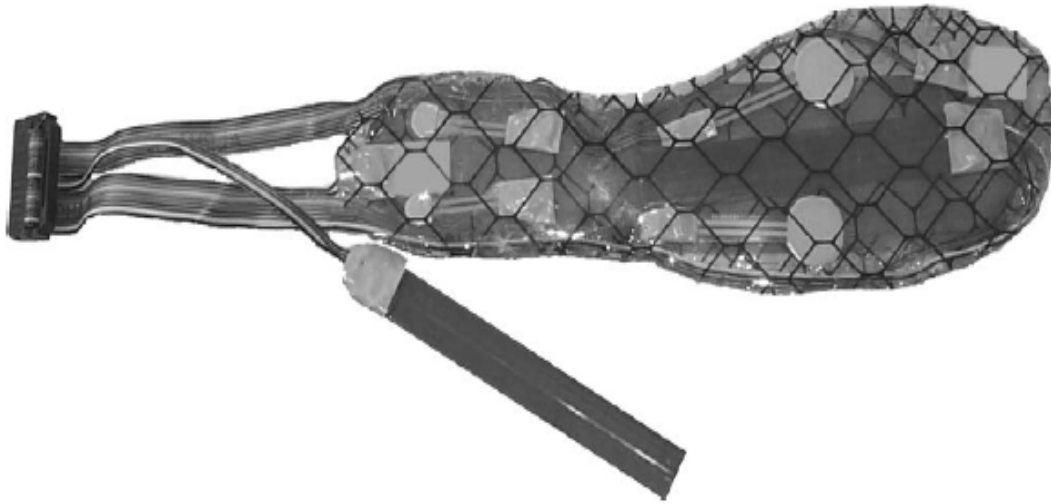
Şekil 2.3. Örnek bir Yürüme Analiz yazılımı (Pediturk, 2018).

Kullanılan bu analiz yöntemlerinin bazı kısıtları bulunmaktadır. Sabit kuvvet plakaları yönteminde birden fazla adım ölçülemez. Sürekli yürüyüş çalışmalarında çok fazla kuvvet plakası kullanmak gerekir. Bir koşu bandı altında iki kuvvet plakası döşeyerek oluşan dinamo metrik platformda ise hastalar düz bir çizgi boyunca yürüdükleri için dönüş veya yön değişikliklerini gerçekleştiremezler (Tao vd., 2012). Ayrıca diğer kısıtları ise sadece özel laboratuvarlarda kullanımı, uzman teknisyen

gerekliliđi, maliyeti ve dođal yryř lmenin zorluđudur (Yavuzer, 2009, 2014; Maeda vd., 2018).

Bu kısıtları ortadan kaldırmak iin tabanına giyilebilir sensrler yerleřtirilmiř ayakkabılar da eřitli alıřmalarda kullanılmıřtır. alıřmamızda tabanına sensrler yerleřtirilmiř ve nesnelerin interneti yntemiyle kablosuz haberleřen kinetik analiz iin zel bir ayakkabı retilmiřtir.

Moris ve Paradisonun 2002; Bamberg ve arkadaşlarının 2008 yıllarında yayınladıkları alıřmalarında řekil 2.4.1. ve 2.4.2. deki ayakkabıyı kullanmıřlardır. Sistem ivmeler, jiroskop, kuvvet sensrleri, dinamik basın sensr ve elektrik alan ykseklik sensrlerinden oluřmaktadır. Sistem yryře mdahale edilmeden ayakkabıya yerleřtirilerek gerekleřtirilmiřtir. řekil 2.4.1. ve 2.4.2. de sensrlerin tabanlıđa yerleřtirilmiř hali ve sistemin son hali gsterilmektedir (Morris ve Paradiso, 2002; Bamberg vd., 2008).

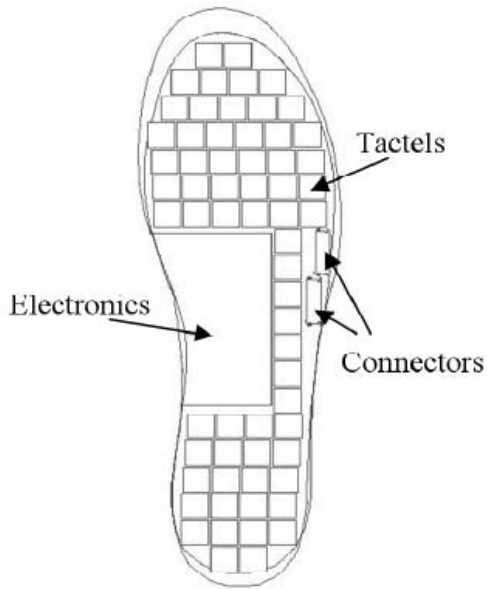


řekil 2.4.1. Sensrlerin ayak tabanına yerleřtirilmesi



Şekil 2.4.2. Çözümün ayakkabıya monte edilmiş hali

De Rossi ve arkadaşları 2011 yılında yayınladıkları çalışmalarında şekil 2.5. de gösterilen ayakkabıyı kullanmışlardır. Geliştirilen ayakkabı yürüme esnasında basınç dağılımını izlemek için 64 adet hassas matristen oluşmaktadır. Şekil 2.5. de tasarlanan tabanlık gösterilmiştir (De Rossi vd., 2011).



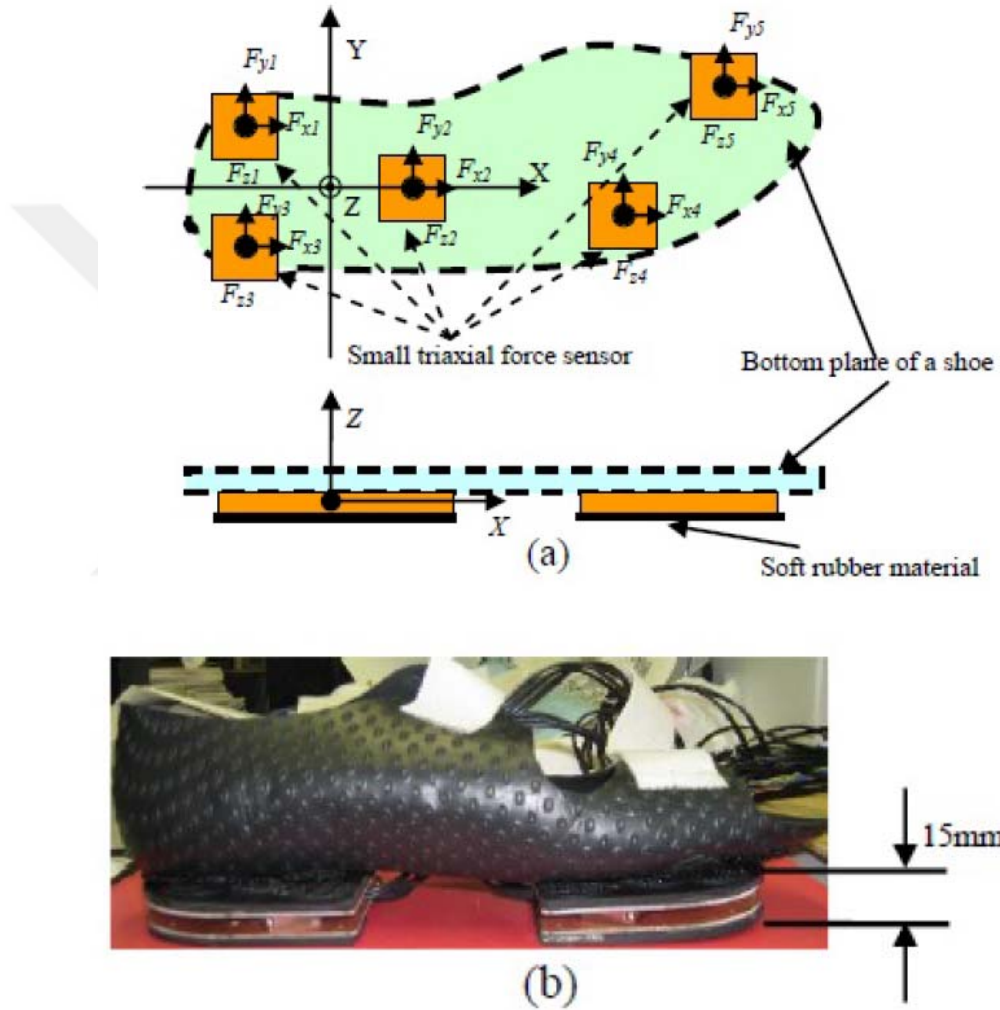
(a)



(b)

Şekil 2.5. Sensör kartının yerleşimi (Elektronics= Elektronik kart, Connectors= konektörler, Tactels=naylon-plastik taban)

Liu ve arkadaşları 2010 ve 2012 yıllarında yayınladıkları çalışmalarında şekil 2.6. daki ayakkabıyı kullanmışlardır. Kullanılan ayakkabı 5 adet basınç sensöründen oluşmaktadır. Bu çalışma zemin reaksiyon kuvvetini (GRF) ve basınç merkezini (CoP) doğrulamak için 7 gönüllü katılımcı üzerinde denenmiştir. Şekil 2.6. da sensörlerin çalışma prensibi ve ayakkabının son hali gösterilmektedir (Liu vd., 2010; Tao vd., 2012).



Şekil 2.6. Beş küçük üç eksenli kuvvet kullanılarak inşa edilen bir giyilebilir GRF sensör sistemi ve sensörleri. (a) Sensör' ün ordinat sisteminin ve sensör mekanizmasının tanımı; (b) Prototip Sağ ayak için bir ayakkabı. (Small triaxial force sensor=Küçük üç eksenli kuvvet sensörü, Bottom plane of a shoe=Ayakkabının alt düzlemi, Soft rubber material= yumuşak kauçuk malzeme)

Howell ve arkadaşları 2013 yılında yaptıkları çalışmada şekil 2.7. deki ayakkabıyı kullanmışlardır. Bu çalışma FSR (Force Sensitive Resistors) kullanılarak gerçekleştirilmiştir. Hareket analiz laboratuvarları çoğu birey için pahalı ve erişilemezdir. Geliştirilen düşük maliyetli ve giyilebilir iç tabanlı yürüme analiz sistemi

6 gönüllü katılımcı üzerinde test edilmiştir. Şekil 2.7. de sensörlerin tabanlığa yerleşimi gösterilmektedir (Howell vd., 2013).

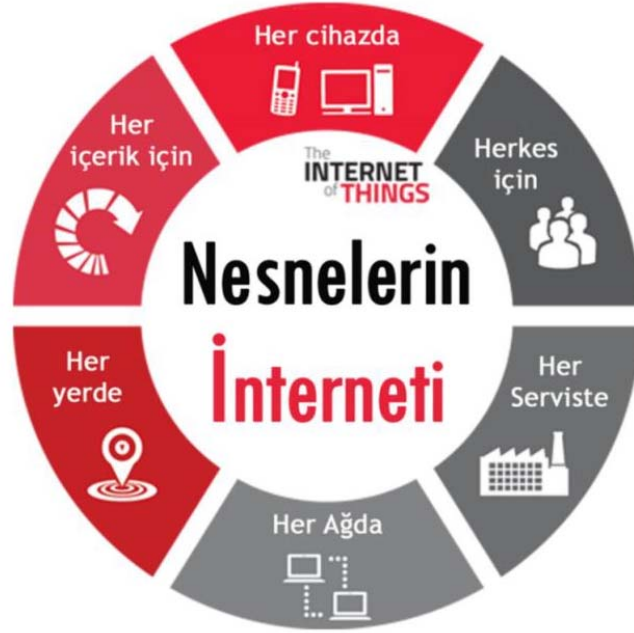


Şekil 2.7. 12 FSR (kuvvet sensörü) den oluşan kinetik analiz çözümü

2.2. Nesnelerin İnterneti

Nesnelerin İnterneti (Internet of Things - IoT); nesnelerin aralarında verileri transfer edip, analiz ettikten sonra karar verebilmeleri için onlara birbirleri ile iletişim kurma imkânı sağlar. Nesnelerin İnterneti; gömülü sistemler, haberleşme ve internet protokolleri gibi temel teknolojilerin geleneksel olandan akıllı olana dönüşmesidir (Atzori vd. 2010; Al-Fuqaha vd., 2015). Nesnelerin İnterneti: İnsan kontrolüne ve bilgi girişine gerek olmaksızın cihazların aralarında iletişim kurarak veri transferi gerçekleştirdiği, veriler toplayıp toplanan veriler ile karar verebildiği bir ağ olarak tanımlanabilir (Atzori vd., 2010).

Şekil 2.8. de Nesnelerin İnterneti genel yapısı yer almaktadır (Arslan ve Kırbaş, 2016).



Şekil 2.8. Nesnelerin İnterneti genel yapısı

2.2.1. İnternetin aşamaları

İnternet birçok aşama sonrasında bugünkü halini almıştır. Bu bölümde bu aşamalardan bahsedilecektir.

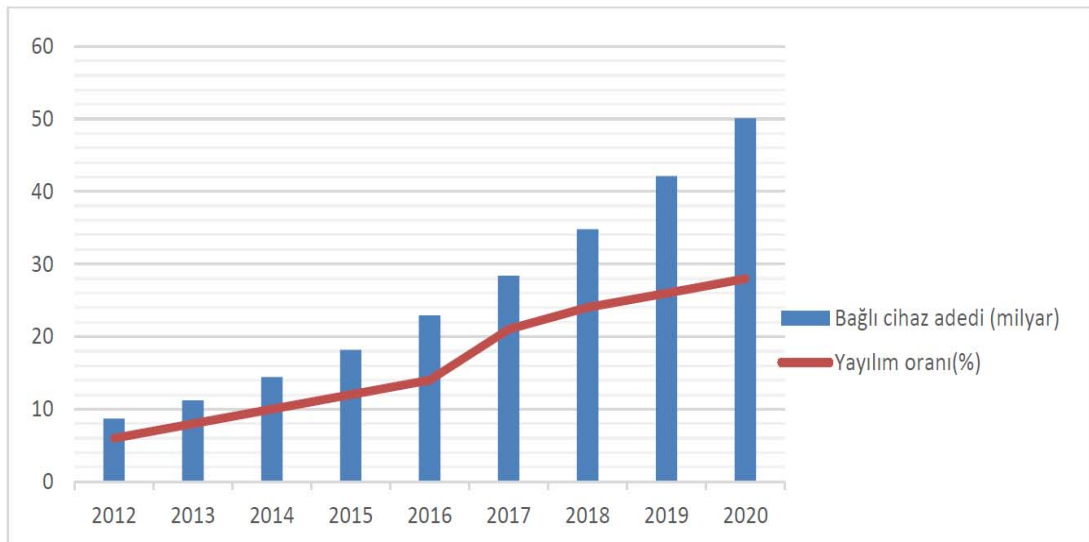
1. Aşama: Araştırma dönemi; ARPANET, ABD savunma bakanlığı bünyesine bağlı ARPA tarafından geliştirilmiş ilk paket dağıtım ağıdır. Halen kullanmaya devam ettiğimiz TCP/IP protokolünün temelleri Stanford Üniversitesi'nde gerçekleştirilen Internetworking projesiyle 1973 yılında atılmıştır. 1978 yılına kadar IPv1, IPv2 ve IPv3 olmak üzere 3 standart geliştirilmiştir. 1980 yılında IPv4 protokolü ortaya çıkmıştır.
2. Aşama: İnternet sitelerinin kullanıcı etkileşimi olmadan sadece bilgilendirme yapmak için kullanıldıkları dönemdir.
3. Aşama: İnternetin kullanıcı etkileşimli, interaktif olmaya başladığı dönemdir. Ürün satın alma işlemlerinin gerçek zamanlı gerçekleşmeye başladığı dönemdir. Bu dönem de Amazon ve eBay türevi birçok ticari uygulama geliştirilmiştir.
4. Aşama: "Sosyal Medya, Semantik Web" kavramları yaygınlaşmaya başlamıştır. YouTube, Facebook, Twitter gibi sosyal medya ağları popüler olmaya başlamıştır.

5. Aşama: Nesnelerin İnternetiyle birlikte “her şey” kavramı değer kazanmış ve tüm cihazların internete bağlanabileceği fikri ortaya çıkmıştır (Özvural, 2015).

Nesnelerin İnterneti’ne ilk kez Kevin Ashton tarafından 1999 yılında Procter & Gamble şirketi için hazırlanan bir tanıtımda değinilmiştir. Bu tanıtımda firmanın tedarik zincirinde Radyo Frekans ile tanımlama (RFID) teknolojisinin kullanımının firmaya faydaları sıralanmakta ve kullanımı tavsiye edilmektedir. Nesnelerin İnterneti 3 temel kavramdan oluşur; nesneler, nesneleri birbirine bağlayan haberleşme ağları ve bilgisayar sistemleridir (Kutup, 2011; López-de-Armentia vd., 2012).

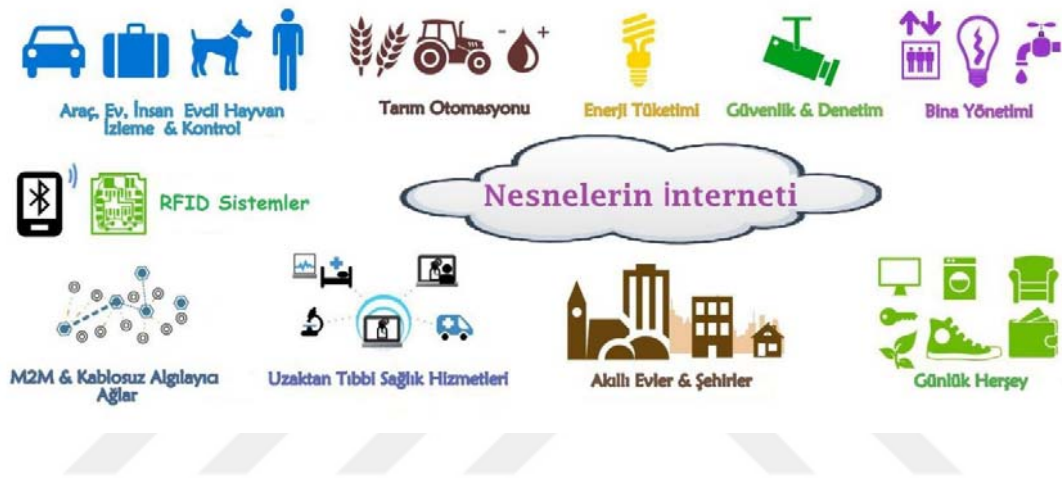
2.2.2. Uygulama alanları

IoT: İnsan müdahalesine ve bilgi girişine ihtiyaç duymadan nesnelerin aralarında iletişim kurduğu ve veri transferi gerçekleştirebildiği ağ yapısı olarak tanımlanmaktadır. Bir diğer tanıma göre ise cihazların aralarında oluşturduğu bir ağ ve bu ağdaki cihazların tanımlı bir protokol eşliğinde birbirleriyle iletişim kurmaları olarak tanımlanabilir. Mevcut bilgiler ışığında günümüzde 10 milyardan fazla nesne internete bağlı iken 2020 yılına gelindiği zaman bu değer 50 milyar cihaz seviyesinde olacağı düşünülmektedir (AKTAŞ vd., 2016). Şekil 2.9 da CISCO verilerine dayandırılarak Nesnelerin İnterneti yayılım beklentisi gösterilmektedir (Evans, 2011).



Şekil 2.9. IOT yayılım beklentisi

IOT: Sağlık, çevrecilik, tarım, hayvancılık, enerji, şehircilik, ölçüm sistemleri, güvenlik, alışveriş, lojistik, E-Sağlık, Ev otomasyonu, lojistik ve ticaret gibi birçok alanda kullanılmaktadır. Bu alanlarda verimlilik ve üretkenliğin artması sonucu daha kaliteli hizmet vermek adına sensörlerden veriler alınır. Alınan bu veriler Büyük Veriyi oluşturur, bu Büyük Veri Bulut Bilişim sistemlerinde depolanır. Çeşitli yöntemlerle veriler analiz edilir. Analiz edilen bu veriler sistemde gerekli iyileştirmelerin yapılmasına katkı sağlar (Gökrem ve Bozuklu 2016). Şekil 2.10. da IoT'in uygulama alanları gösterilmektedir (Aktaş vd., 2014, 2016).



Şekil 2.10. IoT uygulama alanları

Nesnelerin İnterneti'nin kullanımı adına bakım ve sağlık hizmetleri en cazip uygulama alanlarıdır (Pang, 2013). Bu alanda gerçekleştirilen kapsamlı bir çalışmayla Nesnelerin İnterneti'nin sağlık alanında kullanımı ile ortaya koyacağı avantajlar belirtilmiş ve Nesnelerin İnternetinin sağlık alanına sağladığı çözümler belirtilmiştir (İslam vd., 2015).

2.2.3. Nesnelerin İnterneti genel ölçütleri

IoT temel olarak kabul görmüş bir mimari model - katman yapısına sahip olmasa da bazı ölçütleri sağlaması gerekmektedir. Bu ölçütler; güvenilirlik, heterojenlik, ölçeklenebilirlik, birlikte çalışabilirlik ve güvenlik/gizlilik ölçütleridir (Çavdar ve Öztürk, 2018).

2.2.3.1. Güvenilirlik

Bir noktadan diğer noktaya verinin eksiksiz ve doğru iletildiğinden emin olunması veya gönderici ile alıcı nesnelerinde olan verilerin birbirleri ile tutarlılığıdır. Güvenilirlik IoT ortamına bağlı cihazların ve tüm sistemin doğru ve akıcı bir biçimde haberleşmesini, veri iletiminde oluşabilecek gecikme ve hataları azaltmayı amaçlar. Ortama bağlı bir nesnenin hata vermesi durumunda bir diğer nesne bu hatayı giderebilmeli, oluşabilecek veri kaybını önlemeli ve karar sürecinin uzamasına veya hatalı sonuçların elde edilmesine engel olmalıdır. Bu hata olasılıklarının önceden tahmin edilip verinin alıcı nesneye sağlıklı bir biçimde ulaştırılması sağlanmalıdır (Macedo vd., 2014).

2.2.3.2. Kullanılabilirlik (Availability)

IoT her zaman ve her yerde servis hizmeti sunabilmelidir. Uygulamalar aynı anda farklı konumda bulunan çok fazla sayıda farklı kullanıcıya hizmet verme yeteneğine sahip olmalıdır. IPv6 gibi protokoller ile kullanılabilirlik çerçevesi genişletilir. Bu durum IoT'in fonksiyonel bir yapıya sahip olduğunun da göstergesidir (Al-Fuqaha vd., 2015).

2.2.3.3. Taşınabilirlik (Mobility)

Sürekli bir biçimde kullanıcı veya nesne ile bağlantı halinde olmak ve veri iletişimini sağlamak son derece önemlidir. Bir ağdan diğer bir ağa geçiş sırasında kesintiler meydana gelebilmektedir. Bu gibi durumlarda Frieder Ganz ve arkadaşları caching ve tunneling modlarının kullanılabileceği bir model önerisinde bulunmuşlardır. Bu yaklaşım sayesinde veriler önce önbelleğe alınır. Geçici bir kesinti durumunda veri iletiminin kesintisiz gerçekleştirilmesi amaçlanmıştır (Ganz vd., 2012).

2.2.3.4. Ölçeklenebilirlik (Scalability)

Büyük ve karmaşık platformların bulunduğu heterojen bir ağ sisteminde ölçeklemenin yapılması kaçınılmazdır. Verilerin ölçeklenebilmesi için tek tip formatlara dönüştürülmeleri gerekmektedir (Sarkar, 2015).

2.2.3.5. Heterojenlik (Heterogeneity)

Çok sayıda farklı teknolojik altyapı kullanan nesnelerin bağlı oldukları ağda heterojenlik probleminin ortadan kaldırılması gerekmektedir. Bu nedenle sistemin platform bağımsız olması gerekmektedir. Bu nedenle ölçeklenebilirlik ve heterojenlik ölçütleri bir bütün olarak değerlendirilebilir (Çavdar ve Öztürk, 2018).

2.2.3.6. Birlikte çalışabilirlik (Interoperability)

Kurulacak olan sistemlerin platform bağımsız olması gerekmektedir. Bu sayede farklı mimari yapılara sahip nesneler ve farklı dil veya uygulamalar aynı platformda çalışabileceklerdir. Örneğin; WiFi, NFC ve GSM operatörleri platform bağımsız olarak çalışmaktadırlar (Al-Fuqaha vd., 2015).

2.2.3.7. Güvenlik/Gizlilik (Security/Privacy)

Veri iletiminde verinin gizliliği ve güvenliği büyük önem teşkil etmektedir. Ancak IoT geliştirmekte olan bir yapı olduğundan halen güvenlik ve gizlilik konusunda eksiklikleri bulunmaktadır. Temel veya referans kabul edilen bir mimari modelin eksikliği görülmektedir (Çavdar ve Öztürk, 2018).

2.2.3.8. Performans (Performance)

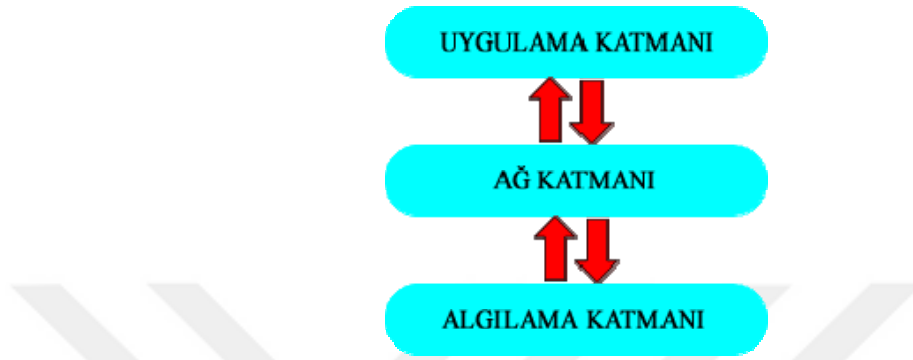
Binlerce nesnenin ve birden fazla sayıda platform ve teknolojinin birbirine bağlandığı IoT' de performans ölçütü üzerinde yoğunlaşmak gerekmektedir. Çünkü sistemin her bir parçasının performansı sistemi etkileyecektir fakat teknolojik altyapı farklılıklarından ve nesnelerden kaynaklanan problemlerin mimari yapıyı etkilememesi gerekmektedir (Al-Fuqaha vd., 2015).

2.2.4. Nesnelerin internetinde kullanılan mimari modeller

Literatürde IoT için mimari model önerisinde bulunan birçok çalışma vardır. Bu bölümde bazı mimari modeller anlatılacaktır.

2.2.4.1. Üç Katmanlı Mimari model

Nesnelerin İnterneti mimari modellerinden genel amaçlı olarak temel kabul edilen 3 katmandan oluşan mimari modeldir. Şekil 2.11. de 3 katmanlı mimari model gösterilmiştir.



Şekil 2.11. Üç Katmanlı Mimari model

Algılama Katmanı (Perception Layer): Bu katman Nesnelerin İnterneti' nin duyu organı gibidir. Veri toplama ve nesnelerin algılandığı katman olarak özetlenebilir. RFID, Zigbee ve NFC gibi protokoller bu katmanda kullanılır (Yang vd., 2011).

Ağ Katmanı (Network Layer): Veri iletimi ve veri işleminin gerçekleştiği katmandır. Algılama katmanından gelen verileri işlenir ve bir üst katmana iletilir. IPv6, LowPAN, UDP, ICMP gibi protokoller bu katmanda kullanılır (Yang vd., 2011).

Uygulama Katmanı (Application Layer): Tamamlayıcı katmandır. Verilerin kullanılabilir hale geldiği ve sonuçlarının gözlemlenebildiği katmandır. Uygulama katmanında CoAP protokolü kullanılır (Yang vd., 2011).

2.2.4.2. Servis Odaklı Mimari (SOA)

Servis Odaklı Mimari temelinde 5 katmandan oluşur. Şekil 2.12. de Servis Odaklı Mimari modelin genel yapısı ve katmanlar arası iletişim gösterilmektedir.



Şekil 2.12. Servis Odaklı Mimari Model

Nesne Katmanı (Object Layer): Mimari yapının ilk katmanıdır. Fiziksel işlemlerin gerçekleştiği, algılayıcılardan verilerin alındığı katmandır. Alınan analog veriler bu katmanda dijital sinyallere dönüştürülerek bir sonraki katmana iletilir. Büyük veriler bu katmanda oluşmaya başlar (Gantz ve Reinsel, 2012).

Nesne Soyutlama Katmanı (Object Abstraction Layer): Verilerin çeşitli kanallara yönlendirildiği katmandır. GSM, WiFi, 3G, Zigbee gibi çeşitli teknolojilerde kullanılacak verilerin dönüşümü bu katmanda gerçekleşir. Veri yönetimi ve bulut bilişim ilk kez karşımıza bu katmanda çıkar (Gantz ve Reinsel, 2012).

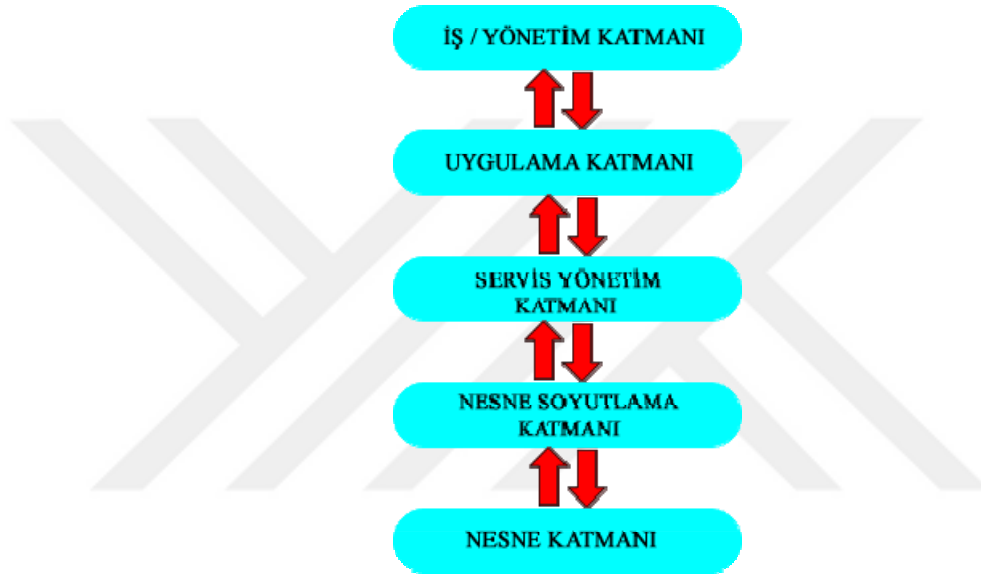
Servis Yönetim Katmanı (Service Management Layer): Bu katman verileri alır, işler, karar verir ve istemciye gitmek üzere servis birleştirme katmanına yönlendirir (Gantz ve Reinsel, 2012).

Servis Birleştirme Katmanı (Service Composition Layer): Gelen verileri ve servisleri birleştirerek toplu bir hizmet sunan katmandır. Ayrıca bu katmanda birlikte bir anlam taşıyan servisler birden fazla uygulama için kullanılabilir (Gantz ve Reinsel, 2012).

Uygulama Katmanı (Application Layer): İşlenmiş veriyi kullanıcıya sunan ve kullanmasına imkân veren katmandır (Gantz ve Reinsel, 2012).

2.2.4.3. Beş Katmanlı Mimari

Mimari yapılar içinde en temel yapılardan bir tanesidir. ITU (International Telecommunication Union) tarafından geliştirilmiş bir mimari modeldir. Bu mimari yapının en önemli özelliği bir yönetim katmanının bulunmasıdır. Yönetim katmanında, diğer katmanlarda meydana gelen aksaklıklara veya yönlendirmelere müdahale edilebilir (Al-Fuqaha vd., 2015). Şekil 2.13. de Beş Katmanlı Mimari modelin genel yapısı ve katmanlar arası iletişim gösterilmektedir.



Şekil 2.13. Beş katmanlı mimari model

Nesne Katmanı (Object Layer): Algılama katmanı olarak tanımlanabilir. Diğer mimari modellerde de olduğu gibi ısı, kuvvet, hız, titreşim, ivme gibi fiziksel veya analog veriler bu katmanda algılanır ve verilerin ilk işlenmeye başlandığı katmandır. Büyük veri (Big Data) bu katmanda oluşmaya başlar (Al-Fuqaha vd., 2015).

Nesne Soyutlama Katmanı (Object Abstraction Layer): Alt katmandan gelen verilerin dönüştürülerek servis yönetim katmanına iletildiği katmandır. Veriler Kızılötesi, Bluetooth, WiFi gibi çeşitli teknolojilerde kullanılmak üzere bu katmanda dönüştürülür. Bulut Bilişim (Cloud Computing) ve veri yönetimi işlemleri bu katmanda gerçekleşir (Al-Fuqaha vd., 2015).

Servis Yönetimi Katmanı (Service Management Layer): Adres ve toplanan verilerin istemciden gelen isteğe bağlı olarak bir servis yardımıyla karşılaştırıldığı

katmandır. Verileri alınır, işlenir ve uygulama katmanına yönlendirilir (Al-Fuqaha vd., 2015).

Uygulama Katmanı (Application Layer): Kullanıcılara verilerin anlaşılabilir bir şekilde sunulduğu katmandır. En önemli özelliği kullanıcılara istedikleri gibi verilerin gösterilebileceği uygulama ve ortamların olmasıdır (Al-Fuqaha vd., 2015).

İş / Yönetim Katmanı (Business Layer): Diğer katmanları kontrol eden katmandır. Yönetim ve gerektiğinde yönlendirilmelerin yapılabildiği bu katmanda her katmanın çıktısı karşılaştırılır ve sistemin doğru bir biçimde işleyişini sağlar. Sistemdeki büyük veri kümelerinin analizi, işlenmesini ve karar vermesini mümkün kılar (Taylor, 2013; Khan vd., 2012).

2.2.4.4. Dört Katmanlı Mimari model

Cihaz, ağ, servis, uygulama olmak üzere 4 katmandan oluşan bu mimari model P. Furtado NV. Lopes ve arkadaşları tarafından engellilere yardımcı olabilmek amacıyla geliştirdikleri bir mimari modeldir (Lopes vd., 2014).

2.2.4.5. Altı Katmanlı Mimari model

Kodlama, bilgi toplama, bilgi erişimi, ağ, bilgi birleştirme, uygulama olmak üzere toplam 6 katmandan oluşan bu mimari model 2012 yılında X. Cheng ve arkadaşları tarafından geliştirilmiştir (Zhang vd., 2012).

2.2.5. Haberleşme protokolleri

Nesnelerin İnterneti uygulamaları için en önemli bileşenlerden biri de haberleşme kurallarının belirlendiği haberleşme protokolleridir. Bir haberleşme protokolü aynı veya farklı ağlarda bulunan uç birimlerin birbirleriyle haberleşebilmeleri amacıyla geliştirilmiş özel kurallar ve kısıtlamalar şeklinde tanımlanabilir (Arslan ve Kırbas, 2016). Pek çok haberleşme protokolü vardır. Bunlardan en yaygın kullanılanı MQTT ve CoAP tır.

MQTT (Message Queue Telemetry Transport): MQTT istemci sunucu mimarisini kullanıldığı mesaj yayını yapan ve bu yayınlara sadece abone olan cihazların bağlanabildiği bir mesaj transfer protokolüdür. Sınırlı kaynaklara sahip cihazlar

tarafından kolayca uygulanabilmesi amacıyla açık, sade ve çalıştırma yükü düşük bir protokol olarak geliştirilmiştir. Bu protokol TCP/IP gibi kayıpsız ve çift yönlü haberleşmeyi destekleyen ağ protokolleri üzerinde çalışabilir. Bire çok mesaj dağıtımını destekler (Arslan ve Kırbaş, 2016).

CoAP (Constraint Application Protocol) protokolü: Düşük enerji, kayıplı iletişim gibi sınırlı kaynaklara sahip düğümler ve ağlar için geliştirilmiş bir web transfer protokolüdür. Düğümler genellikle 8 bitlik mikro işlemcilere ve düşük miktarlarda RAM ve ROM kapasitesine sahiptirler. Haberleşme hızı tipik olarak 10 Kbps değerini aşmaz ve yüksek hata oranına sahiptirler. Daha çok akıllı bina, akıllı şehir gibi M2M uygulamaları için tasarlanmıştır (Arslan ve Kırbaş, 2016).

2.2.6. Nesnelerin İnterneti'nde kullanılan haberleşme teknolojileri

Nesnelerin İnterneti teknolojisinin önemli bir parçası nesnelerin iletişimi ve bilgi paylaşımıdır. Sistem için uygun haberleşme teknolojisinin seçimi; verimlilik, zaman, ekonomi ve güvenlik anlamında farklı başlıklar ele alınarak yapılmalıdır. Doğru seçilmiş teknoloji sisteme fayda sağlayacaktır. Nesneler arası haberleşme için farklı teknolojilerden faydalanılmaktadır. Bu bağlantı teknolojilerinden en çok kullanılanları Bluetooth, Zigbee, WiFi, 6LoWPAN, Hücreli iletişim olarak sıralanabilir (Bozdoğan, 2015).

2.2.6.1. Zigbee

IEEE 802.15.4 standardını baz alan ve düşük güç tüketimi özelliğine sahip kablosuz iletişim teknolojisidir. Arılardan esinlenerek isimlendirilmiştir. Dört katmandan oluşan Zigbee yapısı ayrıca yıldız, noktadan noktaya ve ağaç ağ topolojileri gibi çeşitli yapıları desteklemektedir (Uğuz vd., 2013).

2.2.6.2. Bluetooth

IEEE 802.15.1 olarak standartlaştırılmış kablosuz haberleşme teknolojisidir. Kısa mesafeli çalışmalar için tercih edilip, özellikleri: Düşük ücret gereksinimi ve düşük güç tüketimi olarak sıralanabilir. Veri iletimi için iki farklı mod kullanılmaktadır. Bu

modlar master ve slave modlarıdır. Radyo Frekans tabanlı olan bu teknoloji noktadan noktaya iletim ara yüzü olarak kullanılmaktadır (Ulaş, 2015).

2.2.6.3. 6LoWpan

İnternet Protokolü (IP), ağ üzerinde ki aygıtların iletişim kurabilmeleri için kullandıkları protokoldür. Aygıtların ya da nesnelerin internet ortamından iletişim kurabilmeleri için her aygıtın bir adresi olmalıdır. Bu adrese IP adres denir. Nesnelerin interneti teknolojisi ile çok fazla nesne kullanılmakta ve bu durumda çok fazla IP adres ihtiyacı doğurmaktadır. Günümüzde IPv4 standardı yeterli aygıt adresleyemediği için IPv4'ün yerine daha gelişmiş özelliklere sahip IPv6 sistemine geçilmiştir. Böylece daha fazla nesnenin adreslemesi sağlanacaktır (Bozdoğan, 2015).

İnternet üzerinde araştırmalar yapan bir grubun adı olan 6LoWPAN, IPv6'nın kısaltmasıdır. IEEE 802.15.4 standardına uygun olarak çalışan bir teknolojidir. Paketleri gönderme, alma, kapsülleme ve sıkıştırma özelliklerini barındırmaktadır. Özellikleri: Düşük güç tüketimi ve küçük cihazlar üzerinde çalışabilmesidir (Ulaş, 2015).

2.2.6.4. Wifi

IEEE 802.11 baz alan WiFi teknolojisi nerdeyse tüm akıllı cihazlarda ve bilgisayarlarda kullanılmaktadır. Büyük boyutlu veri aktarımı yapılabilmektedir. Buna rağmen fazla güç tüketimi nedeniyle nesnelerin internetinde diğer teknolojiler kadar çok tercih edilmemektedir (Özköse, 2014).

2.2.6.5. Hücresel

Nesnelerin internetinde uzun mesafeli iletişim için GSM/GPRS teknolojilerinden faydalanılmaktadır. Birçok uygulamanın uzaktan izlenmesi, uzaktan kontrol edilmesi ve otomatik olarak işlemlerin yapılmasında bu teknoloji kullanılmaktadır. Özellikle düşük maliyetli sistemlerde daha çok tercih edilmektedir. Uygulamanın veri iletim hızı 2G, 3G ve 4G hücresel iletişim seçeneklerinde farklı olmaktadır. Genelde bu sistemler için uzun ömürlü, fiziksel koşullara dayanıklı ve gömülü olarak kullanılan SIM kartlar tercih edilmektedir (Ulaş, 2015).

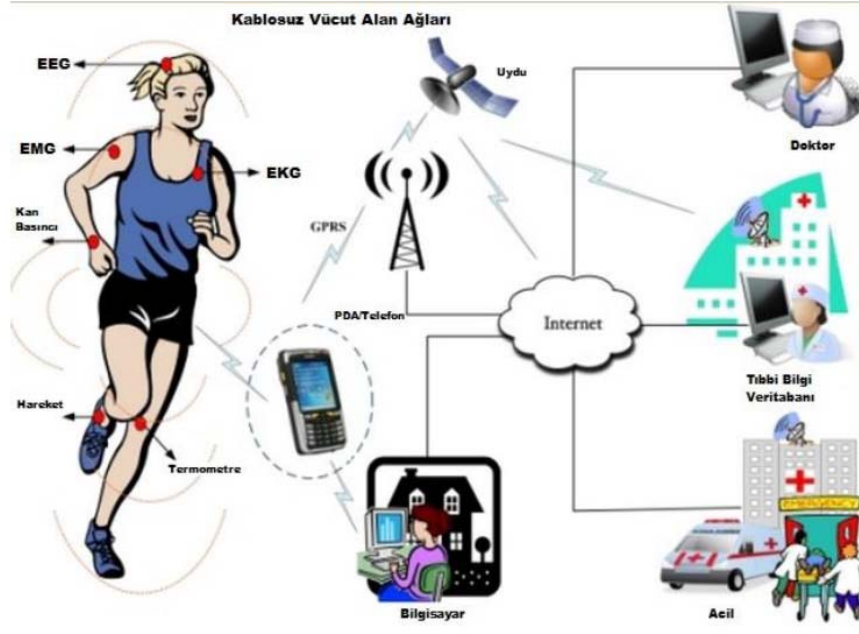
Çizelge 2.1. de haberleşme teknolojilerinin karşılaştırılması gösterilmiştir (Bozdoğan, 2015; Ulaş, 2015).

Çizelge 2.1. Haberleşme teknolojilerinin karşılaştırılması

Teknoloji	Zigbee	Gsm/Gprs	Wifi	Bluetooth
Odaklanma alanı	İzleme ve kontrol	Geniş alan ses ve veri	Web, e-posta, görüntü	Kablo yerine
Sistem kaynağı	4-32 Kb	16 Mb +	1 Mb +	250 Kb +
Pil ömrü (gün)	100-1000	1-7	0,5-5	1-7
Ağ boyutu (adet)	Sınırsız	1	32	7
Ağ veri genişliği (kb/s)	100-1000	64-128	11000	720
Kapsama alanı (metre)	1-100	1000	1-100	1-10
Frekans	2.4 Ghz	900/1800/1900/2100 Mhz	2,4 Ghz, 5 Ghz	2,4 Ghz
Veri transferi	250 Kbps	3-10 Mbps (4G) 600 Kbps – 10 Mbps (3G)	Max 600 Mbps	1 Mbps
Başarı alanları	Dayanıklılık, maliyet, güç tüketimi	Ulaşılabilirlik, kalite	Hız, esneklik	Maliyet, rahatlık

2.2.7. Kablosuz Vücut Alan Ağları (KVAA)

Kablosuz Vücut Alan Ağları (KVAA): İnsanlara ait fizyolojik verileri alabilen, kablosuz haberleşebilen ve verileri işleyebilen akıllı cihazlardan oluşmaktadır. KVAA çalışmalarında kullanılan donanımlar genellikle vücuda yerleştirilebilen algılayıcı cihazlar, hap şeklinde kameralar, yapay retina, vücuda takılabilen EKG/EMG/EEG/SpO2, kan basıncı ve sıcaklık ölçer gibi aygıtlardır (Kırbaş, 2013). Şekil 2.14. de örnek bir KVAA gösterilmiştir (Aktaş vd., 2014).



Şekil 2.14. Kablosuz Vücut Alan Ağları (KVAA) (EKG= Elektrokardiyogram, EEG= Elektroensefalografi, EMG= Elektromiyografi)

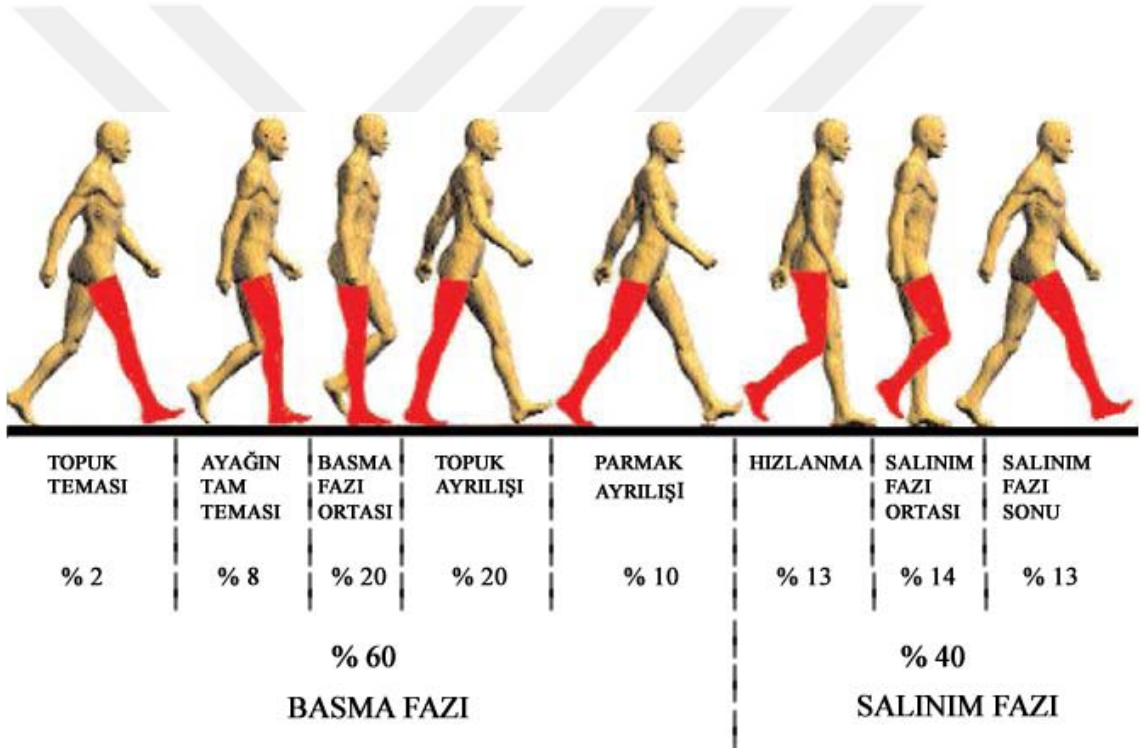
2.3. Yürüme

Yürüme, canlıların bir noktadan bir noktaya hareket etmesi esnasında gövdenin hareket edilecek noktaya doğru ilerletilmesi için kol ve bacakların koordineli ve tekrarlanan hareketlerinin bütünüdür (Özaras ve Yalçın, 2001). Normal yürüme için sinir sistemi ve kaslar birbiriyle ve diğer sistemlerle uyumlu çalışmak zorundadır. Bu yapılarda ki herhangi bir bozukluk patolojik yürümeye neden olabilmektedir (Yavuzer, 2009).

2.3.1. Yürüme döngüsü

Yürüme esnasında, vücudu ilerletebilmek için bir ayak topuğunun zemine değmesi ile aynı ayağın havalanıp tekrar zemine değmesi arasında bacaklarda bazı hareketler oluşur ve bu hareketler aynı şekilde sürekli devam eder. Bir düzen içinde sürekli tekrarlanan bu hareketlere yürüme döngüsü denir. Yürüme esnasında tek ayak yerle temas etmek zorundadır. İnsan yürüme esnasında önce bacağının bir tanesini öne atar; onun üzerinden ağırlığına zemine aktardıktan sonra diğer bacağın yerden kaldırır ve öne doğru ilerletir. İnsan yürüme eylemini gerçekleştirirken hem dengede durma hem de vücudu öne ilerletmeye çalışmaktadır. Her iki ekstremite göz önüne alındığında;

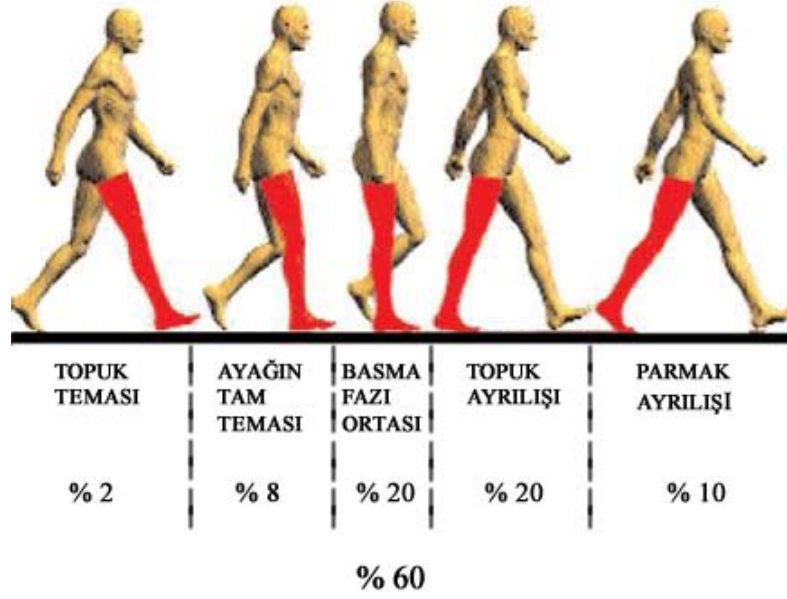
Sadece tek bir ayağın yerde olduğu faz tek destek fazı, iki ayağın aynı anda yere basması ise çift destek fazı olarak adlandırılır. Sadece tek bir bacak göz önüne alındığında ise; bir bacağın yerle temas ettiği süre basma fazını, havada askıda kaldığı süre ise salınım fazını oluşturur (Milli Eğitim Bakanlığı 1, 2011; Tunca vd., 2017). Şekil 2.15. de tek destek fazını oluşturan basma ve salınım fazları ve alt fazlar ile bu fazların ortalama bir yürüme döngüsünde yürümenin yüzde kaçını oluşturdukları detaylı bir şekilde gösterilmektedir. Görüldüğü gibi basma fazı beş alt fazdan oluşmaktadır. Bu alt fazlar: Topuk teması (% 2), ayağın tam teması (% 8), basma fazı ortası (% 20), topuk ayrılığı (% 20) ve parmak ayrılığı (% 10). Salınım fazı ise üç alt fazdan oluşur. Bunlar: Hızlanma (% 13), salınım fazı ortası (% 14), salınım fazı sonudur (% 13). Yürüme döngüsünün yüzde 60' ı basma, yüzde 40' ı salınım fazından oluşmaktadır.



Şekil 2.15. Tek Destek Fazı ve alt fazları

2.3.1.1. Basma fazı ve evreleri

Basma fazı, topuk teması ile başlar, parmak ayrılığı ile sona erer (İmrenk, 2011). Yürüme döngüsü %60 basma fazından oluşur. Beş alt fazdan meydana gelir. Bunlar: Topuk teması, ayağın tam teması, basma fazı ortası, topuk ayrılığı ve parmak ayrılığıdır. Şekil 2.16. da basma fazı alt evreleri ve yüzdelik dilimleri detaylı bir şekilde verilmiştir.



Şekil 2.16. Basma fazı ve alt fazları

Çalışmamızda ayak tabanındaki kuvvet dağılımlarını ölçeceğimiz için tabana sensör yerleşimi çalışmamızın önemli bir unsurudur. Bu konuda basma fazı ve alt fazlarında ayak tabanının hangi bölümlerine kuvvet uygulanıyor bilinmesi gereken önemli bir konudur. Şekil 2.17. de basma fazı alt evrelerinde ayak tabanının yerle temas eden bölümleri detaylı bir şekilde gösterilmiştir (Milli Eğitim Bakanlığı 1, 2011).

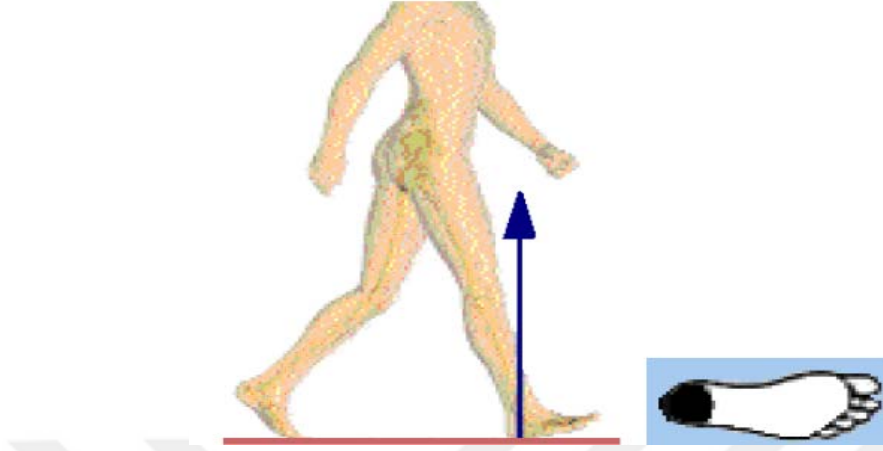


Şekil 2.17. Basma fazında ayak tabanının yer ile teması

2.3.1.1.1. Topuk teması

Basma fazının ilk evresini oluşturur. Bir ayağa ait topuk yere değdiğinde basma fazı başlar. Yürüme döngüsünün % 2 sini oluşturan bu evre basma fazının başlangıcıdır. Amaç, ayağı yere indirmektir. Topuk yer ile ilk temas ettiği zamanda vücut ayağın gerisinde, Vücut Ağırlık Merkezi (VAM) en yüksek hızına ulaşmıştır. Yer Tepkimesi Kuvvet Vektörü (YTKV) kalçanın önündedir (Milli Eğitim Bakanlığı 1, 2011). Şekil

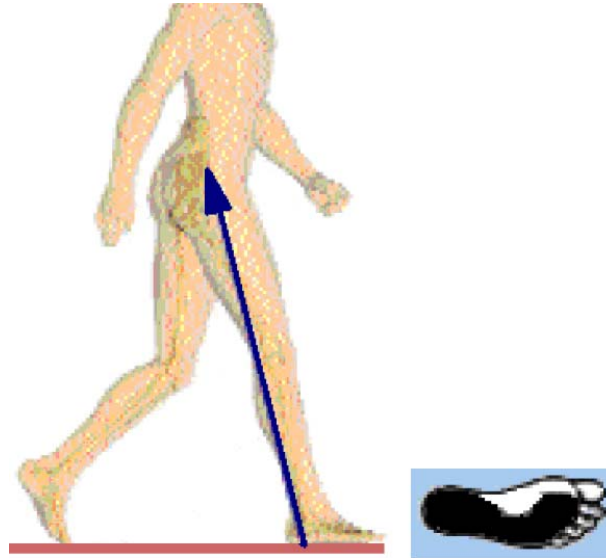
2.18. de topuk teması fazında yer tepkimesi kuvvet vektörü ve ayak tabanı zemin teması gösterilmektedir (Milli Eğitim Bakanlığı 1, 2011).



Şekil 2.18. Topuk teması fazında YTKV' in konumu

2.3.1.1.2. Ayağın tam teması

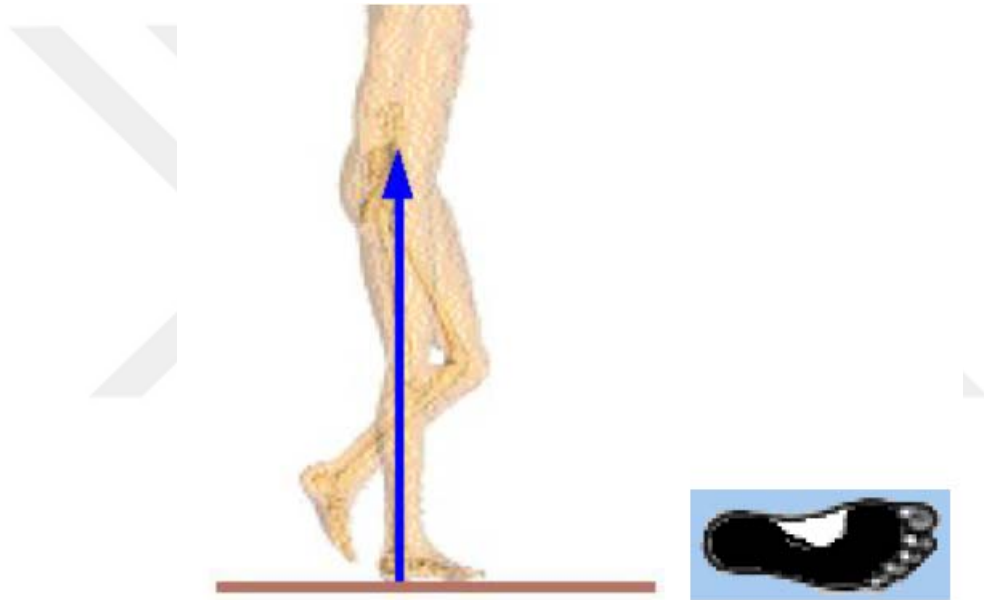
Topuğun yerle temasından bir süre sonra ayak tabanı da yere temas eder. Yürüme döngüsünün % 8'lik kısmını oluşturur. Diğer ayak zeminle temasını kesene kadar vücut ağırlığı bu ayak üzerine aktarılır. Amaç ayağın tamamen zemine indirilmesi ve gövde ağırlığının üstlenilmesidir. Vücut Ağırlık Merkezi (VAM) yükselmeye başlar. Şekil 2.19. da tam temas fazında vücut ağırlık merkezinin konumu gösterilmiştir (Milli Eğitim Bakanlığı 1, 2011).



Şekil 2.19. Ayağın tam teması ve VAM' ın konumu

2.3.1.1.3. Basma fazı ortası

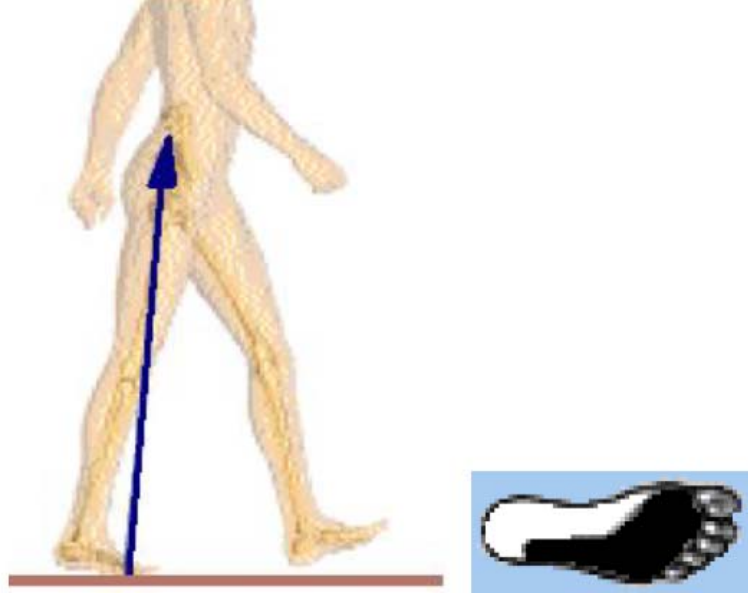
Vücut ağırlık merkezi, dikey şekilde sabit ayak üzerindedir. Yürüme döngüsünün % 20'lik kısmını oluşturur. Tek basma fazının başlangıcıdır. Salınım fazını gerçekleştiren bacak, basma fazını gerçekleştiren bacağın yanından geçer. Amaç yerle temas eden ayak üzerinde vücudu öne doğru ilerletmektir. Bu dönemde Vücut Ağırlık Merkezi'nin öne doğru hızı en az seviyeye inerken yüksekliği en üst ve en dış noktaya ulaşır. Yer Tepkimesi Kuvveti Vektörü ayak bileğinin önünden, kalçanın ortasından geçtiği için kalça kasları çalışmaz. Şekil 2.20. de basma fazı ortası evresinde YTKV' in konumu, ayak tabanı kuvvet dağılımı gösterilmektedir (Milli Eğitim Bakanlığı 1, 2011).



Şekil 2.20. Basma fazı ortası

2.3.1.1.4. Topuk ayrılışı

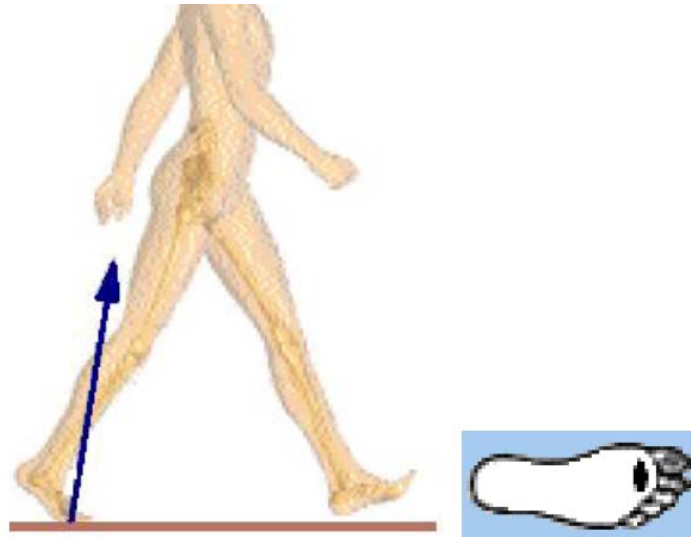
Yürüme döngüsünün % 20 sini oluşturur. Tek basma fazı bu bölümde bitmektedir. Amaç, basma fazındaki bacağın zeminle temasının kaybolmasıdır. Vücut Ağırlık Merkezi'nin yüksekliği ve yana kayması giderek azalır. Yer Tepkimesi Kuvveti Vektörü dizin ve ayak bileğinin önünde, kalçanın arkasında yer almaktadır (Milli Eğitim Bakanlığı 1, 2011). Şekil 2.21. de topuk ayrılışı fazı gösterilmektedir (Milli Eğitim Bakanlığı 1, 2011).



Şekil 2.21. Topuk ayrılışı fazı

2.3.1.1.5. Parmak ayrılışı

Yürüme döngüsünün % 10'unu oluşturan bölümdür. Basma fazının bitip salınım fazının başladığı, diğer bacağın ise basma fazına hazırlandığı dönemdir. Diğer bacak yere temas ettiğinde başlar ve yerdeki bacağın parmaklarının yerden kesilmesiyle sonlanır. Vücut ağırlığı bacak üzerinden kalkar. Amaç, bacağın salınım fazına hazırlanmasıdır. Ayak parmakları zeminle teması kesince Yer Tepkimesi Kuvveti Vektörü kaybolmaktadır (Milli Eğitim Bakanlığı 1, 2011). Şekil 2.22. de parmak ayrılışı fazı gösterilmektedir (Milli Eğitim Bakanlığı 1, 2011).



Şekil 2.22. Parmak ayrılışı fazı

2.3.1.2. Salınım Fazı ve Evreleri

Bu faz, parmakların zeminle temasının kesilmesiyle başlar ve topuğun zemine temas etmesiyle sona erer (Milli Eğitim Bakanlığı 1, 2011). Ayağın yerle temas etmediği, basma fazına hazırlandığı evredir. Yürüme döngüsünün yüzde 40'ını oluşturan bölümdür. Üç alt bölümden oluşur. Bunlar: Hızlanma (% 13), salınım fazı ortası (% 14), salınım fazı sonudur (% 13). Şekil 2.23. de salınım fazı alt evreleri ve yürüme döngüsündeki yüzdeleri yer almaktadır (Milli Eğitim Bakanlığı 1, 2011).



Şekil 2.23. Salınım fazı ve alt evreleri

2.3.1.2.1. Hızlanma

Hızlanma fazı, parmağın zeminle temasının kesildiği anda başlar. Topuğun tekrar zeminle temas edebilmesi için ayak hızlanmalıdır. Yürüme döngüsünün % 13'ünü oluşturur. Ayağın zeminle temasının kesilmesiyle başlayan bu faz ayak diğer bacağın yanına geldiğinde sonlanır (Milli Eğitim Bakanlığı 1, 2011). Şekil 2.24. de hızlanma fazı gösterilmektedir (Milli Eğitim Bakanlığı 1, 2011).



Şekil 2.24. Hızlanma fazı

2.3.1.2.2. Salınım fazı ortası

Havadaki ayağın öne doğru hızlanması ile bu faz başlar. Bu esnada, ayağın zemine temas etmeyecek şekilde havada olması gerekir. Yürüme döngüsünün % 14'ünü oluşturur. Amaç ayağın yere değmeden öne aktarılmasıdır (Milli Eğitim Bakanlığı 1, 2011). Şekil 2.25. de salınım fazı ortasında salınım halindeki ayağın durumu gösterilmektedir (Milli Eğitim Bakanlığı 1, 2011).



Şekil 2.25. Salınım fazı ortası

2.3.1.2.3. Salınım fazı sonu

Yürüme döngüsünün % 13'lük kısmını oluşturur. Salınım fazında ki bacak, basma fazında ki bacağı geçtiğinde başlar; ayağın zeminle temas ettiği zamana kadar

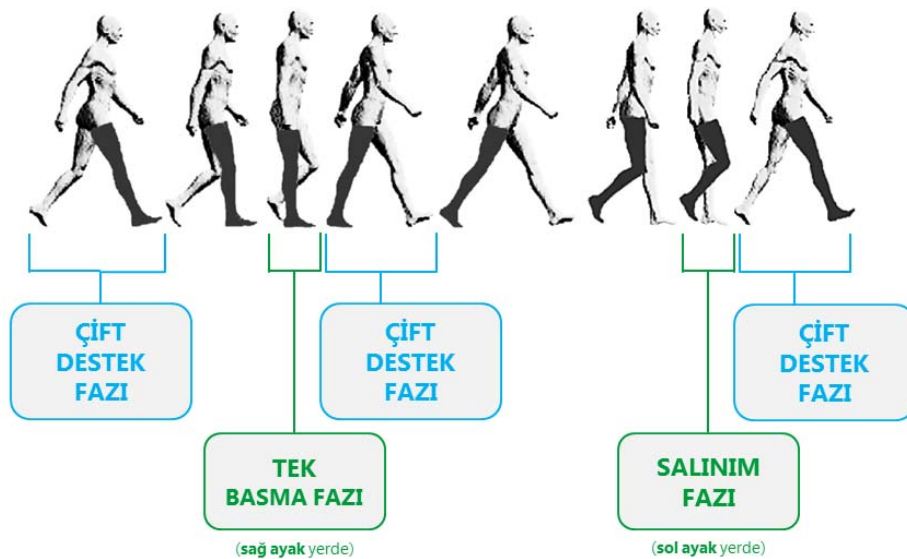
devam eder. Amaç, ayağın yere basmaya hazırlanmasıdır (Milli Eğitim Bakanlığı 1, 2011). Şekil 2.26. da salınım fazı sonu gösterilmiştir (Milli Eğitim Bakanlığı 1, 2011).



Şekil 2.26. Salınım fazı sonu

2.3.1.3. Çift Destek Fazı

Yürüme döngüsünde her iki bacağın da zeminle temas halinde olduğu faza çift destek fazı denir. Çift Destek Fazı, bir ayağın geçiş hali ile parmak kalkışı ve diğer ayağın topuk teması ve ayağın zemine tam teması arasında oluşur. Bu fazda gövde ağırlığı bir bacedan diğer bacağa aktarılır. Yürüme hızı artarsa çift destek fazı kısalır, salınım fazı ise uzar (Milli Eğitim Bakanlığı 1, 2011). Şekil 2.27. de Çift Destek Fazı detaylı olarak gösterilmektedir (Esentürk, 2014).



Şekil 2.27. Çift Destek Fazı

2.3.2. Yürümenin incelenmesinde kullanılan terimler

Yürümenin incelenmesinde kullanılan birçok terim vardır. Bu değerler yürüme analizinde elde edilmesi gereken değerlerdir. Çalışmamızda gönüllü katılımcılardan bu bilgilerden bazıları alınacaktır.

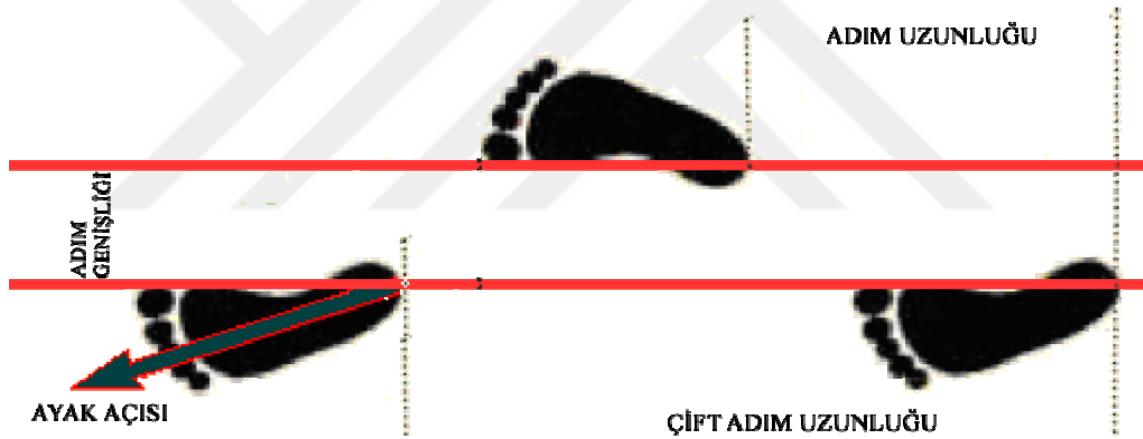
Adım: Bir ayağın yerle temas halinde iken diğer ayağın yerle temasa geçme eylemi.

Adım Uzunluğu: Bir adımda kat edilen mesafe.

Adım Genişliği: Ayak topuklarının, yerle temas ettikleri bölümler arasında yürüme doğrultusuna dik olarak ölçülen uzaklık.

Çift Adım Uzunluğu: Tek ayağın iki topuk vuruşu aralığındaki mesafedir.

Ayak Açısı: Gidiş yönü ile ayağın kesiştiği açıdır (Kanatlı vd., 2006; İmrenk, 2011; Özmanevra, 2015). Şekil 2.28. de bu terimler açıklanmıştır.



Şekil 2.28. Yürümenin incelenmesinde kullanılan terimler

Kadans: Birim zamanda atılan adım sayısı (adım / zaman).

Hız: Birim sürede kat edilen mesafe (uzaklık/zaman). Normal hayatta bir insanın rahat yürüme hızı ortalama 80 m/dk'dır. Dolayısıyla bir yürüme döngüsü 1 saniyeyi biraz geçmektedir (Kanatlı vd., 2006; İmrenk, 2011; Özmanevra, 2015).

Çizelge 2.2. de bazı yürüme değerleri kadın ve erkeklerde ortalama olarak verilmiştir (İmrenk, 2011).

Çizelge 2.2. Ortalama yürüme değerleri

	ERKEK	KADIN
Adım Uzunluğu(cm)	79	66
Çift Adım Uzunluğu(cm)	158	132
Dakikadaki Adım Sayısı(adım/dk)	117	117
Hız(m/sn)	1,54	1,31
Adım Genişliği(cm)	8,1	7,1
Adım Açısı	7	6

Vücut Ağırlık Merkezi: İnsan vücudunun yerçekiminden etkilenecek yere doğru inen ağırlık kuvvet vektörüdür.

Destek Alanı Merkezi: Ayağın yerle temas eden kısmının orta noktasıdır. Yürüme boyunca sürekli yer değiştirir.

Yer Tepkimesi Kuvveti: Newton'un kanunlarının üçüncüsüne göre ayakta durmakta olan bir insanın yerde oluşturduğu kuvvete yer de aynı büyüklükte ve ters yönde bir kuvvet ile karşılık verir. Bu vektöre Yer Tepkimesi Kuvveti Vektörü denir.

Dış Moment: Yer Tepkimesi Kuvvet Vektörü ayak bileğini, kalça ve dizi harekete zorlar. Oluşan bu etkiye dış moment denmektedir.

İç Moment: Dış momente karşı koymak ve hareketi gerçekleştirmek maksadı ile kasların kasılması sonucu eklemde oluşan momenttir (İmrenk, 2011; Özmanevra, 2015).

2.3.3. Anatomik düzlemler

Yürümede önemli olgulardan biride anatomik düzlemlerdir. İnsan anatomisi Sajital, koronal ve transvers olmak üzere 3 düzlemde ele alınır. Çalışmamızın çeşitli yerlerinde bu terimler geçtiği için bu düzlemleri bilmemiz gerekir.

- **Sajital Düzlem:** Vücudu yürüme yönüne göre sağ ve sol olarak iki eşit parçaya bölen düzlemdir. Bir insana yürüme yönüne dik bir doğrultudan bakıldığında Sajital Düzlem görünür.
 - **Koronal veya Frontal düzlem:** Vücudu yürüme yönüne dik bir doğrultuda eşit olarak bölen düzlemdir. Bir insana ön veya arka taraftan bakıldığında Frontal Düzlem görünür.
 - **Transvers:** Vücudu alt ve üst olmak üzere iki parçaya bölen düzlemdir. Bir insana tepeden bakıldığında Transvers Düzlem görünür (İmrenk, 2011).
- Şekil 2.29. da anatomik düzlemler gösterilmektedir (İmrenk, 2011).



Şekil 2.29. Anatomik düzlemler

2.3.4. Patolojik yürüme

Patolojik yürüme gövdenin tekrarlı doğal hareketler ile ileriye doğru taşınması sırasında oluşan aksaklıklar bütünü olarak tanımlanmaktadır. Yürüme siklusunu olumsuz şekilde etkilemekte olan patolojik değerler: Denge, azalmış duyuşal ya da motor kontrol, ağrı, deformite olarak sıralanmaktadır. Ağrı genellikle yürüme döngüsünde hızın azalmasına; deformite, yürüme döngüsü esnasında gerekli eklem hareket açıklığının sağlanamamasına denir (Sarıkaya ve İnan, 2014).

2.4. Yürüme Analizi

Hareket bir nesnenin sabit bir noktaya göre yer değiştirmesidir. Varlıkların hareketlerini mekanik prensipler kullanarak açıklayan bilim dalına biyomekanik adı verilir (Hatze, 1974). Bir başka deyişle, biyomekanik fiziksel hareketlerin sayısal veriye dönüştürülerek incelendiği bir disiplin olarak ifade edilebilir. Ana unsuru hareket olması nedeniyle ülkemizde terim olarak hareket analizi, biyomekanik ile sıklıkla anlamdaş olarak kullanılır (Dönmez vd., 2014).

Yürüme bir noktadan bir noktaya doğru vücudu taşımak maksadı ile gövdenin ilettilmesidir. Uzun vadede yorulmadan yürüyüşü gerçekleştirebilmek için beyin, kaslar, eklemler, kemikler, omurilik, sinirler birlikte çalışmalı, eklem hareketlerinin kasılma zamanı ve gücü yeterli olmak zorundadır. Normal yürüme karmaşıklığı ve

gözle değerlendirmenin güç oluşu bilim insanlarını güvenilir inceleme yöntemleri geliştirmeye zorlamıştır. Modern yürüme analizi: Yürümenin sayısal verilere dönüştürülüp, bu verilerin değerlendirilmesi ve yorumlanması olarak tanımlanmaktadır. Deneyimli doktorlar tarafından yürüme gözle değerlendirilebilse bile bir insan saniyede sadece 12–14 adet görüntüyü algılayabileceğinden yürüme esnasında milisaniyelerle gerçekleşen hareketler tam olarak değerlendirilememektedir. Bunların yanında yürüme hareketi sadece eklem hareketlerinden oluşmayıp, gözle anlaşılamayacak kas aktivitelerini, kuvvet ve moment değerlerini de içermektedir (Özaras ve Yalçın, 2001). Doğru tanıyı koyabilmek ve başarılı bir tedavi gerçekleştirebilmek için normal yürüme argümanları bilinmeli ve anormal olandan ayırt edilmelidir. Bu farkları ayırt edebilmek için yürümenin tüm bileşenlerini sayısal verilere dönüştürüp, kaydedecek, klinik muayene ile kıyaslamaya ve tedavi sonrası veya zaman içinde oluşan değişiklikleri karşılaştırıp değerlendirmeye olanak sağlayacak sistemler gereklidir. Günümüzde modern analizler, hareket analiz laboratuvarlarında gerçekleştirilmektedir (Kala ve Bol, 2014).

Yürüme analizi sonucunda elde edilen veriler, görsel yürüme analizinden farklı olarak, sayısal, gözlemciden bağımsız, yansız bilgiler verir. Bu veriler kayıt altına alınabilir, aynı hasta için değişik zamanlarda alınan veriler ile karşılaştırılabilir. Yürüme analizi sonuçları tanı koymak, tedavi planlaması ve değerlendirilmesi için; ortez ve protez tasarlama gibi alanlarda yardımcı bir araç olarak kullanılmaktadır (Tönük, 2014).

2.4.1. Yürüme analizinin önemi

Yürüme analizi bize yürüme bozukluğunun teşhis ve tedavi yönteminin belirlenmesinde birçok fayda sağlar. Bu bölümde yürüme analizinin sağladığı bu faydalardan bahsedilecektir. Patolojik yürümeyi belirlemede objektif veriler sağlar, tedavi yönteminin seçimine yardımcı olur. Tedavi öncesi ve sonrasındaki sayısal verilerin karşılaştırılmasını sağlar. Yürümede hangi kasların hangi yürüme fazında patolojik kasıldığının tespitini sağlar, böylece hangi kas grubuna tedavi ya da cerrahi müdahale yapılacağına karar verilmesini sağlar. Eklem fonksiyonlarında patolojik bulgular varsa bunun tespitini sağlar (Zorer, 2001). Ne kadar deneyimli olursanız olun gözlemsel analizde çeşitli faktörler yanılma riskini arttırır, gözlemsel analizden farklı olarak yürüme analizinde veriler sayısal bilgi içerdiği için gözlemciden bağımsız sonuçlar verir. Verilerin kayıt altına alınabilir olması istendiği zaman verilere

ulaşılabilmesini sağlar. Ayrıca ortez ve protez tasarlamada yardımcı bir araç olarak kullanılmaktadır (Tönük, 2014).

2.4.2. Yürüme analizinde kullanılan donanımlar

Yürüme analiz laboratuvarlarında genellikle kullanılan donanımları şöyledir. Tepki kuvvetlerini ölçen bir kuvvet platformları ve yürüme bantları; vücuda yerleştirilen işaretleri hareket sırasında izleyen kameralar; bacağa ait bileşenlerin hızlanması ve yavaşlaması anında ivmeyi ölçen ve kaydeden ivmeölçerler; kaslardaki elektriksel etkinliği ölçen EMG; bütün bu sayısal verilerin analiz edildiği ve saklandığı bilgisayar sistemlerinden oluşur (Dönmez vd., 2014).

2.4.3. Yürüme analizinde kullanılan yöntemler

Gözleme Dayalı Analiz: Yürüyüşün sadece göz ile incelenmesidir.

Video Analiz: Gözlemin kaydedilerek daha sonra izlenmesidir.

Kinematik Analiz: Hareketi oluşturan kuvvetleri dikkate almadan yalnızca hareketin incelenmesidir.

Dinamik Elektromiyografi: Dinamik EMG yürüme esnasında hangi kasın yürümenin hangi evresinde kasıldığını belirler.

Kinetik Analiz: Hareketi oluşturan kuvvetlerin incelenmesidir (Bozer, 2007).

2.4.3.1. Gözleme dayalı analiz

Yürüme analizinde bilgisayarlı yürüme analizi temel standarttır, gözleme dayalı analiz ise sadece klinik bir değerlendirmedir. Diğer yürüme analiz yöntemlerine göre daha subjektif veriler sunar. Yürüme esnasında farklı eklemlerde eş zamanlı ve üç boyutlu hareketlerden oluşur aynı zamanda insan gözüyle tespit edilmesi çok zordur. Bu nedenle gözleme dayalı analizin başarılı ve güvenilir olması gözlemi yapan hekimin deneyimlerine bağlıdır (Gage vd., 2009). Yürüme belli bir sırada, öncelikle ön profilden sonrasında ise yan taraflardan gözlenmelidir. Yürüme parkurunun uzunluğu 8–10 m, parkurun genişliği ise en az 3 m olmalıdır (Yalçın S. 2001). Gözleme dayalı analizin kısıtlılıkları; verilerin subjektif olması, sayısal veriler olmadığından verilerin saklanamaması ve çok sayıda vücut bileşenini birlikte hareket halindeyken incelemenin

güçlüğüdür. Bu gibi durumlarda video analiz kullanılabilir (Yalçın, 2001; Yavuzer, 2014).

2.4.3.2. Video analiz

Gözleme dayalı analiz, kameralar yardımı ile yürüyüş kayıt altına alınarak daha gelişmiş bir analiz haline gelebilir. Bu analiz çeşidine de video analiz denir. Video analiz, farklı açılardan çekilen kayıtların tekrar izlenmesi sayesinde, gözleme dayalı analize göre daha avantajlıdır. Video analiz özel olarak ayrılmış bir alanda yapılmalıdır. Kayıtlar eş zamanlı olarak önden, arkadan ve her iki yandan alınmalıdır. Kayıtlar sırasında hastanın günlük yürüme şeklinde ve hızında yürümesi önemlidir. Genellikle kayıtlar 1–2 dakika süreyle alınır; bazı durumlarda 3 dakika üzerinde çekim yapılabilir. Bu kayıtların bilgisayar ortamında saklanması gereklidir. Bu şekilde her muayenede yeni kayıtlarla beraber eski kayıtlar da izlenebilmeli ve ilk değerlendirme ile yeni yapılan değerlendirme arasındaki değişiklikler kaydedilmelidir (Miller, 2005). Şekil 2.30. da bir video analiz ortamı verilmiştir (Full Balance, 2016).

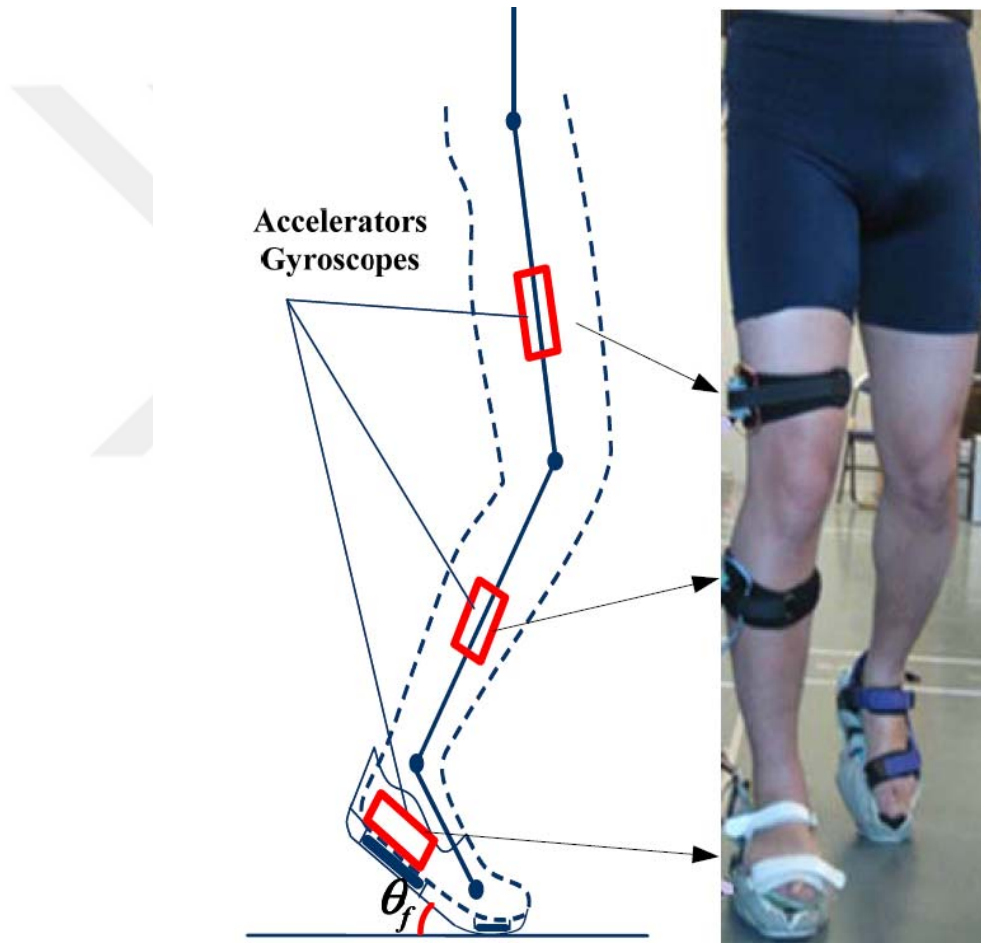


Şekil 2.30. Video analiz

2.4.3.3. Kinematik analiz

Hareketi meydana getiren kuvvetleri dikkate almaksızın sadece hareketin incelendiği analiz türüdür. Analiz ile vücudun hareketi incelenir. Kinematik analiz sırasında gövde, pelvis, kalça, diz ve ayak bileği eklemlerinin üç düzlemde (sagittal,

koronal ve transvers) pozisyonu, hızı, eklem açıları ve ivme değerleri ölçülerek elde edilen sayısal veriler kaydedilir. Bu veriler yürüme hızına bağlı olarak değişir (Whittle, 2007). Hareketin üç boyutlu değerlendirilmesi ve kayıt sistemlerinde verilerin kayıt altına alınabilmesi için en az beş adet kayıt cihazı kullanımı önerilir. Yazılımlar, hasta üzerinde bulunan işaret cihazlarından gelen verilerin yer değişimi ile eklem açılarında ki değişiklikler hesaplanır. Kinematik veriler, kameralar, kayıt ediciler, üzerinde sensörler bulunan yürüyüş parkurları ve elektrogonyometreler kullanılarak kayıt altına alınabilir (Yavuzer, 2014). Şekil 2.31. de kinematik analiz uygulama örneği verilmiştir (Tao vd., 2012).



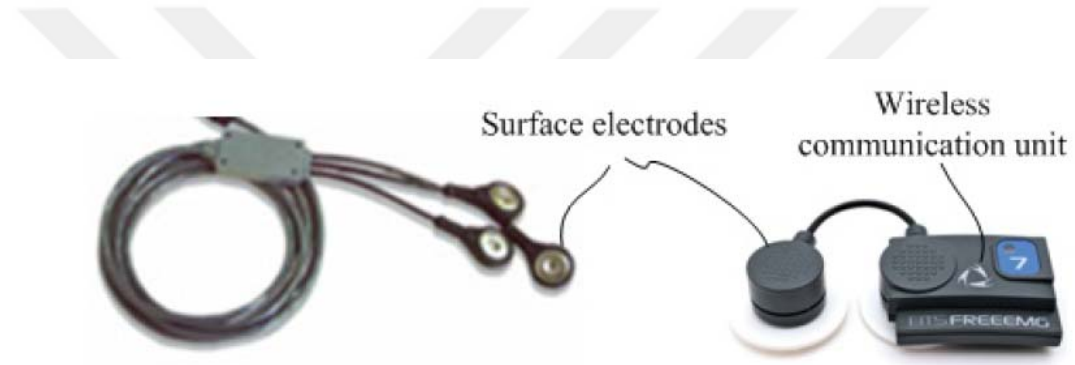
Şekil 2.31. Kinematik analiz - Accelerators Gyroscopes (Hızlandırıcı Jiroskoplara)

2.4.3.4. Dinamik Elektromiyografi (EMG)

Dinamik EMG yürüyüş esnasında kaslarda oluşan aktivitelerin elektrotlar yardımıyla ölçülüp kaydedilmesine denir. Dinamik EMG, kasların ne zaman kasıldığı, kasılma süre ve kasılma şiddeti bilgilerine ulaşmamızı sağlar. Kaydedilen sinyaller

kablolar ya da Radyo Frekans (RF) sistemi ile bir bilgisayara aktarılır, filtreleme işlemlerinden sonra veriler bilgisayar sistemlerine kayıt edilir. EMG verilerini kaydetmenin önemi: Kas aktivitelerinin, yürüme döngüsünün hangi anında oluştuğunu göstermesidir. EMG, kas aktivitesinin gerçekleştiği zamana dair objektif bilgi vermekle birlikte, Kinematik Analiz olmadan patolojik yürümeyi sağlıklı yürümeden ayırt etme olanağı sağlamaz (Whittle, 1996; Yavuzer, 2014). Dinamik EMG de Yüzeysel ve İğne Elektrotlar olmak üzere iki çeşit elektrot kullanılır. Yüzeysel Elektrotlar sayesinde kas aktivitesi hakkında bilgi edinilirken iğne elektrotlar ise kasa ait aktiviteyi değerlendirmede kullanılmaktadır (Davis, 1988).

Şekil 2.32. de kablosuz veri gönderen bir EMG sistemi görülmektedir (Tao vd., 2012).



Şekil 2.32. Dinamik EMG (Surface electrodes= Yüzey elektrotları, Wireless communication unit= kablosuz haberleşme ünitesi)

2.4.3.5. Kinetik analiz

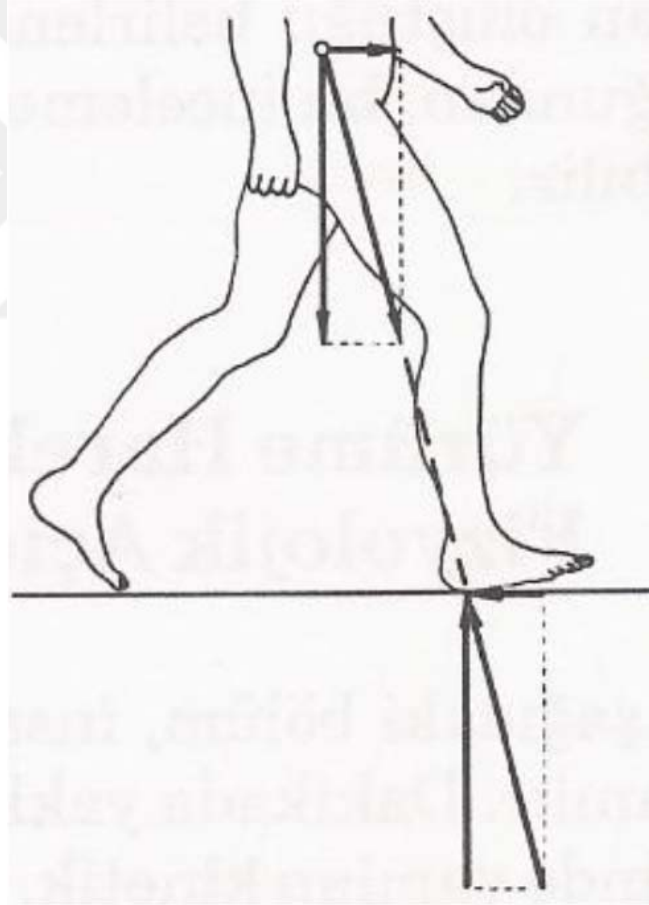
İnsan yürüme hareketinin incelenmesinde bulunan kuvvetler: Dış kuvvetler ve iç kuvvetler olmak üzere iki çeşittir.

2.4.3.5.1. Dış kuvvetler

Yer çekiminin neden olduğu kuvvetlerdir. Kinetik analiz ile çeşitli yürüyüş fazlarında meydana gelen ve bacak üzerinde etkisi gözlenen dış kuvvetlerin şiddeti ve yönü ölçülebilmektedir. Diğer taraftan kinematik analiz yardımı ile de eklemlerin konumları ve duruşları belirlenebilir. İki analiz yöntemi de kullanılmak şartı ile dış kuvvetlerin çeşitli eklemler üzerine yaptıkları etkiler hesaplanabilmektedir (Milli Eğitim Bakanlığı 2, 2011).

2.4.3.5.2. İç kuvvetler

Kasların kasılmaları sonucu meydana gelen kuvvetlerdir. İç kuvvetlerin sayısal olarak ölçülmesini sağlayacak uygun bir yöntem bugüne kadar geliştirilememiştir. Yer çekimi, ayağın yere ilk değmesinden hemen sonra diz ekleminin bükülmesine etki etmektedir (dış kuvvet). Ekleme karşı bir kuvvet etki etmese de bu durum meydana gelmektedir. Karşı kuvvet, üst bacağın ön tarafında yer alan bacağın uzanma ve bükülme hareketlerini birlikte yapan kas grubu (quadriceps) tarafından oluşturulur (iç kuvvet). Şekil 2.33. de dış kuvvet ve iç kuvvet gösterilmiştir (Milli Eğitim Bakanlığı 2, 2011).



Şekil 2.33. Dış ve iç kuvvetler

Kinetik analiz yer tepkimesi kuvvetleri, eklem momentleri, eklem güçleri gibi hareketi oluşturan kuvvetlerin analiz edilmesidir. Kinetik analiz yönteminde ölçülebilen tek veri yer tepkimesi kuvveti vektörüdür. Bu vektör ayağın yere uyguladığı toplam

kuvveti ölçebilen basınca duyarlı plakalarla ölçülür. Hazırlanan bu platformun doğal yürüyüşü bozmaması gerekir. Böylece ayak bileği, diz ve kalça eklemine etki eden momentler ve eklemlerde meydana gelen kuvvetler hesaplanabilir. Sonrasında bilgisayar ortamında kinematik veriler ile bir araya getirilir (Milli Eğitim Bakanlığı 2, 2011).

Yük taşıma yeteneği, yer tepkime kuvvetleri açısından kuvvet plakalarıyla güvenilir bir şekilde ölçülebilir (Yavuzer, 2009).

2.4.4. Yürüme Analizinin kısıtları

Yürüme Analizi'nin kısıtları ve gereksinimleri incelendiğinde bunların, laboratuvarla kısa yürüme mesafesinde kameralar ve elektrotlar ile yapılan yürümenin doğal yürümeden farklılığı, çok fazla emek, çok fazla maliyet gerektirmesi ve eğitimli bir ekip ihtiyacının olmasıdır (Delisa ve Joel, 1998). Verilerin toplanması, analiz edilip yorumlanması sırasında son derece dikkatli olunmalıdır. Potansiyel hata kaynaklarının tespiti yapılmalıdır. Potansiyel hata kaynakları genelde, elektrot türleri, büyüklüğü ve vücuda yerleşimi; yaş, vücut yapısı, katılımcının stresinin verileri üzerine olan etkisi; sistem kaynaklı hatalar ve yanlış kalibrasyondan kaynaklı hatalar ve değerlendirmelerin yanlış bir şekilde yapılmasıdır. Bunların çoğu dikkatsizlik ya da yeterli olmayan eğitimden kaynaklanmaktadır (Gök vd., 2002; Yavuzer vd., 2008).

Farklı yürüme analiz laboratuvarları arasında kabul görmüş standartlar yoktur. Yürümenin kinetik, kinematik ve motor özelliklerin karmaşık yapısıyla ilgili daha iyi eğitim verilmeli; standart bir terminoloji oluşturulmalıdır. Bu çözümler verilerin daha kolay anlaşılmasını sağlayacaktır. Farklı yürüme bozukluklarının değerlendirilmesi ve tedavisi için kullanılacak yürüme analiz yöntemlerinin seçimi ve uygulanması hakkında kaynaklara ihtiyaç vardır (Yavuzer, 2009).

3. MATERYAL VE YÖNTEM

Bu bölümde sistemde kullanılan donanımlar, yazılımlar, sistemin Nesnelerin İnterneti teknolojisine göre haberleşme esasları, sistemin çalışması ile ilgili şemalar ve sistemin katılımcılar üzerinde kullanılma adımları anlatılmıştır.

3.1. Kullanılan Donanımlar

Bu bölümde projemizde kullandığımız elektronik kartlar, pcb kartlar, sensörler, iletişim için kullanılan algılayıcılar ve bazı materyallerin üretiminde kullanılan 3D yazıcı hakkında bilgi verilmiştir.

3.1.1. Arduino

Arduino çevresiyle kolayca etkileşime girebilen sistemler tasarlamamızı sağlayan açık kaynak kodlu geliştirme platformudur. Arduino'nun programlama dili C++ ile çok benzerlik gösterir. Yazılan programlar Arduino IDE aracılığıyla Arduino kartlarına kolayca yüklenebilir. Arduino dijital ve analog girişleri sayesinde analog ve dijital verileri harici bir dönüştürücü kullanmadan işleyebilme olanağına sahip kartlarıdır (Diyot.net, 2015). Arduino kartlarının birçok çeşidi vardır:

- Arduino Uno
- Arduino Mega 2560
- Arduino Lilypad
- Arduino Mega ADK
- Arduino Ethernet
- Arduino Bluetooth
- Arduino Mini ve Mini Pro
- Arduino Nano
- Arduino Fio (Robotik Sistem 1, 2018).

3.1.1.1. Arduino kartlarının genel özellikleri ve aralarındaki farklar

Birçok çeşidi bulunan Arduino kartlarının genel olarak benzer bileşenler bütün kartlarda yer almaktadır. Fakat kartların I/O sayılarında, kullanılan denetleyici

çeşitlerinde ve çalışma gerilimleri ile dahili modüllerin sayısında farklılıklar bulunmaktadır. Çizelge 3.1. de Arduino kartlarının özelliklerinden bahsedilmektedir.

Arduino'nun temel bileşenlerinden bahsedecek olursak, bunlar:

- Mikroişlemci: Kartın beynidir.
- Giriş Voltajı: Kart için önerilen voltaj aralığıdır.
- Sistem Voltajı: Kartta bulunan mikroişlemcinin çalıştığı voltajdır.
- UART: Arduino'nun desteklediği, birbirinden ayrı seri bağlantı çizgileri sayıdır.
- PWM: Sinyal üretebilme kapasitesine sahip olan dijital I/O'ların sayıdır. PWM sinyalleri analog çıkışlar gibidir.
- Dijital I/O: Arduino kartında bulunan dijital I/O'ların sayıdır. Bunların her biri giriş ya da çıkış olarak bazıları ise PWM olabilecek şekilde tasarlanmıştır
- Analog Giriş: Arduinoda bulunan analog cihazların bağlanabildiği kullanılabilir analog girişlerin toplam sayıdır. Analog pinler "A" harfi isimlendirilir. A0,A1,,,
- Saat Hızı: Mikroişlemcinin hızına bağlı frekans aralığıdır.
- Bootloader: Bootloader ATmega'nın içinde seri port yardımı ile donanımların programlanabilmesi için yükleme yapılmasını sağlar.
- Programlama Ara yüzü: Arduino kartını programlamak için bilgisayar ile bağlantı kurmayı sağlayan ara yüzdür (Robotik Sistem 1, 2018).

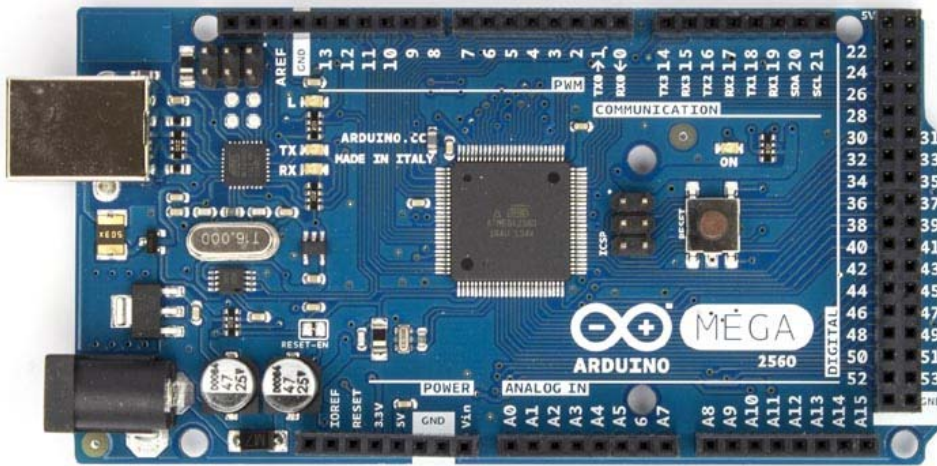
Çizelge 3.1. Arduino kartlarının özellikleri

Arduino	Çalışma Voltajı	Giriş Voltajı	UART	PWM	Dijital Pin	Analog Pin
Uno	5v	7-12v	1	6	14	6
Fio	3.3v	3.3-12v	1	6	14	8
Nano	5v	7-12v	1	6	14	8
Leonardo	5v	3.3-5v	1	7	25	12
Mini	5v	7-9v	1	6	14	8
Mega 2560	5v	7-18v	4	14	54	16

3.1.1.2. Arduino Mega 2560

ATmega2560 micro işlemci tabanlı bir geliştirme kartıdır. 54 adet dijital giriş/çıkış pini bulunmaktadır. Bu pinlerden 14 tanesi PWM çıkışı olarak kullanılabilir. 16 analog giriş, 4 UART (serial port), 16 MHz Kristal Osilatör, USB bağlantısı, adaptör giriş, ICSP çıkış ve bir reset butonuna sahip geliştirme kartıdır

(Alpat, 2012). Projemizde bu kartı seçme nedimiz piyasadaki en fazla analog girişe sahip kartlardan biri olmasıdır.



Şekil 3.1. Arduino Mega 2560

3.1.1.2.1 Arduino Mega 2560 özellikleri

Çizelge 3.2. de kullanılan Arduino kartının özellikleri verilmiştir (Alpat, 2012).

Çizelge 3.2. Arduino Mega 2560 genel özellikleri

Mikrodenetleyici	ATmega2560
Çalışma Gerilimi	5V
Besleme Voltajı (Önerilen)	7-12V
Besleme Voltajı (Limit)	6-20V
Dijital I/O Pinleri	54 (14ü PWM çıkışı)
Analog Giriş Pinleri	16
I/O Pinlerinin Akımı	40 mA
3.3V Pini Akımı	50 mA
Flash Bellek	256 KB (8kB'ını bootloader kullanıyor)
SRAM	8 KB
EEPROM	4 KB
Clock Frekansı	16 MHz

3.1.1.2.2. Input / output

54 adet dijital I/O pinlerinin tamamı pinMode(), digitalWrite(), ve digitalRead() gibi fonksiyonlar kullanılarak giriş veya çıkış olarak tanımlanabilir. 5 volt gerilim ile çalışan I/O maksimum 40 mA giriş veya çıkış sağlar. Pinlerde 20-50

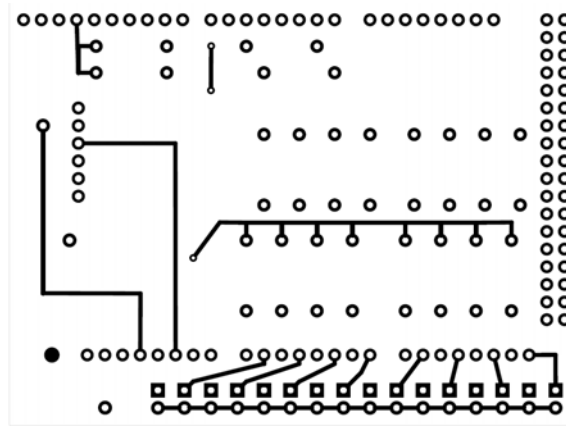
kohm pull-up dirençleri bulunmaktadır. Bu pinlerin haricinde özel göreve sahip pinler de mevcuttur (Robotik Sistem 2, 2018).

- Seri: 0 (RX) ve 1 (TX); Serial 1: 19 (RX) ve 18 (TX); Serial 2: 17 (RX) ve 16 (TX); Serial 3: 15 (RX) ve 14 (TX). Seri veri almak (RX) ve seri veri göndermek (TX) için kullanılmaktadır.
- Harici Kesme: 2 (interrupt 0), 3 (interrupt 1), 21 (interrupt 2), 20 (interrupt 3), 19 (interrupt 4) ve 18 (interrupt 5).
- PWM: 0 to 13. 8-bit PWM çıkış verir. analogWrite() fonksiyonu kullanılır.
- SPI: 50 (MISO), 51 (MOSI), 52 (SCK), 53 (SS).
- LED: 13. 13 nolu pinde bir LED bulunmaktadır. Çıkış High edildiğinde LED yanmaktadır, LOW edildiğinde ise led sönmektedir.
- TWI: 20 (SDA) and 21 (SCL) (Robotik Sistem 2, 2018).

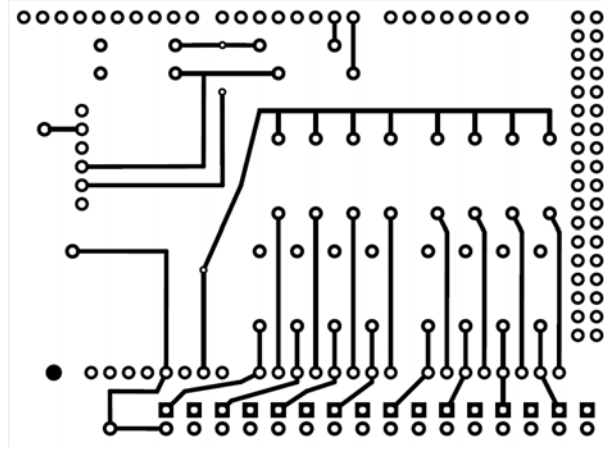
Mega 2560'ın her biri 10 bit çözünürlüğe sahip 16 adet analog girişi bulunmaktadır. Standart olarak 0-5V aralığında çalışmaktadır (Robotik Sistem 2, 2018).

3.1.2. Pcb kartı

Sensörlerin Arduino ile Arduino'nun Bluetooth Modülü ile bağlantısını sağlayan aynı zamanda gerilimin sensörlere iletilmesini sağlayan karttır. Kartın çizimi fritzing programı ile yapılmıştır. Bu program Arduino'ya uyumlu devre kartları tasarlamak için çok kullanışlı bir programdır. Kartımız Arduino'nun girişlerine uyumlu şekilde çizilmiştir. Şekil 3.2. de pcb kartımızın baskı öncesi alttan ve üstten görünüşü yer almaktadır.



a)



b)

Şekil 3.2. Baskı öncesi pcb kart katman görünümüleri a) üst katman b) alt katman

3.1.2.1. Pcb kartın özellikleri

Pcb kart tasarımında çizelge 3.3. deki özellikler kullanılmıştır.

Çizelge 3.3. Pcb kart özellikleri

Ölçü	61x82 mm
Katman sayısı	2
Malzeme	Fr-4
Kalınlık	1.6 mm
Yol kalınlığı	0.6 mm
Yollar arası boşluk	0.6 mm
Çember kalınlığı	0.3 mm

3.1.2.2. Pcb kartın son hali

Pcb kart arduino ile diğer donanımlar arasındaki bağlantıyı sağlamakta aynı zamanda sensörlere de enerji Arduino'nun 5v pininden besleme alınarak pcb kart aracılığı ile iletilmektedir. Şekil 3.3. de pcb kart ve üzerine yerleştirilmiş bluetooth modülü; Arduino Mega 2560 üzerine takılmış halde görülmektedir.



Şekil 3.3. Pcb kart son hali

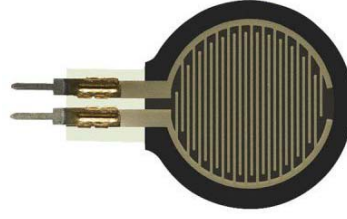
3.1.3. Kuvvet Sensörü

Aslen kuvvete duyarlı dirençlerdir. Bu dirençler aktif dairesel alana uygulanan kuvvetle orantılı olarak direnç değerlerinde düşüş gösterirler. Bu sayede kuvvet veya basıncın algılanabilmektedir. Dokunma kontrollü elektronik cihazlar için birkaç gramdan 10 kilograma kadar olan kuvvetlerin algılanabilmesi için optimize edilmiştir (Robotistan 1, 2018).

Ürün üzerinde yapılan testlerde, ürün üzerine kuvvet uygulanmadığında $1\text{ M}\Omega$, 10 kilogram gibi kuvvetler uygulandığında ise $100\text{k}\Omega$ direnç değerlerine ulaşmıştır. Uygulanan kuvvet karşılığında oluşan değerler yüksek tutarlılık göstermektedir. Tekrarlı sabit kuvvetler uygulandığında ürünün aynı değerleri ürettiği yapılan testlerde görülmüştür (Dirençnet 1, 2018). Şekil 3.4. de sistemde kullanılan kuvvet sensörü gösterilmektedir.

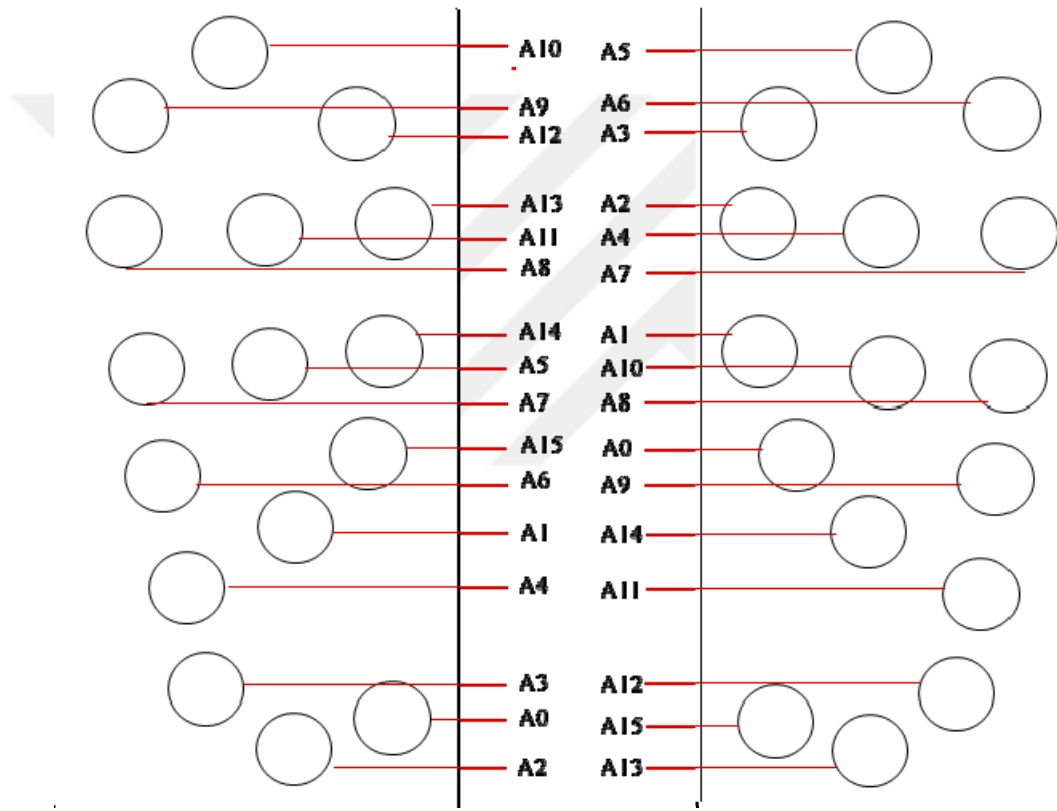
Ürün Boyutları:

- Pad Çapı: 1,83 cm
- Algılama Alanı Çapı: 1,27 cm
- Ağırlık: 0,25 gr
- Kalınlık: 0,51 mm dir (Dirençnet 1, 2018).



Şekil 3.4. Kuvvet sensörü

Şekil 3.5 de sensörlerin ayak tabanına yerleşimi ve sensörlerin Arduino Mega 2560 analog pinlerine bağlantısı gösterilmektedir.



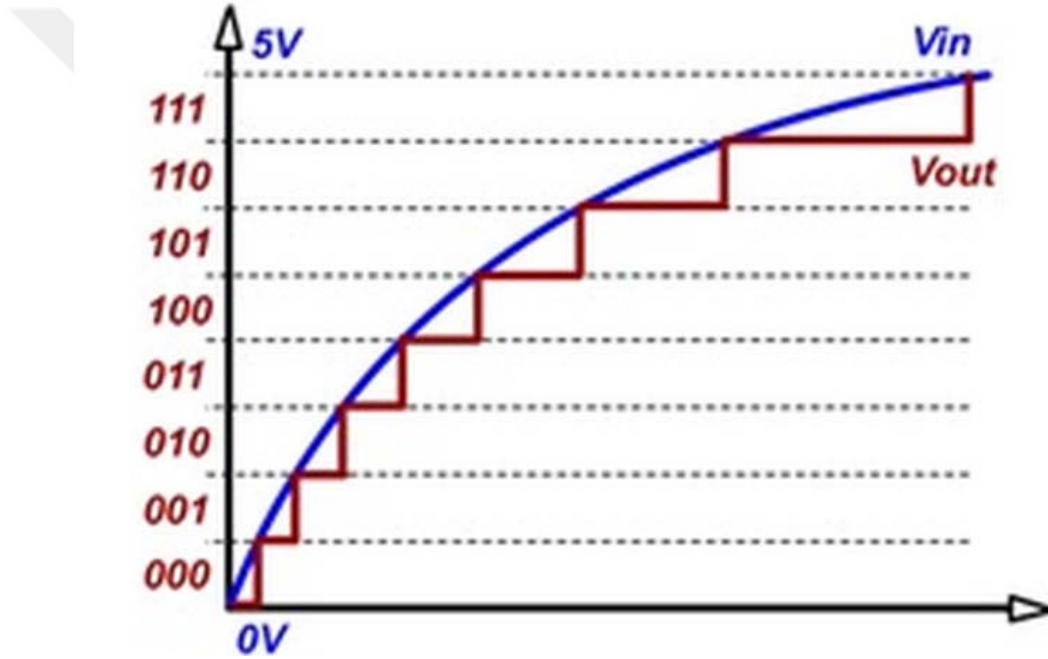
Şekil 3.5. Sensörlerin analog girişlere bağlantısı

3.1.3.1. Analog veri

Yaşadığımız dünyada her etkinin bir analog sinyal olarak karşılığı vardır. Bu değerleri ölçebilen sensörler de genellikle analog çıkış üretmektedir fakat Arduino geliştirme kartları analog sinyalleri işleyememektedir. Bu nedenler analog sinyallere

karşılık gelen dijital değerlerin bulunması gerekmektedir. Bu işleme ADC (Analog Digital Converter) denilmektedir.

Şekil 3.6. deki gibi 0 ve 5 volt arasında değeri değişen analog bir sinyalinizi olduğunu düşünelim. Bu analog sinyalin dijitale çevrilebilmesi için Arduino'da 10 bitlik bir saklayıcı yer almaktadır. Bu saklayıcı 0 volt giriş için 0, 5 volt giriş için ise 1023 değerini üretmektedir. Bu artış doğrusal bir artıştır yani girişteki 0,005 voltluk bir değişim saklayıcının değerini bir artırmaktadır. Eğer giriş sinyali 3,3 volt olacak olur ise okunacak değer yaklaşık olarak 675'tir. Kısacası ADC, 0 ve 5 volt arasındaki sinyali oranlayarak 0 ve 1023 arasında sayısal bir değer döndürmektedir (Turkcell, 2015).



Şekil 3.6. Analog sinyallerin değişimi

Sensörlerden gelen verileri 0 – 1023 arası dijital değil; 0 – 5 arası analog değerler olarak da kullanabiliriz. Bu dönüşüm denklem 3.1. de gösterilmiştir.

A: Analog veri

D: Dijital veri

$$A = D * (5 / 1023)$$

(3.1.)

3.1.4. HC-05 Bluetooth Modülü

HC-05 Bluetooth Modül Kartı kablosuz seri haberleşme için geliştirilmiş bir karttır. Breadboardlarda, Arduino ve çeşitli devrelerde rahatça kullanılabilmesi adına gerekli olan pinler devre kartı aracılığı ile kartın dışına aktarılmıştır. Bluetooth 2.0 teknolojisini desteklemekte olan bu kart, 2.4 GHz frekansında haberleşme olanak sağlayıp, yaklaşık olarak 10 metrelik bir haberleşme mesafesi vardır (Direncnet 2, 2018). Çoğu Bluetooth modülünden farklı olarak Master Mod'unu da desteklemektedir. Slave Modda Bluetooth Modülüne dışarıdan bir cihaz tarafından bağlantı yapılabilirken, modül ile başka bir Bluetooth modüle ilk bağlantı yapılamaz. Kısaca HC-06 ile başka bir cihaza bağlanamazken, HC-06 modülüne dışarıdan bir cihaz kolaylıkla bağlanabilir. Master moda ise elinizdeki modül yardımı ile dışarıdaki bir Bluetooth cihaza direk olarak ilk bağlantı yapılabilir. HC-05 hem master hem de slave modda çalışabileceği için HC-06 modülüne göre çok daha fazla tercih edilmektedir (muhendisbeyinler.net). Şekil 3.7. de HC-05 modülünün arkadan ve önden görüntüsü yer almaktadır (Robotistan 2, 2018).

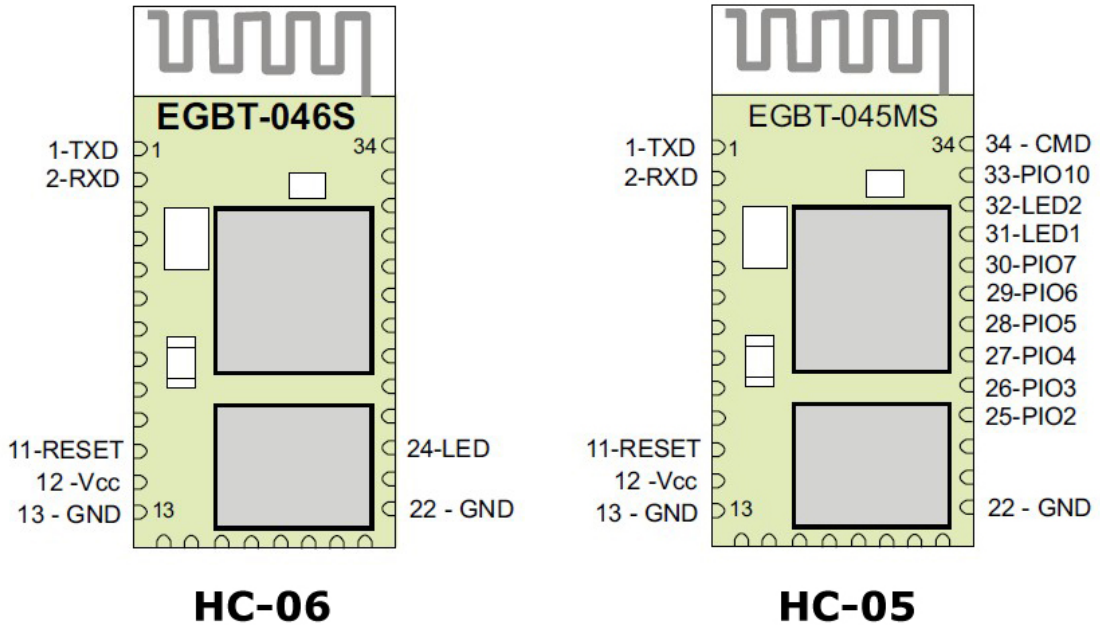


Şekil 3.7. HC – 05 Bluetooth modülü

3.1.4.1. Çalışma mantığı

Çalışma mantığı tamamen seri haberleşme esasına dayanmaktadır. Tx pini veri göndermek için, Rx pini ise veri almak için kullanılır. 3.3 V, 5 V gerilimlerde ve bu gerilimlerin arasında bir gerilimde veri transferi yapılabilir ama veri alış verişi 3.3V ile yapmaktadır. Çünkü 5 V gerilim verilirse, Tx pininden gönderilen veriyi mikro denetleyici algılar ama mikro denetleyicinin çıkış voltajı 5V olduğu için bluetooth modülüne zarar verebilir (Gül, 2015).

Şekil 3.8. de HC-05, HC-06 modüllerinin pin numaraları gösterilmektedir (İzgöl, 2016).



Şekil 3.8. HC – 05 ve HC – 06 pin yapıları

3.1.4.2 HC – 05 Bluetooth Modülü özellikleri

HC-05 genel özellikleri:

- Çalışma Gerilimi: 3.3V
- Bluetooth Protokolü: Bluetooth 2.0+EDR(Gelişmiş Veri Hızı)
- haberleşme frekansı: 2.4GHz
- Hassasiyet: ≤ -80 dBm
- Çıkış Gücü: $\leq +4$ dBm
- Asenkron Hız: 2.1 MBps/160 KBps
- Senkron Hız: 1 MBps/1 MBps
- Güvenlik: Kimlik Doğrulama ve Şifreleme
- Akım: 50 mA
- Boyutları: 43x16x7mm (Robotkutus 1, 2018).

3.1.4.3. AT komutları

Modülün konfigürasyonu için modülü Arduino'muza bağladığımızda ilk olarak modül adı, baud rate ve şifre değerlerini değiştirmemiz güvenlik açısından önemlidir. HC-05 bluetooth modülünün konfigürasyon ayarlarını yapabilmemiz için 5V gerilim

bağlantısı yapıldığı sırada modül üzerindeki butonu basılı tutmamız gerekmektedir. Eğer modül üzerindeki led 3'er saniyelik aralıklarla yanıp sönüyorsa, modül konfigürasyon moduna girmiş demektir. LED daha sık yanıp sönüyorsa ise bu modülün iletişim modunda olduğunu göstermektedir. İletişim modunda olan cihaz diğer cihazlar tarafından yapılan Bluetooth taramalarında listelenir (İzgöl, 2015).

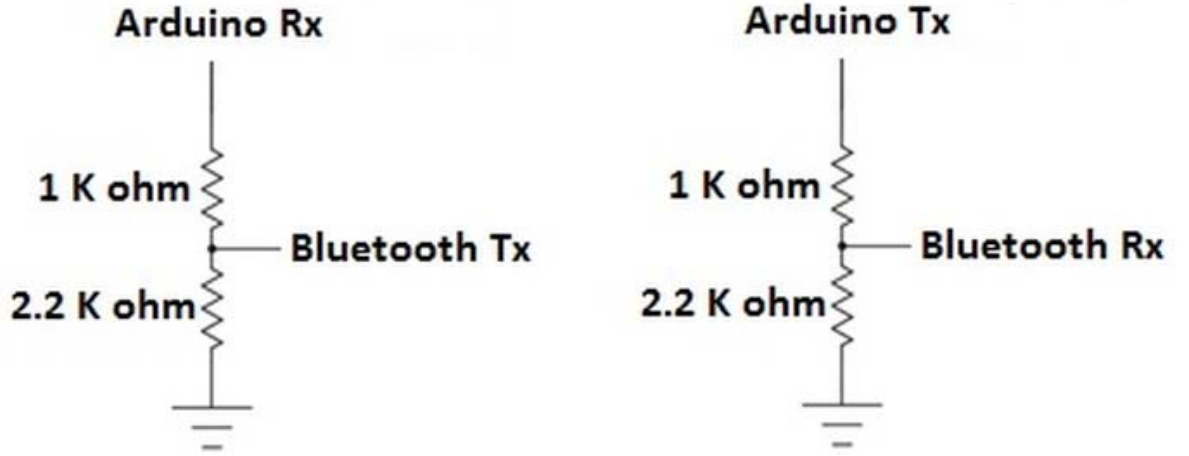
Konfigürasyonu yapabilmek için AT komutları bilinmelidir. Seri port ekranına "AT" yazıp gönderdiğimizde cevap olarak "OK" ifadesi geliyorsa modül AT komutlarını almaya hazır demektir. Bazı "AT" komutları şu şekildedir:

- AT+ROLE: ifadesi modülün hangi modda olduğunu gösterir.
- AT+NAME: Default ismi gösterir.
- AT+ADDR: Default adresi gösterir.
- AT+VERSION: Versiyonu Gösterir.
- AT+UART: Ayarlanmış olan Baud Rate' yi gösterir.
- AT+ROLE: Master ya da Slave olma durumunu gösterir(1=master/0=slave).
- AT+RESET: Ayarları resetler ve AT moddan çıkar.
- AT+ORGL: Fabrika ayarlarına geri döner.
- AT+PSWD: Default parolayı gösterir (Gül, 2016).

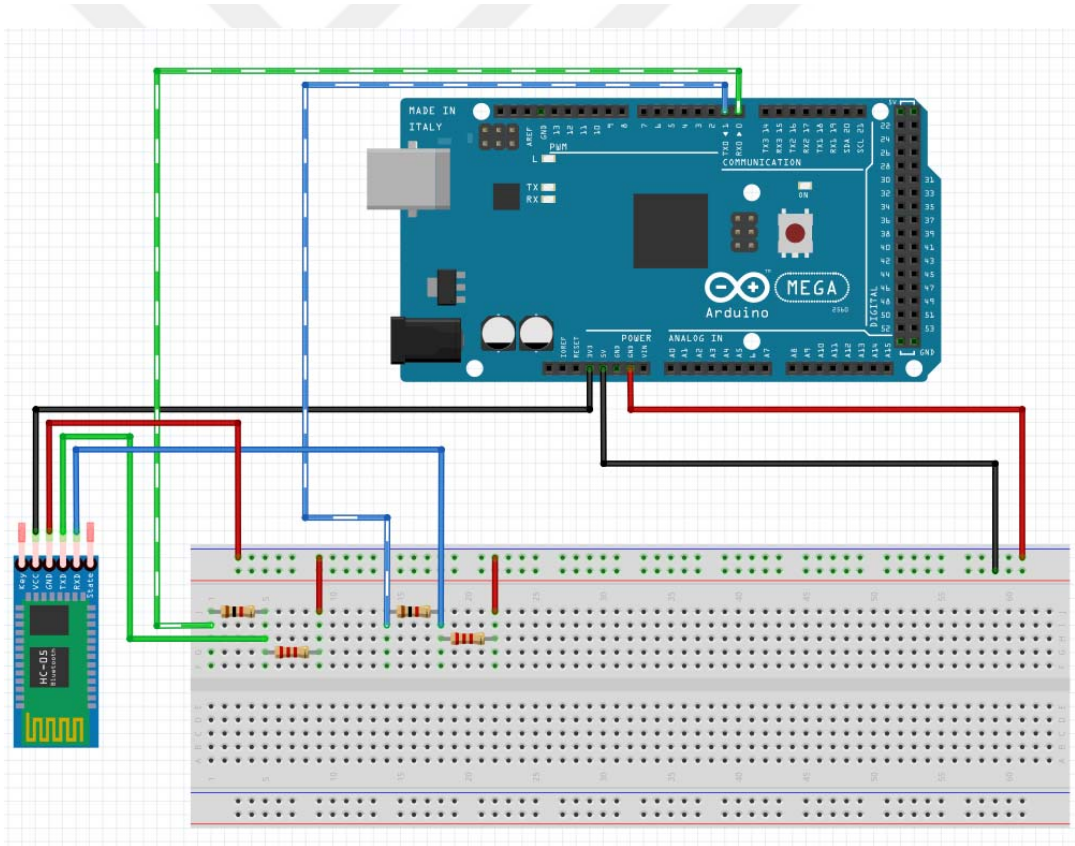
Baud rate hızı 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200, 460800, 921600, 1382400 değerleri seçilebilir fakat alıcı tarafta da aynı baud rate ayarlanmış olmalıdır.

3.1.4.4. Bluetooth Modülü'nün devreye bağlantısı

Projenin nesnelerin interneti yöntemi ile çalışması için önemli parçalardan bir tanesi de Bluetooth Modülüdür. Verilerin kablosuz iletilmesi, iletim hızı yani sensörlerin yazılımla kablosuz haberleşmesi bu modül sayesinde gerçekleşir. Şekil 3.9. ve 3.10. da Bluetooth Modülü'nün Arduino Mega 2560 ile bağlantıları gösterilmiştir.



Şekil 3.9. HC - 05 elektronik bağlantı şeması



Şekil 3.10. HC – 05 ve arduino bağlantısı

3.1.4.5. HC – 05 Modülünün bilgisayara tanıtımı

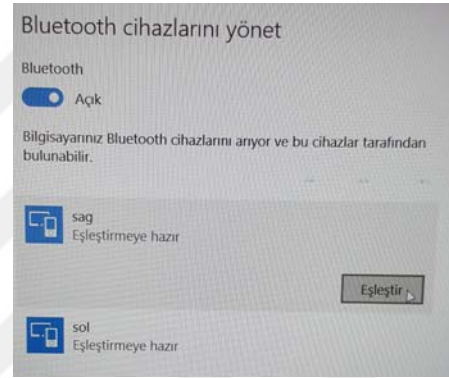
Bluetooth modüllerinin bilgisayar sistemine tanıtılması için çizelge 3.4. deki ayarlar kullanılmıştır. Şekil 3.11. deki adımlar sırasıyla izlenerek bağlantı kurulur.

Çizelge 3.4. Bluetooth bilgileri

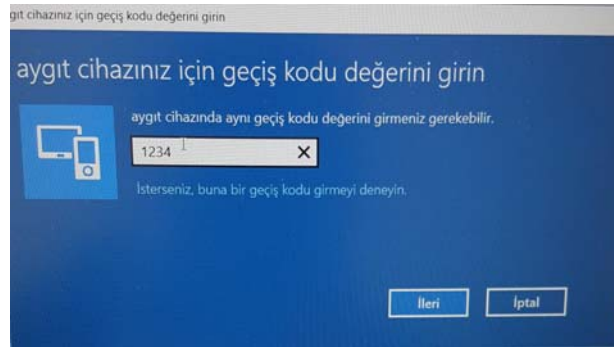
	Cihaz adı	Şifre	Com portu
Sağ ayak	Sag	1234	Bilgisayar tarafından atanacaktır. Bluetooth ayarlarından hangi com portu olduğu öğrenilmelidir.
Sol ayak	Sol	1234	



a)



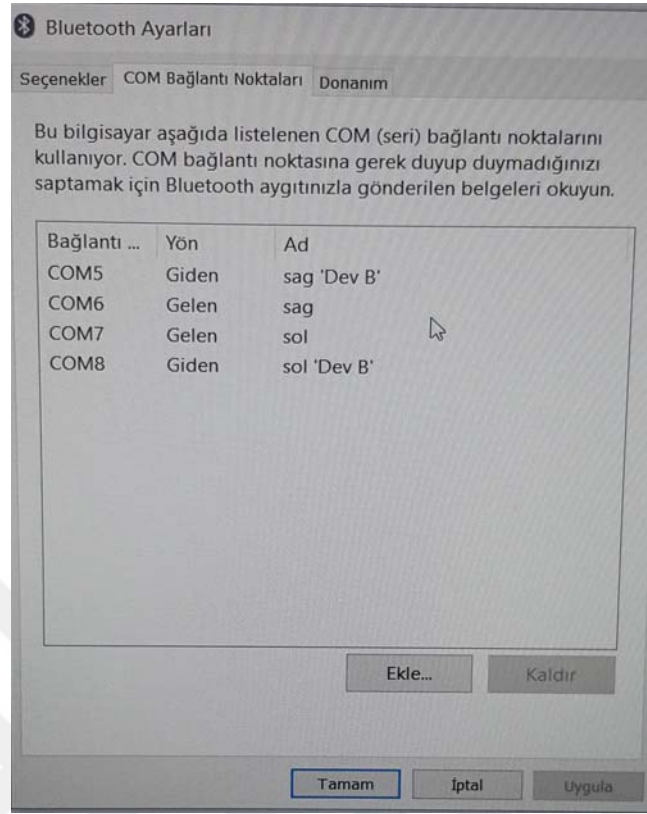
b)



c)

Şekil 3.11. a), b), c) sistem bluetooth bağlantısı

Cihazların hangi com portunu kullandığı bluetooth ayarları menüsünden görülebilir. Sistem yazılımında seri port bağlantısı kurulurken HC – 05 Modüllerinin bilgisayarın hangi com portuna bağlandığını bilmemiz gerekir. Şekil 3.12. de Bluetooth ayarları menüsünde Modüllerin kullandığı com portlar gösterilmektedir.



Şekil 3.12. Bluetooth ayarları

3.1.5. 3D yazıcı

3D yazıcılar birçok farklı ham maddeyi kullanarak üretim yapabilmektedir. Günümüzde 3D yazıcıların kullandığı en popüler yöntem FDM (Fused Deposition Modelling) ya da birleştirmeli yığma teknolojisidir. Bu yöntemde ısı verilerek şekillendirilebilen plastik türevi malzemeler (PLA, ABS) kullanılmaktadır. Bu teknoloji ile sarkıt tipi havada asılı duran nesnelerin üretilmesi oldukça sıkıntılı olabilmektedir. En alt katmanda basılan malzemenin az olması nedeni ile üzerine basılan diğer katmanları taşıyamaması ve yamulması söz konusu olabilir. 3D yazıcıların tamamı katmanları üst üste yığma prensibi ile çalışır (3Dprintertr, 2018).

Projenin baskı işlemleri Ultimaker Original + 3d yazıcı ile gerçekleştirildi. Çizelge 3.5 de kullandığımız yazıcının teknik özellikleri gösterilmektedir (3Dörtgen, 2018).

Çizelge 3.5. 3d yazıcı özellikleri

Basım Alanı	21 x 21 x 20,5 cm
Teknoloji	FDM
Katman Kalınlığı	20 mikron
Hammadde	PLA / ABS / CPE
Kafa Sayısı	1
Hammadde Çapı	2.85 mm
Nozzle Çapı	0.4 mm
İskelet	Ahşap
Ebatları	35 x 35 x 39 cm
Ağırlık	11,5 kg
Platform	Isıtmalı Cam
Güç Kaynağı	120W
Veri Girişi	SD kart / USB Kablo
Yazdırma Yazılımı	Cura

Şekil 3.13. de projede kullanılan Ultimaker Original + 3d yazıcı gösterilmektedir (3Dörtgen, 2018).



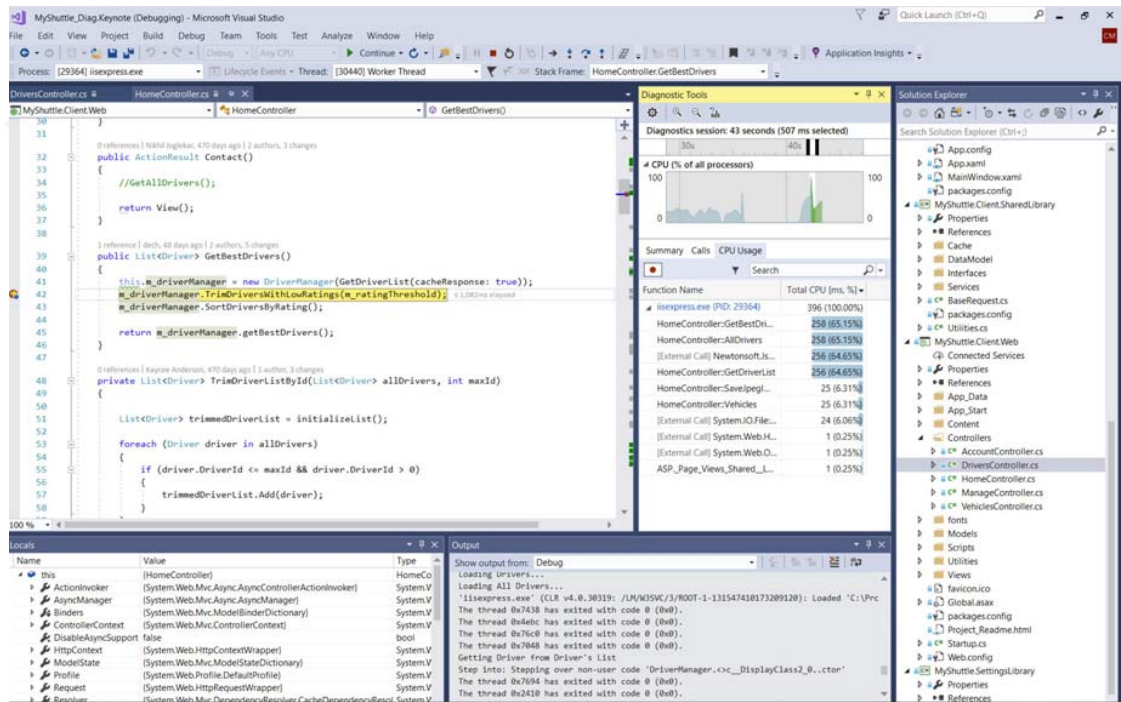
Şekil 3.13. Ultimaker Original + 3d yazıcı

3.2. Kullanılan Yazılımlar

Projemizde birçok yazılım kullanılmıştır. Bunlar Visual studio, c# programlama dili, arduino ide yazılımı, Autodesk 3ds max ve cura yazılımlarından bahsedilecektir.

3.2.1. Microsoft Visual studio

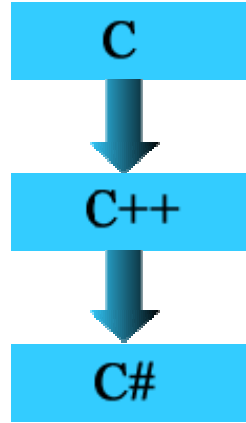
Microsoft firması tarafından geliştirilen bir yazılım geliştirme ortamıdır. Windows Form uygulamaları, web servisleri ve web uygulamalarıyla beraber grafiksel ara yüz ve konsol kullanıcı arabirimi uygulamaları geliştirmek için kullanılmaktadır. Visual Studio, farklı programlama dillerine destek sağlar (Tok, 2017). Entegre diller C#, VB.NET, C/C++, F#. Şekil 3.14. de Microsoft Visual Studio ortamı gösterilmektedir.



Şekil 3.14. Microsoft Visual Studio

3.2.2. C#

C#, 1990'ların sonlarında Microsoft firması tarafından ortaya çıkarılmış ve Microsoft' un. NET projesinin bir parçası olmuş c temelli bir programlama dilidir. C# baş üreticisi Anders Hejlsberg'dir (Tok, 2017). C#, C kökenli bir dildir. C# söz dizimini ve operatörlerini C'den almıştır. C#, C++ ile tanımlanan nesne modeli temel alınarak oluşturulmuş ve bu nesne modelini geliştirmiştir. Şekil 3.15. te c# soy ağacı gösterilmektedir.



Şekil 3.15. C# soy ağacı

Sharp 2009 yılında yayınladığı çalışmasında C# programlama dilinin diğer programlama dillerinden ayıran özellikleri sıralamıştır:

- Kolay öğrenilebilen bir programlama dilidir.
- Yüksek verimle çalışır.
- Nesne yönelimli programlama teknolojisini destekler.
- Güç ve kolaylık arasında denge sağlar. Güçlü ve hızlı uygulamalar geliştirmek için uygundur.
- XML’i destekler (Sharp, 2009).

C# programlama dili her bilgisayar platformunda çalışacak yazılımlar geliştirmek için ideal bir dildir ve C# ile yapılabileceklerden bazıları şunlardır:

- C#, Asp.Net uygulamaları geliştirme.
- Windows ile uyumlu, güçlü, hızlı ve güvenli kodlar yazma.
- Web uygulamaları geliştirme.
- Web üzerinden hizmet verebilen programlar yazma.
- Mobil uygulama geliştirme.
- .Net desteği olan diller için hızlı ve esnek DLL (Dynamic Link Library) bileşenleri yazma.
- WPF uygulamaları geliştirme.
- Dokunmatik donanım kontrolü için kolay kütüphane barındırma ve yazmadır (Algan, 2009).

3.2.2.1. Kullanılan formlar

Projemizde kullanılan bazı formlar ve kodlar bu bölümde açıklanacaktır.

3.2.2.1.1. Kayıt formu

Hasta kayıt formu: hasta bilgileri veri tabanına bu form aracılığı ile kayıt edilir. Veri tabanı olarak Microsoft Access veri tabanı kullanılmıştır. Şekil 3.16 da hasta kayıt formu gösterilmektedir ayrıca şekil 3.17 deki kod parçacığı ile de veri tabanındaki kayıtların listelenmesini sağlayan kod gösterilmektedir.

	id	tc	ad_soyad	yas	sehir
▶	1			33	konya
	2			32	konya
	3			20	ANKARA
	4			18	TRABZON
	5			10	İSTANBUL
*					

Şekil 3.16. Hasta kayıt formu

```
void griddoldur()
{
    baglanti = new OleDbConnection("Provider=Microsoft.ACE.Oledb.12.0;Data Source=analiz.accdb");
    da = new OleDbDataAdapter("SElect *from hasta", baglanti);
    ds = new DataSet();
    baglanti.Open();
    da.Fill(ds, "hasta");
    dataGridView1.DataSource = ds.Tables["hasta"];
    baglanti.Close();
}
```

Şekil 3.17. Veri tabanı bağlantı kodları

3.2.2.1.2. Seri port bağlantı formu

Bluetooth Modülü'nden algılayıcı modüle gelen veriler seri porttan alınır yani bluetooth teknolojisi seri portu kullanarak haberleşir. Visual studio da seri port haberleşmesi için serialport bileşeni kullanılır. Seri port haberleşme yapılırken dikkat edilmesi gereken gönderici ve alıcı tarafta baud rate değerlerinin aynı olmasıdır. Bu nedenle arduino tarafında ve c# tarafında baud rate değeri aynı değer, 9600 seçilmiştir. Şekil 3.19. daki kod parçacığı seri portu kontrol edip bağlantı kurmayı sağlar. Şekil 3.18. de seri port bağlantı form görüntüsü şekil 3.19. da seri port bağlantı için kod parçacığı gösterilmektedir.

Şekil 3.18. Seriport bağlantı formu

```

if (serialPort1.IsOpen == false)
{
    if (comboBox1.Text == "")
        return;
    if (comboBox3.Text == "")
        return;
    serialPort1.PortName = comboBox1.Text;
    serialPort2.PortName = comboBox3.Text;
    serialPort1.BaudRate = Convert.ToInt32(comboBox2.Text);
    serialPort2.BaudRate = Convert.ToInt32(comboBox2.Text);
    try
    {
        serialPort1.Open(); //Haberleşme için port açılıyor
        serialPort2.Open();
        label3.ForeColor = Color.Green;
        label3.Text = "Bağlantı Açık";
    }
    catch (Exception hata)
    {
        MessageBox.Show("Hata:" + hata.Message);
    }
}
else
{
    label3.Text = "Bağlantı kurulu !!!";
}

```

Şekil 3.19. Seri port bağlantı kodu

üretilmiştir. Şekil 3.22. deki kod parçasığında multi threading ile seri port okuma gösterilmiştir.

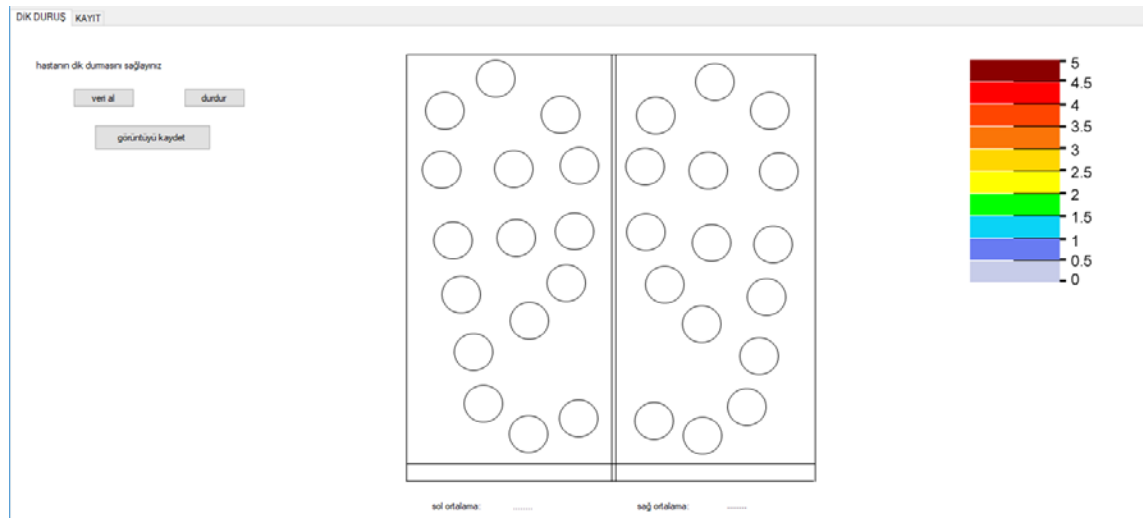
```
void sensport_DataReceived(object sender, SerialDataReceivedEventArgs e)
{
    sl++;
    Thread paralel1 = new Thread(new ThreadStart(soloku));
    paralel1.Start();
    paralel1.Join();
}

void sensport_DataReceived2(object sender, SerialDataReceivedEventArgs e)
{
    sg++;
    Thread paralel2 = new Thread(new ThreadStart(sagoku));
    paralel2.Start();
    paralel2.Join();
}
```

Şekil 3.22. Paralel programlama

3.2.2.1.4. Dik duruş veri alma formu

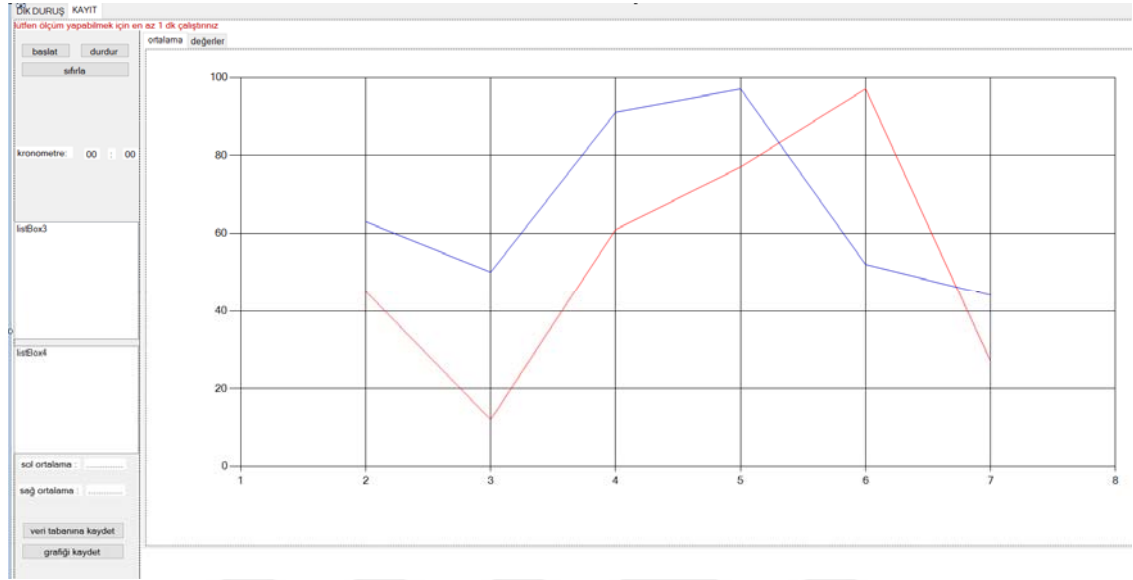
Aşağıdaki form ile ilk olarak hastanın ayakta dik ve sabit durması istenip değerler alınır. Dik duruşta ayak tabanındaki kuvvet dağılımları gözlemlenebilir. Sensörlerden okunan değer Aralığı 0-1023 olmasına rağmen daha kolay gözlemlenebilmesi adına 0-5 Aralığına indirildi. Bu dönüşüm 3.1.3.1. Analog veri bölümünde anlatılmıştır. Şekil 3.23. de dik duruş veri alma formu gösterilmektedir.



Şekil 3.23. Dik duruş veri alma formu

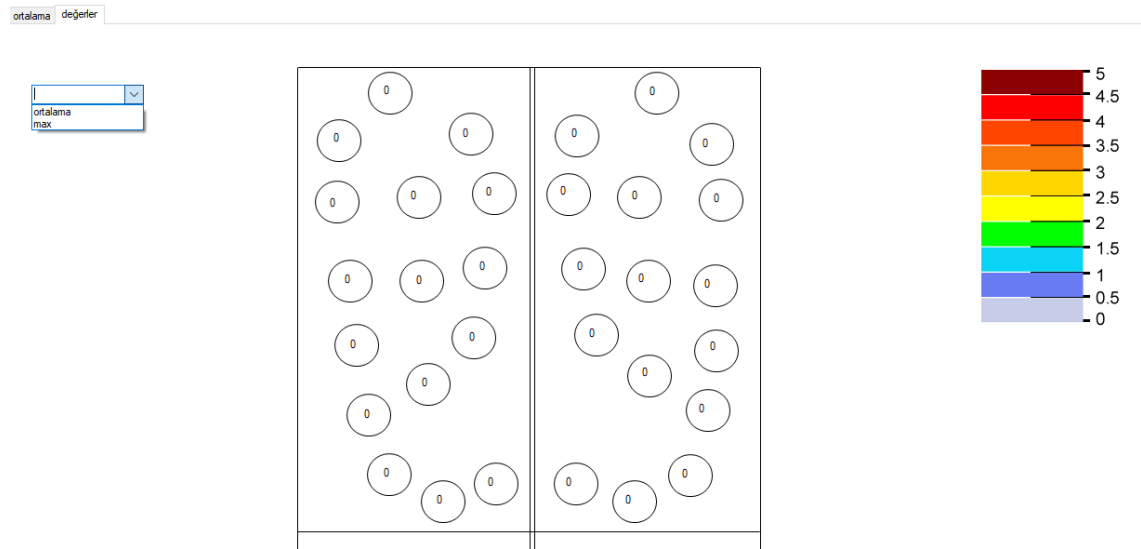
3.2.2.1.5. Yürüme boyunca veri alacak formlar

Şekil 3.24 de ki grafik yürüme esnasında sağ ve sol ayakta ortalama kuvvet değerlerini göstermektedir.



Şekil 3.24. Ortalama değerleri gösteren grafik

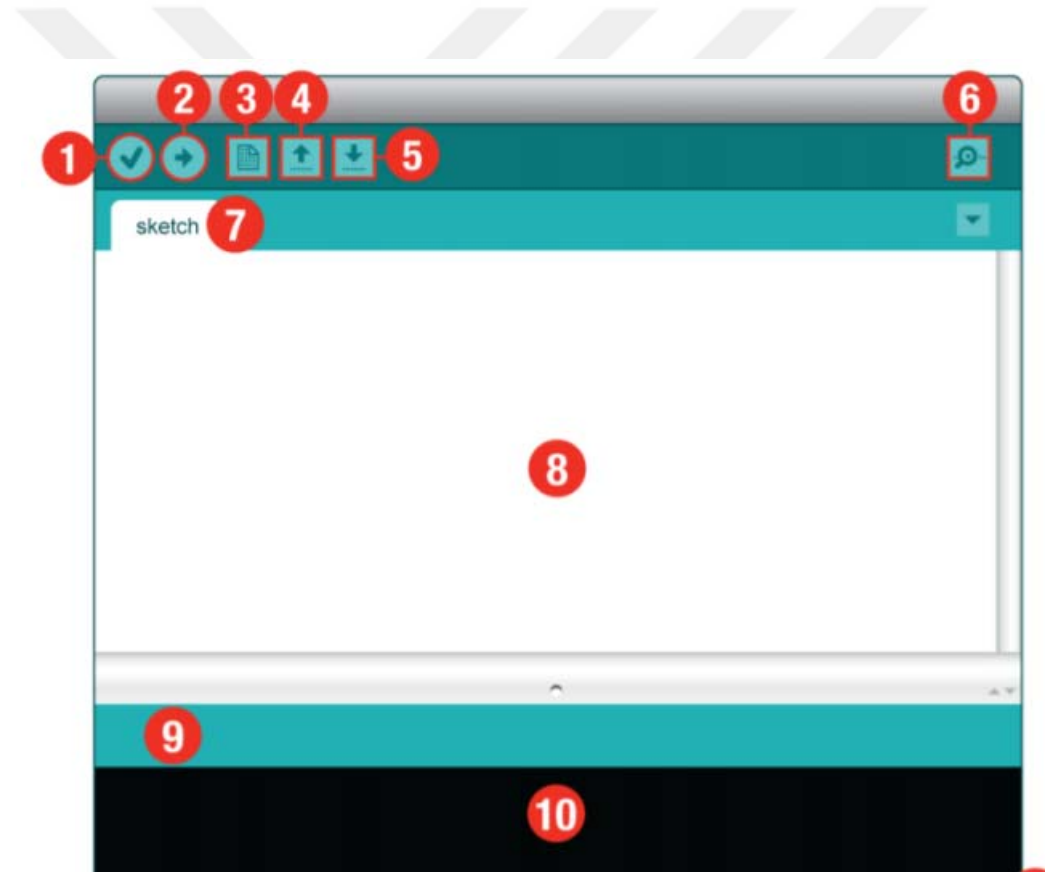
Şekil 3.25. deki formda ise sensör noktalarındaki kuvvet dağılımlarının ortalama, max ve minimum değerleri uzman kişi tarafından gözlemlenebilir.



Şekil 3.25. Kayıt formu

3.2.3. Arduino IDE

Mikro denetleyicileri programlamak için çeşitli diller kullanılmaktadır. Bu diller Assembly, C gibi dillerdir. Bu iki dil mikro denetleyicileri programlamak için kuvvetli dillerden ikisidir. Yalnız Assembly dili ilkel bir dil olduğu için anlaşılması zordur. Üstelik alt seviye bir dil olduğu için diğer üst seviye dillere göre oldukça uzun kod yazmak gerekmektedir. Arduino bir geliştirme ortamı (IDE) ve kütüphanelerden oluşur. IDE, Java dilinde yazılmış bir geliştirme ortamıdır. Kütüphaneler ise C ve C++ dillerinde yazılmıştır (Robotik Sistem 1, 2018). Açık kaynaklı ve platform bağımsız bir yazılımdır yani Linux, Windows, Mac platformlarında çalışmaktadır. Şekil 3.26 da Arduino Ide Ara yüzü gösterilmektedir (Sparkfun, 2018).



Şekil 3.26. Arduino ide

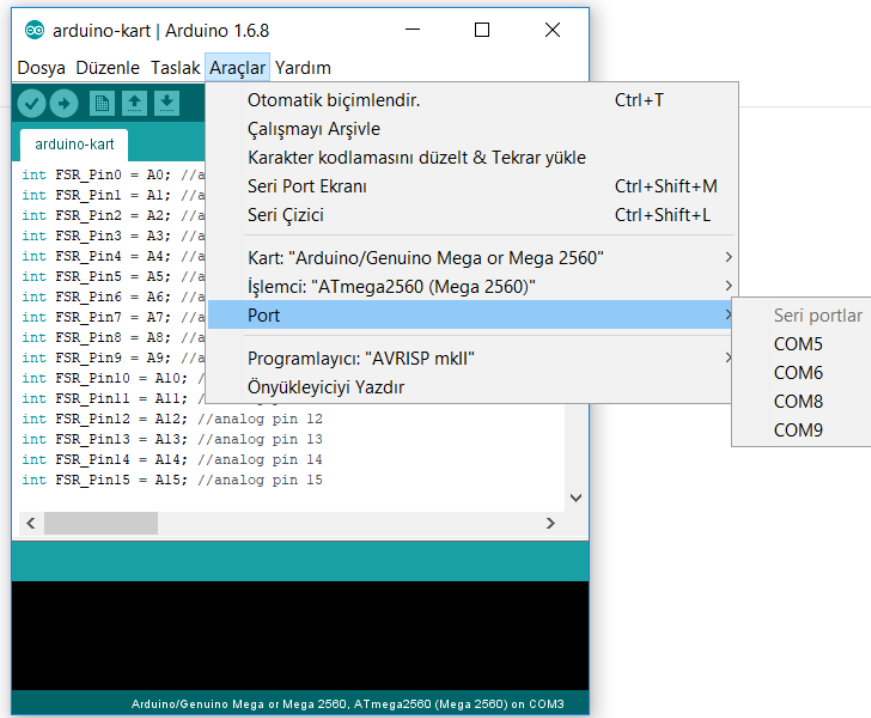
Şekil 3.26. gösterilen menüleri sırası ile açıklayalım.

1. Kontrol et: Yazdığımız programda ki hataları bulur.
2. Yükle: Yazdığımız kodu Arduino'ya yükler.
3. Yeni Belge: Yeni boş belge açar.

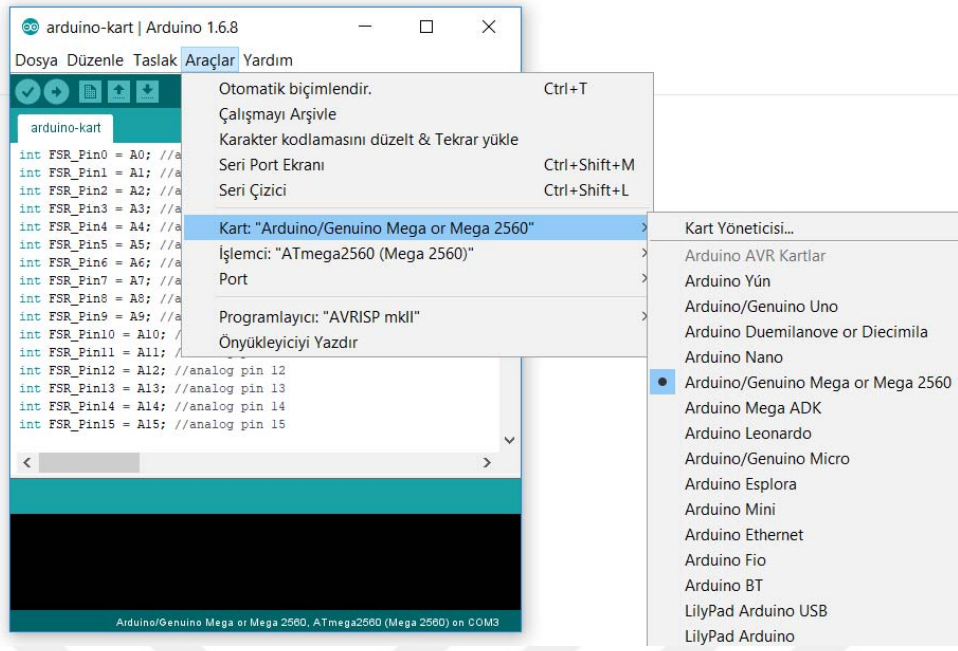
4. Aç: Daha önce kaydedilmiş bir belgeyi açar.
5. Kaydet: Programı kaydeder.
6. Seri Port Ekranı: Seri iletişim yaparken kullanılır.
7. İsim: Yazdığımız dosyanın ismi.
8. Yazma alanı: Kodların yazıldığı alan.
9. Gösterge çubuğu: Yapılan işlemin durumunu gösterir.
10. Rapor: Derleme sonucunda yapılan hataların veya yükleme bilgilerinin gösterildiği alan (Sparkfun, 2018).

3.2.3.1. Arduino ve Arduino IDE arasında bağlantı kurma

Arduino'yu USB kablo ile bilgisayara bağladıktan sonra Arduino IDE'i çalıştırıyoruz. Kodlarımızı yazmadan önce şekil 3.27. deki adımları izlemeliyiz.



(a)



(b)

Şekil 3.27. (a), (b): Arduino ide ve kart bağlantıları

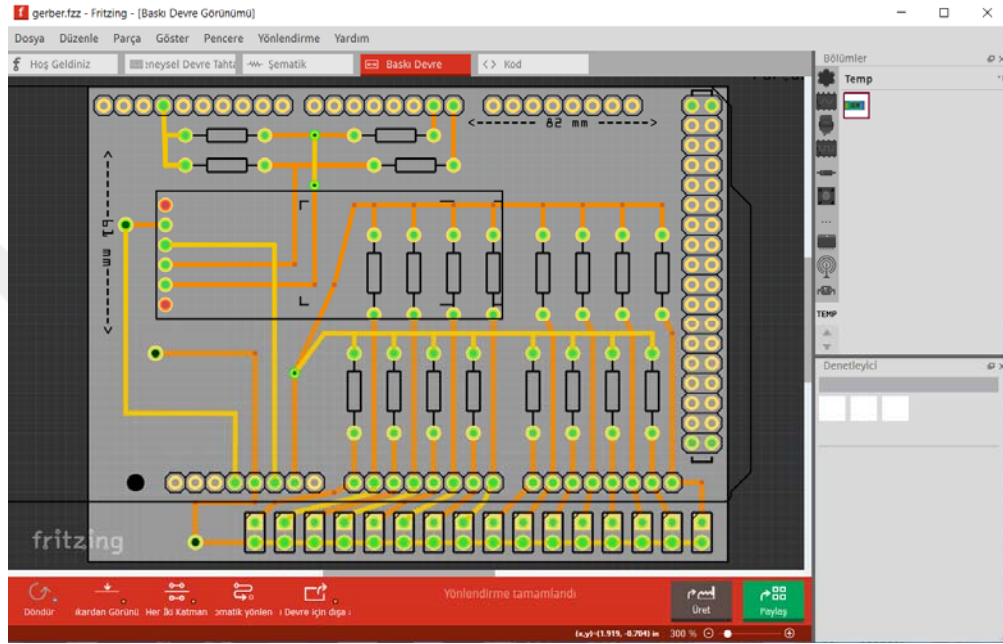
3.2.4. Fritzing

Açık kaynak veya “Open Source” olarak isimlendirilen “Free Software” türü, üzerinde değişiklikler yapabildiğimiz, kendi kütüphanenizi oluşturabileceğiniz, ücretsiz yazılımlara verilen isimdir; “Özgürlük Felsefesi” olarak ta tanımlanır. Bu felsefe sadece yazılımlarla sınırlı değildir. Bunun en iyi örneği “Arduino” geliştirme kartlarıdır (Emre, 2015). Açık kaynak kodlu sistemlere geçindikten sonra elektronik devre çizim programlarına değinecek olursak. Bunlardan bazıları:

1. AltiumDesign
2. Proteus
3. MultiSim
4. Eagle

Bu programlar sayesinde rahatlıkla elektronik devre çizimi yapılabilir ve baskı devre çıkartabiliriz. Fakat son zamanlarda çizimlerin içerisine Arduino vb. geliştirme kartları ve bunların harici donanımları da girince bu çizimleri bu programlarla gerçekleştirmek zorlaştı. Bu durumda karşımıza büyük bir alternatif olarak Fritzing çıkmaktadır.

Fritzing açık kaynak kodlu bir elektronik devre çizim ve simülasyon programıdır. İçerisinde birçok geliştirme kartının ana boardları ve harici donanımları, direnç, led, vb. demirbaşlar haricinde röleler ve anahtarlama elemanları da bulunmaktadır (Emre, 2015). Şekil 3.28. de Fritzing programında çizilmiş sisteme ait elektronik devre kartının çizimi ve programın ara yüzü gösterilmektedir.



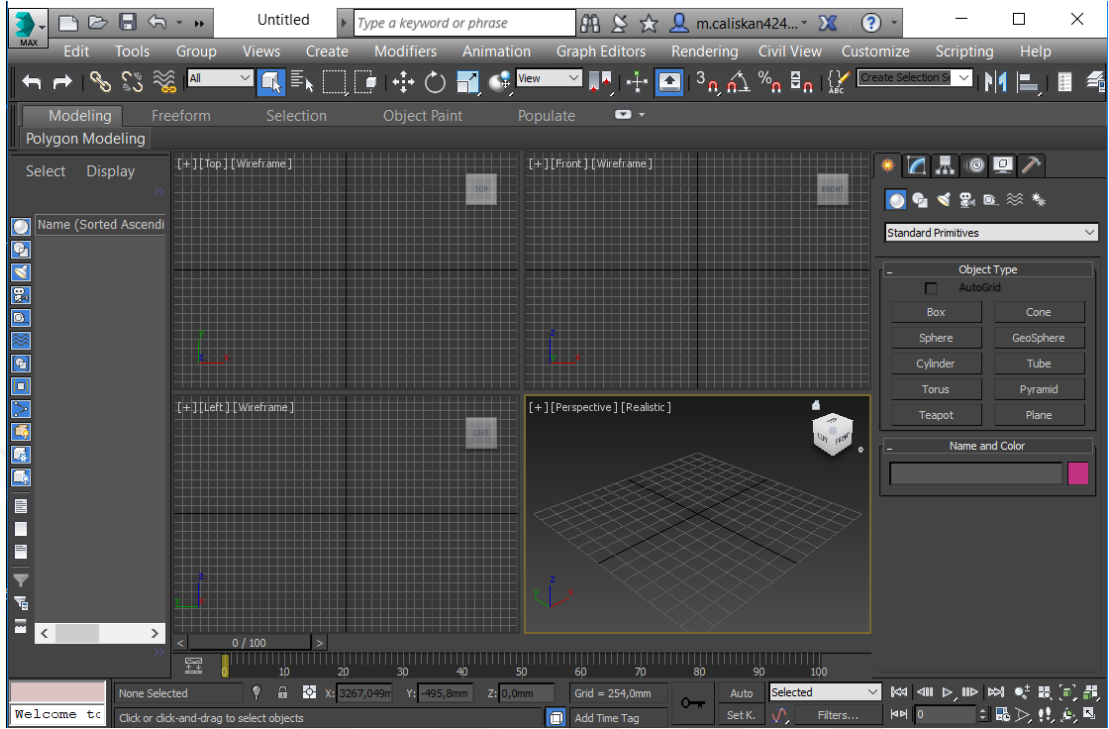
Şekil 3.28. Fritzing programı

3.2.5. Autodesk 3Ds Max

Autodesk 3Ds Max' in ilk örneği 1988 yılında test edilmiştir. İlk program 4 programcı tarafından yazılmıştır. Bu 4 kişilik yazılım ekibinin başında 3D Max' in fikir babası Tom Hudson vardır. 3D Max 1 Ekim 1990 tarihinde piyasaya sürülmüştür. DOS ortamında çalışmakta olan ilk sürüm, Yost Group tarafından geliştirilmiş ve dağıtımcılığı Autodesk firması tarafından yapılmıştır. 3D Studio 4. Sürümünden sonra Windows NT platformu için yeniden kodlanarak 3D Studio Max ismi ile piyasaya çıkmıştır (3Dakademi, 2018).

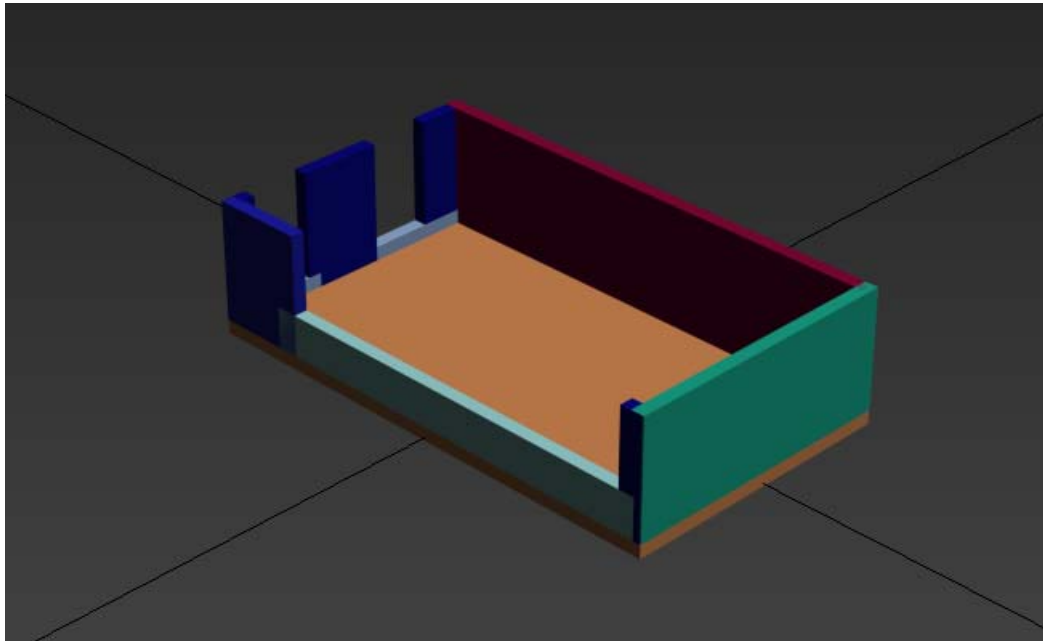
3Ds Max, oldukça popüler ve en çok tercih edilen 3D modelleme ve animasyon programlarının başında gelir. Tam adı "3D Studio Max" tır. Gelişmiş eklenti desteği, güçlü modelleme kabiliyeti ile modelleme yazılımları arasında en yaygın kullanıma

sahip uygulamalardan birisi; birçok hazır model ve eklentilere sahip programlardan birisidir (3Dakademi, 2018). Şekil 3.29. da 3ds Max Arayüzü gösterilmektedir.



Şekil 3.29. 3ds max arayüzü

Şekil 3.30 3ds Max' ta çalışmamız için tasarlanmış bir çizim gösterilmektedir.



Şekil 3.30. Proje için 3 boyutlu çizim

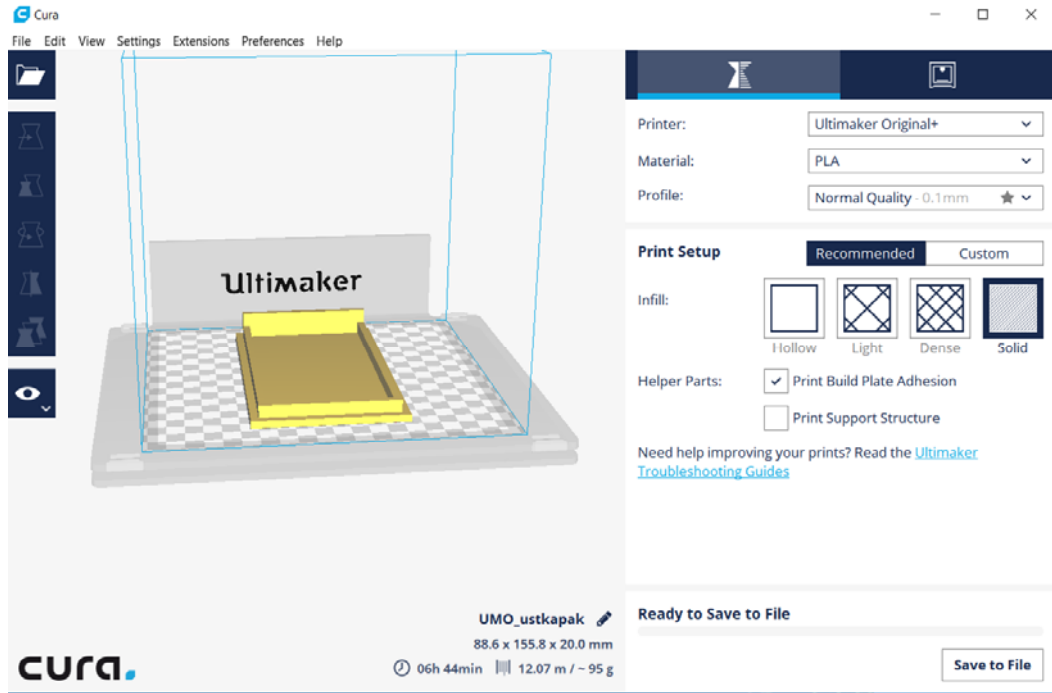
3.2.6. Cura

Ultimaker cura modelleri 3d baskıya hazırlamak için STL, 3MF ve OBJ dosya formatlarını destekleyen; açık kaynak ve tamamen ücretsiz bir programdır (Ultimaker, 2018). Cura yazılım ile baskı almak için gerekli adımlar: 3ds max ile çizilen modeller .stl uzantısı ile kaydedilir, yeni dosya cura programında açılıp gerekli baskı ayarları yapıldıktan sonra .gcode uzantısıyla kaydettikten sonra 3d yazıcıya gönderilir ve 3d yazıcıdan baskı alınır.

Cura yazılımının iki önemli işlevi bulunmaktadır. Bunlar:

1. Gcode uzantılı dosyaların kolay bir şekilde oluşturulmasına olanak sağlar
2. Hazırlanan gcode uzantılı dosyaların baskı alınması için 3D yazıcıya USB kablo ile ya da SD kart ile aktarılabilir.

CURA, David Braam (Daid) tarafından Ultimaker desteğiyle geliştirilen açık kaynak kodlu bir yazılımdır (Rigid3d, 2018). Şekil 3.31. de cura yazılım ara yüzü ve çalışmamız için geliştirilen örnek çizim gösterilmektedir.



Şekil 3.31. Cura yazılımı

3.2.7. Microsoft Office Access

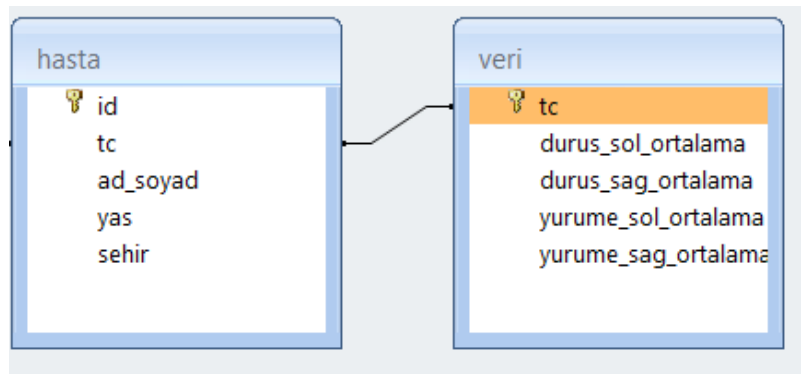
Sistemde veri tabanı olarak Microsoft Office Access programı kullanılmıştır. Microsoft Access veri tabanı programı İlişkisel veri tabanı yönetim sistemi ile çalışmaktadır. Access ile bir veri tabanını oluşturmak ve oluşturulan veri tabanını kullanmak diğer veri tabanlarına göre çok daha kolaydır. Bunun temel nedeni Access'in Windows Grafiksel Kullanıcı Arabiriminin sağladığı avantajlardan yararlanma imkânı sağlamasıdır (Bilmeslek, 2011).

3.2.7.1. Veritabanı nesneler

Bir veri tabanında verileri kaydetmek kadar, kaydettiğiniz verilere en hızlı şekilde ulaşabilmek kadar önemli bir noktadır. Kayıt altına alınan bilgilerin tablolar içerisinde saklanması işi çok kolaylaştırmaktadır fakat farklı özelliklere de ihtiyaç duyulmaktadır. Bu özellikler:

- Tablolar: MS Access programında oluşturulan veri tabanının en önemli unsuru tablolardır. Sütunlara alan, satırlara da kayıt denilmektedir.
- Raporlar: Verilerin etkin bir şekilde sunulmasını sağlar.
- Formlar: Veri tabanı kullanıcılarının kolay bir şekilde veri girişi yapmalarına olanak sağlar.
- Sorgular: Verilere hızlı erişilmesini sağlayan bir araçtır (Chip, 2007).

Şekil 3.32. de sistemde kullanılan tablolar arasındaki ilişki gösterilmiştir.



Şekil 3.32. Tablolar arası ilişki

3.3. Nesnelerin İnterneti (IoT)

3.3.1. Seçilen mimari ve sistemdeki işleyişi

Bölüm 2.2. de bahsedilen mimarilerden geliştirilecek sistem için 5 katmanlı mimari seçilmiştir.

Nesne Katmanı (Object Layer): Kuvvet sensörleri kuvvet değerlerinden oluşan analog verileri bu katmanda algılar. Algılanan analog veriler HC-05 Bluetooth Modülü'ne gönderilmek üzere Arduino' ya alınır ve burada tutulur.

Nesne Soyutlama Katmanı (Object Abstraction Layer): Alt katmandan (nesne katmanını) gelen veriler Bluetooth teknolojisinde kullanılmak üzere bu katmanda dönüştürülür. HC-05 Bluetooth Modülü'ne gelen veriler Bluetooth algılayıcı modüle yönlendirilir.

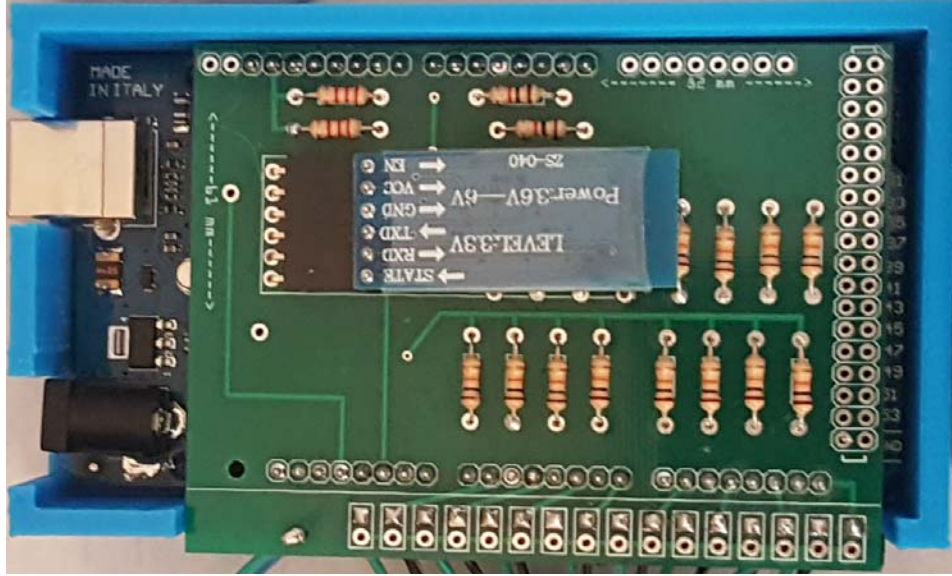
Servis Yönetimi Katmanı (Service Management Layer): Seri porttan paket halinde alınan veriler parçalara ayrılıp bu katmanda işlenir.

Uygulama Katmanı (Application Layer): İşlenen verilerin, kullanıcıların anlayabileceği görsel verilere, grafik vb. bu katmanda dönüştürülür. Bu katmanda Visual Studio platformu verileri; grafiklere ve görsel öğelere dönüştürmek için etkili bir şekilde çalışır.

İş / Yönetim Katmanı (Business Layer): Sistemle ilgili bu bölümde ne yapıldıysa onu yazalım

3.3.2. IoT platform örneği

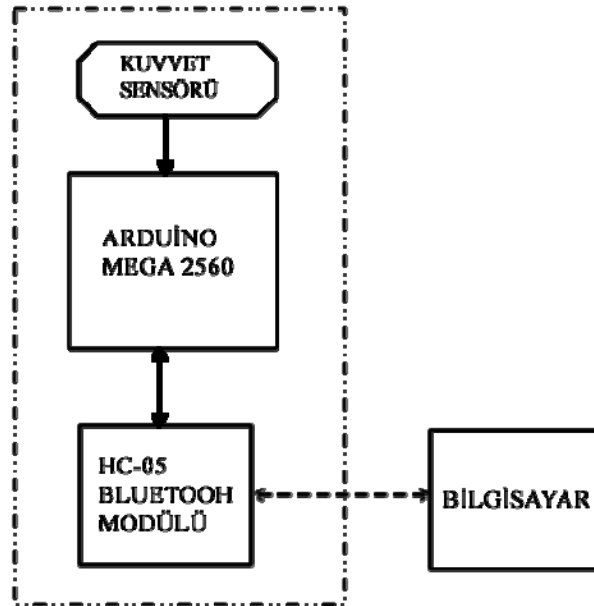
Bu çalışmada, Arduino Mega 2560 için uyumlu bir pcb kart tasarlanmıştır. Bu platform üzerinde analog verileri toplayabilme yeteneğine sahip algılayıcıların bağlanabilmesi için analog uçlar bulunmaktadır. Arduino' ya ait ide ara yüz seri port bileşeni ile veriler çoklu olarak izlenebilmektedir. Gömülü sistem boardları üzerine Bluetooth hc-05 ve hc-06 modülleri gibi haberleşme modülleri kolay bir şekilde bağlanabilmektedir. Platform aracılığı ile hastaya ait sayısal veriler uzman tarafından eş zamanlı olarak izlenebilmektedir. Şekil 3.34. de tasarlanan platform gösterilmektedir.



Şekil 3.34. IOT platformu

3.3.3. Sistemin blok diyagramı

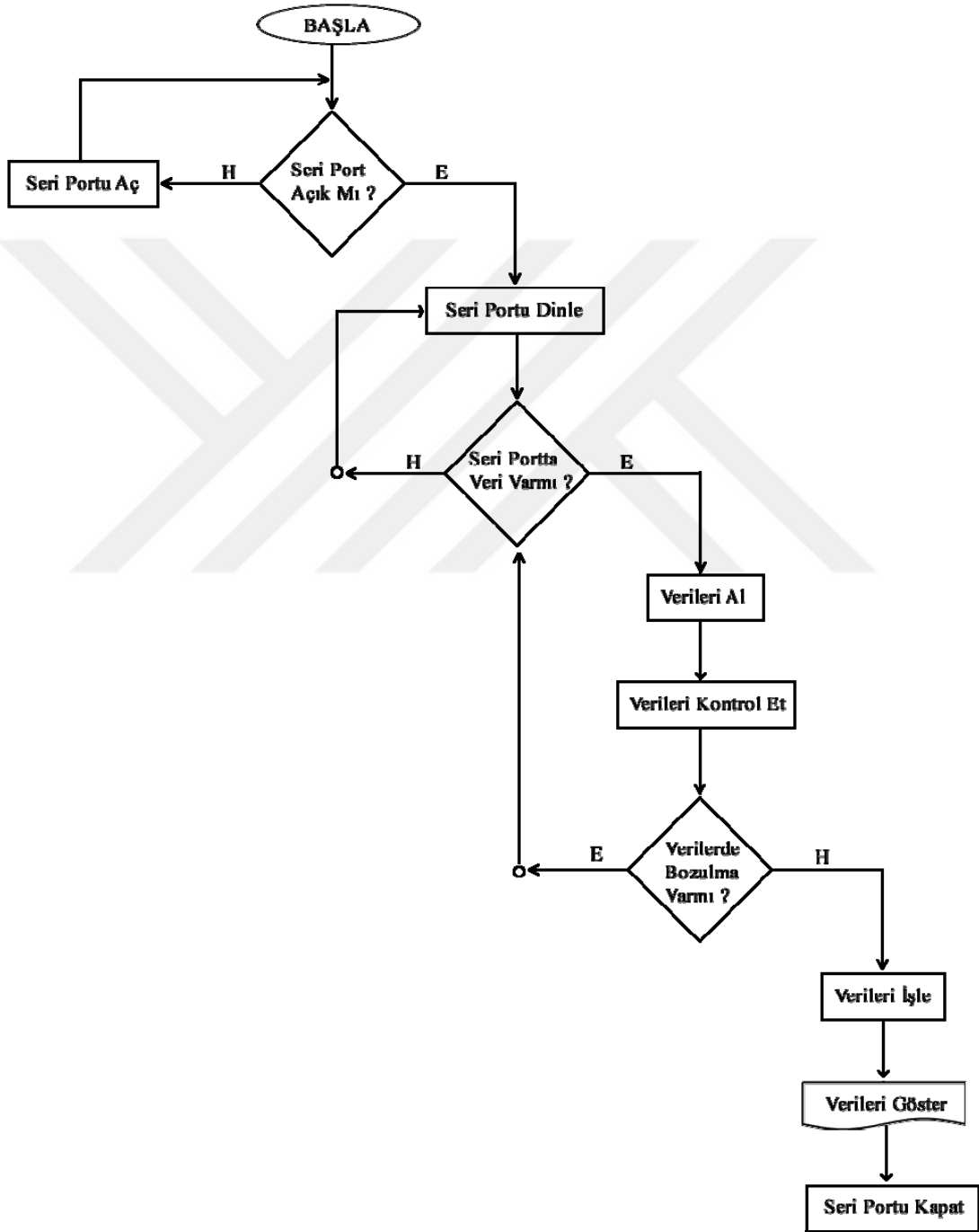
Bu çalışmada geliştirilen sistem temel olarak bir kuvvet sensör'ü, alınan verileri işleyip kablosuz olarak veri aktarma yeteneğine sahip bir işlem birimi ve kablosuz olarak aktarılan verileri işleyip görsel olarak gösterme yeteneğine sahip bir cihazdan oluşmaktadır. Şekil 3.35. de sistemin blok diyagramı gösterilmektedir.



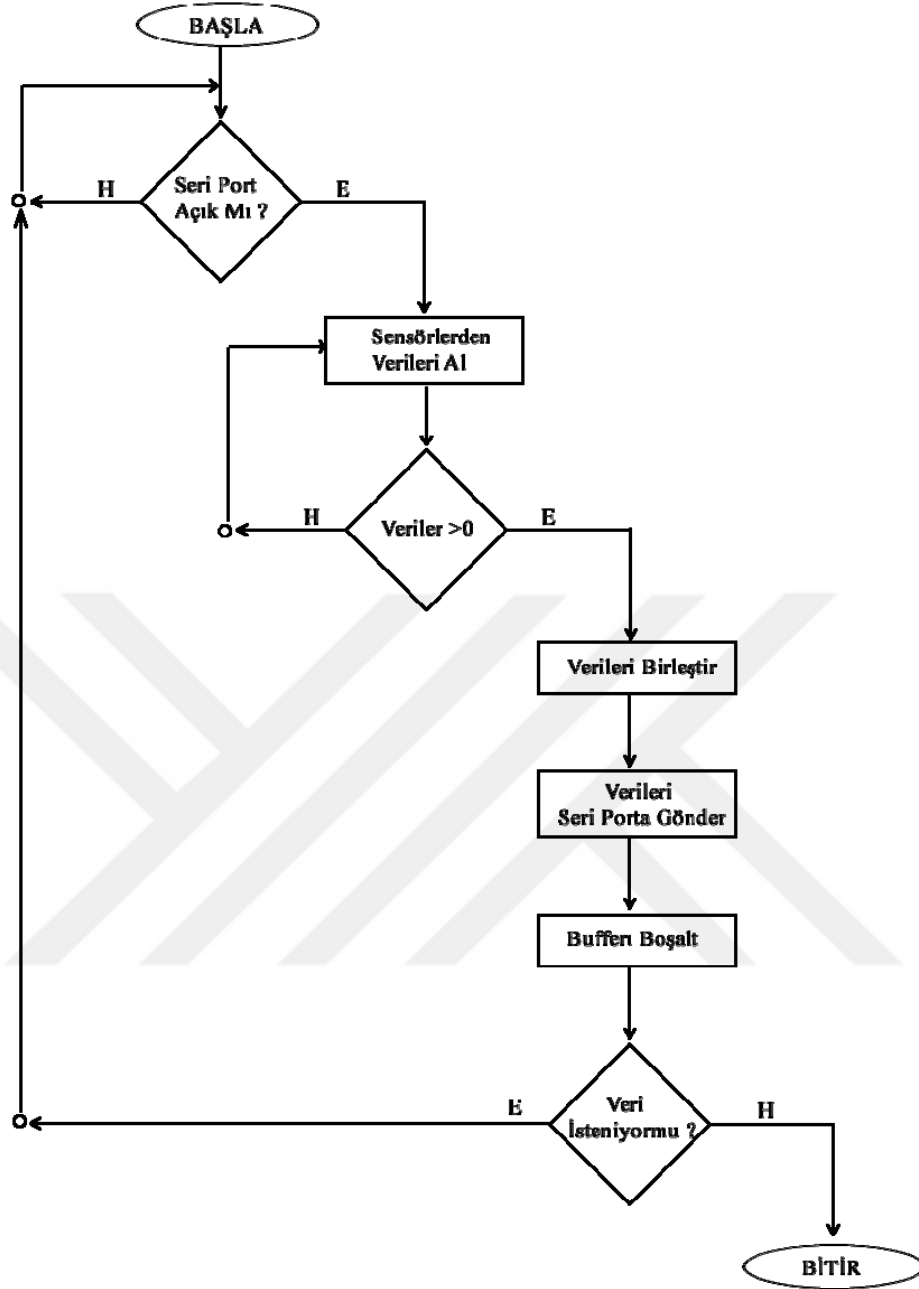
Şekil 3.35. Sistemin blok diyagramı

3.3.4. Sistemde kullanılan yazılımların akış diyagramları

Sistemde kullanılan yazılımların akış diyagramları şekil 3.36. ve 3.37. de gösterilmektedir. Şekil 3.36 C# yazılımının akış diyagramı, 3.37. de ise Arduino IDE akış diyagramıdır.



Şekil 3.36. Sistemin akış diyagramı

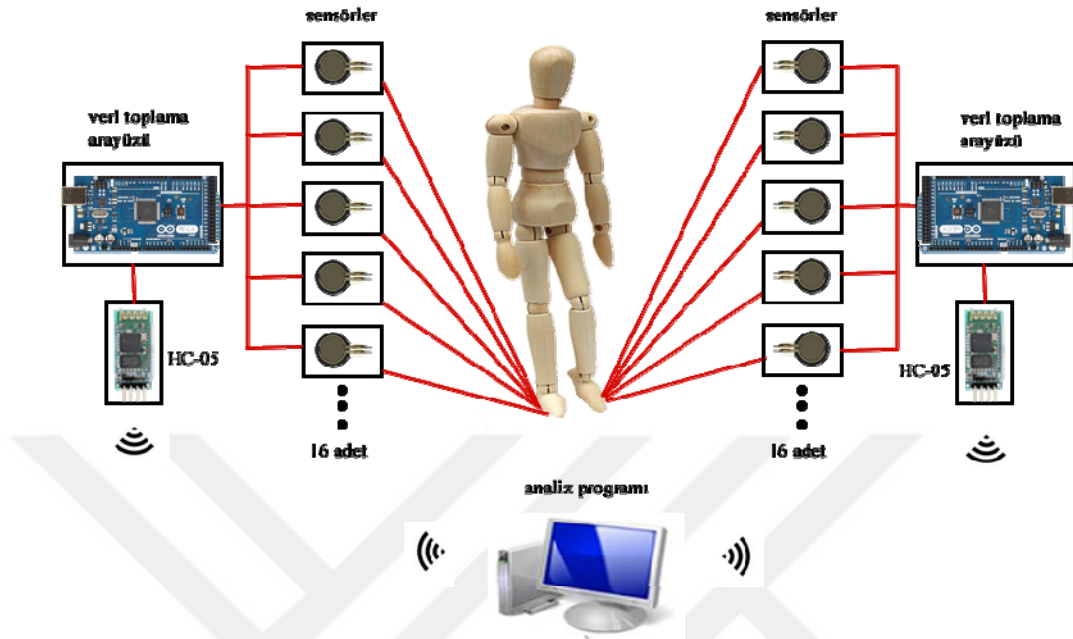


Şekil 3.37. Arduino IDE akış diyagramı

3.3.5. Önerilen sistemin genel yapısı

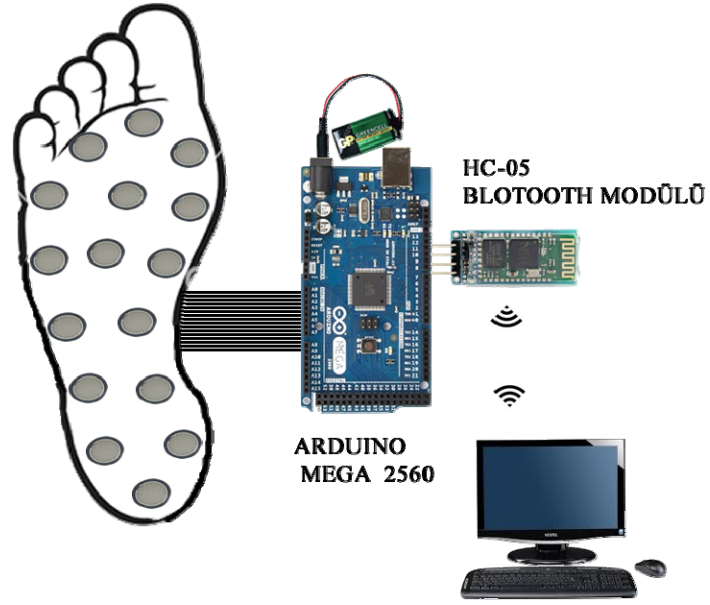
Şekil 3.38. de geliştirilen sistemin çalışmasını gösteren bir şema yer almaktadır. Düğümlerden elde edilen ölçüm sonuçları bilgisayar, akıllı telefon gibi cihazlar kullanılarak kayıtların tutulduğu yazılımlara bağlanılabilir ve düğümlerin takibi ve çıkış değerlerinin kontrolü gerçekleştirilebilir. Sistem 2 ayakta 16 şar adet olmak üzere toplamda 32 sensör, sensörler' den gelen verilerin işlenmeden ilk olarak toplandığı 2

adet Arduino Mega 2560, 2 adet HC – 05 Bluetooth Modülü, tüm bunların birbiri ile iletişimini sağlayan 2 adet pcb kart ve bilgisayar sisteminden oluşmaktadır.



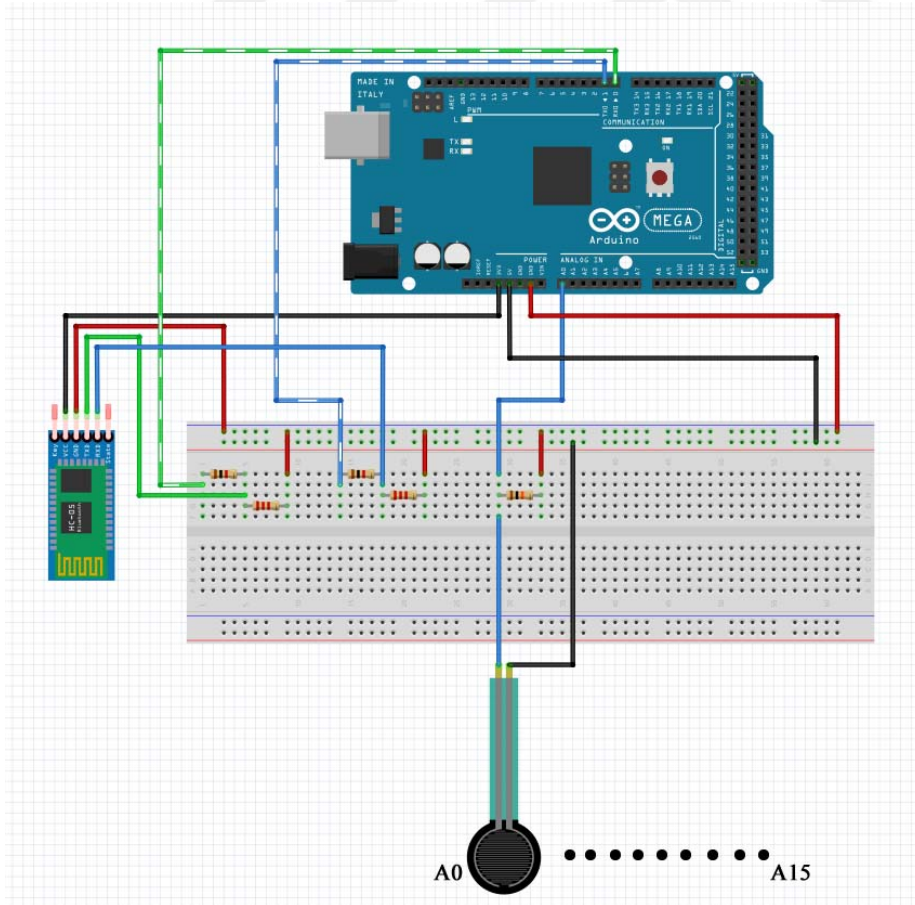
3.4. Sistemin Birleşimi ve Haberleşmesi

Haberleşme için nesnelerin interneti yöntemi kullanılmaktadır. Nesnelerin interneti: Nesnelerin birbiriyle kablosuz haberleşmesi ve oluşan değerlerin bilgisayar yazılımı tarafından analiz edilmesi denilebilir. Şekil 3.39. da sistemin birleşimi ve haberleşmesi görülmektedir.



Şekil 3.39. Sistemin birleşimi

Şekil 3.40. da sistemin fritzing programında çizilmiş devre şeması yer almaktadır.



Şekil 3.40. Sistemin devre şeması

Sistemin devresel kontrolleri ve haberleşme testleri yapıldıktan sonra sistem birleştirilerek son halini alması sağlandı. İlk aşama: Sensörler' in tabanlığa, belirtilen konumlara yerleştirildi ve sensörler RGB kablolar ve klemensler ile pcb karta sabitlendi. İkinci aşama: Devre kartları 3 boyutlu yazıcı ile basılan kutuya yerleştirildi. Son olarak kablolar görülmemesi için dışı deri parçaları ile kaplandı. Şekil 3.41. ve 3.42. de sistemin son hali gösterilmektedir.



Şekil 3.41. Sistemin ayakkabıya yerleştirilmesi



Şekil 3.42. Sistemin son hali

3.5. Sistemin Çalışması ve Katılımcılara Uygulanması

Sistemin testi için 26 – 64 yaş arasında 18 katılımcı katıldı. Katılımcılardan 14'ü erkek 4'ü kadındır. Katılımcılar farklı meslek gruplarından seçilmiştir. Katılımcılara sistem hakkında ve kişisel bilgilerin kullanımı hakkında bilgi verilmiştir. Katılımcılara ait yaş, kilo, ayak numarası gibi bilgiler ölçülüp kayıt altına alınmıştır. Katılımcıların

ölçüm esnasında yürümelerini kısıtlamayacak, rahat hareket imkânı sağlayacak kıyafetler giymeleri sağlanmıştır.

3.5.1. Hazırlık

Sistemin uygulanması için gereken ilk adım yürüme parkurunun belirlenmesidir. Yürüme parkuru: Düz; beton, asfalt ya da karo gibi zeminler tercih edilmelidir. Yürüyüş esnasında katılımcının yürüyeceği başlangıç ve bitiş noktaları belirlenmelidir. Bu mesafe belirlenirken yürüyüş parkuru 20 m den fazla olmamalıdır (Bilgisayar sistemi parkurun tam ortasına konulduğu kabul edilmektedir). Bluetooth Teknolojisi 10 m mesafeye kadar haberleşebilmektedir (Bölüm 2.2.6.). Şekil 3.43. de sistemin test edildiği yürüme parkuru gösterilmektedir.



Şekil 3.43. Yürüme parkuru

Yürüme parkuru belirlendikten sonraki aşama hastanın sistemi kullanmak için hazırlanmasıdır. Sensörlerin yerleştirildiği tabanlık ayakkabı içine yerleştirilir.

Elektronik devre kartının bulunduğu muhafaza kutusu, yürüme esnasında hareket etmeyecek şekilde hastanın bacaklarına sabitlenir. Son olarak Pil bağlantıları yapılarak sisteme gereken enerji verilir. Şekil 3.44. de sistemin hastanın ayağına sabitlenmesi gösterilmiştir.

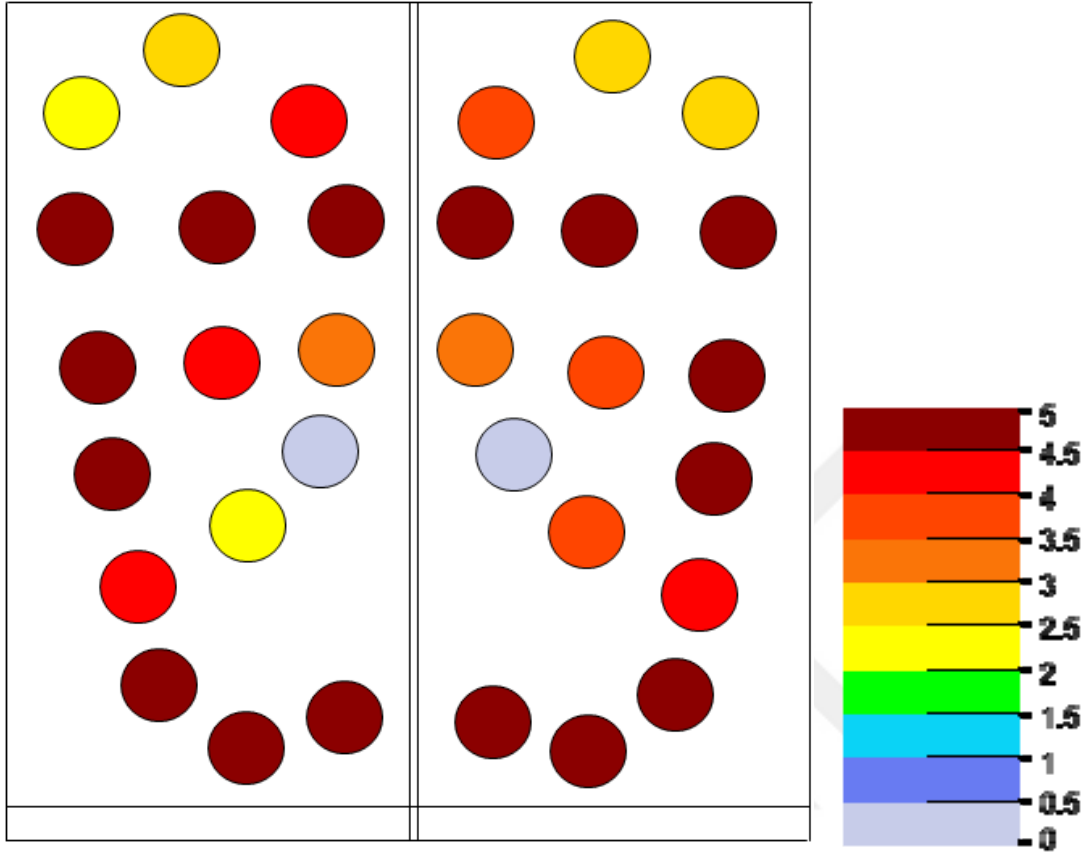


Şekil 3.44. Sistemin hastanın ayağına sabitlenmesi

3.5.2. Sistemin kullanımı

Sistem 2 bölümden oluşmaktadır. Bu bölümler dik duruş ve yürümedir. Dik duruş bölümünde hastanın verilerini almak için hastanın düz zeminde dik durması sağlanır. 0,2 sn de bir sensörlerden veri alan sistem, ayak tabanlarında oluşan kuvvet dağılımlarını göstermektedir. Şekil 3.45 de dik duruş ekran görüntüsü verilmektedir.

Dik duran bir insan vücut ağırlığını iki ayağına de dengeli dağıtmalıdır. İki ayağa uygulanan toplam kuvvetlerin aynı ya da birbirine çok yakın olması beklenir.

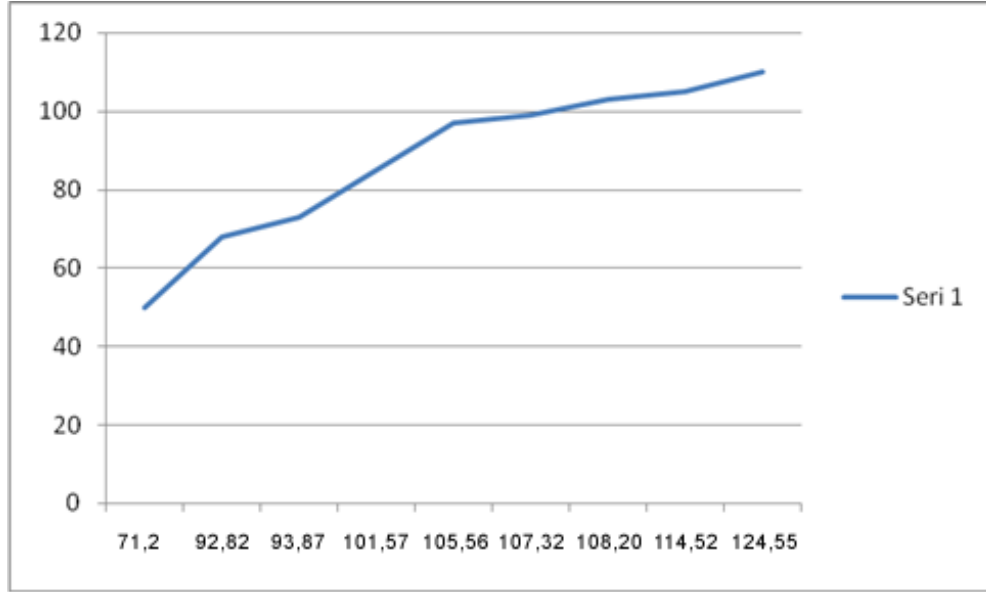


Şekil 3.45. Dik duruş alınan veriler

Gözlenen kuvvet dağılımlarının birbirine yakın değerler olmaması durumunda sistemin diğer bileşenleri de kullanılarak hasta takip edilmeye devam edilir. Burada ki problemin kaynağı araştırılmalıdır. Kuvvet az uygulanan bacakta eklem kaynaklı bir problem ya da ayak tabanında olan bir yara, nasır, topuk diken vb. olabilir.

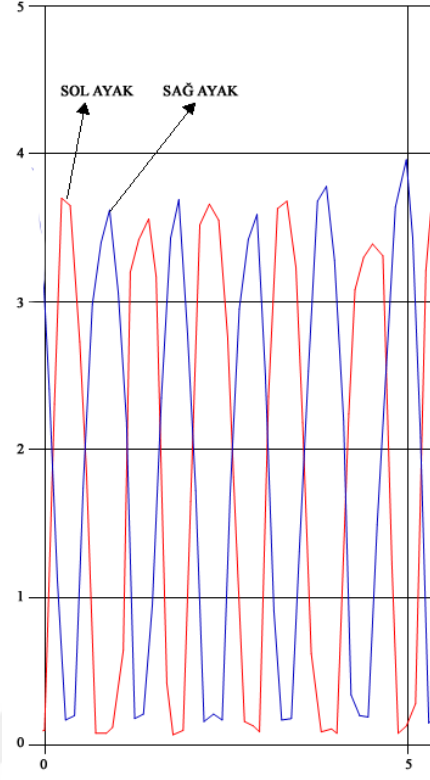
Şekil 3.45. de ayrıca hastanın ayak tabanında en fazla kuvvet uyguladığı noktalarda gözlemlenebilir. Bu sonuçlara göre hastaya uygun tabanlıklar üretilerek hastanın ayak sağlığına katkı sağlanabilir.

Şekil 3.46. da grafikte vücut ağırlığı ile ayak tabanlarında oluşan toplam kuvvet değerleri arasındaki ilişki gösterilmektedir. Grafikte ayak tabanında oluşan kuvvetlerin vücut ağırlığı ile doğru orantılı olmadığı görülmektedir. Bunun nedeni sensörler kuvvet uygulandıkça doğrusal değil parabolik artış olmasıdır (Bölüm 3.1.3.1).



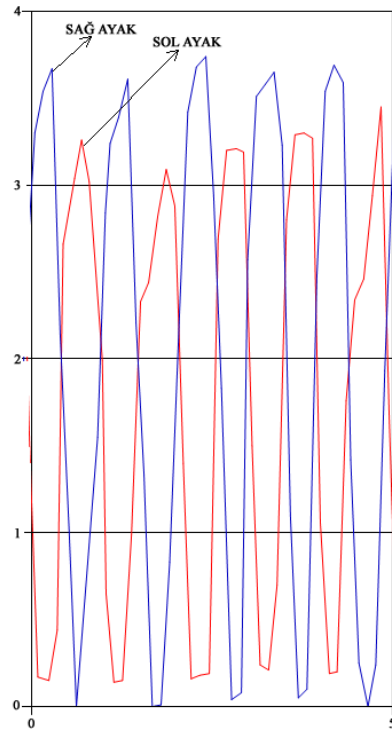
Şekil 3.46. Kuvvet ağırlık oranı grafiği (X eksen: Toplam Kuvvet, Y eksen: Kilo, Seri 1: Toplam Kuvvet, Ağırlık oranını göstermektedir.)

İkinci bölüm; yürüme esnasında alınan verilerin değerlendirildiği bölümdür. Bilgisayar sistemi parkurun ortasında olmak şartıyla maksimum 20 m uzunluğunda bir yürüyüş parkurunda hastanın doğal yürüyüşünü yapması sağlanır. Yürüme boyunca 200 ms (0,2 sn) de bir sensörlerden veriler çekilir. Gelen verilerin ortalaması alınarak birim zamanda ayak tabanında uygulanan ortalama kuvvet hesaplanır. İki ayaktan gelen ortalama değerler hastanın yürüme grafiğini oluşturur. Şekil 3.47. de sistem tarafından üretilen grafik gösterilmektedir. Grafiğin dikey eksen; 0 – 5 Aralığında ortalama kuvvet değerlerini, yatay eksen; sn cinsinden zamanı göstermektedir. Grafikte sol ayak ve sağ ayak olmak üzere iki seri bulunmaktadır. Grafiğin yatay eksen yürüme hızını (dakikadaki adım sayısını) verir. Grafikteki katılımcı dakikada yaklaşık 120 adım atmaktadır. Dakikada atılan adım sayısı ortalama 117 dir (Bölüm 2.3.2.). Grafikte serilerin kesiştiği doğrunun üst kısmı yürüme döngüsünün Basma Fazını, alt kısmı ise Salınım Fazını göstermektedir. Sağlıklı bir yürümede iki ayağın tepe ve dip değerlerinin aynı seviyede olması beklenir (Bölüm 3.1.3.1.).



Şekil 3.47. Yürüme grafiği

Şekil 3.48. de yürüme bozukluğu tespit edilmiş katılımcının grafiği gösterilmektedir.



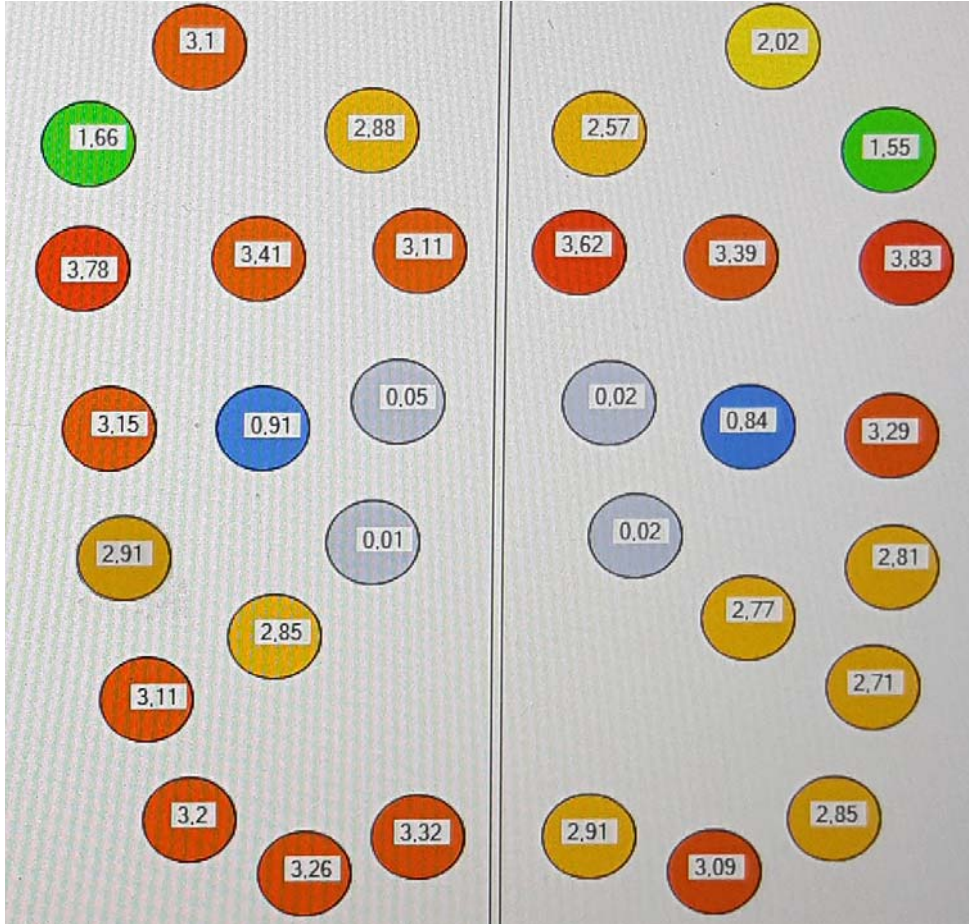
Şekil 3.48. Yürüme bozukluğu tespit edilen hastanın grafiği

Şekil 3.48. deki grafik incelendiğinde hastanın yürüme esnasında sağ ayağına daha fazla kuvvet uyguladığı gözlenmektedir. Ayrıca grafikteki serilerin tepe değerleri incelendiğinde sağ ayağa ait serinin tepe değerlerinin diğer seriye göre daha yüksek olduğu görülmektedir. Yürüme esnasında ayaklarda oluşan ortalama kuvvetler:

- Sol Ayak Ortalama Kuvvet:1,57
- Sağ Ayak Ortalama Kuvvet: 2,07

Bu sonuçlar bize yürüme bozukluğu olabileceğini gösterir fakat bunun nedeni hakkında fikir sahibi olmamız için diğer Yürüme Analiz yöntemleri de kullanılmalıdır (Bölüm 2.4.).

İkinci bölümde seri port okuma işlemi tamamlandıktan sonra yazılım bazı istatistiksel veriler oluşturmaktadır bunlar; yürüme esnasında tek bir sensörde oluşan ortalama kuvvet ve sensöre uygulanan en yüksek kuvvettir. Şekil 3.49. da yürüme boyunca sensörlerde oluşan ortalama kuvvetler gösterilmektedir.



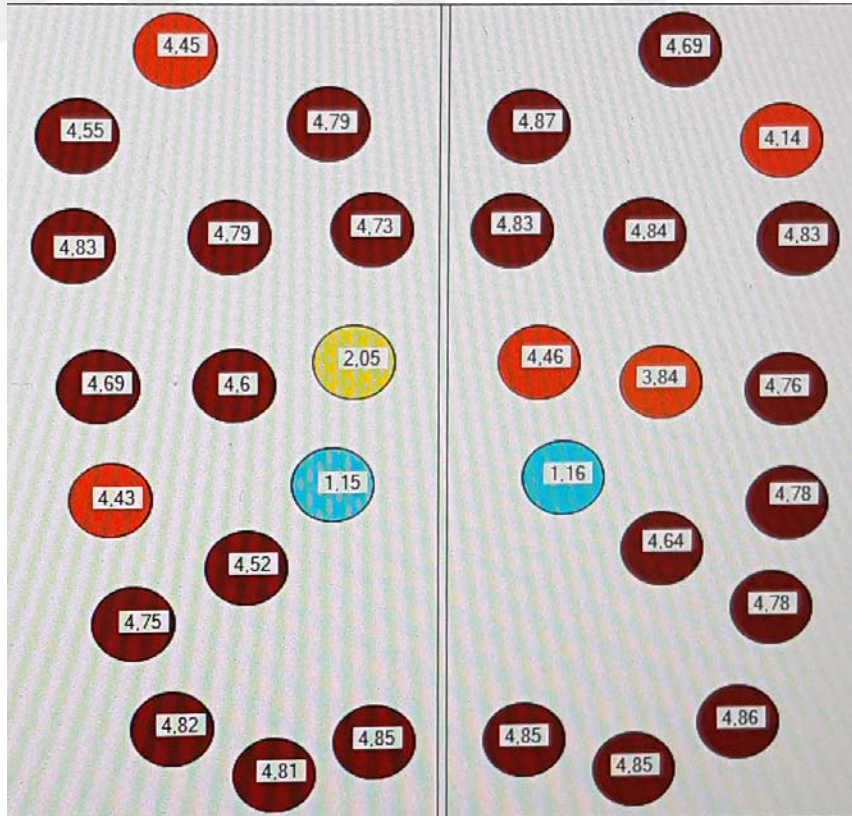
Şekil 3.49. Yürüme boyunca sensörlerde oluşan ortalama kuvvet

Şekil 3.49 da ki katılımcının sol ayak topuğunun dışına sağ ayağın aynı bölgesinden daha fazla kuvvet uyguladığı görülmektedir. Şekil 3.50. de bu katılımcının ayakkabı topukları gösterilmektedir. Şekilde görüldüğü üzere sol topuğun dışı sağ topuğun dışına göre daha çok yıpranmıştır.



Şekil 3.50. Ayak topuğuna uygulanan kuvvetler

Sistem son olarak sensörlere uygulanan maksimum kuvvetleri de vermektedir. Şekil 3.51. de uygulanan maksimum kuvvetler gösterilmektedir.



Şekil 3.51. Maksimum kuvvetler

Çizelge 3.6. da yürüme boyunca katılımcılardan alınan değerler gösterilmektedir.

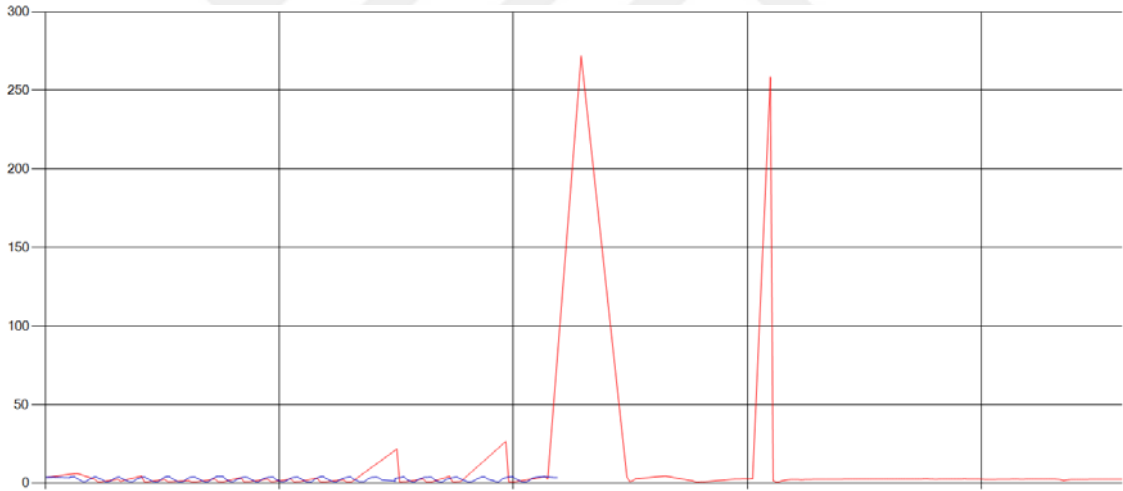
Çizelge 3.6. Katılımcı değerleri

Sn	Yaş	Kilo	Ayak Numarası	Sol Ortalama	Sağ ortalama	Maksimum
1	43	103	43	2,11	2,96	4,86
2	42	67	38	2,21	2,26	4,85
3	32	50	36	1,51	1,49	4,84
4	63	85	42	1,57	2,07	4,82
5	34	97	42	2,4	2,38	4,83
6	64	80	39	1,03	1,06	4,82
7	33	73	40	1,66	1,71	4,86
8	36	94	43	2,02	2,52	4,88
9	34	87	42	1,91	1,8	4,86
10	26	72	42	2,04	2,17	4,80
11	37	68	41	1,7	2,1	4,85
12	30	105	44	2,71	2,99	4,89
13	33	99	43	2,36	2,39	4,87
14	31	104	44	2,46	2,91	4,87
15	47	85	42	1,79	2,02	4,89
16	31	110	43	-----	-----	-----
17	57	85	43	2,18	2,64	4,88
18	59	99	39	-----	-----	-----

4. ARAŞTIRMA SONUÇLARI VE TARTIŞMA

Sistem 20 – 64 yaş arası 18 katılımcıya uygulanmıştır. Sistemin testi sert zeminden oluşan 20 m lik bir yürüme parkurunda gerçekleştirilmiştir. Sistemin uygulanması sonucu karşılaşılan bulgular bu bölümde ele alınacaktır.

Uygulamaya katılan 18 katılımcının tamamında sistem Dik Duruş menüsü sağlıklı çalışmış fakat yürüme bölümünde 2 katılımcının değerleri gözlemlenememiş, hatalı veriler elde edilmiştir. Bu katılımcılardan biri 110 kilo erkektir. Şekil 4.1. de bu kullanıcıya ait grafik gösterilmektedir. Bu gönüllüde sistemin çalışmama nedeni toplam ağırlıktan ziyade tek bir sensöre 10 kg’ dan daha fazla kuvvet uygulanmasıdır. Yapılan gözlemlerde 3 sensöre diğer sensörlere göre çok daha az kuvvet uygulandığı görülmüştür. Bu sensörlere uygulanan kuvvetleri sıfır kabul edip geri kalan 13 sensöre 10 kg uygulandığını düşünürsek sistem teorikte 130 kg’ a kadar çalışmaktadır. Sistemin testinde 105 kg’ a kadar bireyler sağlıklı bir şekilde ölçülmüştür.



Şekil 4.1. Hatalı yürüme grafiği

Ölçüm yapılamayan diğer katılımcı 99 kilo kadındır. Sistemin burada hata verme nedeni katılımcının bacaklarında platin takılı olduğu için elektronik kartlar ve kablosuz haberleşme ünitesinin etkilendiği düşünülmektedir.

Çizelge 4.1. de katılımcıların kiloları ve sensörlere uyguladıkları maksimum değerler gözlenmektedir.

Çizelge 4.1. Maksimum değerler

Sn	Yaş	Kilo	Maksimum Değer
1	43	103	4,89
2	42	67	4,85
3	32	50	4,84
4	63	85	4,82
5	34	97	4,83
6	64	80	4,82
7	33	73	4,86
8	36	94	4,88
9	34	87	4,86
10	26	72	4,80
11	37	68	4,85
12	30	105	4,89
13	33	99	4,87
14	31	104	4,91
15	47	85	4,89
16	31	110	>5
17	57	85	4,88
18	59	99	-----

Sistemde 2 tabanda 16 şar adet olmak üzere toplam 32 adet sensör kullanılmıştır. Çizelge 4.2 de kullanıcıların kuvvet uyguladıkları maksimum sensör sayıları gösterilmektedir.

Çizelge 4.2. Kuvvet uygulanan maksimum sensör sayısı

Sn	Ayak Numarası	Kilo	Maksimum sensör sayısı
1	43	103	14
2	38	67	12
3	36	50	12
4	42	85	14
5	42	97	14
6	39	80	12
7	40	73	13
8	43	94	14
9	42	87	13
10	42	72	14
11	41	68	13
12	44	105	15
13	43	99	14
14	44	104	16
15	42	85	13
16	43	110	16
17	43	85	13
18	39	99	---

17 katılımcıdan ikisi 16 sensöre, katılımcılardan bir tanesi 15 sensöre, 6 tanesi 14 sensöre, 5 tanesi ise 13 sensöre, 3 tanesi 12 sensöre kuvvet uygulamıştır. Çizelge 4.2.

den anlaşılacağı gibi ayak numaraları küçüldükçe etki edilen sensör sayısı azalmaktadır. Bu sorun farklı tabanlar üretilerek ortadan kaldırılabılır. Sensörlerle donatılmış taban elektronik devre kartına klemens ile bağlandığı için devre kartından kolay bir şekilde ayrılıp takılabilir. Tek bir elektronik devre kartı ve kablosuz haberleşme ünitesi farklı tabanlarla birleştirilebilir. Çizelge 4.3. de tasarlanacak taban ölçüleri yer almaktadır.

Çizelge 4.3. Taban ölçüleri

Ayak Numarası	Uzunluk(Ort.)	Taban Ölçüleri	Cinsiyet
20-24	14.5 cm	XS	K/E
25-29	17.5 cm	S	K/E
30-34	19.5 cm	M	K/E
35-39	23 cm	L	K/E
40-44	26 cm	XL	K/E
45-50	29 cm	XXL	K/E

Sağlıklı yürüme için yürüme boyunca iki ayağa eşit veya yakın kuvvet uygulanmalıdır. Çizelge 4.4. de katılımcıların yürüme sonuçları yer almaktadır.

Çizelge 4.4. Katılımcı değerleri

Sn	Yaş	Kilo	Boy	Ayak Numarası	Sol Ortalama	Sağ ortalama	Grafik
1	43	103	1,79	43	2,11	2,96	Uygun değil
2	42	67	1,58	38	2,21	2,26	Uygun
3	32	50	1,6	36	1,51	1,49	Uygun
4	63	85	1,71	42	1,57	2,07	Uygun değil
5	34	97	1,71	42	2,4	2,38	Uygun
6	64	80	1,6	39	1,03	1,06	Uygun
7	33	73	1,65	40	1,66	1,71	Uygun
8	36	94	1,85	43	2,02	2,52	Uygun değil
9	34	87	1,8	42	1,91	1,8	Uygun değil
10	26	72	1,72	42	2,04	2,17	Uygun değil
11	37	68	1,78	41	1,7	2,1	Uygun değil
12	30	105	1,83	44	2,71	2,99	Uygun değil
13	33	99	1,83	43	2,36	2,39	Uygun
14	31	104	1,73	44	2,46	2,91	Uygun değil
15	47	85	1,65	42	1,79	2,02	Uygun değil
16	31	110	1,82	43	-----	-----	-----
17	57	85	1,72	43	2,18	2,64	Uygun değil
18	59	99	1,75	39	-----	-----	-----

Katılımcıların yürüme grafikleri ve sayısal değerler incelenerek; değerlerin grafiklerle örtüştüğü gözlemlenmiştir. Yürüme sağ ortalamaları fazla olan katılımcıların

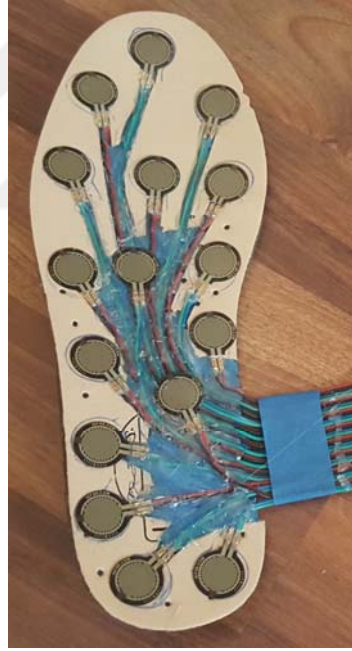
grafiklerinde de sađ ayađa ait serinin daha yksek, yrme sol ortalamaları fazla olan katılımcıların grafiklerinde de sol ayađa ait serinin daha yksek olduđu gzlemlenmiřtir. Katılımcılardan sadece 6 katılımcıda sađlıklı yrme gzlemlenmiřtir. İki katılımcıdan grafik deđerleri alınamamıřtır. Dokuz katılımcının yrme esnasında sađ ayađa daha fazla kuvvet uyguladıđı, bir katılımcının sol ayađa daha fazla kuvvet uyguladıđı gzlemlenmiřtir.



5. SONUÇLAR VE ÖNERİLER

5.1. Sonuçlar

Yürüme kinetik analizinin en büyük kısıtlarından bir tanesi normal yürüyüşü ölçememesidir. Sistem ayakkabı içerisine yerleştirilen sensörler donatılmış bir tabandan oluşmaktadır. Sistem için piyasadaki en fazla analog girişe sahip olan Arduino Mega 2560 kartı kullanılmıştır. Bu kartın bacağa sabitlenmesinden ötürü doğal yürüyüşü kısmen etkilediği gözlemlenmiştir. Katılımcıların küçük bir kısmının karşılaştığı problem ise elektronik devre kartlarının bacağın iç kısmına yerleştirilmesinden ötürü yürümede sıkıntı yaşanabileceğidir. Şekil 5.1. de gösterildiği gibi ayağın iç kısmındaki boşluğa çok fazla sensör yerleştirilmemiş ve bütün sensörlerin kabloları bu bölgeden geçirildiği için devre kartı bacağın iç kısmına yerleştirilmiştir.



Şekil 5.1. Sensör ve kabloların yerleşimi

Çalışmanın bir başka amacı dik duruş da ve yürüme boyunca ayak tabanında fazla kuvvet uygulanan alanların belirlenebilmesidir. Bu noktalar tespit edilip üretilen özel tabanlıklar ile hastanın daha sağlıklı yürümesi sağlanabilir.

Çalışma sonuçları gösteriyor ki hayatımızın büyük bir bölümünde vücudumuzu taşıyan ayaklarımıza gerekli önemi göstermediğimizdir. Bunun başlıca nedenleri Yürüme Analiz sistemlerinin yeterince yaygın ve ulaşılabilir uygun maliyetle

olmamasıdır. Bu alanda yapılacak çalışmalar bu sistemlerin yaygınlığının artmasını ve daha düşük maliyetlerle gerçekleştirilmesini sağlayacaktır.

5.2. Öneriler

Bu bölümde gelecekte yapılabilecek bazı çalışmalara yer verilmiştir. Nesnelerin İnternetinde kablosuz haberleşme için sistemde Bluetooth Modülü kullanılmıştır fakat daha uzun yürüme parkurlarında sistemi kullanabilmek ve hastanın günlük yaşamında ölçümler yapabilmek için diğer haberleşme teknolojileri kullanılabilir (Bölüm 2.2.6.). Çizelge 5.1. de Kablosuz Haberleşme Teknolojilerinin kıyaslanması yapılmıştır.

Çizelge 5.1. Haberleşme teknolojilerinin karşılaştırılması

Teknoloji	Zigbee	Gsm/Gprs	Wifi	Bluetooth
Pil ömrü (gün)	100-1000	1-7	0,5-5	1-7
Ağ boyutu (adet)	Sınırsız	1	32	7
Ağ veri genişliği (kb/s)	100-1000	64-128	11000	720
Kapsama alanı (metre)	1-100	1000	1-100	1-10
Frekans	2.4 Ghz	900/1800/1900/2100 Mhz	2,4 Ghz, 5 Ghz	2,4 Ghz
Veri transferi	250 Kbps	3-10 Mbps (4G) 600 Kbps – 10 Mbps (3G)	Max 600 Mbps	1 Mbps

KAYNAKLAR

- 3Dakademi, 2018, 3DS MAX, 3D Akademi Animasyon ve Görsel Efekt Ltd Şti. <http://3dakademi.com/3ds-max/> [Ziyaret Tarihi: 10.01.2018].
- 3Dörtgen, 2018, Ultimaker Original +, DÖRTGEN, <http://edukkan.3dortgen.com/ultimaker-original-plus> [Ziyaret Tarihi: 10.01.2018].
- 3Dprintertr, 2018, 3D Printer Nedir?, 3D Printer TR, <http://3dprintertr.com/3d-printer-nedir/> [Ziyaret Tarihi: 10.01.2018].
- Aktaş, F., Çeken, C., and Erdemli, Y. E., 2014, Biyomedikal uygulamaları için nesnelerin interneti tabanlı veri toplama ve analiz sistemi. Tıp Teknolojileri Ulusal Kongresi, Kapadokya, Nevşehir, 25-27.
- Aktaş, F., Çeken, C., and Erdemli, Y. E., 2016, Nesnelerin İnterneti Teknolojisinin Biyomedikal Alanındaki Uygulamaları. Düzce Üniversitesi Bilim ve Teknoloji Dergisi, 4(1), 37-54.
- Al-Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M., and Ayyash, M., 2015, Internet of things: A survey on enabling technologies, protocols, and applications. IEEE Communications Surveys & Tutorials, 17(4), 2347-2376.
- Algan, S., 2009, Her Yönüyle C#. Pusula Yayıncılık ve İletişim Ltd. Şti, İstanbul , 780s.
- Alpat A., 2012, Arduino Mega 2560 Nedir?, Arduino Türkiye, <http://arduinoturkiye.com/arduino-mega-2560-nedir/> [Ziyaret Tarihi: 15.01.2018].
- Alsan Ortopedi, 2016, Yürüme Analizi ve Kişiyi Özel Tabanlık, Alsan Ortopedi Ortez – Protez Kliniği, <https://www.alsanortopedi.com/yurume-analizi-kisiye-ozel-tabanlik/> [Ziyaret Tarihi: 10.02.2018].
- Angunsri, N., Ishikawa, K., Yin, M., Omi, E., Shibata, Y., Saito, T., and Itasaka, Y., 2011, Gait instability caused by vestibular disorders—Analysis by tactile sensor. Auris Nasus Larynx, 38(4), 462-468.
- Arslan, K., and Kırbaş, İ., 2016, Nesnelerin İnterneti Uygulamaları İçin Algılayıcı/Eyleyici Kablosuz Düğüm İlk örneği Geliştirme. Mehmet Akif Ersoy Üniversitesi Fen Bilimleri Enstitüsü Dergisi, (1), 35-43.
- Atzori, L., Iera, A., ad Morabito, G., 2010, The internet of things: A survey. Computer networks, Italy, 54(15), 2787-2805.
- Bakbak, S., and Kayacan, R., 2014, Hastalıkların Teşhis ve Takibinde Kullanılan Yürüyüş Analiz Sistemleri. Gazi Üniversitesi Fen Bilimleri Dergisi, Ankara, 2(4), 323-332.
- Baker, C. R., Armijo, K., Belka, S., Benhabib, M., Bhargava, V., Burkhart, N., ... and Ho, C., 2007, May, Wireless sensor networks for home health care. In Advanced

- Information Networking and Applications Workshops, 2007, AINAW'07. 21st International Conference on (Vol. 2, pp. 832-837). IEEE.
- Bamberg, S. J. M., Benbasat, A. Y., Scarborough, D. M., Krebs, D. E., and Paradiso, J. A., 2008, Gait analysis using a shoe-integrated wireless sensor system. IEEE transactions on information technology in biomedicine, 12(4), 413-423.
- Bilmeslek, 2011, Microsoft Accessin Temel Özellikleri, Avantajları – Dezavantajları, bilmeslek internet bilişim hizmetleri, <http://blog.bilmeslek.com/microsoft-accessin-temel-ozellikleri-avantajlari-%E2%80%93dezavantajlari.html> [Ziyaret Tarihi: 10.03.2018].
- Bozdoğan, Z., 2015, Nesnelerin interneti için mimari tasarımı. Yüksek Lisans Tezi, Düzce Üniversitesi, Düzce, Türkiye.
- Bozer, C., 2007, Genç erişkinlerde günlük aktivite sırasında yapılan bazı hareketlerin kinetik analizi, Doktora Tezi, Trakya Üniversitesi Sağlık Bilimleri Enstitüsü, Edirne, 10-15.
- Chip, 2007, MS Access: İlk Ders, Chip Online, https://www.chip.com.tr/bilgisayarkursu/ms-access-ilk-ders_3109.html [Ziyaret Tarihi: 10.03.2018].
- Chunming, W., and Xuemei, Z., 2010, May, Routing algorithm simulation of the patient monitoring system based on ZigBee. In Networking and Digital Society (ICNDS), Wenzhou, China, 2010 2nd International Conference on (Vol. 2, pp. 26-29). IEEE.
- Çavdar, T., and Öztürk, E., 2018, Nesnelerin interneti için yeni bir mimari tasarımı. Sakarya University Journal of Science, 22(1), 39-48.
- Dart Europe, 2018, Dart Europe, <http://www.dart-europe.eu/basic-search.php> [Ziyaret Tarihi: 12.04.2018].
- Davis, R. B., 1988, Clinical gait analysis. IEEE Engineering in Medicine and Biology Magazine, 7(3), 35-40.
- DeLisa, M. D., and Joel, A. D., 1998, Gait Analysis in the Science of Rehabilitation. US Department of Veterans Affairs: Washington, DC, USA.
- Derawi, M. O., Gafurov, D., Larsen, R., Busch, C., and Bours, P., 2010, May, Fusion of gait and fingerprint for user authentication on mobile devices. In Security and Communication Networks (IWSCN), 2010 2nd International Workshop on (pp. 1-6). IEEE.
- De Rossi, S. M. M., Lenzi, T., Vitiello, N., Donati, M., Persichetti, A., Giovacchini, F., and Carrozza, M. C., 2011, August, Development of an in-shoe pressure-sensitive device for gait analysis. In Engineering in Medicine and Biology Society, EMBC, 2011 Annual International Conference of the IEEE (pp. 5637-5640). IEEE.

- Dirençnet 1, 2018, 0.6 inch Kuvvete Duyarlı Dairesel Sensör, İNT-EL ELEKTRONİK SAN. VE TİC. LTD. ŞTİ., <https://www.direnc.net/06-kuvvete-duyarli-dairesel-sensor-pololu> [Ziyaret Tarihi: 15.01.2018].
- Dirençnet 2, 2018, HC05 Arduino Bluetooth Modül, İNT-EL ELEKTRONİK SAN. VE TİC. LTD. ŞTİ., <https://www.direnc.net/hc05-arduino-bluetooth-modul> [Ziyaret Tarihi: 15.01.2018].
- Di Stasi, S. L., Logerstedt, D., Gardinier, E. S., and Snyder-Mackler, L., 2013, Gait patterns differ between ACL-reconstructed athletes who pass return-to-sport criteria and those who fail. The American journal of sports medicine, 41(6), 1310-1318.
- Diyot.net, 2015, Arduino Nedir, Nasıl Kullanılır?, diyot.net, <http://diyot.net/arduino> [Ziyaret Tarihi: 20.03.2018].
- Dönmez, G., Ak, E., Ödek, U., Özberk, N., and Korkusuz, F., 2014, Sporda hareket analizi, Türk Ortopedi ve Travmatoloji Birliği Derneği (TOTBİD) Dergisi, 13: 369–380.
- Emre Y., 2015, Fritzing Nedir, Mühendis Beyinler, <https://www.muhendisbeyinler.net/fritzing-nedir/> [Ziyaret Tarihi: 15.12.2017].
- Esentürk S., 2014, Yürüme Siklusu Nedir?, Podoloji Türkiye, <https://blog.podolojiturkiye.org/2014/10/yurume-siklusu-nedir.html> [Ziyaret Tarihi: 10.03.2018].
- Evans, D., 2011, The internet of things: How the next evolution of the internet is changing everything. CISCO white paper, 1(2011), 1-11.
- Full Balance, 2016, Yürüme Analizi, Full Balance Ortopedik Taban, <http://fullbalanceinsole.com/yurume-analiz-sistemleri-c> [Ziyaret Tarihi: 31.03.2018].
- Gage, J. R., Schwartz, M. H., Koop, S. E., and Novacheck, T. F. (Eds.), 2009, The identification and treatment of gait problems in cerebral palsy (Vol. 4). John Wiley & Sons, p. 198–202.
- Gantz, J., and Reinsel, D., 2012, The digital universe in 2020: Big data, bigger digital shadows, and biggest growth in the far east. IDC iView: IDC Analyze the future, 2007(2012), 1-16.
- Ganz, F., Li, R., Barnaghi, P., and Harai, H., 2012, November, A resource mobility scheme for service-continuity in the Internet of Things. In Green Computing and Communications (GreenCom), 2012 IEEE International Conference On (pp. 261-264). IEEE.
- Gouwanda, D., and Senanayake, S. M. N. A., 2008, Emerging trends of body-mounted sensors in sports and human gait analysis. In 4th Kuala Lumpur International

- Conference on Biomedical Engineering 2008, Springer, Berlin, Heidelberg, (pp. 715-718).
- Gök, H., Ergin, S., and Yavuzer, G., 2002, Reliability of gait measurements in normal subjects. *Romatoloji Ve Tıbbi Rehabilitasyon Dergisi*, 13(2), 76-80.
- Gökrem, L., and Bozuklu, M., 2016, Nesnelerin İnterneti: Yapılan Çalışmalar ve Ülkemizdeki Mevcut Durum, *Gaziosmanpaşa Üniversitesi Fen Bilimleri Enstitüsü, Gaziosmanpaşa Bilimsel Araştırma Dergisi*, Tokat, (13): 47-68.
- Gül M., 2015, Bluetooth Modüller HC-05 ve HC-06, *Mühendis Beyinler*, <https://www.muhendisbeyinler.net/bluetooth-moduller-hc-05-ve-hc-06/> [Ziyaret Tarihi: 15.12.2017].
- Gül S., 2016, HC-05 ile HC-06 Bluetooth Modüllerin Haberleşmesi, *Robimek*, <https://www.robimek.com/hc-05-ile-hc-06-bluetooth-modullerin-haberlesmesi/> [Ziyaret Tarihi: 15.12.2017].
- Gülçimen, B. and Ülkü, S., 2008, İnsan Ayağı Biyomekaniğinin İncelenmesi. *Uludağ University Journal of The Faculty of Engineering, Bursa*, 13(2).
- Hatze, H., 1974, The meaning of the term 'biomechanics'. *Journal of biomechanics*, 7(2), 189-190.
- Howell, A. M., Kobayashi, T., Hayes, H. A., Foreman, K. B., and Bamberg, S. J. M., 2013, Kinetic gait analysis using a low-cost insole. *IEEE Transactions on Biomedical Engineering*, 60(12), 3284-3290.
- Islam, S. R., Kwak, D., Kabir, M. H., Hossain, M., and Kwak, K. S., 2015, The internet of things for health care: a comprehensive survey. *IEEE Access*, 3, 678-708.
- İmrenk, M., 2011, Bir Alt Ekstremitte Ortezinin Kinetik ve Kinematik Analizi, Bitirme Projesi, Dokuz Eylül Üniversitesi Mühendislik Fakültesi, İzmir.
- İzgöl K., 2015, HC-05 Bluetooth Modülü Kullanımı, *maker.robotistan.com*, <http://maker.robotistan.com/arduino-dersleri-17-hc-05-bluetooth-modulu-kullanimi/> [Ziyaret Tarihi: 15.12.2017].
- İzgöl K., 2016, HC05 ve HC06 Bluetooth Seri Modülleri Kullanımı, *maker.robotistan.com*, <http://maker.robotistan.com/hc05-hc06-bluetooth-uart-modulleri-kullanimi/> [Ziyaret Tarihi: 15.12.2017].
- Jara, A. J., Belchi, F. J., Alcolea, A. F., Santa, J., Zamora-Izquierdo, M. A., and Gómez-Skarmeta, A. F., 2010, March, A Pharmaceutical Intelligent Information System to detect allergies and Adverse Drugs Reactions based on internet of things. In *Pervasive Computing and Communications Workshops (PERCOM Workshops)*, 2010 8th IEEE International Conference, Mannheim, Germany, (pp. 809-812). IEEE.

- Kala, Y., and Bol, H., 2014, Yürüme analizi: hasta değerlendirmesi ve hazırlığı, Türk Ortopedi ve Travmatoloji Birliği Derneği (TOTBİD) Dergisi, 13: 325–330.
- Kanatlı, U., Yetkin, H., Songür, M., Öztürk, A., and Bölükbaşı, S., 2006, Yürüme Analizinin Ortopedik Uygulamaları. Türk Ortopedi ve Travmatoloji Birliği Derneği Dergisi, 5(1-2), 53-59.
- Khan, R., Khan, S. U., Zaheer, R., and Khan, S., 2012, December, Future internet: the internet of things architecture, possible applications and key challenges. In Frontiers of Information Technology (FIT), 2012 10th International Conference on (pp. 257-260). IEEE.
- Kırbaş, İ., 2013, Online Kablosuz İnkübatör İzleme ve Kontrol Sistemi Tasarımı ve Uygulaması, Doktora Tezi, Sakarya Üniversitesi Fen Bilimleri Enstitüsü, Sakarya.
- Kutup, N., 2011, Nesnelerin İnterneti; 4H Her yerden, Herkesle, Her zaman, Her nesne ile bağlantı. 16. Türkiye’de İnternet Konferansı inet-tr’11, İzmir-Türkiye, 1-5.
- Lee, D. S., Bhardwaj, S., Alasaarela, E., & Chung, W. Y., 2007, October, An ecg analysis on sensor node for reducing traffic overload in u-healthcare with wireless sensor network. In Sensors, 2007 IEEE (pp. 256-259). IEEE.
- Liu, T., Inoue, Y., and Shibata, K., 2010, A wearable ground reaction force sensor system and its application to the measurement of extrinsic gait variability. Sensors, 10(11), 10240-10255.
- López-de-Armentia, J., Casado-Mansilla, D., and López-de-Ipina, D., 2012, July, Fighting against vampire appliances through eco-aware things. In Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), 2012 Sixth International Conference on (pp. 868-873). IEEE.
- Lopes, N. V., Pinto, F., Furtado, P., & Silva, J. (2014, October). IoT architecture proposal for disabled people. In Wireless and Mobile Computing, Networking and Communications (WiMob), 2014 IEEE 10th International Conference on (pp. 152-158). IEEE.
- Luo, J., Chen, Y., Tang, K., and Luo, J., 2009, December, Remote monitoring information system and its applications based on the Internet of Things. In BioMedical Information Engineering, 2009. FBIE 2009. International Conference on Future (pp. 482-485). IEEE.
- Macedo, D., Guedes, L. A., and Silva, I., 2014, April, A dependability evaluation for Internet of Things incorporating redundancy aspects. In Networking, Sensing and Control (ICNSC), 2014 IEEE 11th International Conference on (pp. 417-422). IEEE.
- Maeda, T., Ishizuka, T., Yamaji, S., and Ohgi, Y., 2018, February, A Force Platform Free Gait Analysis. In Multidisciplinary Digital Publishing Institute Proceedings, Australia, (Vol. 2, No. 6, p. 207).

- Malan, D., Fulford-Jones, T., Welsh, M., and Moulton, S., 2004, April, Codeblue: An ad hoc sensor network infrastructure for emergency medical care. In International workshop on wearable and implantable body sensor networks (Vol. 5, pp. 12-14).
- Miller F., 2005, Cerebral Palsy, 1st edition. Wilmington, DE: Springer. p. 276.
- Milli Eğitim Bakanlığı 1, 2011, Yürüme Analizi I, Ortopedik Protez Ve Ortez, Megep, Ankara.
- Milli Eğitim Bakanlığı 2, 2011, Yürüme Analizi II, Ortopedik Protez Ve Ortez, Megep, Ankara.
- Morris, S. J., and Paradiso, J. A., 2002, October, Shoe-integrated sensor system for wireless gait analysis and real-time feedback. In Engineering in Medicine and Biology, 2002. 24th Annual Conference and the Annual Fall Meeting of the Biomedical Engineering Society EMBS/BMES Conference, 2002. Proceedings of the Second Joint (Vol. 3, pp. 2468-2469). IEEE.
- NTV, 2017, Türk insanının % 63'ü kendini sağlıklı sanıyor, NTV MSNBC, <http://arsiv.ntv.com.tr/news/185361.asp#TOP> [Ziyaret Tarihi: 10.11.2017].
- Özaras N., Yalçın S., 2001, Yürüme Analizi. 1. baskı. İstanbul: Avrupa Tıp Kitapçılık.
- Özköse, H., 2014 Bilişim Uzmanlığı Tezi, Makineler Arası Haberleşme (M2M) ve Türkiye İçin Düzenleyici Öneriler, Bilgi Teknolojileri ve İletişim Kurumu.
- Özmanevra R., 2015, Farklı Materyal Kullanılarak Yapılan Tabanlıkların Yürümenin Kinetik Ve Kinematik Özellikleri Üzerine Etkilerinin Araştırılması, Dokuz Eylül Üniversitesi Tıp Fakültesi Ortopedi ve Travmatoloji Anabilim Dalı, Uzmanlık Tezi, İzmir.
- Özvural, G., 2015, Nesnelerin İnterneti İçin Sistem Tasarımı Ve Kablosuz Kişisel Alan Ağlarında Ağ Kodlama Uygulamaları (Doctoral dissertation, İTÜ Fen Bilimleri Enstitüsü), İstanbul.
- Pang, Z., 2013, Technologies and Architectures of the Internet-of-Things (IoT) for Health and Well-being (Doctoral dissertation, KTH Royal Institute of Technology).
- Pediturk, 2018, Bilgisayarlı Ayak Analizi – Yürüme Analizi, Pediturk Ayak Sağlığı ve Bakım Merkezi, <http://pediturk.com/ayak-analizi-tabanlik/bilgisayarli-ayak-analizi/> [Ziyaret Tarihi: 10.04.2018].
- Rigid3d, 2018, Cura Dilimleme Yazılımı, RIGID3D, <http://www.rigid3d.com/cura-dilimleme-yazilimi/> [Ziyaret Tarihi: 10.03.2018].
- Robotik Sistem 1, 2018, Arduino Nedir?, Robotik Sistem, http://www.robotiksistem.com/arduino_nedir_arduino_ozellikleri.html [Ziyaret Tarihi: 31.03.2018].

- Robotik Sistem 2, 2018, Arduino / Genuino Mega 2560?, Robotik Sistem, http://www.robotiksistem.com/arduino_mega_2560_ozellikleri.html [Ziyaret Tarihi: 15.01.2018].
- Robotistan 1, 2018, 0.6 inch Kuvvete Duyarlı Kısa Saplı Dairesel Sensör - Force-Sensing Resistor - PL-2728, Robotistan Elektronik Ticaret AŞ, <https://www.robotistan.com/06-kuvvete-duyarli-kisa-sapli-dairesel-sensor-force-sensing-resistor> [Ziyaret Tarihi: 15.01.2018].
- Robotistan 2, 2018, HC05 Bluetooth – Serial Modül Kartı, Robotistan Elektronik Ticaret AŞ, <https://www.robotistan.com/hc05-bluetooth-serial-modul-karti-hc05-bluetooth-to-serial-port-module-br> [Ziyaret Tarihi: 15.01.2018].
- Robotkutusu 1, 2018, Bluetooth Modülü – HC - 05, Robotkutusu.com, <https://www.robotkutusu.com/Bluetooth-Modulu-HC-05,PR-55.html> [Ziyaret Tarihi: 15.01.2018].
- Rohokale, V. M., Prasad, N. R., and Prasad, R., 2011, February, A cooperative Internet of Things (IoT) for rural healthcare monitoring and control. In Wireless Communication, Vehicular Technology, Information Theory and Aerospace & Electronic Systems Technology (Wireless VITAE), 2011 2nd International Conference on (pp. 1-6). IEEE.
- Sarıkaya, İ., and İnan, M., 2014, Patolojik yürüme, Türk Ortopedi ve Travmatoloji Birliği Derneği (TOTBİD) Dergisi, 13:344–350.
- Sarkar, C., SN, A. U. N., Prasad, R. V., Rahim, A., Neisse, R., and Baldini, G., 2015, DIAT: A scalable distributed architecture for IoT. IEEE Internet of Things journal, 2(3), 230-239.
- Sharp, J., 2009, Adım Adım Microsoft Visual C# 2008, Arkadaş Yayın Evi, Ankara, 696s.
- Sparkfun, 2018, SIK Experiment Guide for the Arduino 101/Genuino 101 Board, SparkFun Electronics, <https://learn.sparkfun.com/tutorials/sik-experiment-guide-for-the-arduino-101-genuino-101-board/download-and-setup-the-arduino-software> [Ziyaret Tarihi: 15.01.2018].
- Sutherland, D. H., 2005, The evolution of clinical gait analysis part III–kinetics and energy assessment. Gait & Posture, San Diego, USA, 21(4), 447-461.
- Tao, W., Liu, T., Zheng, R., and Feng, H., 2012, Gait analysis using wearable sensors. Sensors, 12(2), 2255-2283.
- Taylor, S., 2013, The next generation of the Internet revolutionizing the way we work, live, play, and learn. CISCO, San Francisco, CA, USA, CISCO Point of View, 12.
- Tok, K., 2017, Hastalık teşhisi için bir yapay sinir ağları yazılımının tasarlanması ve gerçekleştirilmesi (Doctoral dissertation, Selçuk Üniversitesi Fen Bilimleri Enstitüsü), Konya.

- Tönük, E., 2014, Yürüme ve hareket analizi sistemleri: teknik ayrıntılar, Türk Ortopedi ve Travmatoloji Birliği Derneği (TOTBİD) Dergisi, 13:309–313.
- Tunca, C., Pehlivan, N., Ak, N., Arnrich, B., Salur, G., and Ersoy, C., 2017, Inertial sensor-based robust gait analysis in non-hospital settings for neurological disorders. *Sensors*, 17(4), 825.
- Turkcell, 2015, Analog Sinyal Okuma, Turkcell Geleceği yazarlar, <https://gelecegiyazarlar.turkcell.com.tr/konu/arduino/egitim/arduino-201/analog-sinyal-okuma> [Ziyaret Tarihi: 15.12.2017].
- Uğuz, S., Kılıç, B., and Şişeci, M., 2013, Akıllı Ev Otomasyonu Sistemlerinde Zigbee Tabanlı Ağ Uygulamaları, III. Elektrik Tesisat Ulusal Kongresi kapsamında 6. Kontrol Otomasyon ve Yapı Elektronik Sistemleri Sempozyumu, İzmir.
- Ulaş, S., 2015, Nesnelerin interneti ekosisteminde makineler arası özerk iletişim. Gazi Üniversitesi Bilişim Enstitüsü Yüksek Lisans Tezi, Ankara, 105s.
- Ultimaker, 2018, Ultimaker Cura Software, Ultimaker Company, <https://ultimaker.com/en/products/ultimaker-cura-software> [Ziyaret Tarihi: 10.01.2018].
- Ulusal Tez Merkezi, 2018, Yükseköğretim Kurulu Başkanlığı, <https://tez.yok.gov.tr/UlusalTezMerkezi/> [Ziyaret Tarihi:12.04.2018].
- Whittle M. (1996). Clinical gait analysis: a review, *Human Movement Science*, 15: 369–387.
- Whittle, M. W., 2007, Normal gait. *Gait analysis: an introduction*, 4th ed. Edinburgh: Butterworth-Heinemann, p. 47-100.
- Yang, L., Vyas, R., Rida, A., Pan, J., and Tentzeris, M. M., 2008, May, Wearable RFID-enabled sensor nodes for biomedical applications. In *Electronic Components and Technology Conference, USA, 2008. ECTC 2008*. 58th (pp. 2156-2159). IEEE.
- Yang, Z., Yue, Y., Yang, Y., Peng, Y., Wang, X., and Liu, W., 2011, July, Study and application on the architecture and key technologies for IOT. In *Multimedia Technology (ICMT), 2011 International Conference on* (pp. 747-751). IEEE.
- Yang, G., Xie, L., Mäntysalo, M., Zhou, X., Pang, Z., Da Xu, L., and Zheng, L. R., 2014, A health-iot platform based on the integration of intelligent packaging, unobtrusive bio-sensor, and intelligent medicine box. *IEEE transactions on industrial informatics*, 10(4), 2180-2191.
- Yavuzer, G., Öken, Ö., Elhan, A., and Stam, H. J., 2008, Repeatability of lower limb three-dimensional kinematics in patients with stroke. *Gait & posture*, 27(1), 31-35.

- Yavuzer, G., 2009, Üç Boyutlu Niceliksel Yürüme Analizi. *Acta Orthopaedica Traumatologica Turcica*, Ankara, 43(2), 94-101.
- Yavuzer, G., 2014, Yürüme Analizi ve Temel Kavramlar. *Türk Ortopedi ve Travmatoloji Birliği Derneği Dergisi*, İstanbul 13, 304-308.
- Yılmaz, N. A., Erdeo, F., Tat, A. M., and Hayriye, A. L. P., 2017, Vücut Kitle İndeksinin Ayak Taban Basınç Dağılımına Etkisi. *Adnan Menderes Üniversitesi Sağlık Bilimleri Fakültesi Dergisi*, Aydın, 1(2), 33-39.
- Zhang, M., Sun, F., and Cheng, X., 2012, October, Architecture of internet of things and its key technology integration based-on RFID. In *Computational Intelligence and Design (ISCID)*, 2012 Fifth International Symposium on (Vol. 1, pp. 294-297). IEEE.
- Zhang, Z., and Hu, X., 2013, December, ZigBee based wireless sensor networks and their use in medical and health care domain. In *Sensing Technology (ICST)*, 2013 Seventh International Conference on (pp. 756-761). IEEE.
- Zorer, G., 2001, Yürüme Analizinin Yararları, *Türkiye Spastik Çocuklar Vakfı, Spastik Çocuklar Günleri III*, 29-31.

EKLER**EK-1** Arduino IDE Kodları

```

static const uint8_t analog_pins[] =
{A0,A1,A2,A3,A4,A5,A6,A7,A8,A9,A10,A11,A12,A13,A14,A15};
int analog_deger_old[16];

void setup()
{ Serial.begin(9600); }

void loop(){

String s="";
int analog_deger[16];

for (int j = 0; j < 16; j++)
{ analogWrite(analog_pins[j],0); }

for (int i = 0; i < 16; i++)
{
pinMode(analog_pins[i],INPUT);
analog_deger[i]=analogRead(analog_pins[i]);
delay(10);
if(i==15)
{
s=s+analog_deger[i];
}
else
{ s=s+analog_deger[i]+"-"; }
}

bool gonder=0;
for(int i=0;i<16;i++)
{
if (analog_deger[i]!=analog_deger_old[i])
{
gonder=1;
break;
}
}
if(gonder==1)
{
Serial.println(s);
delay(40);
}
for(int i=0;i<16;i++)
{ analog_deger_old[i]=analog_deger[i]; }
}

```

EK-2 C# Form 2 Kodları

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using System.IO.Ports;
using System.Collections;
using System.Data.OleDb;
using System.IO;
using System.Threading.Tasks;
using System.Data.SqlClient;
namespace tez
{
    public partial class Form2 : Form
    {
        public Form2()
        {
            InitializeComponent();

            Form1 frm = new Form1();

            public static string tcno;

            OleDbConnection baglanti;
            OleDbDataAdapter da;
            OleDbCommand komut;
            DataSet ds;

            void griddoldur()
            {
                baglanti = new OleDbConnection("Provider=Microsoft.ACE.Oledb.12.0;Data
Source=analiz.accdb");
                da = new OleDbDataAdapter("SElect *from hasta", baglanti);
                ds = new DataSet();
                baglanti.Open();
                da.Fill(ds, "hasta");
                dataGridView1.DataSource = ds.Tables["hasta"];
                baglanti.Close();
            }

            private void Form2_Load(object sender, EventArgs e)
            {
                griddoldur();
            }
            private void button5_Click(object sender, EventArgs e)

```

```

    {
        baglanti = new OleDbConnection("Provider=Microsoft.ACE.Oledb.12.0;Data
Source=analiz.accdb");
        string sorgu = "Insert into hasta (tc,ad_soyad,yas,sehir) values
(@tc,@ad,@yas,@sehir)";

        komut = new OleDbCommand(sorgu, baglanti);
        komut.Parameters.AddWithValue("@tc", tc.Text);
        komut.Parameters.AddWithValue("@ad", ad.Text);
        komut.Parameters.AddWithValue("@yas", yas.SelectedItem.ToString());
        komut.Parameters.AddWithValue("@sehir", sehir.SelectedItem.ToString());
        baglanti.Open();
        komut.ExecuteNonQuery();
        baglanti.Close();
        griddoldur();
    }

private void button6_Click(object sender, EventArgs e)
{
    ad.Text = "";
    tc.Text = "";
    yas.SelectedIndex = 0;
    sehir.SelectedIndex = 0;
}

private void button7_Click(object sender, EventArgs e)
{
    baglanti = new OleDbConnection("Provider=Microsoft.ACE.Oledb.12.0;Data
Source=analiz.accdb");
    da = new OleDbDataAdapter("SElect *from hasta where tc like " + tc.Text +
"%%", baglanti);
    ds = new DataSet();
    baglanti.Open();
    da.Fill(ds, "hasta");
    dataGridView1.DataSource = ds.Tables["hasta"];
    baglanti.Close();
}

private void button1_Click(object sender, EventArgs e)
{
    Form2.tcnno= tc.Text;
    this.Hide();
    frm.ShowDialog();
    this.Close();
}
}
}

```

EK-3 C# Form 1 Kodları

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using System.IO.Ports;
using System.Collections;
using System.Data.OleDb;
using System.IO;
using System.Threading.Tasks;
using System.Threading;
using System.Windows.Forms.DataVisualization.Charting;
using System.Drawing.Imaging;

namespace tez
{
    public partial class Form1 : Form
    {
        string[] ports = SerialPort.GetPortNames(); //Port Numaralarını ports isimli diziye
        atıyoruz.

        OleDbConnection baglanti;
        OleDbDataAdapter da;
        OleDbCommand komut;
        DataSet ds;

        int sl = 0;
        int sg = 0;
        double cvr = 0.0048875855327468d;
        double [] a = {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0};
        double [] b = {0,0,0,0,0,0,0,0,0,0,0,0,0,0,0};
        bool abool=false;
        bool bbool=false;

        bool xbool = false;
        bool ybool = false;

        double rt = 0;
        Boolean i = false;

        string oku;
        string oku2;

```

```

string[] veri;
string[] veri2;

public Form1()
{
    InitializeComponent();
    Control.CheckForIllegalCrossThreadCalls = false;
}

private void Form1_Load(object sender, EventArgs e)
{
    this.Text = Form2.teno;
    sifirla.Enabled = false;
    durdur.Enabled = false;
    foreach (string port in ports)
    {
        comboBox1.Items.Add(port); // Port isimlerini combobox1'de gösteriyoruz.
        comboBox1.SelectedIndex = 0;
    }

    foreach (string port2 in ports)
    {
        comboBox3.Items.Add(port2); // Port isimlerini combobox1'de gösteriyoruz.
        comboBox3.SelectedIndex = 0;
    }
    comboBox2.Items.Add("2400"); // Baudrate'leri kendimiz combobox2'ye
giriyoruz.
    comboBox2.Items.Add("4800");
    comboBox2.Items.Add("9600");
    comboBox2.Items.Add("19200");
    comboBox2.Items.Add("115200");
    comboBox2.Items.Add("460800");
    comboBox2.SelectedIndex = 2;

    label3.Text = "Bağlantı Kapalı"; //Bu esnada bağlantı yok.
}

private void timer3_Tick(object sender, EventArgs e)
{
    rt = rt + 1;
}

private void Form1_FormClosed(object sender, FormClosedEventArgs e)
{
    if (serialPort1.IsOpen == true || serialPort2.IsOpen == true)
    {

```



```

        serialPort1.Close();
        serialPort2.Close();
        this.Close();
        Application.Exit();
    }
}

void baglan()
{
    if (serialPort1.IsOpen == false)
    {
        if (comboBox1.Text == "")
            return;
        if (comboBox3.Text == "")
            return;
        serialPort1.PortName = comboBox1.Text;
        serialPort2.PortName = comboBox3.Text;
        serialPort1.BaudRate = Convert.ToInt32(comboBox2.Text);
        serialPort2.BaudRate = Convert.ToInt32(comboBox2.Text);
        try
        {
            serialPort1.Open();
            serialPort2.Open();
            label3.ForeColor = Color.Green;
            label3.Text = "Bağlantı Açık";

        }
        catch (Exception hata)
        {
            MessageBox.Show("Hata:" + hata.Message);
        }
    }
    else
    {
        label3.Text = "Bağlantı kuruldu !!!";
    }
}

private void button1_Click(object sender, EventArgs e) // bağlan butonu
{
    baglan();
}

private void button2_Click(object sender, EventArgs e)

```

```

{
    //BAĞLANTIYI KES BUTONU

    if (serialPort1.IsOpen == true || serialPort2.IsOpen == true)
    {
        serialPort1.Close();
        serialPort2.Close();
        label3.ForeColor = Color.Red;
        label3.Text = "Bağlantı Kapalı";
    }
}

```

```

// -----dik duruş veri al-----

```

```

double solokuort = 0;
int solr = 0;
int sagr = 0;
void soloku()
{
    solr += 1;
    if (solr <= 50)
    {
        abool = true;
        oku = serialPort1.ReadLine();
        int x = 16;
        if (oku.IndexOf('-') > -1)
        {
            veri = oku.Split('-');
            for (int i = 0; i < 16; i++)
            {
                try
                {
                    a[i] = Convert.ToDouble(veri[i]);
                }
                catch (Exception e)
                {
                    a[i] = 0;
                }

                a[i] = a[i] * cvr;
                a[i] = Math.Round(a[i], 2);
                solokuort = solokuort + a[i];
            }

            solokuort = solokuort / 16;
            label16.Text = Math.Round(solokuort, 2).ToString();
            serialPort1.DiscardInBuffer();

```

```
}
```

```
listBox1.Items.Add(a[0].ToString() + "-" + a[1].ToString() + "-" +
a[2].ToString() + "-" + a[3].ToString() + "-" + a[4].ToString() + "-" + a[5].ToString() +
 "-" + a[6].ToString() + "-" + a[7].ToString() + "-" + a[8].ToString() + "-" +
a[9].ToString() + "-" + a[10].ToString() + "-" + a[11].ToString() + "-" +
a[12].ToString() + "-" + a[13].ToString() + "-" + a[14].ToString() + "-" +
a[15].ToString());
```

```
abool = false;
```

```
solokuort = 0;
```

```
}
```

```
else
```

```
{ serialPort1.DataReceived -= sensport_DataReceived; }
```

```
}
```

```
double sagokuort = 0;
```

```
void sagoku()
```

```
{
```

```
sagr += 1;
```

```
if (sagr <= 50)
```

```
{
```

```
bbool = true;
```

```
oku2 = serialPort2.ReadLine();
```

```
int y = 16;
```

```
if (oku2.IndexOf('-') > -1)
```

```
{
```

```
veri2 = oku2.Split('-');
```

```
for (int j = 0; j < 16; j++)
```

```
{
```

```
try
```

```
{
```

```
b[j] = Convert.ToDouble(veri2[j]);
```

```
}
```

```
catch (Exception e)
```

```
{
```

```
b[j] = 0;
```

```
}
```

```
b[j] = b[j] * cvr;
```

```
b[j] = Math.Round(b[j], 2);
```

```
sagokuort = sagokuort + b[j];
```

```
}
```

```
sagokuort = sagokuort / 16;
```

```
label29.Text = Math.Round(sagokuort, 2).ToString();
```

```
serialPort2.DiscardInBuffer();
```

```

    }
    listBox2.Items.Add(b[0].ToString() + "-" + b[1].ToString() + "-" +
b[2].ToString() + "-" + b[3].ToString() + "-" + b[4].ToString() + "-" + b[5].ToString() +
 "-" + b[6].ToString() + "-" + b[7].ToString() + "-" + b[8].ToString() + "-" +
b[9].ToString() + "-" + b[10].ToString() + "-" + b[11].ToString() + "-" +
b[12].ToString() + "-" + b[13].ToString() + "-" + b[14].ToString() + "-" +
b[15].ToString());

```

```

    bbool = false;
    sagokuort = 0;

```

```

    }
    else
    { serialPort2.DataReceived -= sensport_DataReceived2; }
    }
    Thread paralel1;
    Thread paralel2;

    void sensport_DataReceived(object sender, SerialDataReceivedEventArgs e)
    {
        if (abool == false)
        {
            sl++;
            paralel1 = new Thread(new ThreadStart(soloku));
            paralel1.Start();
            paralel1.Join();
        }
    }

```

```

    void sensport_DataReceived2(object sender, SerialDataReceivedEventArgs e)
    {
        if (bbool == false)
        {
            sg++;
            paralel2 = new Thread(new ThreadStart(sagoku));
            paralel2.Start();
            paralel2.Join();
        }
    }

```

```

    string boya(double deger)
    {
        if (deger <= 0.5)
        {
            //ovalShape15.BackColor = Color.LightBlue;

```

```

        return "#C7CCE9";
    }
    else if (deger > 0.5d & deger <= 1d)
    {
        // ovalShape15.BackColor = Color.Blue;
        return "#687BF2";
    }
    else if (deger > 1d & deger <= 1.5d)
    {
        // ovalShape15.BackColor = Color.Aqua;
        return "#0AD3F6";
    }

    else if (deger > 1.5d & deger <= 2d)
    {
        // ovalShape15.BackColor = Color.Green;
        return "#00FF00";
    }
    else if (deger > 2d & deger <= 2.5d)
    {
        // ovalShape15.BackColor = Color.Yellow;
        return "#FFFF00";
    }

    else if (deger > 2.5d & deger <= 3d)
    {
        // ovalShape15.BackColor = Color.Gold;
        return "#FFD700";
    }
    else if (deger > 3d & deger <= 3.5d)
    {
        // ovalShape15.BackColor = Color.Orange;
        return "#FA7508";
    }
    else if (deger > 3.5d & deger <= 4d)
    {
        // ovalShape15.BackColor = Color.OrangeRed;
        return "#FF4500";
    }

    else if (deger > 4d & deger <= 4.5d)
    {
        // ovalShape15.BackColor = Color.Red;
        return "#FF0000";
    }
    else
    {
        // ovalShape15.BackColor = Color.DarkRed;
        return "#8B0000";
    }
}

```

```
}
```

```
int kontrol;
```

```
private void button8_Click(object sender, EventArgs e)
{
```

```
    baglan();
```

```
    serialPort1.DataReceived += sensport_DataReceived;
    serialPort2.DataReceived += sensport_DataReceived2;
```

```
}
```

```
int dur;
```

```
private void button5_Click(object sender, EventArgs e)
{
```

```
    if (serialPort1.IsOpen == true)
    {
```

```
        serialPort1.DiscardInBuffer();
```

```
        serialPort2.DiscardInBuffer();
```

```
        serialPort1.Close();
```

```
        serialPort2.Close();
```

```
        dur = 1;
```

```
        label3.ForeColor = Color.Red;
```

```
        label3.Text = "Bağlantı Kapalı";
```

```
        ovalShape22.BackColor =
```

```
        System.Drawing.ColorTranslator.FromHtml(boya(a[15]));
```

```
        ovalShape29.BackColor =
```

```
        System.Drawing.ColorTranslator.FromHtml(boya(a[14]));
```

```
        ovalShape28.BackColor =
```

```
        System.Drawing.ColorTranslator.FromHtml(boya(a[13]));
```

```
        ovalShape25.BackColor =
```

```
        System.Drawing.ColorTranslator.FromHtml(boya(a[12]));
```

```
        ovalShape26.BackColor =
```

```
        System.Drawing.ColorTranslator.FromHtml(boya(a[11]));
```

```
        ovalShape27.BackColor =
```

```
        System.Drawing.ColorTranslator.FromHtml(boya(a[10]));
```

```
        ovalShape24.BackColor =
```

```
        System.Drawing.ColorTranslator.FromHtml(boya(a[9]));
```

```
        ovalShape31.BackColor =
```

```
        System.Drawing.ColorTranslator.FromHtml(boya(a[8]));
```

```
        ovalShape32.BackColor =
```

```
        System.Drawing.ColorTranslator.FromHtml(boya(a[7]));
```

```

        ovalShape23.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(a[6]));
        ovalShape30.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(a[5]));
        ovalShape21.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(a[4]));
        ovalShape20.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(a[3]));
        ovalShape19.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(a[2]));
        ovalShape18.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(a[1]));
        ovalShape17.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(a[0]));

        ovalShape1.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(b[15]));
        ovalShape2.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(b[14]));
        ovalShape3.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(b[13]));
        ovalShape4.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(b[12]));
        ovalShape5.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(b[11]));
        ovalShape6.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(b[10]));
        ovalShape7.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(b[9]));
        ovalShape8.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(b[8]));
        ovalShape9.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(b[7]));
        ovalShape10.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(b[6]));
        ovalShape11.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(b[5]));
        ovalShape12.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(b[4]));
        ovalShape13.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(b[3]));
        ovalShape14.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(b[2]));
        ovalShape15.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(b[1]));
        ovalShape16.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(b[0]));

    }
}

```

//-----yürüyüş anında veri alma-----

```
private void timer2_Tick_1(object sender, EventArgs e)
{
    if (int.Parse(saniye.Text) < 9)
    {
        saniye.Text = "0" + (int.Parse(saniye.Text) + 1).ToString();
    }
    else
    {
        saniye.Text = (int.Parse(saniye.Text) + 1).ToString();
    }
    if (int.Parse(saniye.Text) == 60)
    {
        saniye.Text = "00";
        if (int.Parse(dakika.Text) < 9)
        {
            dakika.Text = "0" + (int.Parse(dakika.Text) + 1).ToString();
        }
        else
        {
            dakika.Text = (int.Parse(dakika.Text) + 1).ToString();
        }
    }
}
```

```
string sloku;
string[] solveri;
string sgoku;
string[] sagveri;
```

```
int solkontrol;
int sagkontrol;
double slort = 0;
double sgort = 0;
```

```
void s1DataReceived(object sender, SerialDataReceivedEventArgs e)
{
    if (xbool == false)
    {
        Thread paralel1 = new Thread(new ThreadStart(solyuru));
        paralel1.Start();
        paralel1.Join();
    }
}
```



```

    }
    int solyr = 0;
    int sagyr = 0;
    void solyuru()
    {
        solyr += 1;
        if (solyr <= 300)
        {
            xbool = true;
            sloku = serialPort1.ReadLine();
            solkontrol = 1;
            if (sloku.IndexOf('-') > -1)
            {

                solveri = sloku.Split('-');

                for (int si = 0; si < 16; si++)
                {
                    try
                    {
                        a[si] = Convert.ToDouble(solveri[si]);
                    }
                    catch (Exception e)
                    {
                        a[si] = 0;
                    }
                }

                a[si] = a[si] * cvr; // a[i] = a[i] * (5 / 1023);
                a[si] = Math.Round(a[si], 2);

            }
        }

        slort = (a[0] + a[1] + a[2] + a[3] + a[4] + a[5] + a[6] + a[7] + a[8] + a[9] + a[10]
+ a[11] + a[12] + a[13] + a[14] + a[15]) / 16;
        slort = Math.Round(slort, 2);
        listBox3.Items.Add(a[0].ToString() + "-" + a[1].ToString() + "-" +
a[2].ToString() + "-" + a[3].ToString() + "-" + a[4].ToString() + "-" + a[5].ToString() +
 "-" + a[6].ToString() + "-" + a[7].ToString() + "-" + a[8].ToString() + "-" +
a[9].ToString() + "-" + a[10].ToString() + "-" + a[11].ToString() + "-" +
a[12].ToString() + "-" + a[13].ToString() + "-" + a[14].ToString() + "-" +
a[15].ToString() + "-" + slort.ToString());
        solkontrol = 0;
        this.chart1.Series["sol ayak"].Points.AddXY(DateTime.Now, slort);

        xbool = false;

    }
    else
    { serialPort1.DataReceived -= s1DataReceived; }

```

```

    }

    void s2DataReceived(object sender, SerialDataReceivedEventArgs e)
    {
        if (ybool == false)
        {

            Thread paralel2 = new Thread(new ThreadStart(sagyuru));
            paralel2.Start();
            paralel2.Join();
        }
    }

    void sagyuru()
    {
        sagyr+=1;
        if(sagyr<=300)
        {
            ybool = true;
            sgoku = serialPort2.ReadLine();
            sagkontrol = 1;
            if (sgoku.IndexOf('-') > -1)
            {

                sagveri = sgoku.Split('-');

                for (int gi = 0; gi < 16; gi++)
                {
                    try
                    {
                        b[gi] = Convert.ToDouble(sagveri[gi]);

                    }
                    catch (Exception e)
                    {
                        b[gi] = 0;
                    }

                    b[gi] = b[gi] * cvr; // a[i] = a[i] * (5 / 1023);
                    b[gi] = Math.Round(b[gi], 2);

                }
            }

            sgort = (b[0] + b[1] + b[2] + b[3] + b[4] + b[5] + b[6] + b[7] + b[8] + b[9] +
            b[10] + b[11] + b[12] + b[13] + b[14] + b[15]) / 16;
            sgort = Math.Round(sgort,2);

```

```

        listBox4.Items.Add(b[0].ToString() + "-" + b[1].ToString() + "-" +
b[2].ToString() + "-" + b[3].ToString() + "-" + b[4].ToString() + "-" + b[5].ToString() +
 "-" + b[6].ToString() + "-" + b[7].ToString() + "-" + b[8].ToString() + "-" +
b[9].ToString() + "-" + b[10].ToString() + "-" + b[11].ToString() + "-" +
b[12].ToString() + "-" + b[13].ToString() + "-" + b[14].ToString() + "-" +
b[15].ToString() + "-" + sgort.ToString());

```

```

        sagkontrol = 0;
        this.chart1.Series["sağ ayak"].Points.AddXY(DateTime.Now, sgort);

```

```

        ybool = false;

```

```

    }
    else
    { serialPort2.DataReceived -= s2DataReceived; }
}

```

```

private void baslat_Click(object sender, EventArgs e)
{
    timer2.Start();
    timer2.Interval = 1000;
    sifirla.Enabled = false;
    baslat.Enabled = false;
    durdur.Enabled = true;
}

```

```

        baglan();

```

```

        timer3.Start();

```

```

        serialPort1.DataReceived += s1DataReceived;
        serialPort2.DataReceived += s2DataReceived;
    }
}

```

```

private void durdur_Click(object sender, EventArgs e)
{
    timer2.Stop();
    timer3.Stop();
    label3.ForeColor = Color.Red;
    label3.Text = "Bağlantı Kapalı";
    sifirla.Enabled = true;
    baslat.Text = "sürdür";
    baslat.Enabled = true;
    durdur.Enabled = false;
}

```

```

        if (solkontrol == 0 && sagkontrol == 0)
        {

```

```

        if (serialPort1.IsOpen == true && serialPort2.IsOpen == true)
        {
            serialPort1.Close();
            serialPort2.Close();
            analiz();
        }
    }

}

private void sifirla_Click(object sender, EventArgs e)
{
    saniye.Text = "00";
    dakika.Text = "00";
    baslat.Enabled = true;
    baslat.Text = "başlat";
    durdur.Enabled = false;
    sifirla.Enabled = false;
}

string [] solanl;
string [] saganl;
string solort;
string sagort;
double sollort;
double saggort;
double sola = 0;
double saga = 0;

void analiz()
{
    for (int i = 0; i < listBox3.Items.Count; i++)
    {
        solort = listBox3.Items[i].ToString();
        solanl = solort.Split('-');
        sollort = Convert.ToDouble(solanl[16]);

        sola = sola + sollort;
    }

    for (int i = 0; i < listBox4.Items.Count; i++)
    {
        sagort = listBox4.Items[i].ToString();
        saganl = sagort.Split('-');
        saggort = Convert.ToDouble(saganl[16]);
        saga = saga + saggort;
    }
}

```

```

sola = sola / listBox3.Items.Count;
saga = saga / listBox4.Items.Count;

label47.Text = Math.Round(sola, 2).ToString();
label49.Text = Math.Round(saga, 2).ToString();
}

```

```

string[,] matrissol;
string solgecici;
double[] solortalama = { 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 };
double[] solmax = { 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 };
double[] solmin = { 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 };
double[] sll = { 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 };

string[,] matrissag;
string saggecici;
double[] sagortalama = { 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 };
double[] sagmax = { 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 };
double[] sagmin = { 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 };
double[] sgg = { 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 };

void degerler()
{
    matrissol = new string[listBox3.Items.Count, 16];
    matrissag = new string[listBox4.Items.Count, 16];

    for (int i = 0; i < listBox3.Items.Count; i++)
    {
        solgecici = listBox3.Items[i].ToString();

        for (int j = 0; j < 16; j++)
        {
            matrissol[i,j]=solgecici.Split('-')[j];
            solortalama[j] = solortalama[j] + Convert.ToDouble(matrissol[i, j]);

            if (Convert.ToDouble(matrissol[i, j]) > solmax[j])
                solmax[j] = Convert.ToDouble(matrissol[i, j]);

            if (Convert.ToDouble(matrissol[i, j]) < solmin[j])
                solmin[j] = Convert.ToDouble(matrissol[i, j]);

        }
    }

    for (int i = 0; i < listBox4.Items.Count; i++)
    {
        saggecici = listBox4.Items[i].ToString();
    }
}

```

```

for (int j = 0; j < 16; j++)
{
    matrissag[i, j] = saggecici.Split('-')[j];
    sagortalama[j] = sagortalama[j] +
        Convert.ToDouble(matrissag[i, j]);

    if (Convert.ToDouble(matrissag[i, j]) > sagmax[j])
        sagmax[j] = Convert.ToDouble(matrissag[i, j]);

    if (Convert.ToDouble(matrissag[i, j]) < sagmin[j])
        sagmin[j] = Convert.ToDouble(matrissag[i, j]);
}
}

for (int k = 0; k < 16; k++)
{
    solortalama[k] = solortalama[k] / listBox3.Items.Count;
    sagortalama[k] = sagortalama[k] / listBox4.Items.Count;
}

if (comboBox4.SelectedIndex == 0)
{
    for (int i = 0; i < 16; i++)
    {
        sll[i] = Math.Round(solortalama[i], 2);
        sgg[i] = Math.Round(sagortalama[i], 2);
    }
}
else if (comboBox4.SelectedIndex == 1)
{
    for (int i = 0; i < 16; i++)
    {
        sll[i] = solmax[i];
        sgg[i] = sagmax[i];
    }
}

}

label10.Text = sll[0].ToString();
label11.Text = sll[1].ToString();
label12.Text = sll[2].ToString();
label14.Text = sll[3].ToString();
label17.Text = sll[4].ToString();
label18.Text = sll[5].ToString();
label19.Text = sll[6].ToString();

```

```

label20.Text = sll[7].ToString();
label21.Text = sll[8].ToString();
label22.Text = sll[9].ToString();
label23.Text = sll[10].ToString();
label24.Text = sll[11].ToString();
label25.Text = sll[12].ToString();
label26.Text = sll[13].ToString();
label27.Text = sll[14].ToString();
label28.Text = sll[15].ToString();

```

```

label30.Text = sgg[0].ToString();
label31.Text = sgg[1].ToString();
label32.Text = sgg[2].ToString();
label33.Text = sgg[3].ToString();
label34.Text = sgg[4].ToString();
label35.Text = sgg[5].ToString();
label36.Text = sgg[6].ToString();
label37.Text = sgg[7].ToString();
label38.Text = sgg[8].ToString();
label39.Text = sgg[9].ToString();
label40.Text = sgg[10].ToString();
label41.Text = sgg[11].ToString();
label42.Text = sgg[12].ToString();
label43.Text = sgg[13].ToString();
label44.Text = sgg[14].ToString();
label45.Text = sgg[15].ToString();

```

```

ovalShape49.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(sll[0]));
ovalShape50.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(sll[1]));
ovalShape51.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(sll[2]));
ovalShape52.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(sll[3]));
ovalShape53.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(sll[4]));
ovalShape62.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(sll[5]));
ovalShape55.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(sll[6]));
ovalShape64.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(sll[7]));
ovalShape63.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(sll[8]));
ovalShape56.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(sll[9]));
ovalShape59.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(sll[10]));

```

```

        ovalShape58.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(sll[11]));
        ovalShape57.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(sll[12]));
        ovalShape60.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(sll[13]));
        ovalShape61.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(sll[14]));
        ovalShape54.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(sll[15]));

```

```

        ovalShape33.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(sgg[0]));
        ovalShape34.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(sgg[1]));
        ovalShape35.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(sgg[2]));
        ovalShape36.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(sgg[3]));
        ovalShape37.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(sgg[4]));
        ovalShape38.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(sgg[5]));
        ovalShape39.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(sgg[6]));
        ovalShape40.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(sgg[7]));
        ovalShape41.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(sgg[8]));
        ovalShape42.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(sgg[9]));
        ovalShape43.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(sgg[10]));
        ovalShape44.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(sgg[11]));
        ovalShape45.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(sgg[12]));
        ovalShape46.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(sgg[13]));
        ovalShape47.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(sgg[14]));
        ovalShape48.BackColor =
System.Drawing.ColorTranslator.FromHtml(boya(sgg[15]));

```

```

    }

```

```

private void comboBox4_SelectedIndexChanged(object sender, EventArgs e)
{
    degerler();
}

```



```

private void button7_Click(object sender, EventArgs e)
{
    baglanti = new OleDbConnection("Provider=Microsoft.ACE.Oledb.12.0;Data
Source=analiz.accdb");
    string sorgu = "Insert into veri
(tc,durus_sol_ortalama,durus_sag_ortalama,yurume_sol_ortalama,yurume_sag_ortalam
a) values (@tc,@d_sol_ort,@d_sag_ort,@y_sol_ort,@y_sag_ort)";

    komut = new OleDbCommand(sorgu, baglanti);
    komut.Parameters.AddWithValue("@tc", Form2.tcno);
    komut.Parameters.AddWithValue("@d_sol_ort", label16.Text);
    komut.Parameters.AddWithValue("@d_sag_ort", label29.Text);
    komut.Parameters.AddWithValue("@y_sol_ort", label47.Text);
    komut.Parameters.AddWithValue("@y_sag_ort", label49.Text);

    baglanti.Open();
    komut.ExecuteNonQuery();
    baglanti.Close();
}

private void button6_Click(object sender, EventArgs e)
{
    try
    {
        string kayıtyeri = @"\" + Form2.tcno + "\";
        this.chart1.SaveImage((kayıtyeri+".png"), ChartImageFormat.Png); // bu
komut o andaki resmi çekip kaydeder

        MessageBox.Show("Grafik kaydedildi ");
    }
    catch (Exception ex3)
    {
        MessageBox.Show(ex3.Message, "Hata Mesajı");
    }
}

private void button9_Click(object sender, EventArgs e)
{
    int ekran_x = Screen.GetBounds(new Point(0, 0)).Width;
    int ekran_y = Screen.GetBounds(new Point(0, 0)).Height;
    Bitmap screenShot = new Bitmap(ekran_x, ekran_y);
    Graphics grp = Graphics.FromImage((Image)screenShot);
    grp.CopyFromScreen(0, 0, 0, 0, new Size(ekran_x, ekran_y));
    screenShot.Save(Form2.tcno+"-dik-durus.png", ImageFormat.Png);
    MessageBox.Show("işlem tamam ");
}
}
}

```