



**MARMARA
ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**



**DÜŞÜK AYAK HASTALIĞINA
YÖNELİK ELEKTRİK STİMULATÖRÜ
GELİŞTİRİLMESİ**

MEHMET ALINBAY

YÜKSEK LİSANS TEZİ
Mekatronik Anabilim Dalı

DANIŞMAN
Doç. Dr. Haluk KÜÇÜK

İSTANBUL, 2013



**MARMARA
ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**



**DÜŞÜK AYAK HASTALIĞINA
YÖNELİK FONKSİYONEL ELEKTRİK
STİMULATÖRÜ GELİŞTİRİLMESİ**

MEHMET ALINBAY
(526210010)

YÜKSEK LİSANS TEZİ
Mekatronik Anabilim Dalı

DANIŞMAN
Doç. Dr. Haluk KÜÇÜK

İSTANBUL, 2013

TEŞEKKÜR

Bu çalışmanın başlamasına vesile olan ve tez süresi boyunca sabır ve desteğini esirgemeyen, eleştirileri ile tezime yön veren danışmanım Sayın Haluk Küçük'e,

Beni yetiştiren, hayatım boyunca bana destek olan değerli Anne ve Babama,

Tez çalışmamın her aşamasında fikir ve önerileri ile yardımlarını esirgemeyen emeği geçen tüm arkadaşlarıma teşekkürlerimi sunarım.

Haziran, 2013

Mehmet ALINBAY

İÇİNDEKİLER

	SAYFA
TEŞEKKÜR.....	2
İÇİNDEKİLER	iii
ÖZET.....	v
ABSTRACT	vii
SEMBOLLER.....	ix
KISALTMALAR	xi
ŞEKİL LİSTESİ.....	xiii
1.GİRİŞ VE AMAÇ	1
1.1. Düşük Ayak Hastalığı	1
1.2. Fonksiyonel Elektrik Stimulatorleri	3
1.3. Amaç	6
2. GENEL BİLGİLER	7
2.1. Yürüme	7
2.1.1. Yürümenin Fonksiyonel Görevleri	7
2.1.2. Yürümenin Fazları	8
2.2. Yürüme Fazlarının Tespiti.....	9
2.2.1. Yürüme Fazlarının Tespiti: Sensörler.....	9
2.2.1. Yürüme Fazlarının Tespit Metodları	11
3. DÜŞÜK AYAK STİMULATÖRÜ SİSTEMİ	14
3.1. Sisteme Genel Bakış.....	14
3.2. Gömülü Sistemler.....	15
3.2.1. PIC Mikrodenetleyicileri ve Genel Özellikleri	17
3.2.2. PIC12F683 ve Genel Özellikleri	17

3.2.3. Programlama Dilleri ve Mikrodenetleyiciler	19
3.2.3.1 C programlama dili	20
3.2.3.2. Mikro-C derleyicisi	21
3.3. İvme Sensörleri.....	22
3.4. Stimulasyon Jeneratörü	25
3.5. Stimulator Programı Algoritması	30
3.6. Şarj Cihazı	36
3.7. Bilgisayar Programı.....	38
3.7.1. Haberleşme Sistemi	39
3.7.2. Bilgisayar Programı Arayüzü	40
4. SONUÇLAR VE DEĞERLENDİRME	43
KAYNAKLAR	45
EKLER.....	50
EK-A	50
STIMULATOR PROGRAMI.....	50
EK- B	68
BİLGİSAYAR PROGRAMI	68
ÖZGEÇMİŞ	74

ÖZET

Düşük ayak hastalığının elektrik stimülasyonu ile tedavisi yaklaşık 50 yıldır kullanılmaktadır. Düşük ayak stimulatörleri kaybedilmiş olan motor fonksiyonları ve yürümeyi geliştirmek için kullanılmaktadır. Yakın zamanda gelişen teknoloji ile stimulatörler ticarileşmiştir.

Bu çalışmada, peroneal siniri elektrik stimülasyonu ile uyararak dorsifleksiyon hareketinin sağlanması amaçlanmıştır. Çalışmanın uygulama kısmında pil ile beslenen taşınabilir elektronik devre tasarlanmıştır. Temel olarak, elektronik devre bir sensör, bir mikroişlemci ve güç katından oluşmaktadır. Sensörden alınan bilgiler ile bacak konum ve açı bilgisi bulunmakta ve yürüyüş paternine uygun olarak peroneal siniri stimule ederek dorsifleksiyonu sağlamaktadır. Ayrıca bir bilgisayar programı yardımı ile devrenin stimülasyon parametreleri ayarlanabilmektedir.

Tasarlanan düşük ayak stimulatörü ile yapılan uygulamalar hastalar üzerinde başarılı sonuçlar vermektedir. Sistem, mevcut stimulatörlerden daha hafiftir. Ayrıca sistem herhangi bir dış sensöre ihtiyaç duymadan kendi başına çalışabilir ve herhangi bir cerrahi operasyona gerek duymadan uygulanabilir.

Haziran, 2013

Mehmet Alınbay

ABSTRACT

Drop foot disease treatment with electrical stimulation is used for about 50 years. Drop foot stimulator is used to improve lost motor function and walking. Stimulator with a recently developed technology is commercialized.

This study is aimed to ensure the movement of dorsiflexion stimulating the common peroneal nerve with electrical stimulation. Battery-powered portable electronic circuit used in the application part of the study. Basically, the electronic circuit is composed of a sensor, a microprocessor and power stage. With data from the sensor, microprocessor unit obtains the knowledge of location and angle of the leg, evaluates data and provides dorsiflexion by stimulating common peroneal nerve in accordance with the walking pattern. Also stimulation parameters of the circuit can be adjusted with the help of a computer program.

Applications with designed drop foot stimulator gives successful results on patients. The system is lighter than the existing stimulator. Also, the system can operate on its own without the need of any external sensor and can be applied without requiring any surgical operation.

June, 2013

Mehmet Alınbay

SEMBOLLER

°	: Celcius
Hz	: Hertz
V	: gerilim şiddet birimi
A	: akım şiddet birimi
g	: yerçekimi ivmesi

KISALTMALAR

DFS	: Baskılı Devre Kartı
FES	: Fonksiyonel Elektrik Stimulatörü
TENS	: Transcutaneous Electrical Nerve Stimulation
CP	: Cerebral Palsy
AFO	: Grafiksel Programlama Dili
WHO	: National Instruments firması
PCI	: Fizyolojik Maliyet Endeksi
ANSI	: Otomatik Denetleme
LED	: Işık Yayan Diyod
AC	: Alternatif Akım
DC	: Doğru Akım
ANSI	: American National Standards Institute
ICD	: In-Circuit Debug
SPI	: Serial Peripheral Interface
I2C	: Inter-Integrated Circuit
SOIC	: Small-outline integrated Circuit
PDIP	: Plastic Dual Inline Package
QFN	: Quad Flat No Leads
PWM	: Pulse Width Modulation
RAM	: Random Access Memory
EEPROM	: Electrically Erasable Programmable Read Only Memory
CMOS	: Complementary Metal Oxide Semiconductor
ROM	: Read Only Memory

ŞEKİL LİSTESİ

	SAYFA
Şekil 1.1 Dorsifleksiyon hareketi	2
Şekil 1.2 AFO Örneği	2
Şekil 1.3 Liberson DFS [5]	5
Şekil 2.1 Yürüme Fazları [22].....	9
Şekil 2.2 Temel Yürüme	12
Şekil 2.3 Yürümenin Sonlu Durum Analizi [22]	12
Şekil 3.1 DFS Sistemi Blok Şema	14
Şekil 3.2 Tasarlanan DFS ve Şarj Cihazı	15
Şekil 3.3 Mikrodenetleyici	16
Şekil 3.4 Çeşitli PIC mikrodenetleyicileri	17
Şekil 3.5 PIC12F683 Bacak Bağlantıları[42].....	18
Şekil 3.6 Mikro-C Derleyicisi	21
Şekil 3.7 MXD2020E.....	23
Şekil 3.8 MXD2020E Bacak Bağlantıları.....	23
Şekil 3.9 MXD2020E ile Mikrodenetleyici Bağlantı Şeması	24
Şekil 3.10 MXD2020E İvme Çıkış Bilgisi[45].....	24
Şekil 3.11 Bifazik Stimulasyon Akımı	25
Şekil 3.12 Temel Flyback DCDC Devresi.....	27
Şekil 3.13 Flyback Devresi Voltaj ve Akım Sinyalleri.....	27
Şekil 3.14 Çıkış Devresi Şeması	28
Şekil 3.15 Bifazik Çıkış Sinyali (SPICE)	29
Şekil 3.16 Bifazik Çıkış Sinyali	29
Şekil 3.17 İvme Sensörü Yerleşimi[46]	30
Şekil 3.18 İvme Sensörü Çıkış Eğrisi [46].....	31
Şekil 3.19 Hazırlanan Baskıdevre ve İvme Sensörü Ekseni	32
Şekil 3.20 Yürüme Analizi.....	34
Şekil 3.21 DFS Sistemi Şeması.....	35
Şekil 3.22 Program Algoritması Akış Şeması	36

Şekil 3.23 Li-Ion Pil Şarj Eğrileri[43].....	37
Şekil 3.24 MCP73831 Uygulama Şeması[8]	38
Şekil 3.25 Visual Basic Kod Geliştirme Ortamı	39
Şekil 3.26 Sanal Seri Port Bağlantı Şeması	40
Şekil 3.27 Stimulatör Kontrol Programı Arayüzü.....	41

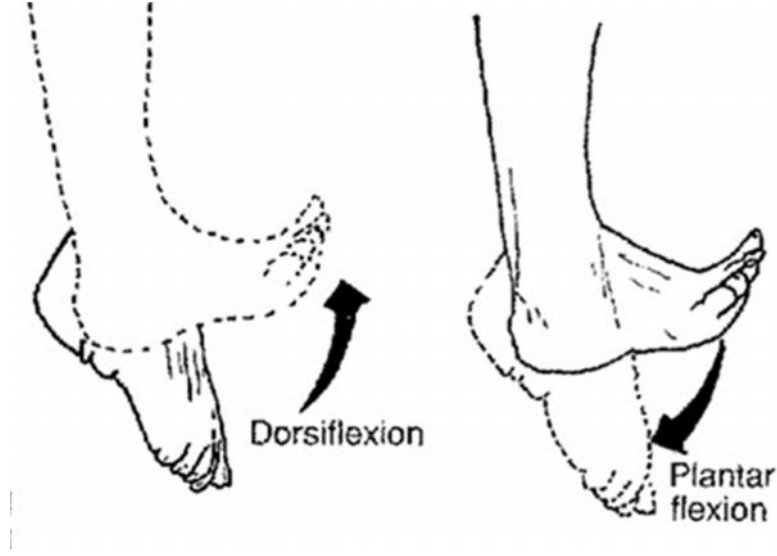
1.GİRİŞ VE AMAÇ

1.1. Düşük Ayak Hastalığı

Dünya Sağlık Örgütü'ne göre her yıl dünya çapında 15 milyon kişi inme geçirmektedir [1].Birleşik Krallıkta ise bu sayı yaklaşık olarak yıllık 100.000'dir [2]. Ayrıca Birleşik Krallık'ta inme kaynaklı ölüm oranı yaklaşık olarak %20, WHO istatistiklerine göre dünya genelinde bu oran %33'tür. Kurtulanların yaklaşık %75'inde ise yaşam kalitesinde düşme vardır. Kesin olarak bir bilgi olmasa da tahmini olarak inme hastalarının %20sinde düşük ayak hastalığı oraya çıkmaktadır [3].

İnmenin rehabilitasyonun kritik bileşenlerinden biri de yürümeyi sağlamaktır. 804 akut inme hastasında yapılan bir araştırmaya göre rehabilitasyon öncesi katılımcıların %51'inde yürüme hiç sağlanamamakta, %12'sinde ise yardım gerekmektedir. Rehabilitasyon sonrası ise yürüyememe oranı %18'e yardım alarak yürümenin oranı ise %11'e gerilemiştir. Bu araştırmada da ortaya yaklaşık olarak %20lik bir grubun üst motor nöron-düşük ayak hastalığının kalıcı olarak kaldığı gözlemlenmiştir [4].

Düşük Ayak Hastalığı; peroneal sinir hasarlarına veya peroneal sinire giden iletinin kesilmesine bağlı olarak ortaya çıkmaktadır. Peroneal sinir, dizin dış-yan-alt kısmından bacağın dış-yanı boyunca ayak bileğine kadar uzanır ve yürüme sırasında ayağın uygun zamanda doğru yöne kaldırılmasıyla görevli kasları kontrol eder. Bu kontrol görevinin yeterince yapılamadığı durumlarda; sıklıkla parmaklar yerde sürüklenir, ayak bileği içe/dışa hareket ettirilemez, ayak yerden kaldırılamaz ve düşük ayak ortaya çıkar [5]. Azalmış veya bozulmuş hareket yeteneği kişinin yaşam kalitesini olumsuz yönde etkilediği gibi, günlük aktivitelerini de sınırlamaktadır.



Şekil 1.1 Dorsifleksiyon hareketi

Düşük Ayak hastalığı inmenin yanı sıra cerebral palsy(CP) ve multipl skleroz (MS) hastalarında, başarısız bel cerrahisi sonucu, herhangi bir şekilde spinal kordun zarar görmesi veya travmatik beyin hasarları sonucu da ortaya çıkabilmektedir.

Düşük Ayak Hastalığının tedavisinde ayak bileği ortezleri veya destekleri (AFO - Ankle Foot Orthotic) en sık kullanılan ve tavsiye edilen seçeneklerdir. AFO genellikle plastik veya daha hafif (karbon gibi) malzemelerden yapılan ayak ve alt bacak kısmına giydirilen bir ortezdir(Şekil1.2). Bu ortez, ayak bileğine 90 derecelik açıda destek olarak ve ayağın yere takılmasını engelleyerek yürümeye yardımcı olmaktadır. Bu tür bir ortezin rahat kullanımı için uygun ayakkabılar ile beraber giyilmesi gerekmektedir. Yumuşak ayakkabılar, sandaletler veya arkası açık ayakkabılar ile AFO giyilememektedir.



Şekil 1.2 AFO Örneği

Fakat AFO çözümü pasif tedavi yöntemlerindendir: sinir-kas sisteminin aktif kullanımını sağlayamamakta ve bununla beraber ayak bileği hareket aralığını sınırlamaktadır. Ek olarak, AFO'lar düzgün yerleştirilmez ise dokulara aşırı baskı uygulayarak zarar verebilmektedir. Alternatif olarak DFS (Drop Foot Stimulator – Düşük Ayak Simülatörleri) sistemleri hastanın mevcut sinir yollarını ve kaslarını kullanarak AFO'ların yerini alabileceği düşünülmektedir. DFS sistemleri kullanılarak AFO'ların oluşturduğu kısıtlar ortadan kaldırılabilir. Mevcut kas sisteminin kullanılması ile yürüme yorgunluğu AFOların kullanımına göre azaltılabilir [6].

AFO'ların yanı sıra yeni bir uygulama olarak Botoks tedavisi de kullanılabilmektedir. Ayrıca düşük oranda da olsa cerrahi operasyonda yapılmaktadır [7].

1.2. Fonksiyonel Elektrik Stimulatörleri

Nörostimulasyon, anlık elektronik sinyaller kullanılarak sinir sisteminin uyarılmasıdır. Stimulasyon yönteminde elektrik sinyalleri deri üzerinden elektrotlar vasıtasıyla dolaylı olarak ya da bir implant ile doğrudan sinir sistemini uyarmaktadır. Terapinin amacı beyin veya spinal kordun zarar görmesi sonucu kaybedilmiş motor fonksiyonların yeniden kazandırılmasıdır. Bu işlemleri yerine getiren cihazlar Fonksiyonel Elektrik Stimulatörü (FES) olarak adlandırılır.

Üst motor nöron lezyonlarının önemli bir özelliği ilgili periferik sinirlerin elektriksel uyarımlara karşı tepki verebilmesidir, bu sayede FES kullanımı ile motor fonksiyonları tekrar kazanılabilir veya artırılabilir.

FES'lerin altında yatan nörofizyolojik etken, zarar görmemiş sinirlere düşük seviye palsler halinde elektrik akımları uygulayarak zarar görmemiş alt seviye nöronlarda aksiyon potansiyelinin oluşmasını sağlamaktır. Kas kasılmaları efferent(motor) sinirleri veya afferent(duyu) sinirlerini uyararak yapay olarak sağlanabilir. Örnek olarak, çekme refleksinin ortaya çıkarılması için peroneal sinirin uyarılmasıdır.

FES tarafından oluşturulan kas gücü çalıştırılan motor ünite sayısı ve aktivasyon oranına bağlıdır. Motor ünitesi tek motor nöronu ve bir grup kas lifini içerir. Elektrik palsi yükü (pals genişliği x pals genliği) yeteri kadar yüksek olduğunda ve sinir

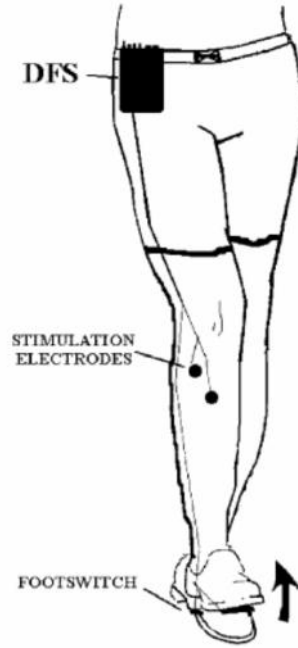
elektroda yeteri kadar yakınsa, motor nöron eşik değerinin üstünde polarize olur ve elektrik aksiyon potansiyeli oluşur. Kas gücü etkilenen motor ünitesinin sayısı oranında artar bu nedenle pals genişliğinin veya pals genliğinin kontrolü ile kas gücü ayarlanabilir. Kas gücü aynı zamanda stimülasyon frekansının kontrolü ile de ayarlanabilir. Geleneksel nöromüsküler stimülasyonu bir dizi kare dalga stimülasyon akım palslerinden oluşur.

FES'ler güçlü ve efektif kas kasılmaları ortaya çıkarsa da önemli limitleri de vardır. Normalde, kaslar çeşitli motor ünite tiplerinin karışımlarından oluşur. Motor üniteleri yavaş kasılan dirençli (tip-1) hızlı kasılan dirençli (tip-2a) ve hızlı kasılan çabuk yorulan (tip-2b) olarak ayrılırlar. Düşük kas gücü istendiği zaman ve buna bağlı olarak düşük yoğunluklu stimülasyon uygulandığında, elektroda yakın olan çoğu hızlı yorulan büyük motor üniteleri aktifleşir. Bunun nedeni ise hızlı yorulan tip-2B motor ünitelerinin daha büyük çaplı sinir aksonlarına sahip olmasıdır. Sinyal yoğunluğunu artırarak, elektroda daha uzakta kalan nöronlar ve daha küçük nöronlar (tip-1 ve tip-2a) aktifleşebilir. Buna ek olarak, sabit stimülasyon yoğunluğu ve sabit elektrod pozisyonu ile aynı motor üniteleri eşzamanlı olarak aktifleşir. FES sistemleri ile çeşitli tip motor üniteleri aynı zamanda tetiklenir ki merkezi sinir sisteminde çeşitli tip motor üniteleri asenkron olarak aktifleşir.

Düşük Ayak Stimulatörleri (DFS) yürümenin salınım fazında peroneal siniri uyarak dorsefleksiyonu sağlamaktadır. Bu özellikleri itibariyle hastaya fonksiyonel destek sağlamaktadırlar. Fakat stimulatörler beyin veya sinir sistemini tedavi etme özelliğine sahip değildir. Bu stimulatörler sayesinde yürüme paterni (ritmi) birçok hasta için gelişme göstermekte [8, 9, 10, 11], fakat sadece sınırlı sayıda hastalık durumlarında düşük ayak tamamen tedavi edilebilmektedir. Ayrıca simulatörün düzenli kullanımı kasların sürekli çalışmasını sağladığından atrofiyi engellediği [12], spastisiteyi düşürdüğü [13, 14] kardiyobasküler sistemi geliştirdiği [15, 16, 17] gözlemlenmiştir.

Düşük ayak hastalığına yönelik ilk FES uygulaması Liberson [18] tarafından 1961'de yapılmıştır. Liberson'un uygulamasında topuğun altına yerleştirilen bir anahtar ile stimülasyon tetiklemesi yapılmış ve peroneal sinir uyartılmıştır(Şekil1.3). Yürüme esnasında, anahtar üzerindeki baskının kalktığı anda başlayan stimülasyon ile dorsefleksiyon sağlanarak salınım fazında hastanın ayağının düşmesi veya yere sürmesi engellenir. Stimülasyon için deri üstü yüzey elektrotlar kullanılmıştır. DFS'ler hastanın

sürekli kullanabileceği bir cihaz olabileceği gibi rehabilitasyon programının bir parçası olarakta kullanılabilirler.



Şekil 1.3 Liberson DFS [5]

Yakın tarihte, Dr. Richard Stein yeni bir stimulator tasarlamıştır (WalkAide2) ve bu sistemde ayakaltına konulan bir anahtar yerine ivme sensörü kullanılarak yürüyüş paterni algılanmıştır [19, 20]. Klasik DFS sistemlerdeki topuk sensörünün kaldırılması ile klinik kullanımdaki bir takım sorunların önüne geçilmiştir. Örneğin topuk sensörüne giden kablolar kullanımı kısıtlandırmaktadır. Walkaide2’de de Liberson’un stimülasyon sistemi izlenmiş deri üstü elektrotlar ile stimülasyon sağlanmıştır. Ayrıca Walkaide2’nin stimülasyon sistemini implante elektrotlar ile değiştiren bir klinik çalışmada mevcuttur [21]. Bu çalışmada fizyolojik maliyet endeksi (PCI) incelenmiş ve yüzey stimülasyona göre daha başarılı olduğu ortaya konmuştur.

Bunların dışında Otto Bock firması implante elektrotlar kullanmaktadır. Sistemde kablosuz topuk sensörü ile konum bilgisi algılanmakta ve stimulator deri üstünden implante elektroda stimülasyon sinyallerini göndermektedir. Ayrıca Bioness firmasının ürettiği L300 DFS sisteminde de kablosuz topuk sensörü vardır ancak stimülasyon deri üstünden yapılmıştır.

1.3. Amaç

Yapılan çalışmalar göstermiş ki DFSler çeşitli nedenlerle dorsefleksiyon yeteneğini kaybetmiş hastaların rehabilitasyonuna yardımcı sistemlerdir. Hastaların yürüme hızında ve dengeleri sağlamalarında gelişmeler görülmüştür.

Bu çalışmada, ivme sensörü ile yürüyüş paternini algılayan düşük ayak hastalığına yönelik bir FES tasarlanması amaçlanmıştır. Yapılacak sistemde ivme sensörü kullanılarak topuk sensörü olmadan tek başına çalışabilen ve yüzey elektrot stimülasyonu yapabilen bir sistem tasarlanması amaçlanmıştır.

Stimülasyon sinyalleri bifazik olacak ve puls genişliği ile frekansı ayarlanabilecektir. Stimülasyon parametreleri bilgisayar programı tarafından ayarlanabilecektir. Stimulatör ile bilgisayar bağlantısı standart USB üzerinden olacaktır.

Şarj edilebilen bir pil yardımı ile sistem beslenecek ve pil şarj ünitesi tasarlanacaktır. Stimülasyon çıkış ünitesi ise ana sistemden izole tasarlanacaktır.

2. GENEL BİLGİLER

2.1. Yürüme

Yürüyüş kısa mesafelerde seyahat etmek insan için en uygun yoldur. Vücut ileri giderken, bir bacak temel olarak diğer bacak önde iken gerekli desteği sağlamaktadır. Adım devri (AD) en basit olarak salınım ve durma fazlarını içerir. Durma temelde 3 bölüme ayrılabilir, ilk durum olarak çift bacak destek, tek bacak destek ve son durum çift bacak destek. Her bir çift-bacak destek AD periyodunun yaklaşık %10'unu içermekte, salınım fazı %40 içermekte ve tek bacak destek periyodu %40'ını içermektedir. Adım hızına bağlı olarak durma ve salınım fazlarının oranları değişmektedir. Yürüme hızı arttıkça durma fazının oranı düşmektedir ve çift-bacak destek süresi düşmektedir. Koşma ise çift-bacak destek sürelerinin olmadığı durumdur. Ipsilateral terimi vücudun aynı tarafını kontralateral terimi ise vücudun diğer tarafını ya da diğer bacağı tanımlamak için kullanılır. Konuyla ilgili en yaygın terimler Perry [22] tarafından yapılmıştır ve bu tezde de bu tanımlamalar kullanılacaktır.

2.1.1. Yürümenin Fonksiyonel Görevleri

Yürüme sekiz fonksiyonel parçaya ayrılabilir, bunlar fazların alt bölümleridir. Her bir alt-fazın bir hedefi vardır ve hepsinin sıralı düzenli bir biçimde çalışması ile yürüme oluşur. Adım fazlarının sıralı kombinasyonu temelde üç basit görevden oluşur: Ağırlık Kabul (AK), Tek-Bacak Destek (TBD), Bacak İlerlemesi (Bİ).

AK ilk görevdir. Bu bölümde İlk-Temas ve Yükleme-Cevabı olarak adlandırılan iki adet alt-faz bulunmaktadır.

İkinci görev TBD'dir. Bu bölümde iki alt-faz bulunmaktadır; ortadurma ve son durma. Kontralateral ayak salınım fazına başladığında, tek-bacak desteği durma konumundadır(orta durma). Orta durma TBD'nin ilk fazıdır, kontralateral ayağın kalkması ile başlar ve önayağın yere hizalanması ile sona erer. Son durma ise arkada kalan diğer ayağında yere vurması ile sona erer.

Bİ dört alt fazdan oluşur: Ön-salınım, ilk salınım, orta salınım, son salınım. Ön salınım fazı durmanın son fazıdır. Ağırlık bir taraftan diğerine aktarılır ve bacak salınım için hazırlanır. İlk salınım bacağın yerden kalkması ile başlar ve salınımın diğer bacağın

konumuna gelmesi ile biter, orta-salınım başlar. Orta-salınım salınım yapan bacağın dik pozisyona gelmesine kadar devam eder. Son-salınım ise bacağın yere vurmasına kadar sürer.

2.1.2. Yürümenin Fazları

Adım döngüsünün fazları; ilk temas, yükleme cevabı, orta durma, son durma, ilk salınım, orta salınım, son salınım olarak sıralandırılır. Durma periyodu ilk beş fazı içermektedir: ilk temas, yükleme cevabı, orta durma, son durma, ilk salınım. Salınım periyodu ise kalan fazları içerir. Şekil 2.1’de tüm fazlar gösterilmiştir.

İlk-Temas; ayağın yere değdiği andır. Bu anda bacakta gerçekleşen motor fonksiyonların birçoğu yükleme cevabına hazırlık içindir.

Yükleme cevabı; adım periyodunun ilk %10unu kapsar ve çift-bacak desteği içerir. Bu faz dönemimde bütün ayak yere basmaktadır ve tüm vücut ağırlığı durma bacaklarına aktarılır. Bu faz kontralateral bacak yerden kalktığı zaman biter.

Orta-bekleme; tek-bacak desteğin ilk yarısını kapsar. Bu faz adım periyodunun %10 ile %30’u arasındadır. Kontralateral bacağın kalkması ile başlar ve vücut ağırlığının ayağın uzunluğu boyunca ilerleyip ayağın ön tarafına hizalanması ile biter.

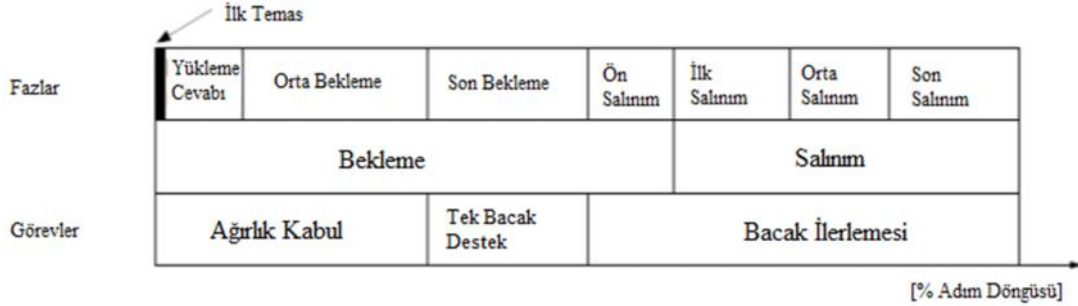
Son bekleme; tek-bacak desteğin ikinci yarısını kapsar. Bu faz topuğun yerden kesilmesi ile başlar ve kontralateral bacağın ayak yere basıncaya kadar devam eder. Son bekleme yaklaşık olarak adım periyodunun %30’undan ile %50’sine kadar olan süreyi kapsar. Bu faz süresince vücut ağırlığı ileride olan bacağa doğru gider.

Ön-salınım; çift-bacak desteğin sonudur ve bu faz durma fazının en az %12’sini yani periyodun %50-%62 arası olan süreyi kapsar. Bu faz kontralateral ayak yerde iken başlar ve parmaklar yerden kalkınca biter. Bu faz süresince, vücut ağırlığı hareketin olmadığı yöne aktarılır.

İlk-salınım; adım periyodunun %62’den %75’ine kadar olan süreyi kapsar. Bu faz parmakların yerden kalkması ile başlar ve dizde maksimum fleksiyon sağlanıncaya kadar devam eder.

Orta-salınım; salınım periyodunun ikinci fazıdır ve adım periyodunun %75’inden %85’ine kadar olan süreyi kapsar. Bu faz dizin maksimum fleksiyonundan kaval kemiği dik pozisyona geçinceye kadar olan süreyi kapsar.

Son-salınım; adım periyodunun %85'inden %100'üne kadar olan süreyi kapsar. Bu faz kaval kemiğinin sagittal oryantasyonda dik konumun ötesine geçmesi ile başlar ve dizin topuk teması için tamamen uzanması ile biter.



Şekil 2.1 Yürüme Fazları [22]

2.2. Yürüme Fazlarının Tespiti

Yürüme fazlarının tespiti DFS sistemlerinde elektriksel stimülasyonun zamanlamasını ayarlamak içindir. Özellikle günlük yaşamda kullanılması nedeniyle DFS sistemleri için fazların sağlam ve doğru bir şekilde tespit edilebilmesi gerekmektedir. DFS için faz tespitleri günlük yaşamdaki yürüme dışındaki diğer aktiviteleri de takip edebilmelidir. Yürüme fazlarının tespitinde yürüme döngüsü iki veya daha fazla aşamaya ayrılmıştır. Daha önceki bölümde de belirtildiği gibi yürüme her bir bacak için sekiz fazdan oluşmaktadır. Tüm fazları tek bir sensör girişinden takip edebilmek ise oldukça zordur. Genelde kullanılan sensöre bağlı olarak dörtten daha az faz tespit edilebilmektedir.

2.2.1. Yürüme Fazlarının Tespiti: Sensörler

Faz tespiti için ilk kullanılan sensör topukta bulunan bir anahtardır. Ayağın altında bulunan bu anahtarın zamanla performansında düşme olmaktadır[100] ve bu yapı oldukça mekanik bir arızaya açıktır. Mekanik anahtarların yetersizliği yapılan çalışmalarla ortaya konmuştur [23]. Ortalama olarak, %1,3 - %7,6 oranında topukta bulunan anahtar zamanlama hatası yapmaktadır. Odstock DFS sisteminin kullanıcıları arasında yapılan bir ankette ise yaklaşık %40 oranında hasta stimulatörün anahtar sistemi nedeniyle sorun yaşamıştır [24].

Bu veriler yürüme fazlarının tespiti için daha güvenilir bir sensör sisteminin kullanılması gerektiğini göstermiştir. Fakat sistemin doğruluğu ve fazlar arasında geçişin tanımlanması oldukça zordur bu bakımdan sistemin bağımsız bir kişi tarafından da yorumlanması ve fazları manuel olarak tanımlaması, örneğin video aracılığı ile, gerekebilir. Bu nedenle, hiçbir sensör tüm yürüme fazlarını ve faz geçişlerini tek başına tanımlayabilecek olarak düşünülmemelidir.

Basit topuk anahtarı dışında, mekanik anahtarların kombinasyonları, force sensitive resistors (FSR), eğim sensörleri, açıölçerler(goniometre), jiroskoplar, ivme sensörleri ve elektronörografi (ENG) sensör sistemleri gibi sistemler yürüme fazlarının tespiti ve dolayısıyla stimülasyon dizilerinin zamanlama kontrolü için kullanılmıştır. İki ya da üç anahtarın kullanımı [25] ve ayak tabanından basınç ölçümü yöntemlerinin [26] çalışmaları yapılmıştır. Daha fazla sensör donanımı ile yapılan bir çalışma da ise topuk anahtarlarına ek olarak, goniometreler ile açı ölçümü yapılmış ve kalça ekleminde ivme ölçülmüştür [27]. Tüm bu bilgiler makine öğrenmesinde kullanılmıştır. Başka bir çalışmada bacak bölgesinin radyal ve teğet ivmelerinin ölçülmesi için dört adet ivme sensörü kullanılmıştır [28]. Özellikle peroneal sinirin stimülasyonu için önemli olan topuğun yerden kesilmesinin tespiti olmak üzere dört ayrı yürüme fazının tespiti için kural tabanlı bir algoritma geliştirilmiştir. İvme sensörleri kullanılarak yapılan bir başka yaklaşım Williamson ve Andrews tarafından tanımlanmıştır [29]. Üç sensörde farklı oryantasyonlarda kaval kemiğine yerleştirilmiştir. Ayrıca basınç sensörleri ile yürüme fazları için referans sinyaller üretilmiştir. Bu yapı ile oldukça yüksek yüksek oranda doğruluk sağlanmıştır. Veriler 33 Hz ile örnekleştir.

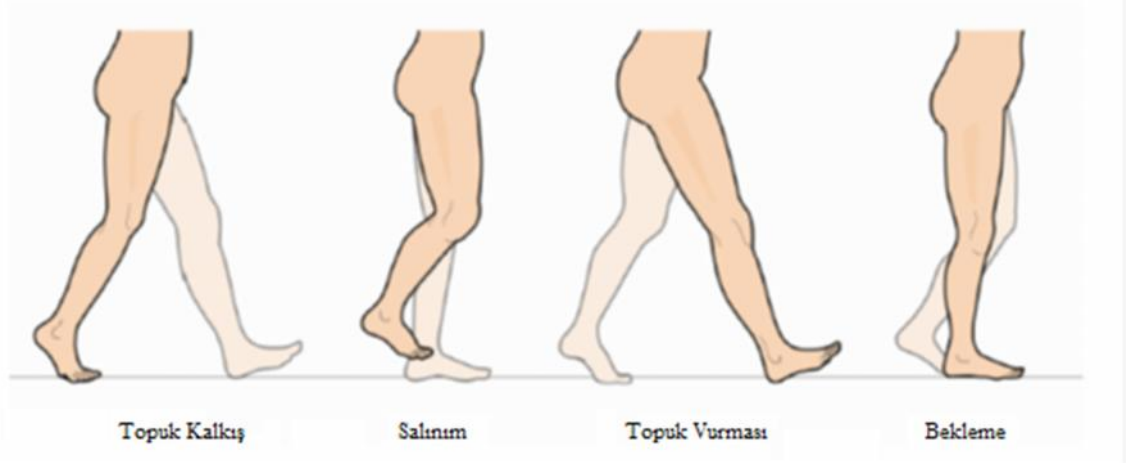
Başka bir alternatif olarak; kalça, diz ve ayak bileğinin açıları goniometreler ile ölçülerek stimülasyon zamanlaması ayarlanmıştır [30]. Yakın zamanda jiroskop ve ivme ölçerlerin kombinasyonundan oluşan ISU (inertial sensor system) kullanılarak yürüme fazlarının tespitine yönelik bir çalışma Kotiadis tarafından yapılmıştır [31]. Bir ISU sisteminde 3 eksen jiroskop ve 3 eksen ivmeölçer bulursa da sadece sagittal düzlemde bulunan 2 ivmeölçer ve jiroskop kullanılmıştır. Bu tezde çalışmanın aksi olarak sensör sistemi hemen dizaltına yerleştirilmiştir. Sabatani yapılan bir başka çalışmada ise iki ivme sensörü ve bir jiroskop kullanılmıştır [32]. ISU'nun tüm eksenlerini kullanmak yürüme fazları hakkında daha doğru bir tahmin yürütülmesini

sağlayabilir fakat yapılan literatür taramasında şu ana kadar tüm jiroskop ve ivmeölçerlerin tüm eksenlerini kullanan bir sistem ile karşılaşılmamıştır.

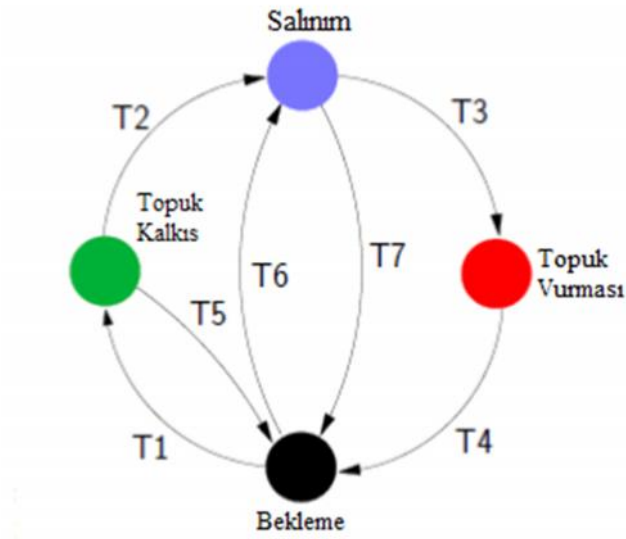
Başka bir metod olarak; Stange ve Hoffer motor (afferent) sinirden gelen verileri kullanarak yürüme fazlarını belirlemeye çalışmıştır [33]. Kediler üzerinde yapılan çalışmada implante bir kelepçe tip elektrot ile fazlar arası geçiş %99 oranında doğrulukla tespit edilmiştir. Daha sonra bu yaklaşım insanların üstünde uygulanmıştır [34]. Peroneal sinire bağlanan kelepçe elektrot ile tespit ve stimülasyonda aynı elektrot üstünden yapılmıştır [34]. Fakat bu sistemler sadece iki fazı tespit için kullanılabilir ve uygulaması zordur. Başka bir uygulama da ise EMG sinyalleri kullanılmıştır [35]. Ön tibialis üzerindeki elektrotlar ile hareket isteği okunmuştur. Elde edilen sinyal üzerinden yapılan işlemlerden sonra eğer eşik değeri geçer ise stimülasyon diğer iki elektrot üzerinden başlatılmış ve EMG sinyali var olduğu sürece devam etmiştir.

2.2.1. Yürüme Fazlarının Tespit Metodları

Temel olarak metodolojide yürüme fazlarının tespitinde iki yöntem kullanılmıştır. İlk olarak kural tabanlı bir yaklaşım ile yürüme fazları sonlu durum analizindeki durumlar olarak kabul edilmiş ve durumlar arası geçişler sensörlerden gelen girişler olarak tanımlanmıştır [36, 31, 37, 38, 28]. Buna basit bir örnek olacak çalışma Papas tarafından yapılmıştır [37]. Bu uygulamada bir jiroskop ve 3 adet FSR kullanılmıştır. Bir FSR topuk altına diğer iki FSR ise neredeyse simetrik olarak yüklenmiş metatarsal uçlara yerleştirilmiştir. Jiroskop topuğa bağlanmıştır ve yere göre açının hesaplanması için kullanılmıştır. Eğim tüm FSRler yere bastığı zaman sıfırlanmıştır. Yürüme fazları; durma, topuğun yerden kesilmesi, salınım ve topuğun yere vurması Şekil 2.1 olmak üzere dörde ayrılmıştır. Bu fazlar durum olarak belirlenmiştir. Düzenlenen algorithmada durumlar arasında 7 geçiş vardır(Şekil 2.3).



Şekil 2.2 Temel Yürüme



Şekil 2.3 Yürümenin Sonlu Durum Analizi [22]

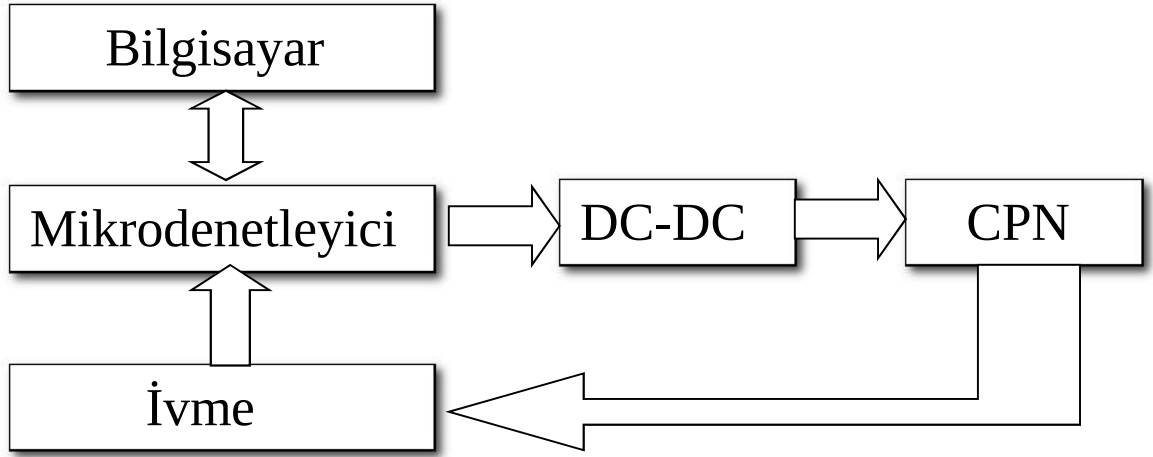
İkinci yaklaşım ise birinciye göre tamamen değişiktir. Bu yaklaşımda belli kurallar veya durumlar yerine sistemde bir kara kutu ya tüm girişler yapılır ve çıkışlar alınır. Bu tip sistemler genellikle makine öğrenmesi, bulanık mantık ve yapay sinir ağlarında kullanılır. Genel olarak bu tip sistemlerin bilinen bir referansa göre eğitilmeleri gerekmektedir. Örneğin yürüme videoları bu tip bir eğitim için referans olabilirler. Bu sistemlerin en büyük avantajı doğru olarak eğitilirse güvenilir bir şekilde fazları tespit edebilirler. Dezavantajları ise her bir konu için sistemlerin manuela olarak eğitilmesi gerekmektedir. Örneğin bir mekaniksel anahtar ile elle girdi verilmelidir. Bu ise oldukça zaman harcayan bir prosedürdür ve her seferinde yeniden

uygulanması gerekmektedir ki klinik uygulamalar için yerine getirmek oldukça zor olacaktır.

3. DÜŞÜK AYAK STİMLATÖRÜ SİSTEMİ

3.1. Sisteme Genel Bakış

Bu bölümde tezin konusu olan düşük ayak stimulatörün ve ölçüm sistemleri anlatılacaktır. Tasarlanması planlanan DFS sistemi yüzey elektrotlar ile stimulasyon yapacak, ivme sensörü ile yürüme fazlarını takip edecek, şarj edilebilen bir batarya ile beslemesini sağlayacak ve stimulasyon parametreleri bir bilgisayar programı aracılığı ile ayarlanacaktır. Sistemi blok şeması Şekil 3.1'deki gibidir.



Şekil 3.1 DFS Sistemi Blok Şema

Stimulatör piyasada varolan düşük ayak stimulatörlerine göre oldukça ufak boyutlardaki bir kutuya sığdırılmıştır. Ek olarak pil şarj devresi de tasarlanmış ve piyasada rahatlıkla bulunabilen bir adaptör kutusuna uygun olarak baskı devresi tasarlanmıştır(Şekil 3.2).



Şekil 3.2 Tasarlanan DFS ve Şarj Cihazı

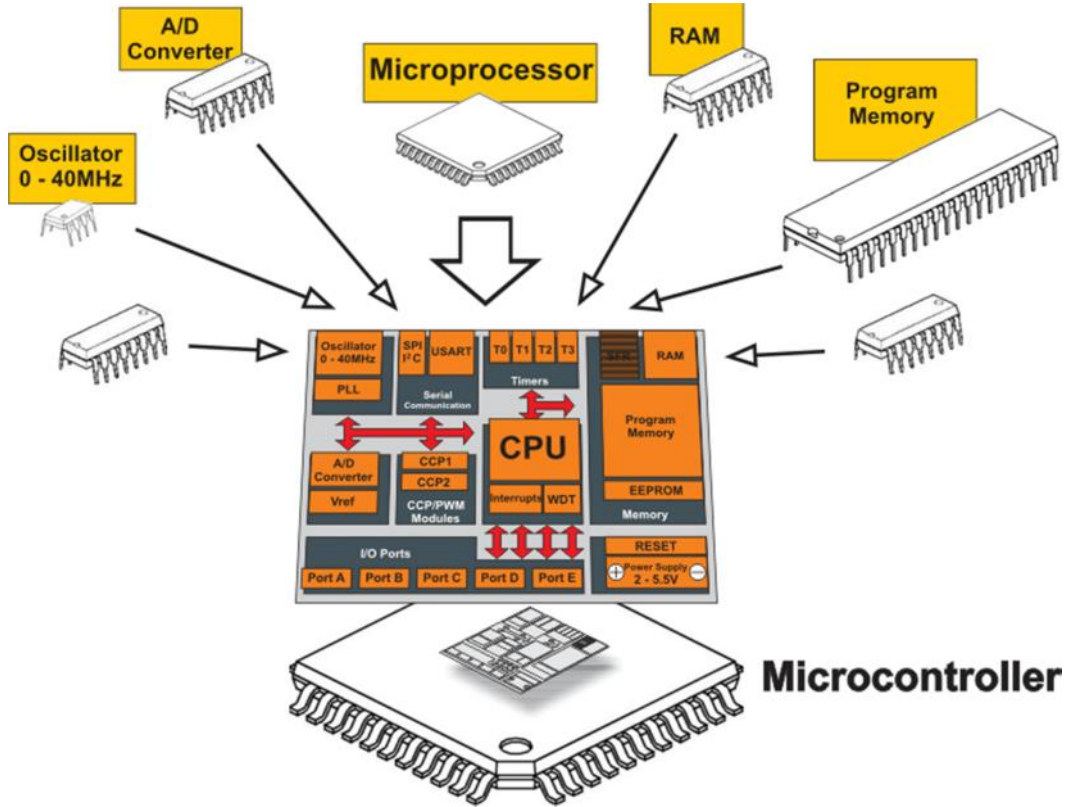
Tezin bundan sonraki bölümlerinde önce stimulator devresinde kullanılan mikrodenetleyici, ivme sensörü ve DC-DC dönüştürücü devresi açıklanacak ve Stimulator programının algoritmasına değinilecektir. Sonra bilgisayar bölümünde kullanılan materyal ve yöntem açıklanacaktır.

3.2. Gömülü Sistemler

Gömülü sistemler, en genel tanımıyla belirli bir işi yapması için tasarlanmış, mikroişlemci veya mikrodenetleyici tabanlı sistemlerdir. Bir bilgisayarı gömülü sistem olmaktan çıkaran nüans ise; bilgisayarlar kullanıcının her türlü işlemini yapabilmesini sağlamak üzere tasarlanmış sistemlerdir [39]. Örneğin buzdolabının, çamaşır makinesinin kontrol kartı gömülü bir sistemdir. Kullanım alanı daraltılmıştır; motor sürer, kullanıcının kontrol arayüzünde (butonlar, düğmeler) yaptığı değişiklikleri yorumlarlar vs. Kısaca belirli bir amacı gerçekleştirmeye yönelik tasarlanan sisteme gömülü sistem denir.

Gömülü sistemlerin çekirdeğini oluşturan mikroişlemciler veya mikrodenetleyicilerdir. Mikroişlemciler içerisinde çok fazla sayıda transistör bulunan, silikondan yapılmış bilgisayarların ihtiyaç duyduğu matematiksel ve lojik işlemleri

bünyesinde gerçekleştiren entegre devreye denir(Şekil 3.3). Verilerin alınması, işlenmesi ve bir kontrol uygulamasına dâhil edilmesinden sorumlu yapıdır. Mikrodenetleyiciler ise yapılarında mikroişlemci ünitesi, mikroişlemci ünitesinin temel çevre donanımlarını (RAM-ROM-EEPROM) ve çeşitli çevrebirim donanımlarını (analog dijital çevirici, PWM modülü gibi) içeren entegre devrelerdir. Bu yapıyla mikrodenetleyiciler baskıdevre de kazanım sağlarlar ve daha güvenilir yapıdadırlar.



Şekil 3.3 Mikrodenetleyici

Çeşitli yonga üreticileri farklı çevrebirimler ekleyerek çeşitli kılıflarda mikrodenetleyici üretmektedirler. Bu sayede uygulamalara yönelik uygun bir mikrodenetleyici seçilebilir ve başka bir uygulamada başka bir mikrodenetleyici kullanılabilir.

Programcılar belli bir kurallar çerçevesinde yazdığı mnemonicleri, üreticilerin veya çeşitli yazılım firmalarının ürettiği derleyiciler ile mikrodenetleyicilerin anlayacağı tek bir dosyadan oluşan opcodelere çevirir. Oluşan bu dosya genellikle bir programlayıcı donanım vasıtası ile bilgisayardan mikrodenetleyicinin program

hafızasına aktarılır. Her bir mikrodnetleyici ailesinin kendine özel bir programlayıcısı vardır.

Bu çalışmada ülkemizde sıkça kullanılan Microchip firmasının PIC ailesi kullanılacaktır.

3.2.1. PIC Mikrodnetleyicileri ve Genel Özellikleri

PIC mikrodnetleyicileri, Amerikan Microchip firması tarafından üretilmektedir. Profesyonel ve amatör elektronikçiler tarafından en çok tercih edilen 8-bit mikrodnetleyici serisidir. Fiyatlarının düşük olması, kolay temin edilebilmeleri, geniş bir kullanıcı kitlesinin olması, yeterli kaynak kitap ve uygulama notlarının mevcut olması, düşük fiyatlđ geliştirme kitlerinin bulunması ve seri programlanabilmeleri PIC mikrodnetleyicilerini rakiplerinden ayıran en temel avantajlarıdır [40].

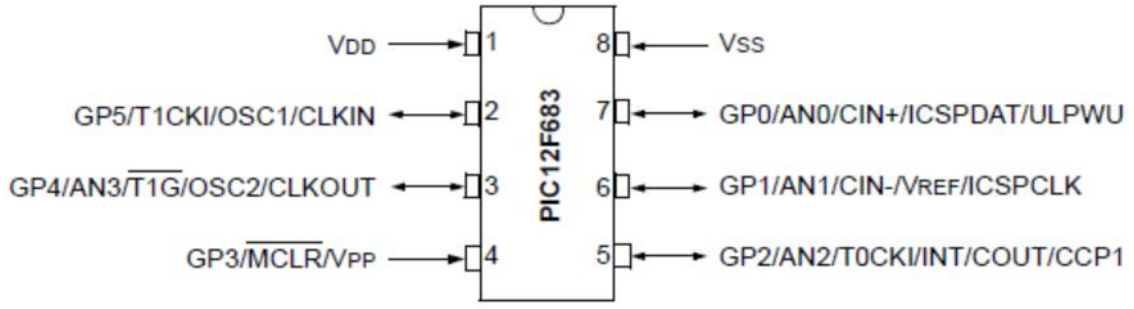
Günümüzde PIC mikrodnetleyicileri 8-bit, 16-bit ve 32-bit'lik üretilmektedir. Microchip firması tarafından üretilen 400'den fazla mikrodnetleyici modeli mevcuttur [41]. Ayrıca 8 bacaklı kılıftan 100 bacaklı kılıfa kadar yaygın bir aralıđı vardır(Şekil 3.4).



Şekil 3.4 Çeşitli PIC mikrodnetleyicileri

3.2.2. PIC12F683 ve Genel Özellikleri

Çalışmamızda PIC ailesinde PIC12F683 kullanılmıştır. PIC12F683 8-bit, 8 bacak nanowatt teknolojisine sahip bir mikrodnetleyicidir [42]. Şekil 3.5'te PIC12F683'ün bacak bağlantıları gösterilmiştir.



Şekil 3.5 PIC12F683 Bacak Bağlantıları[42]

PIC12F683'ün genel özellikleri aşağıdaki gibidir[42].

- Güç yönetim seçenekleri;
- Aktif (220 uA-4MHz, 11uA-32kHz)
- Uyku (işlemci ve çevresel üniteler pasif, tüketim 50 nA),
- Uyku watchdog timer ile 1uA
- Çevresel birimleri besleme akımı maksimum 95 mA
- Programlanabilir 6 dijital giriş-çıkış
- Tüm giriş çıkışlar kesme kaynağı olabilir.
- 4 kanal 10-bit A/D dönüştürücü.
- İki adet 8-bit zamanlayıcı
- Bir adet 16-bit zamanlayıcı
- Bir adet CCP (Capture/Compare/PWM, Yaklama/Karşılaştırma/PWM) modülü
- Sadece iki bacak ile devre üstünde programlama yapılabilme (ICSP)
- 2048 word Flash Program hafızası
- 128 byte RAM
- 256 byte EEPROM
- Geniş çalışma voltaj aralığı (2.0v – 5,5v)
- Dahili ayarlanabilir kristal seçeneği (125 kHz – 8MHz)
- Maksimum 20 MHz çalışabilme (harici osilatör devresi ile)
- Programlanabilir Brown-out Reset fonksiyonu ile ani gerilim düşümlerinde resetlenebilme
- Programlanabilir Watchdog zamanlayıcısı ile yazılımsal kilitleme durumunda resetlenebilme

- Üç farklı kılıf seçeneği; PDIP, SOIC ve DFN

PIC12f683'ün çalışmamızda kullanılmasının sebepleri ise şöyledir:

- Düşük Güç tüketimi
- Oldukça ufak SOIC kılıf
- Programlanmasının kolay olması
- Dahili osilatör
- Dahili EEPROM (özellikle düşük bacak sayısına bağlı kılıflarda genellikle EEPROM bulunmamaktadır).

3.2.3. Programlama Dilleri ve Mikrodenetleyiciler

Programlama dili; bir algoritmayı ifade etmek amacıyla, bir bilgisayara ne yapmasını istediğini anlatmasının tek tipleştirilmiş yoludur. Programlama dilleri, yazılımcının bilgisayara hangi veri üzerinde işlem yapacağını, verinin nasıl depolanıp iletileceğini, hangi koşullarda hangi işlemlerin yapılacağını tam olarak anlatmasını sağlar.

Şu ana kadar 150'den fazla programlama dili yapılmıştır [44]. Bunlardan bazıları Pascal, Basic, C, C#, C++, Java, Cobol, Perl, Python, Ada, Fortran, Delphi ve 1C Enterprise'dır.

Programlama dilleri çeşitli şekillerde gruplandırılabilir. Örneğin, programlama dilleri nesne yönelimli olması veya olmaması durumuna göre ayrılabilir. Nesne yönelimli diller oldukça hızlı ve kolay programlama imkanı verirken özellikle mikrodenetleyicilerin sahip olduğu kısıtlı donanım imkanları nedeniyle tercih edilmezler.

Programlama dilleri arasındaki en sık gruplandırma ise şöyledir:

- Yüksek Seviyeli Diller
- Orta Seviye Diller
- Düşük Seviye Diller

Yüksek seviye diller; öğrenilmeleri kolay ve program yazmanın kolay olduğu dillerdir. Derleyiciler sahip olduğu özellikler sayesinde programcıların üstündeki yükleri kaldırarak oldukça kolay bir hale getirmişlerdir. Bu dillerin avantajı kolaylık

iken dezavantajı ise yeteri kadar esnek olamamaları nedeniyle özellikle gömülü sistemlerde programcıları kısıtlamaktadırlar.

Orta seviye diller; en yaygın olarak kullanılan dillerdir. Yüksek seviye dillere göre daha esnek bir yapıya sahiptirler. Düşük seviye dillerdeki karışık ayarlamalar ile uğraşmalarına ise gerek yoktur. Örneğin Assembly dilinde adresleme yöntemleri ile de uğraşmanız gerekmektedir fakat C dilinde adreslemeler otomatik olarak linker tarafından yürütülür. Kullanıcılar istedikleri kütüphaneleri kullanarak programlarını yazarlar. Yapılan araştırmalara göre dünyada en yaygın kullanılan dil C dilidir[survey].

Düşük seviye dillere aslında tek verilebilecek örnek Assembly dilidir. Bu dilde mikrodenetleyici / mikroişlemci ünitesinin direk olarak algıladığı ifadeler (op-code) mnemoniclerle ifade edilmiştir. Assemblyde tüm yük kullanıcıların omzundadır. Derleyici herhangi bir şekilde faydası olmaz sadece yazılan kodu derler. Örneğin yazılacak bir alt fonksiyona gönderilecek parametrelerin adresleri, alt fonksiyonun üst fonksiyona göre uzaklığı ve buna bağlı yapılacak adreslemeler gibi. Düşük seviye dillerin en büyük avantajı hızı ve az yer kaplamasıdır. Fakat bu avantaj özellikle projenin karmaşık bir hale gelmesi durumunda kaybedilebilmektedir.

3.2.3.1 C programlama dili

1970'lerin başında Ken Thompson ve Dennis Ritchie tarafından UNIX İşletim Sistemi için 1972 yılında geliştirilmiş bir C programlama dili, günümüzde neredeyse tüm işletim sistemlerinde kullanılmaktadır. Dünya'nın en çok kullanılan sistem programlama dilidir.

C'nin ilk gelişme safhaları 1969 ile 1974 arasında AT&T Bell Laboratuvarları'nda gerçekleşti. Ritchie'ye göre en yaratıcı devre 1972 idi. Dilin pek çok özelliği "B" adlı bir dilden türediği için, yeni dile "C" adı verildi. 1983'te Amerikan Ulusal Standartlar Enstitüsü (ANSI) bir C standardı oluşturmak için bir kurul oluşturdu ve bu kurul standardı 1989'da tamamladı ve standart ANSI X3.159-1989 " Programming Language C (C Programlama Dili) " olarak yayımlandı. Dilin bu versiyonu genellikle ANSI C olarak adlandırılır.

Günümüzde yazılmakta olan C programlarının çoğunluğu ANSI C standardına uygun olarak yazılmaktadır. Yalnızca standart C kullanılarak yazılmış bir program, standarda uyumlu her derleyici ile doğru bir biçimde derlenip çalıştırılabilir. Ancak,

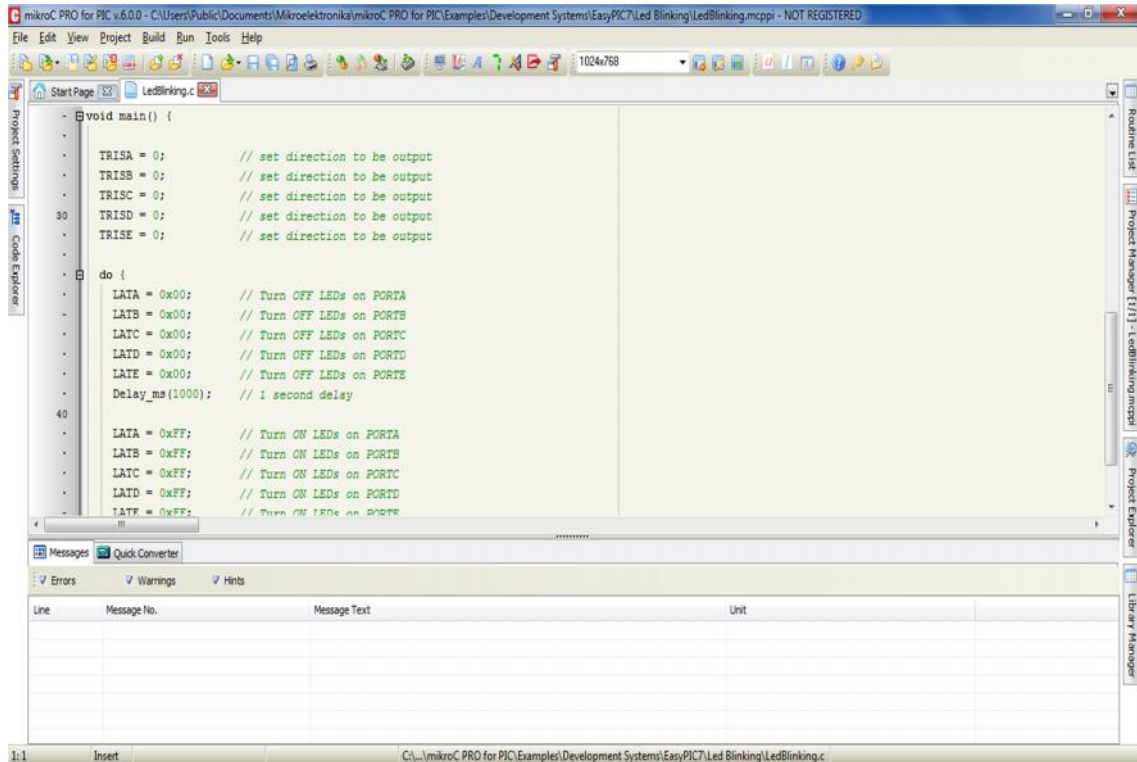
standart olmayan kütüphaneler kullanılarak yazılmış programlar belli bir platform ya da derleyici gerektirebilirler.

Gömülü sistemler için yazılmış C derleyicileri, programcının yazmış olduğu C kodlarını öncelikle Assembly diline çevirir ve daha sonra derler. Derleyicinin ürettiği Assembly kodu ne kadar az ise derleyicinin kalitesi o kadar artar. Bazı derleyiciler arasındaki fiyat farkı sadece üretilen kod miktarı arasındaki farktan kaynaklanmaktadır.

3.2.3.2. Mikro-C derleyicisi

PIC mikrodeneleyicileri için birçok dilde derleyiciler vardır. Örneğin MeLabs firmasının PICBASIC dili, IAR firmasının C derleyicisi, Microchip'in kendi C ve Assembly derleyicileri CCS firmasının PIC-C derleyicisi ve MikroElektronika firmasının Mikro-C ve Mikro-BASIC derleyicileri gibi. Bunlar dışında da çeşitli derleyiciler vardır. Bu çalışmada Mikro-C derleyicisi kullanılacaktır.

Mikro-C içinde barındırdığı 500'den fazla kütüphane ile C dilinde oldukça başarılı kod yazmayı sağlamaktadır. Ayrıca MikroElektronika'nın geliştirdiği kod geliştirme ortamı ile PIC mikrodeneleyicisine yönelik çeşitli ayarlamalar oldukça kolay bir şekilde ayarlanmaktadır(Şekil 3.6).



Şekil 3.6 Mikro-C Derleyicisi

3.3. İvme Sensörleri

İvme sensörleri üzerlerine düşen statik ve dinamik ivmeleri ölçüp çeşitli şekillerde elektriksel verilere dönüştüren sensörlerdir. Örneğin düz bir yüzeyde yerleştirilen bir ivme sensörü çıkış olarak yer çekimi ivmesini gösterir, serbest düşmede ise sıfır değerini gösterir. İvme sensörleri çeşitli konfigürasyonlarda üretilebilirler. Fakat özellikle son yıllarda Mikro-elektronik mekanik üretim teknikleri ile oldukça küçük yapılarda üretilebilmişlerdir. Mikroelektro-mekanik sistemler (MEMS) günümüzde var olan mekanik ve elektrik sistemlerin entegre ve minyatürize versiyonları olup mikron boyutlarında olan bu sistemleri nanoelektromekanik sistemler (NEMS) vasıtası ile nanoteknoloji uygulamaları için de kullanmak da mümkündür [44]. MEMS kavramı ilk olarak 1987 yılında bir mikrodinamik çalışmayı esnasında telaffuz edilmiştir. Fakat MEMS kavramının ortaya çıkması esas olarak entegre devre çalışmalarında yaşanan gelişmeler ışığında olmuştur.

MEMS ivme sensörleri birkaç g kuvvetlerinden birkaç bin g kuvvetlerini ölçebilecek çeşitlikte üretilmektedir. Uygulamaya yönelik aralıkta ivme sensörü seçilerek en yüksek çözünürlük elde edilebilir. Ayrıca tek bir kılıfta 3 eksen ölçüm yapabilen ivme sensörleri de mevcuttur.

İvme sensörlerinin geniş uygulama alanı vardır. Özellikle savunma sanayinde kullanılmaktadır. Son yıllarda akıllı telefonlarda kullanıcı arayüzünün yönlendirmesi uygulaması ile ön plana çıkmaktadır.

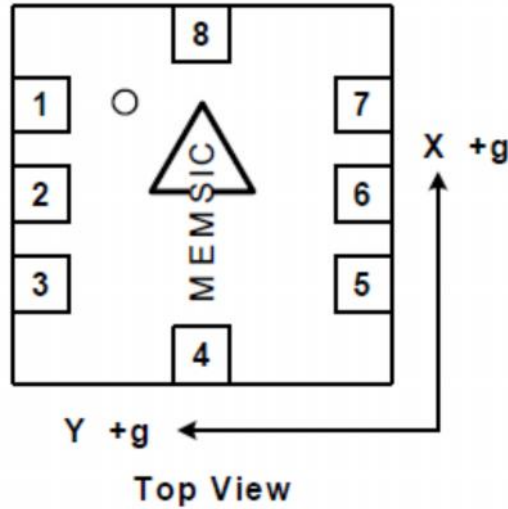
İvme sensörlerinin çeşitli elektronik çıkış opsiyonları vardır:

- Analog çıkış: İvme değerini genellikle voltaj olarak çıkış verirler. Örneğin MMA7260Q; bir g kuvvetine karşılık 800 mV çıkış üretmektedir.
- Pals-Genişlik Çıkışı: İvme değerini bir pals genişliği olarak verirler. Örneğin MXD2020E; bir g kuvvetine karşılık %20 oranında çıkış palsini değiştirmektedir.
- Dijital Çıkış: Bu sensörler standart olan SPI veya I2C çıkış protokollerini kullanırlar. Elde edilen ivme bilgisi sensörde dâhili bulunan bir A/D dönüştürücü ile dijital hale çevrilir ve çıkış olarak verilir. Örneğin LIS3DH sensöründe 1 bitlik veri 1mglik değişime eşittir.

Bu çalışmada MEMSIC firmasının ürettiği MXD2020E ivmeölçeri kullanılmıştır(Şekil 3.7). Bu ivmeölçerin seçilme nedeni özellikle geniş çalışma voltaj aralığı nedeniyle sistemde kullanılması düşünülen lityum-polimer piller ile direk olarak çalışabileceklerdir. Ayrıca çıkış sistemleri filtre edilerek PWM sinyalleri olarak verilmektedir [45]. Bu sayede A/D çevrim ve filtrasyonu ile uğraşılmamaktadır. Sensör çift eksendir, çalışmamızdaki uygulamamız için yeterlidir. Şekil 3.8'te MXD2020E'nin bacak bağlantıları verilmiştir.



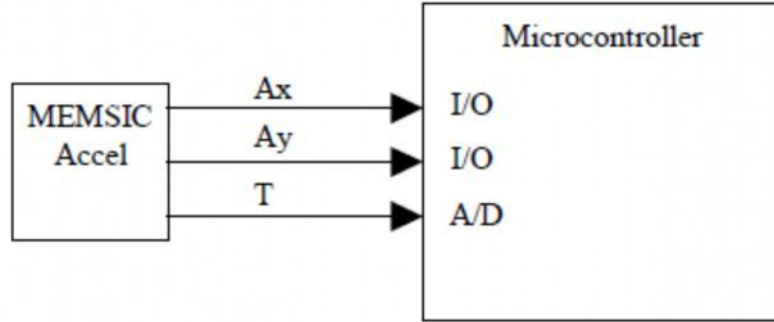
Şekil 3.7 MXD2020E



Şekil 3.8 MXD2020E Bacak Bağlantıları

Her bir eksen için olan çıkışlar iki ayrı bacadan verilmiştir. Opsiyonel olarak entegre devrenin sıcaklığı da başka bir bacadan verilmiştir. Bu sayede özellikle yüksek

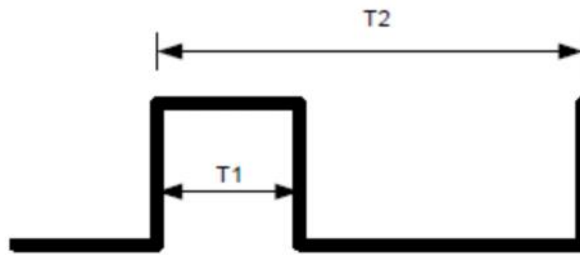
sıcaklıklarda çalışırken oluşabilecek değişikliklerin kompanzasyonu sağlanabilir [46]. Örnek bir bağlantı şeması Şekil 3.9’da verilmiştir.



Şekil 3.9 MXD2020E ile Mikrodenetleyici Bağlantı Şeması

MXD2020E’nin çıkış sinyali belirli bir frekansa sahip kare dalgadır(Şekil 3.10). Çıkış sinyalinin frekansı MXD2020E için 100Hz, MXD2020F için 400Hzdir. Sensör eksenlerine uygulanan ivmenin yönüne ve şiddetine bağlı olarak çıkış sinyalinin genişliği azalmakta veya artmaktadır. Eksenlerde herhangi ivme yokken çıkış sinyal genişliği periyodun yarısıdır. Eksenlere uygulanacak her bir g kuvveti sinyalin %20 oranında artmasını veya azalmasını sağlar. Bu durumda aşağıdaki formülle eksen üzerindeki ivme hesaplanabilir.

$$A(g) = \left[\frac{\left[\frac{T_1}{T_2} \right]^{-0,5}}{0,2} \right] \quad (3.1)$$



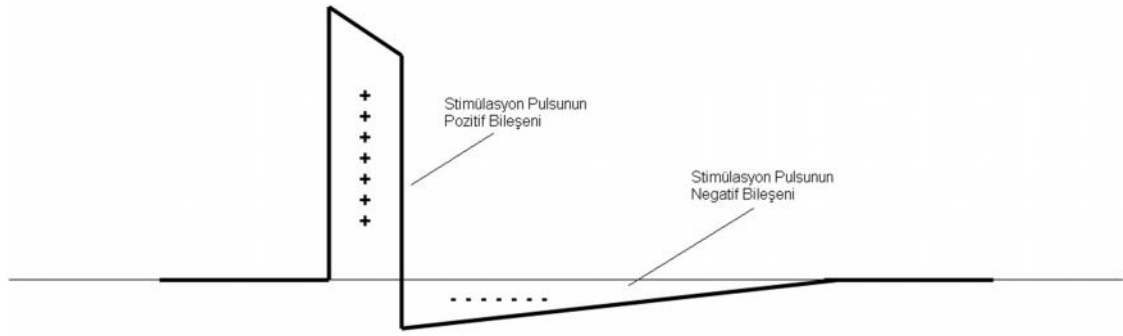
Şekil 3.10 MXD2020E İvme Çıkış Bilgisi[45]

MXD2020 serisinin en büyük dezavantajı ise sıcaklıktan etkilenmesidir. Bununla ilgili olarak öncelikle ürünün dizgisi yapılırken dikkat edilmelidir. Çalışmamızda devre yüksek sıcaklık altında çalışmayacağı için sıcaklık kompanzasyonunun uygulanmasına gerek görülmemiştir.

3.4. Stimulasyon Jeneratörü

Bu çalışmada deri üstü stimulasyon yapılacaktır. Deri üstü stimulasyonda stimulasyon sinyali sinirlere yüzey elektrotları ile aktarılır. Deri üstü stimulasyonun avantajı; cerrahi müdahale olmadığı için kolay uygulanabilmesidir, dezavantaj olarak sinir lokalizasyonun doğru bir biçimde sağlanamaması durumunda sonucun başarısız olmasıdır.

Temel olarak iki tip stimulasyon sinyali vardır: Monofazik ve Bifazik sinyaller. Monofazik sinyaller tek yönlü olarak elektrik akımı verilen sinyallere denir. Bifazik sinyallerde elektrik akımı hem pozitif hem de negatif olarak uygulanır (Şekil 3.11). Bu konuda yapılan karşılaştırma amaçlı çalışmalarda net bir sonuç ortaya konmamakla birlikte asimetrik bifazik stimulasyon peroneal nöromusküler uygulamalar için tercih sebebidir. Bifazik stimulasyonda ortalama DC akımın sıfır olması nedeniyle monofazik uygulamalara göre öne çıkmaktadır. Özellikle bifazik stimulasyonda ortalama DC akımın sıfır olması nedeniyle öne çıkmaktadır.



Şekil 3.11 Bifazik Stimulasyon Akımı

Deri üstü stimulasyonu gerçekleştirebilmek için çıkış sinyalinin implante stimulasyon tekniklerine göre oldukça yüksek olması gerekmektedir. Örneğin implante bir sistemde 4-5 V gibi değerler bile yüksek kabul edilirken, harici stimulasyonda en yüksek değer olarak 100-120 V kabul edilmektedir [47]. Ancak tasarlanacak sistemde kullanılacak lityum-polimer pilin çıkış voltajı ortalama olarak 3,7 V'tur. Bu durumda pil voltajının yükseltilmesi için bir adet DC-DC çevirici gerekmektedir.

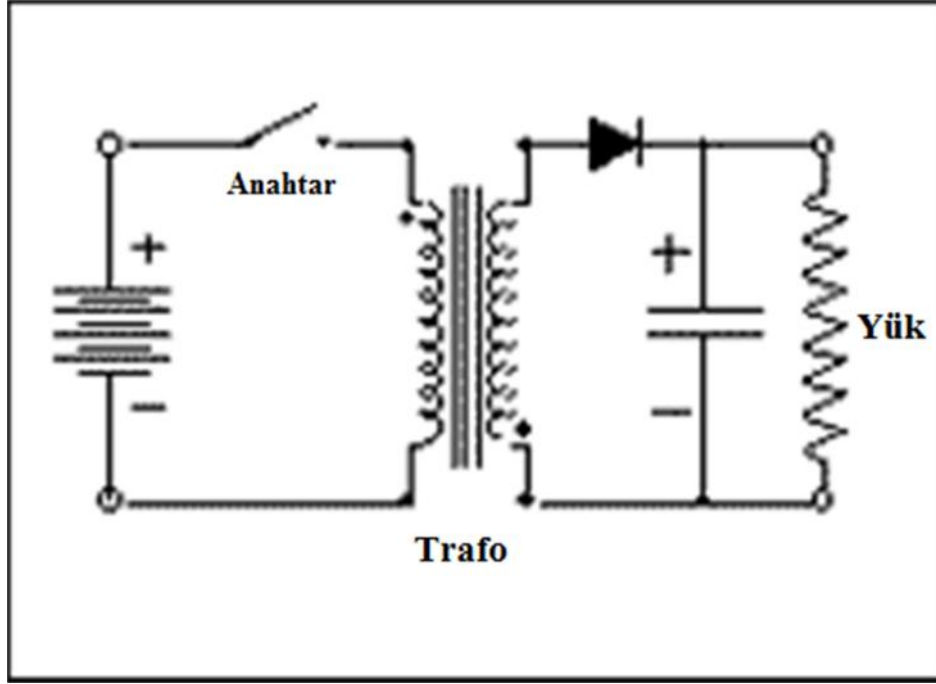
DC-DC dönüştürücüler genel olarak belirli bir seviyedeki gerilimi artırır veya azaltırlar. Özellikle mobil sistemler için değişken olan batarya voltajını bir seviyeden sabit bir seviyeye çıkarmak veya indirmek önemlidir. Bu sayede regüle bir voltaj

kaynağı da elde edilmiş olur. DC-DC dönüştürücüler temelde lineer ve switch-mode olarak ikiye ayrılırlar.

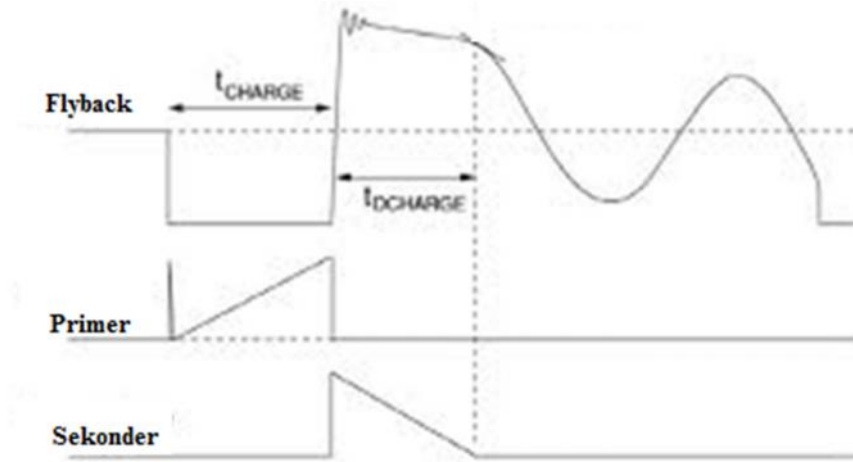
Lineer regülatörler sadece gerilimi değerini düşürme amaçlı kullanılabilirler. Ucuz ve kurulumları basit olmasına karşılık, özellikle yüksek voltaj düşümelerinde oldukça verimsizdirler ve aşırı ısı oluştururlar.

Switch-mode dönüştürücüler ise hem voltaj yükseltme hem de voltaj düşürme işlemlerini yapabilirler. Çalışma prensipleri ise elektrik enerjisini kısa süreli olarak manyetik (bobin veya trafo) veya kapasitif bir elemanda saklayıp daha sonra çıkışa aktarmalarıdır. Bu çevrim yöntemi lineer regülatörlere göre oldukça verimlidir ve aynı zamanda atık ısı üretmezler. Switch mode regülatörlerde daha fazla eleman kullanılmaktadır ve yapıları daha komplekstir ancak özellikle son yıllarda gelişen entegre devre teknolojisi ile aktif elektronik elemanları tek bir kılıfta toplanmıştır. Çeşitli switch mode teknolojileri vardır: Forward, Push-Pull, Half-Bridge, Full-Bridge ve Flyback gibi.

Özellikle FlyBack tekniği özellikle giriş ile çıkış arasında galvanik izolasyon sağlaması ile ön plana çıkmaktadır. Bu sayede özellikle medikal sektöründeki bir çok switch mode regülatör flyback tekniği ile oluşturulmaktadır. Temel olarak Flyback devresi bir anahtarlama elemanı, bir trafo, bir diyod ve çıkış kapasitesinden oluşmaktadır(Şekil 3.12). Anahtarlama elemanı belli bir frekansta trafonun primerini tetikler ve sekonderde oluşan akım çıkışa seri bağlı olan diyod üzerinden kapasitörü şarj eder. Çıkış geriliminin ayarlamak için anahtarlama sinyalinin duty-cycle oranı değiştirilir. Şekil 3.13'te Flyback devresinin çıkış akım ve voltaj değerleri verilmiştir.



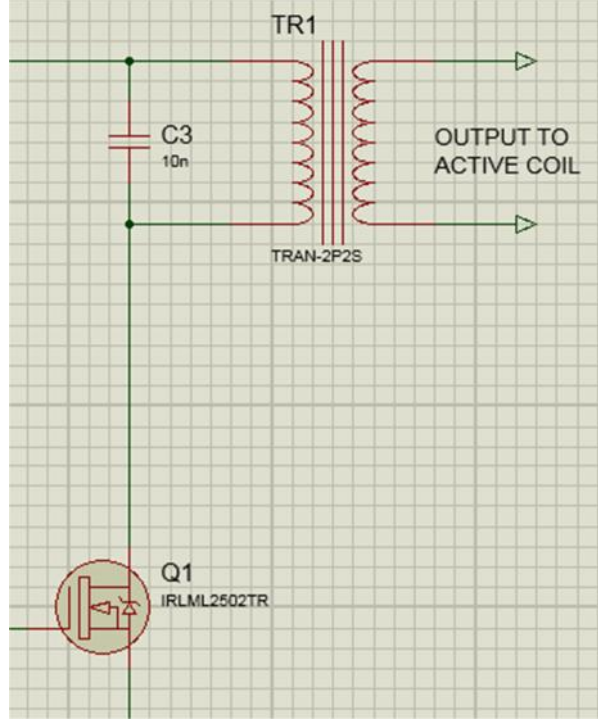
Şekil 3.12 Temel Flyback DCDC Devresi



Şekil 3.13 Flyback Devresi Voltaj ve Akım Sinyalleri

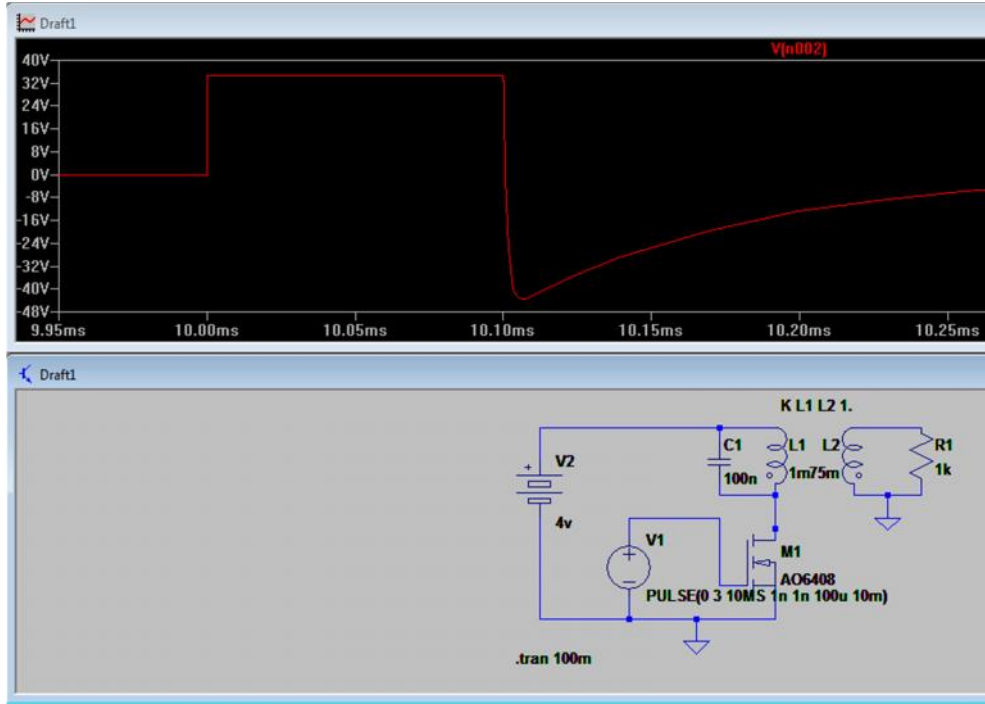
Geliştirdiğimiz sistemde negatif alternansları kesmemesi için çıkış diyodunu devreden çıkardık. Bu sayede istenilen bifazik çıkış voltajı elde edilmiş olundu. Her bir palsin süresi ve pals dizilerinden oluşan toplam stimülasyon süresi mikrodenetleyici tarafından ayarlanacaktır. Bu devre için 1:10 kazanç oranında trafo kullanılmıştır. Anahtarlama elemanı olarak IRLML2502 N kanal MOSFET kullanılmıştır. IRLML

serisi mosfetler özellikle lojik seviyedeki gerilimler iletime geçebilirler, bu sayede mikrodnetleyicilerin çıkışlarından direk olarak sürülebilir, araya herhangi bir sürücü elemana gerek duymaz. Şekil 3.14'te çıkış devresi şeması verilmiştir.

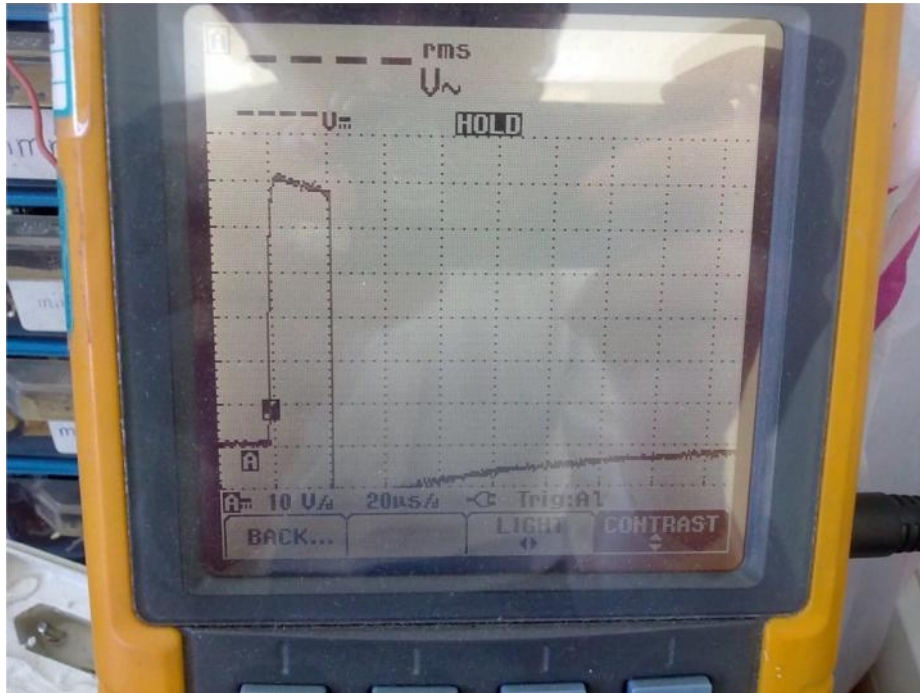


Şekil 3.14 Çıkış Devresi Şeması

1 k Ω üzerinde oluşan çıkış sinyalinin SPICE simülasyon çıkışı Şekil 3.15'de uygulama çıkışı Şekil 3.16'daki gibidir.



Şekil 3.15 Bifazik Çıkış Sinyali (SPICE)

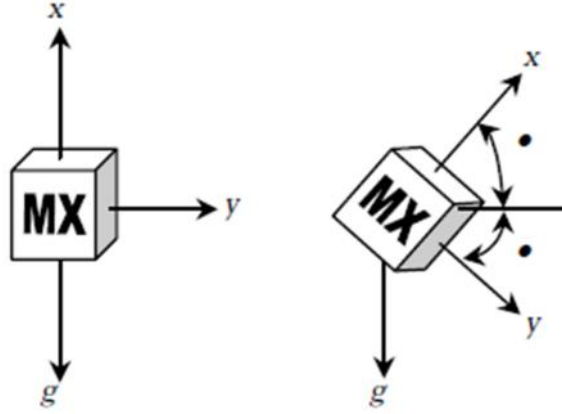


Şekil 3.16 Bifazik Çıkış Sinyali

3.5. Stimulatör Programı Algoritması

Stimulatör temel olarak ivme sensörü ile ön yükleme yürüme fazını tespit ederek belirlenen stimulasyon parametrelerine göre stimulasyonu başlatmalıdır.

Ön yükleme yürüme fazı; diz altı bölgesinin açısını tespit edilerek bulunabilir. Bu nedenle ivme sensörü ile açı tespiti yapılmalıdır. Bu konuyla ilgili literatür taraması ise ivme sensörü ile açı ölçümünün, ivme sensörünün yerçekimi ivmesine göre olan açısının değişmesi sonucunda ivme sensörünün eksenleri üstünde oluşan değişmeyi yorumlayarak yapılabileceğini göstermektedir [46]. Ön yükleme fazı özellikle düşük ayak hastaları söz konusu olunca yüksek frekanslı ve AP ekseninde yüksek ivmeli bir hareket değildir. Bu nedenle sadece yerçekimi ivmesine göre olan ivme sensöründeki eksenleri okumak yeterli olacaktır.



Şekil 3.17 İvme Sensörü yerleşimi[46]

Şekil 3.16’da gibi bir sensör yerleşiminde sensörden okunan ivme verilerine göre açı formülü her bir eksen için $\text{Alpha} = \sin^{-1}(A_x/g)$ formülüyle hesaplanmaktadır. Açı bilgisi bu sayede tek eksen ile okunabilir. Şekil 3.17’de görüldüğü üzere $\pm 60^\circ$ için sinüs fonksiyonu neredeyse lineerdir ki bu durum sistemimiz için yeterlidir. Eğer $\pm 60^\circ$ dışına çıkılmak istenirse iki eksen den ölçüm yapılması gerekmektedir çünkü 60° üstü bölgede değişim oldukça küçüktür. Bu durumda hesaplama aşağıdaki gibi olur:

$$A_x = g \cdot \sin(\delta) \quad (3.2)$$

$$A_y = g \cdot \sin(\gamma) \quad (3.3)$$

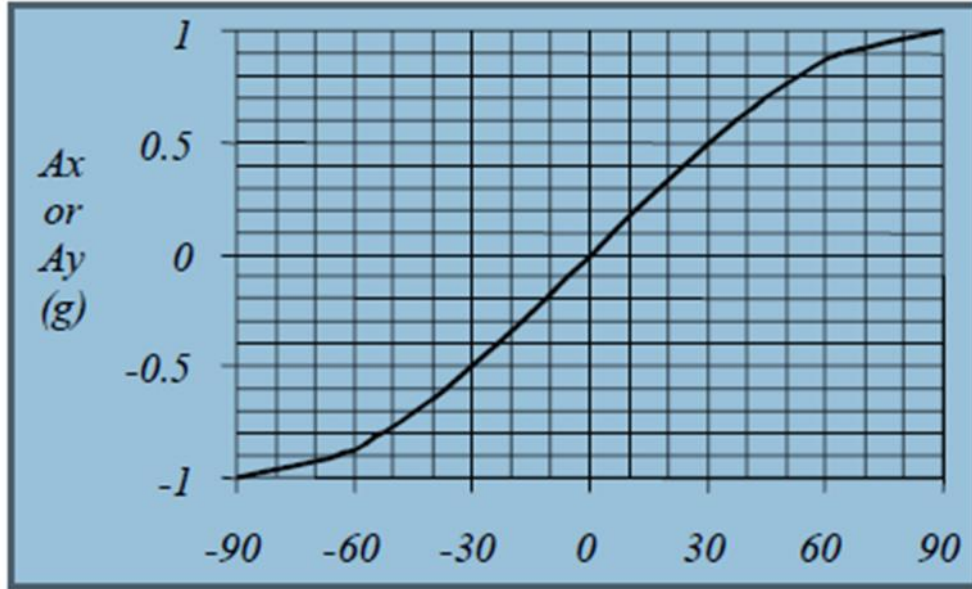
$$(\delta + \gamma) = 90^\circ \quad (3.4)$$

$$A_x = g \cdot \sin(-\gamma + 90^\circ \text{arc}) = g \cdot \cos(\gamma) \quad (3.5)$$

$$\left[\frac{A_y}{A_x}\right] = \left[\frac{g.\sin(\gamma)}{g.\cos(\gamma)}\right] = \tan(\gamma) \quad (3.6)$$

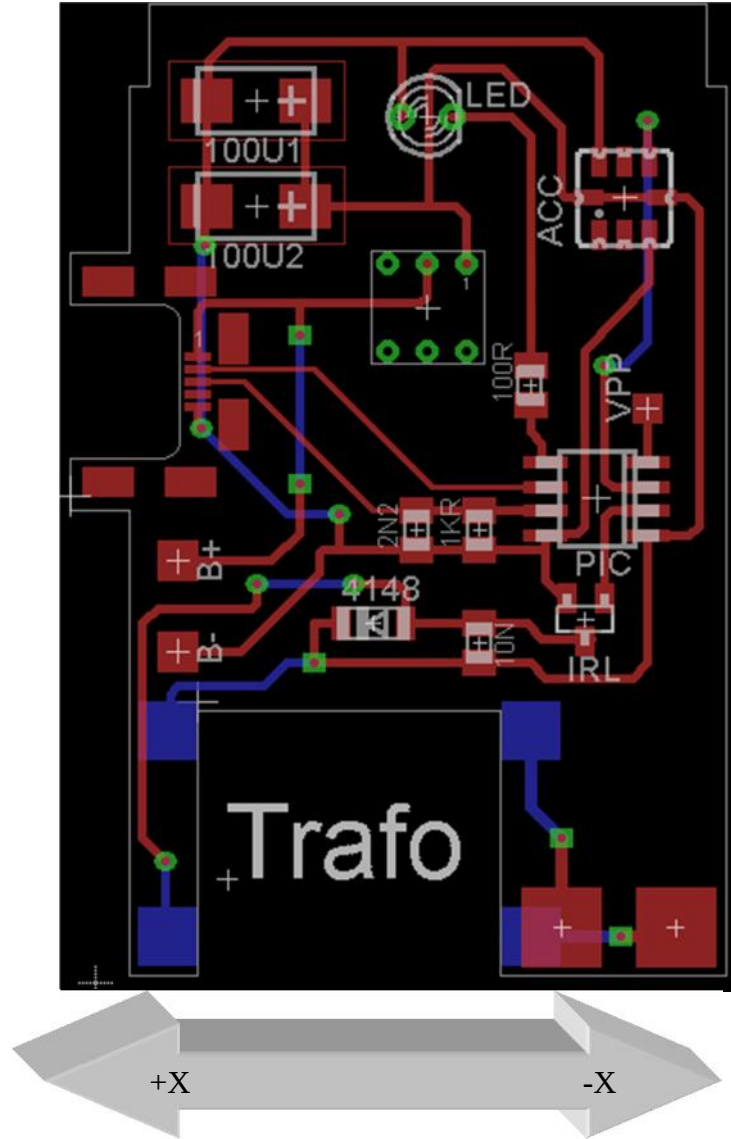
$$\gamma = \tan^{-1} \left[\frac{A_y}{A_x}\right] \quad (3.7)$$

Bu yapı özellikle yerçekimi ivmesinden bağımsızdır ve doğru bir yazılım ile dört bölgede de sonuç verir. Açı ölçümünün yerçekimi ivmesinden bağımsız olması sensörün yerleşimindeki hassasiyetinin önemini azaltır ve mikrodenetleyici yazılımının sensörün yerçekimi ivmesine göre kompanzasyon yapmasına gerek yoktur. Fakat özellikle 8-bit işlemcilerde ters tanjant fonksiyonu oldukça ağır bir kod yüküdür bunun yerine Taylor serileri veya sabit değerlerden oluşan bir tablo kullanılabilir. Çalışmamızda çift eksen ölçüm hem ağır kod yükü nedeniyle kullanılmamış tek eksen ölçüm ile sonuca gidilmiştir.



Şekil 3.18 İvme Sensörü Çıkış Eğrisi [46]

Şekil 3.19’da sağ taraf için devrenin yerleşimi verilmiş ve ivme eksenleri gösterilmiştir. Buna göre, sağ bacak adım atmaya başlayacağı zaman öncelikli olarak x+ değeri artar yani sensörün x çıkışındaki pals genişliği artacaktır. Sol taraf için bu durum tam tersi olacağı için stimulatöre hangi bacağa takıldığı da bilgisayar programı tarafından bildirilmelidir.



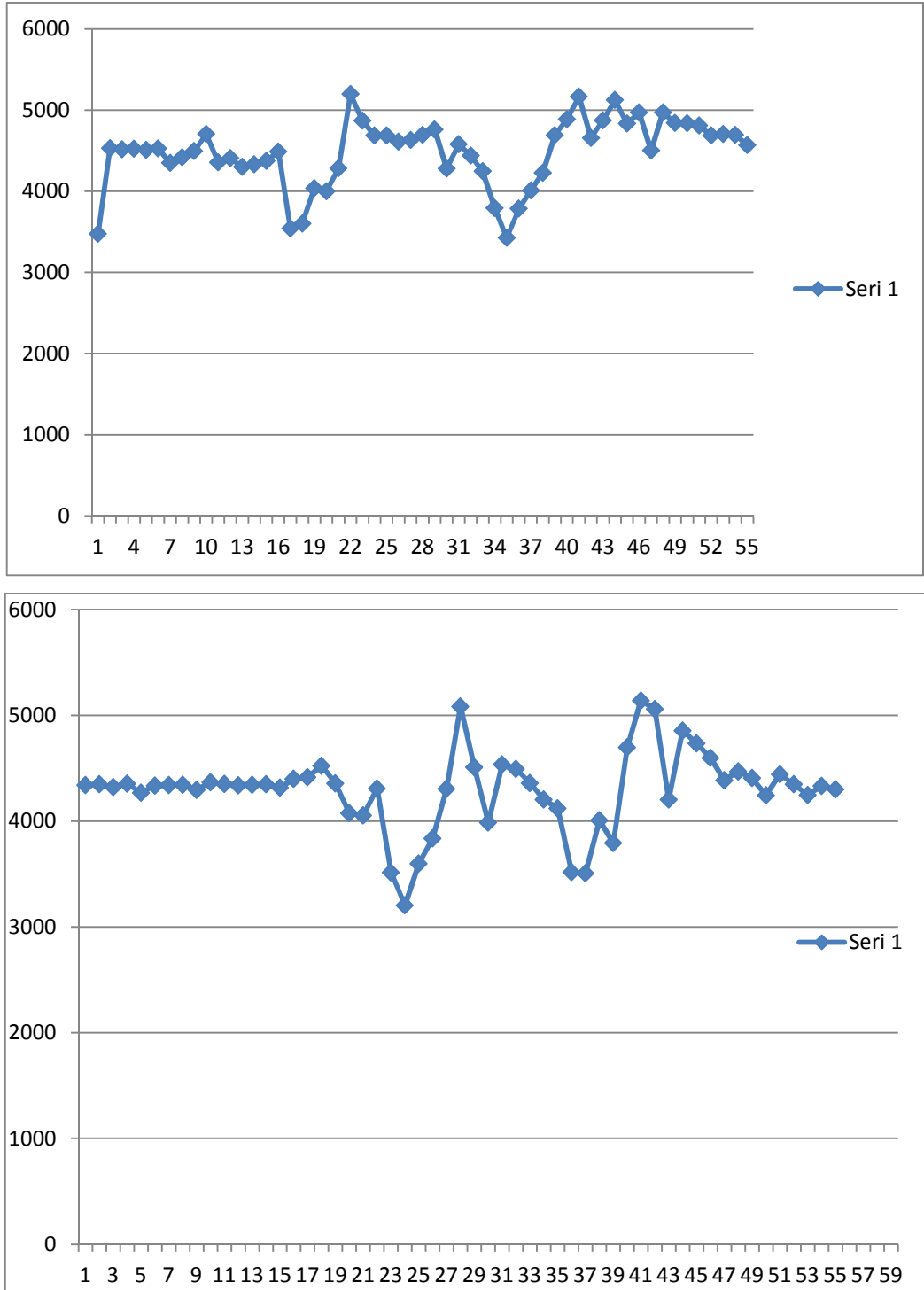
Şekil 3.19 Hazırlanan Baskıdevre ve İvme Sensörü Eksen

Kod yükünü azaltan bir diğer faktör ise pals genişliği değeri üzerinden hareket etmek olmuştur. Örneğin elde edilen değerleri g kuvvetine çevirip açı ölçmektense direk olarak pals genişliği değeri kullanılmıştır. Bu sayede mikro denetleyici gereksiz matematiksel dönüşümler yapmadan açı ölçümü yapmıştır.

Bunun dışında yazılımda ölçümün doğruluğunu artırmak için basit bir ortalama alma kullanılarak alçak geçiren filtre oluşturulmuştur. Yazılımda üç adet ölçümün ortalaması alınmıştır, bu sayede sistem gürültülerden daha az etkilenmektedir. Sistemin hızı ise 33,3 Hz olmuştur.

Her ne kadar tek eksen ivme sensörü ile açı ölçümü konusunda başarılı olursa da uygulamayı geliştirirken başka bir sorun belirmiştir. Ayak yere vurduğu zaman tüm eksenleri etkileyen bir titreşim kuvveti meydana gelmektedir. Bu durumda X değerinde oluşan değişimler mikrodenetleyici tarafından bir açı değişimi olarak algılanabilir. Her ne kadar ortalama alınarak veriler filtrelense de ve bakıldığı zaman ortalama ivme değeri sıfır olsa da ivme sensörü bazı hareketleri kaçırmaktadır ve bu durumda yazılımın gidişatını bozmaktadır. Bu sorunu çözmek için ilk temas fazı da programa dâhil edilmiştir. Program önce açığı hesaplayarak stimulasyonu başlatmakta ve sonra ilk temas fazını beklemektedir. İlk temas fazı programda “fren” olarak adlandırılmıştır. Fren bekleme sistemi basitçe bir biti setleyerek elde edilmiştir. Stimulasyon başladıktan sonra fren biti birleşmiş ve bu andan sonra gelen büyük ivme değişimi ile fren biti sıfırlanmıştır. Fren biti sıfırlandıktan sonra 200mS’lik sabit bir gecikme konulmuştur.

Şekil 3.20’de DFS sistemi ile yapılan ivme değerleri ölçümleri verilmiştir. Görüldüğü üzere ilk başta herhangi bir adım hareketi olmadığı için düz bir çizgi görülmektedir. Daha sonra ters bacağın adım hareketi ivm değerinde ufak bir dalgalanma olmuş ölçüm bacağının hareketlenmesi ile ivme ölçümlerinde bir düşme olmuştur. Salınım süresince açının yerçekimine göre değişmesi nedeniyle ivme değerlerinde artış olmuştur. Ayağın yere vurması ile pozitif bir pik ivme değeri ölçülmüştür. Daha sonra tekrardan ters bacak hareket etmiş ve ivme de küçük bir düşme gözlemlenmiştir ve tekrar ölçüm yaağının hareketi ile yüksek bir negatif ivme oluşmuş ve salınım süresince pozitif bir artış gözlemlenmiş, pozitif bir pik değerle hareket sonlamıştır. Bu pozitif pik değerler yukarıda anlatılan fren hareketini simgelemektedir. Bu fren hareketi sayesinde stimulator yeni adım için bekleme durumuna geçmektedir. Diğer bir önemli durumda ters bacağın yarattığı ivme dalgalanmasıdır. Bu dalgalanma küçük seviyede olduğu için yazılımda belli bir ivme değerinin üstündeki değişimler dikkate alınmıştır. Son bölümde oluşan ivme değerleri dikkate alınmamalıdır, çünkü o sırada sensörü sökme durumuna geçilmiştir o bölgedeki ölçümler dikkate alınmayacaktır.

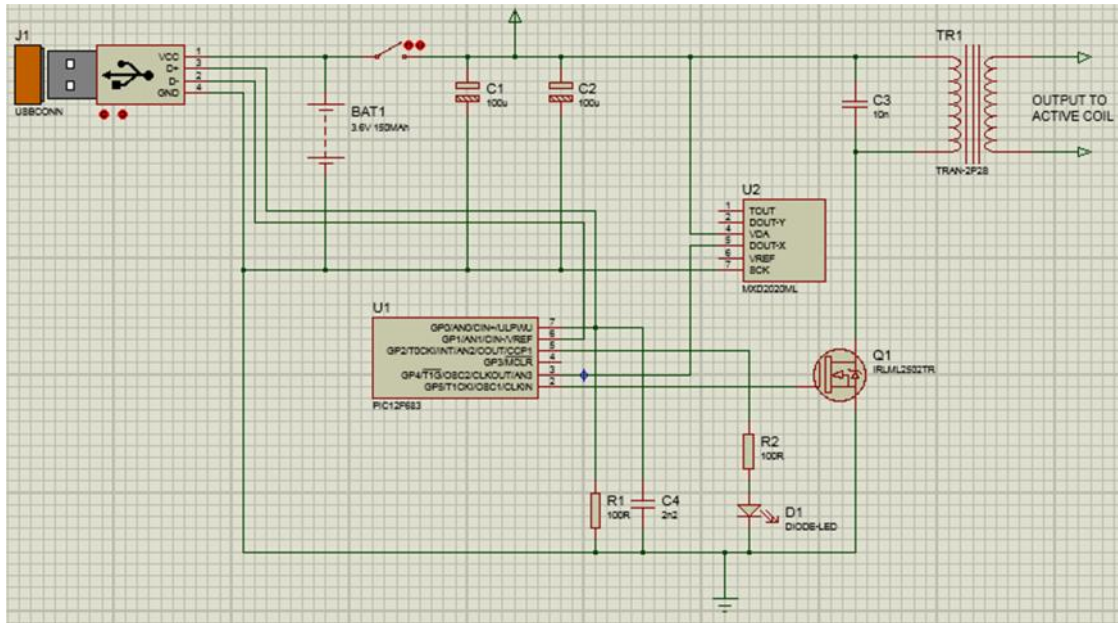


Şekil 3.20 Yürüme Analizi

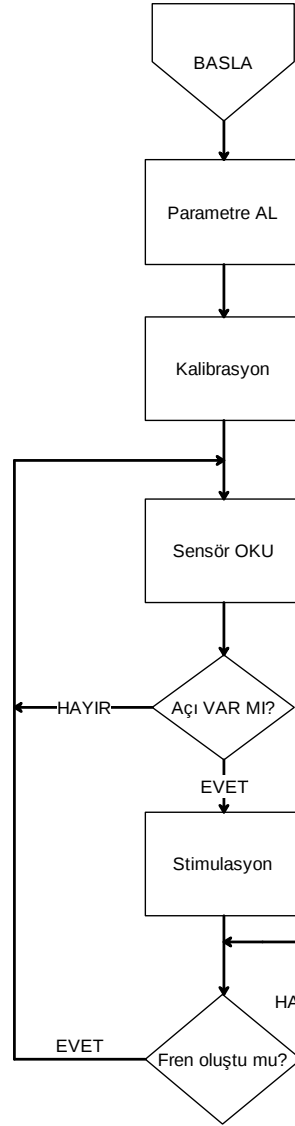
Program ilk açıldığı zaman ilk olarak EEPROM hafızadaki parametreleri okur. Bu parametreler stimülasyon pals genişliği, toplam stimülasyon süresi ve stimülastörün hangi bacağı takılı olduğudur. Parametreler alındıktan sonra mikrodentleyici kalibrasyon işlemine geçer. Kalibrasyonda 100 adet değer okunup ortalaması

alınmaktadır. Kalibrasyon işlemi sensörün X eksen çıkışının açılış değeri bulunmaktadır ve bu sayede hastanın yerçekimine göre stimulatörü yaklaşık olarak ne kadar hata ile yerleştirdiği bulunmaktadır. İdeal olarak Stimulatör yerçekimine tam dik ekseninde takılması gerekse de pratikte bu durum söz konusu değildir. Kalibrasyonunda yaptıktan sonra stimulatör sürekli olarak ölçüm yapmakta ve açısal değişimi ve onu takip eden freni beklemektedir.

Bunların dışında USB haberleşmesini takip edebilmek için bir kesme konmuştur. İşlemcinin 7 numaralı ucu, haberleşme kablosu takıldığı zaman 3,3v'a yükselir ki bu durum bir yükselen kenardır. PIC12F683'ün tüm bacakları kesme kaynağı olarak kullanılabildiği için 7 numaralı bacak üzerinde oluşturulan bir kesme ile USB haberleşme kablosu takıldığı zaman bir kesme oluşur ve stimulatör ana döngüsünden çıkarak haberleşme fonksiyonuna girer. 7 numaralı bacağı 1k ohm ve 2,2nF'lık basit bir RC filtresi konulmuştur bu sayede elektriksel gürültüler nedeniyle gereksiz yere kesme oluşmasının önüne geçilmiştir. Sistem bir anahtar ile açılıp kapatılmaktadır. Stimulatör şematiği Şekil 3.21'de ki gibidir.



Şekil 3.21 DFS Sistemi Şeması



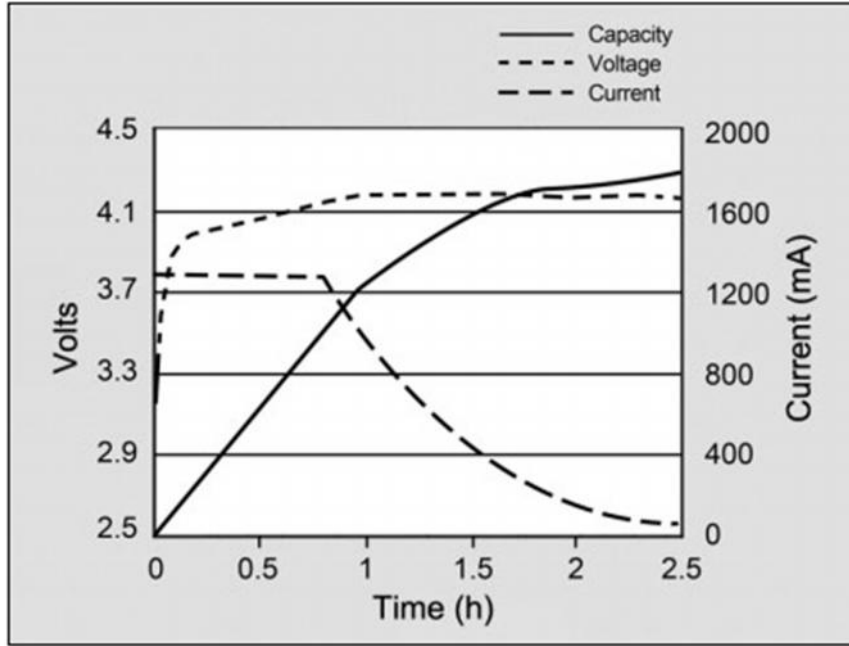
Şekil 3.22 Program Algoritması Akış Şeması

3.6. Şarj Cihazı

Sistemin taşınabilir olması için hafif bir pil sistemine ihtiyaç vardır. Mevcut ürünler standar AA, AAA, 9Vluk pilleri kullanmaktadır. Bu pillerin avantajı ürün ile birlikte herhangi bir şarj cihazına bağlı olmamaları ve pil bittiği zaman hemen yenisi ile değiştirilebilir. Ancak lityum-iyon tabanlı piller diğer pillere göre daha yüksek enerji yoğunluğuna sahiptir[43]. Bu özelliğiyle li-ion piller aynı ağırlıkta daha yüksek enerji depolayabilirler. Fakat doğru bir şarj terminasyonu izlenmezse bu pillerin hücreleri çok çabuk ölür ve hatta pilin patlamasına bile neden olabilir.

Varta firmasının LPP402025 kod numaralı pil boyut olarak uygulama için idealdir. Pilin boyutları 25x20x4.3mm, ağırlığı 3,7 gramdır[18] Pilin hızlı şarj akımı 140mA, normal şarj akımı 70mA'dır. Pilin kapasitesi 150 mAh'dır.

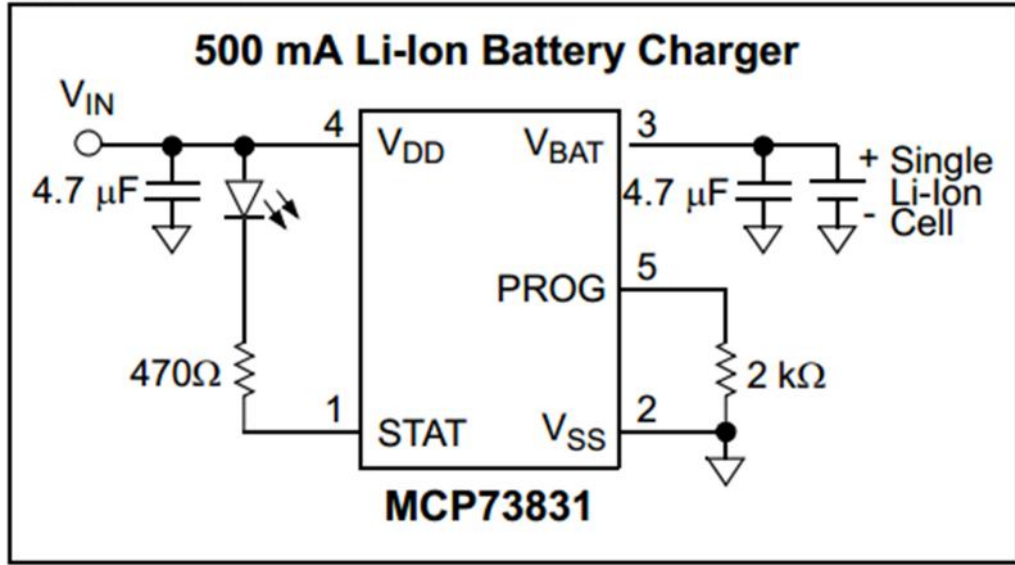
Li-ion pillerin doğru bir şekilde şarj edilebilmesi için “sabit akım-sabit voltaj” tekniği izlenmelidir. Bu teknikte pil voltajı belirlenen üst limite kadar sabit bir akım ile şarj edilmelidir. Üst limit voltajı ise maksimum 4.2V seçilebilir, kimi uygulamalarda pilin ömrünü uzatmak için 4.1V seçilmektedir. Üst limit voltajına ulaşıldığı zaman ise şarj akımı azaltılır ve belli bir seviyenin altında şarj işlemi durdurulur. Şekil 3.23’te şarj tekniğinin voltaj ve akım grafiği verilmiştir.



Şekil 3.23 Li-Ion Pil Şarj Eğrileri[43]

Sabit akım-sabit voltaj tekniği 5-6 elektronik komponent ve bir mikroişlemci sistemi ile gerçekleştirilebilir. Ancak Li-ion piller için hazır pil şarj entegreleri vardır. Bu entegreler pili doğru bir şekilde şarj eder ve sürekli olarak pil voltajını izlerler. Ayrıca yapılarında bulunan çeşitli koruma devreleri de vardır. Örneğin yüksek güçlü sistemler için bit NTC yardımı ile pilin ısısı takip edilebilmektedir.

Uygulamamızda Microchip firmasının ürettiği MCP73831 li-ion pil şarj entegresi kullanılmıştır. MCP73831 500mA'e kadar akımı basabilir, harici bir FET'e ihtiyaç duymaz. Bu özelliği birçok uygulama için yeterlidir. Dahili akım ölçümü vardır bu sayede belirlenen akımın üstüne çıkılmaz. Bir direnç ile şarj akımı ayarlanabilir. Ayrıca ters boşalmaya karşı koruma devresi de vardır. Elle bile kolay lehimlenebilecek SOT-23-5 kılıfıta üretilmektedir[8]. MCP73831 tek bir led ile şarj durumu gösterir. Sadece şarj süresince led yanar diğer durumlarda sönmüştür. MCP73831'e ait uygulama şeması Şekil 3.24'te verilmiştir.

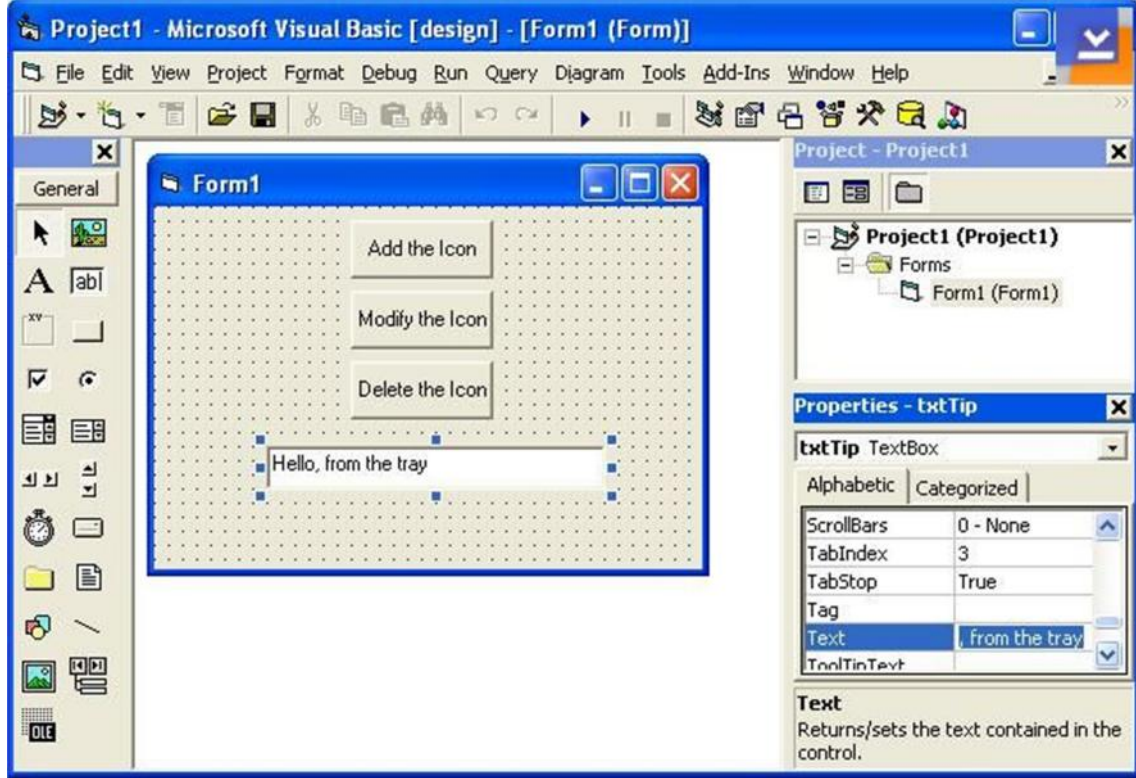


Şekil 3.24 MCP73831 Uygulama Şeması[8]

3.7. Bilgisayar Programı

Stimulasyon parametrelerinin ayarlanabilmesi amacıyla geliştirilen bilgisayar programı Microsoft Visual BASIC ile derlenmiştir. Visual BASIC özellikle nesne yönelim konusunda dünyanın en kolay dilidir. Windows işletim sistemi üzerindeki çeşitli bileşenlerin kullanımı oldukça kolaydır. Ayrıca ActiveX Component gibi bir özelliği kullanma yeteneğine sahiptir. Yani bir işi yapabilecek bir programcık için saatlerce uğraşıp yaptığınızda bunu ActiveX olarak yaparsanız daha sonraları aynı özelliği kullanmanız çok kolaylaşır.

Visual BASIC geliştirme ortamında kullanıcı arayüzü ile ilgili tüm nesneler sürükleyip bırakarak kolayca ayarlanabilir. Ayrıca yerleştirilen nesnelerin olaylarına (event) ulaşım oldukça kolaydır C++'ta ki gibi bir sinyal-slot mekanizması gerektirmez(Şekil 3.25).

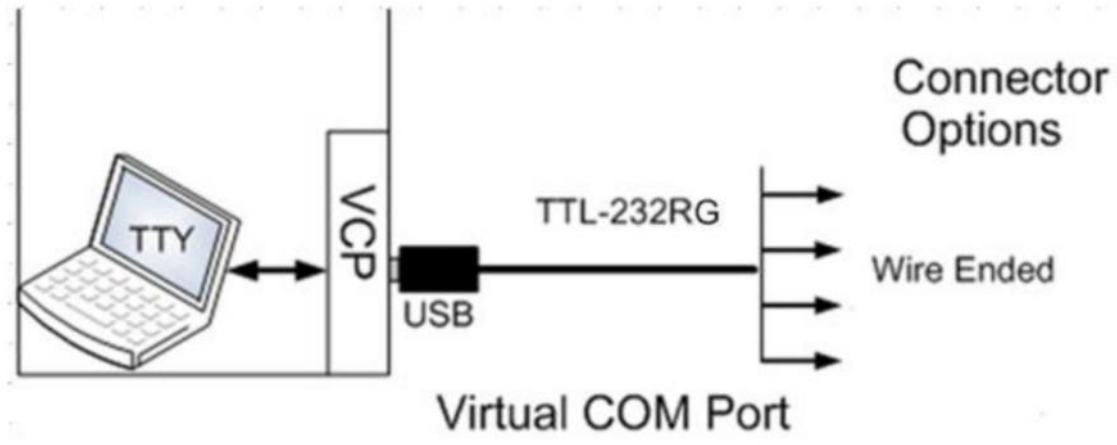


Şekil 3.25 Visual Basic Kod Geliştirme Ortamı

3.7.1. Haberleşme Sistemi

Bilgisayar tarafında USB donanımı üzerinden haberleşmesi yapılması çokta zor olmasa da mikrodnetleyici açısından oldukça zordur. Ayrıca USB donanımını destekleyen PIC mikrodnetleyicilerinin kılıfları oldukça büyüktür ve güç sarfiyatları yüksektir. USB donanımı çözülsede mikrodnetleyiciler için USB protokol kodu yazmak veya hazır bir kod kullanmakta oldukça kompleks bir işlemdir.

Bu sorunu aşmak için FTDI firmasının geliştirdiği USB entegreleri yaygınca kullanılmaktadır. Bu entegreler ile USB donanımı üzerinde yazılımsal olarak seri port oluşturulmaktadır. Seri-port ise programlaması oldukça kolay bir yapıdır. Her ne kadar donanımsal zafiyetleri olsa da USB donanım yapısı üstünde oluşturulduğu için oldukça güvenlidir ve Windows işletim sistemi üzerinde ekstra bir sürücü yüklemeyen direk olarak çalışır. Hem mikrodnetleyici tarafında hem de bilgisayar tarafında seri port programlaması oldukça kolay bir yapıdır. Çalışmamızda FTDI firmasının hazır USB-Seri Port çevirici kablosu TTL-232RG-VREG3V3-WE kullanılmıştır. Kablonun çıkışları TTL seviyededir bu sayede mikrodnetleyici pinlerine direk olarak bağlanabilir(Şekil 3.26).



Şekil 3.26 Sanal Seri Port Bağlantı Şeması

3.7.2. Bilgisayar Programı Arayüzü

Çalışma kapsamında tasarlanan program arayüzü Şekil 3.27’de ki gibidir. Program basit bir arayüze sahiptir. Hasta adı, fizyoterapist adı ve hastalığın adı gibi temel bilgilerin yanı sıra stimülasyon parametreleri olarak pals genişliği (0-100 mikro saniye arası) ve toplam stimülasyon periyodu (200-400mS arasında) seçilebilmektedir. Ayrıca sensörden alınan verilerin doğru yorumlanabilmesi için hangi bacağa takıldığının da belirtilmesi için verilmektedir ki buda programda seçilebilmektedir. Tüm veriler mikrodenetleyici EEPROM hafızasında kaydedilmektedir. “READ” butonu verileri okuma “WRITE” butonu verileri yazma amaçlı kullanılmaktadır.

NeuroTulgar

New Disconnect Calibration Exit

PATIENT'S NAME

PHYSIOTHERAPIST'S NAME

DISEASE

FOOT

DURATION OF EACH STIMULATION

INTENSITY

Şekil 3.27 Stimulator Kontrol Programı Arayüzü

4. SONUÇLAR VE DEĞERLENDİRME

Bu tezde ivme sensörü kullanılarak taşınabilir hafif bir düşük ayak stimulatörü tasarlanmıştır. Geliştirilen Stimulatör 6.5 cm x 4.1 cm x 1.6 cm boyutlarındadır. Bu özelliği ile mevcut stimulatörler arasında en küçüğüdür. Deri üstü stimülasyon uygulaması nedeniyle oldukça kolay bir şekilde uygulanabilir. Stimülasyon parametreleri ise herhangi bir Windows işletim sistemine sahip bir bilgisayarda yapılabilir.

Sistem birkaç hastaya uygulanmış ve başarılı sonuçlar elde edilmiştir fakat şuana kadar herhangi bir klinik çalışmaya tabii tutulamadığı için elimizde matematiksel veriler yoktur, sadece hastaların görüşleri dikkate alınmıştır.

Yapılan uygulamalarda klinik açıdan görülen en büyük eksiklik; sistemin kablosuz veri transferine sahip olmaması nedeniyle her ayar için stimulatörün hasta üzerinden sökülüp bilgisayara bağlanması gerektiğidir. İleriki dönem çalışmalar için kablosuz veri transferi özelliği eklenmelidir.

Bir diğer konu ise tek eksen üzerinden yürüme fazlarının tespiti. Bu durum bazen sistemin adım kaçırmaya neden olmaktadır. Önüne geçmek için ise iki yada üç eksen ölçüm yapılmalı ve ivme sensörünün yanına jiroskopta kullanılmalıdır. Bu durumda daha güçlü bir mikrodenetleyiciye geçilmelidir.

Güç katında istenilen bifazik akım elde edilmiştir. Ancak özellikle ileri derece de olan hastalarda stimülasyon sinyali yetersiz kalmaya başlamaktadır.

Yapılan çalışmanın sonucu olarak tasarlanan Stimulatör; akredite bir kuruluşun incelemesine tabii tutulmuş ve CE belgesi almaya hak kazanmıştır. Ayrıca Sağlık Bakanlığı'nın SUT uygulamasındaki standartları karşılamaktadır ve mevcut uygulamaya göre 2000TL'ye kadar destek kapsamındadır.

KAYNAKLAR

- [1] 2. Mackay J, Mensah G (eds.); “The atlas of heart disease and stroke, Global burden of stroke”, Dünya Sağlık Örgütü (WHO), (2004). www.who.int/cardiovascular_diseases/en/cvd_atlas_15_burden_stroke.pdf. Erişim 13/09/11
- [2] Royal College of Physicians Stroke; Towards better management, London; Royal College of Physicians, (1989)
- [3] Wade D, Wood V, Heller A, Maggs J, Langton Hewer R. Walking after stroke; measurement and recovery over the first 3 months. Scand J Rehabil Med (1987), 19:25-30.
- [4] H.S. Jørgensen, H. Nakayama, H.O. Raaschou, and T.S. Olson; “Recovery of walking function in stroke patients the copenhagen stroke study”, Arch. Phys. Med. Rehabil., 76(1):27–32,
- [5] P.N. Taylor, J. Burridge, A.L. Dunkerley, D.E. Wood, J.A. Norton, C. Sinleton, and I.D. Swain. Clinical Use of the Odsctock Dropped Foot Stimulator: Its Effects on the Speed and Effort of Walking. Arch. Phys. Med. Rehabil., 80:1577–1583, 1999.
- [6] Negard, Nils-Otto. Controlled FES-assisted gait training for hemiplegic stroke patients based on inertial sensors. Diss. Universitätsbibliothek, 2009. Burridge JH, Haugland M, Larsen B, Pickering RM, Svaneborg N, Iversen HK, et al. Phase II trial to evaluate the ActiGait implanted drop-foot stimulator in established hemiplegia. Journal of Rehabilitation Medicine . 2007. 39 (3):212 -8
- [7] <http://ww1.microchip.com/downloads/en/devicedoc/21984e.pdf>
- [8] Liberson W, Holmquest H, Scott M. Functional electrotherapy: stimulation of the common peroneal nerve synchronized with the swing phase of gait of hemiplegic subjects. Arch Phys Med Rehabil 1961; 42: 202-5.
- [9] Burridge J, Taylor P, Hagan SA, Wood DE, Swain ID. The effect on the spasticity of the quadriceps muscles of stimulation of the common peroneal nerve of chronic hemiplegic subjects during walking. Physiotherapy 1997; 83: 82-9.

- [10] Taylor PN, Burridge JH, Dunkerley AL, Lamb A, Wood DE, Norton JA, et al. Patient's perceptions of the Odstock dropped foot stimulator (ODFS). *Clin Rehabil* 1999; 13: 333-40.
- [11] Belanger, M., Stein, R.B., Wheeler, G.D., Gordon, T., and Leduc, B. 2000. Electrical stimulation: can it increase muscle strength and reverse osteopenia in spinal cord injured individuals? *Arch. Phys. Med. Rehabil.* 81: 1090–1098.
- [12] Mirbagheri, M.M., Ladouceur, M., Barbeau, H., and Kearney, R.E. 2002. The effects of long-term FES-assisted walking on intrinsic and reflex dynamic stiffness in spastic spinal-cord-injured subjects. *IEEE Trans. Neural Syst. Rehabil. Eng.* 10: 280–289, (2002)
- [13] Stefanovska, A., Vodovnik, L., Gros, N., Rebersek, S., and Acimovic-Janezic, R. 1989. FES and spasticity. *IEEE Trans. Biomed. Eng.* 36: 738–745.
- [14] Faghri, P.D., Glaser, R.M., and Figoni, S.F. 1992. Functional electrical stimulation leg cycle ergometer exercise: training effects on cardiorespiratory responses of spinal cord injured subjects at rest and during submaximal exercise. *Arch. Phys. Med. Rehabil.* 73: 1085–1093.
- [15] Phillips, W.T., Kiratli, B.J., Sarkarati, M., Weraarchakul, G., Myers, J., Franklin, B.A., Parkash, I., and Froelicher, V. 1998. Effect of spinal cord injury on the heart and cardiovascular fitness. *Curr. Probl. Cardiol.* 23: 641–716.
- [16] Wheeler, G.D., Andrews, B., Lederer, R., Davoodi, R., Natho, K., Weiss, C., Jeon, J., Bhambhani, Y., and Steadward, R.D. 2002. Functional electric stimulation-assisted rowing: increasing cardiovascular fitness through functional electric stimulation rowing training in persons with spinal cord injury. *Arch. Phys. Med. Rehabil.* 83: 1093–1099.
- [17] http://www.varta-microbattery.com/applications/mb_data/documents/data_sheets/DS56416.pdf
- [18] Stein, R.B. 1997. Assembly for functional electrical stimulation during movement. USA Patent No. 5, 643, 332. U.S. Patent and Trademark Office, Alexandria, Va.
- [19] Stein, R.B. 1998. Assembly for functional electrical stimulation during movement. Continuation in part, USA Patent No. 5,814,093. U.S. Patent and Trademark Office, Alexandria, Va.

- [20] Weber, D. J., et al. "Functional electrical stimulation using microstimulators to correct foot drop: a case study." *Canadian journal of physiology and pharmacology* 82.8-9 (2004): 784-792.
- [21] J. Perry. *Gait Analysis: Normal and Pathological Function*. SLACK Incorporated, 1992.
- [22] A. Mansfield and G.M. Lyons. The use of accelerometry to detect heel contact events for use as a sensor in fes assisted walking. *Med. Eng & Phys.*, 25:879–866, 2003.
- [23] P.N. Taylor, J. Burridge, A.L. Dunkerley, A. Lamb, D.E. Wood, J.A. Norton, and I.D. Swain. Patients' perceptions of the Odstock Dropped Foot Stimulator (ODFS). *Clinical Rehabilitation*, 13:439–446, 1999.
- [24] Y.-L. Chen, Y.-C. Li, T.-S. Kuo, and J.-S. Lai. The Development of a Closed-loop Controlled Functional Electrical Stimulation (FES) in Gait Training. *Journal of Medical Engineering & Technology*, 25(2):41–48, Mar./Apr. 2001.
- [25] T. Fuhr and G. Schmidt. Neue Ansätze zur Steuerung und Regelung einer kooperativen Gang-Neuroprothese. *at - Automatisierungstechnik*, 50(7), Jul. 2002.
- [26] N. Fiskekovic and D.B. Popovic. New controller for functional electrical stimulation systems. *Med. Eng & Phys.*, 23:391–399, 2001.
- [27] A.Th.M. Willemsen, F. Bloemhog, and H.B.K. Boom. Automatic stance-swing phase detection from accelerometer data for peroneal nerve stimulation. *IEEE Trans. Biom. Eng.*, 37(12):1201–1208, Dec. 1990.
- [28] R. Williamson and B. Andrews. Gait Event Detection for FES Using Accelerometers and Supervised Machine Learning. *IEEE Trans. Rehab. Eng.*, 8(3):312–319, Sep. 2000.
- [29] S.K. Ng and H.J. Chizeck. Fuzzy Model Identification For Classification of Gait Events in Paraplegics. *IEEE Trans. Fuzzy and Systems*, 5(4):536–544, Nov 1997.
- [30] D. Kotiadis, H. Hermens, P.Veltink, and P. Slyke. Inertial gait phase detection system: Design. In *Proceedings of the 9th Ann. IFESS Conference*, Bournemouth, UK, 2004.
- [31] A.M. Sabatini, C.Martelloni, S. Scapellato, and F. Cavallo. Assessment of Walking features From Foot Inertial Sensing. *IEEE Trans. Biom. Eng.*, 52(3):486–494, Mar. 2005.

- [32] K.D. Strange and J.A. Hoffer. Gait phase information provided by sensory nerve activity during walking: applicability as state controller feedback for fes. IEEE Trans. Biomed. Eng., 46(7):797–809, 1999.
- [33] M. Hansen, M.K.Haugland, and F. Sepulveda. Feasibility of Using Peroneal Nerve Recordings for Deriving Stimulation Timing in a Foot Drop Correction System. Neuromodulation, 6(1):68–77,2003.
- [34] J. Chae, Z.-P. Fang, M. Walker, S. Pourmehdi, and J. Knutson. Intramuscular Electromyographically Controlled Neuromuscular Electrical Stimulation for Ankle Dorsiflexion Recovery in Chronic Hemiplegia. American Journal of Physical Medicine & Rehabilitation, 80(11):842–847, nov2001.
- [35] R. Dai, R.B. Stein, B.J. Andrews, K.B. James, and M. Wickler. Application of Tilt Sensors in Functional Electrical Stimulation. IEEE Trans. Rehab. Eng., 4(4):62–72, 1996.
- [36] I.P.I. Papas, M.R. Popovic, T. Keller, V. Dietz, and M. Morari. A Reliable Gait Phase Detection System. IEEE Trans. Neural Syst. Rehabil. Eng., 9(2):113–125, June 2001.
- [37] A.M. Sabatini, C.Martelloni, S. Scapellato, and F. Cavallo. Assessment ofWalking features From Foot Inertial Sensing. IEEE Trans. Biom. Eng., 52(3):486–494, Mar. 2005.
- [38] Embedded Systems: http://en.wikipedia.org/wiki/Embedded_system, (10.01.2009).
- [39] PIC Microcontrollers.: http://en.wikipedia.org/wiki/PIC_microcontrollers, (11.03.2009).
- [40] 8-bit PIC Microcontroller Products.: www.microchip.com/8bit, (11.03.2009).
- [41] PIC12F683 Datasheet
[http://ww1.microchip.com/downloads/en/devicedoc/41211d .pdf](http://ww1.microchip.com/downloads/en/devicedoc/41211d.pdf)
- [42] http://batteryuniversity.com/learn/article/whats_the_best_battery
- [43] M.C. Roco. “A Frontier for Engineering,”Mech.Eng.123, January, pp. 52–55, (2001).
- [44] MXD2020E Datasheet
http://www.memsic.com/userfiles/files/Datasheets/Accelerometer-Datasheets/MXD2020E_FRevH.pdf
- [45] Inclination Sensing Using Thermal Accelerometers
<http://www.memsic.com/userfiles/files/Application-Notes/an-00mx-007.pdf>

- [46] Ilic, M., Vasiljevic, D., Popovic, D.B. A programmable electronic stimulator for FES systems Rehabilitation Engineering, IEEE Transactions on (Volume:2 , Issue: 4) 234 - 239, Dec 1994

EKLER

EK-A

STIMULATOR PROGRAMI

```
// Önışlemci ayarları
#pragma disablecontexsaving
#define fet GPIO.GP5
#define tedavi_led GPIO.GP2
#include <built_in.h>
```

```
// Global protipler
void puls();
void signal();
void hesap();
void merhaba();
void veri_gonder();
void veri_al();
void e2_yaz();
void e2_oku();
void pc_pic();
void pic_pc();
void pic_reset();
void seri_haberlesme();
void cikis();
void kalibre();
void olcum();
```

```

void b_olcum();
void delta_ayar();

// Intensity Değişkenleri
unsigned short intensity,tur,bacak;
unsigned int i_PV,b_PV;

// DEĞİŞKENLER
unsigned int T1X,mg,im_g;
unsigned int toplam;
int d_mg,delta;
unsigned short fren;

// Veri Alışveriş Değişkenleri
unsigned short err,iveri;
char i_veri[30],veri[4];

// E2 Değişkenleri
unsigned short kk,e2oku_sayac,e2yaz_sayac,son_yer ;

////////////////////////////////////
////////////////// KESME //////////////////
void interrupt() {
unsigned short z=0;
INTCON.GPIE =0;

Soft_UART_Init(&GPIO, 0, 1, 2400, 0);
INTCON=0;                // Kesmeler kapandı
p_mod:                   // Programlama Mod
for(z=0;z<3;z++){        //
tedavi_led=1;            //
delay_ms(200);           // Prog Mod

```

```

tedavi_led=0;           // Anime
delay_ms(200);         //
}                       //

merhaba();
tedavi_led=1;          // Haberleşmeye hazır
seri_haberlesme();     // Haberleşme yapıldı
goto p_mod;            // Yeniden Haberleşme

}

```

```

void main() {
// AYARLAR
unsigned short k;
asm CLRWDT;
OPTION_REG=0b11001111; // Pullups kapalı, yükselen kenar tetikleme, WDT
açık ps=128
WDTCON.SWDTEN = 0;     // WDT kapalı
OSCCON=0b01100111;    // OSC 4mhz internal
TRISIO=0b00011001;    // 1-2-5 çıkış 0-3-4 giriş
GPIO=0;                // PORT temizlendi
ANSEL=0;                // ADC Fosc/2 clock, Tüm pinler dijital
ADCON0.ADON=0;         // ADC kapalı
CMCON0=0x07;           // Comparator kapalı
CCP1CON=0x00;          // Capture PWM kapalı
VRCON.VREN=0;          // CVREF kapalı

// KESME AYARLARI
INTCON=0b11001000;    // Kesme açık, IOC açık
IOC=1;                 // Sadece pin 0 da IOC açık

```

```

// PROGRAM BAŞLIK
tedavi_led=1;           // Cihaz açıldı, Kaibrasyon ve Hesaplama yapılacak

// HESAP
hesap();               // Hesaplamalar yapıldı

// KALİBRASYON
delay_ms(200);
kalibre();             // Kalibrasyon

// ANA PROGRAM
tedavi_led=0;          // Kalibrasyon ve Hesaplamalar yapıldı

ana_dongu:
fren=0;
while(1) {
don:

// ileri hareket
b_olcum();

switch(bacak){
case 10:d_mg=mg-toplam;    // sol bacak ileri hareket
        break;
case 20:d_mg=toplam-mg;    // sağ bacak ileri hareket
        break;
default:d_mg=mg-toplam;    // sol bacak ileri hareket
}

```

```

if (d_mg > delta & fren==0){    // eğer deltadan büyük g değişimi algılandı ise
    signal();                    // sinyal gönder
    delay_ms(500);               // 0.5 sn bekle (adım süresi)
    fren=1;                      // frenleme bekle
}
else if (d_mg > delta & fren==1){ // frenleme g bekle
    fren=0;                      // fren bitini sıfırla
    delay_ms(200);               // 0.2 sn bekle (frenleme adı bekle)
}

}

}

```

```

////////////////////
//////////////////// KALİBRASYON //////////////////////

```

```

void kalibre(){
    unsigned char sayac=0;
    unsigned short Y_mg;
    olcum();
    delay_ms(500);
    toplam=0;
    for (sayac=0; sayac<10; sayac++){
        olcum();
        toplam=toplam+T1X;
    }
    toplam/=10;
    /*
    EEADR=0xfd;
    EEDATA=lo(toplam);
    EECON1.WREN = 1;

```

```

asm {
MOVLW 0x55
MOVWF EECON2
MOVLW 0xAA
MOVWF EECON2
}
EECON1.WR = 1;
delay_ms(100);
EEADR=0xfc;
EEDATA=hi(toplam);
EECON1.WREN = 1;
asm {
MOVLW 0x55
MOVWF EECON2
MOVLW 0xAA
MOVWF EECON2
}
EECON1.WR = 1;
delay_ms(100);

olcum();
mg=Hi(T1X);
Y_mg=Lo(T1X);
EEADR=0xfb;
EEDATA=Y_mg;
EECON1.WREN = 1;
asm {
MOVLW 0x55
MOVWF EECON2
MOVLW 0xAA
MOVWF EECON2
}

```

```

EECON1.WR = 1;
delay_ms(100);

EEADR=0xfa;
EEDATA=mg;
EECON1.WREN = 1;
asm {
    MOVLW 0x55
    MOVWF EECON2
    MOVLW 0xAA
    MOVWF EECON2
}
EECON1.WR = 1;
*/
mg=toplam;

}

```

```

////////////////////////////////////
//////////////// OLCUM //////////////////

```

```

void olcum(){
// TMR1
T1CON=0b00000000;          // TMR1 ps=1, fosc/4, T1 kapalı

TMR1H=0;
TMR1L=0;
/*asm{
T0_OVFL_WAIT:              ;
    BTFSC GP2, GPIO        ; XOUT == 0 ?
    GOTO T0_OVFL_WAIT      ; Hayır, donguye devam

```

```

;
; Timer taşma yaptı
;
}
*/
while(GPIO.GP4==0);
T1CON.TMR1ON=1;          // TIMER1 Aktif
/*asm{
T3_OVFL_WAIT:            ;
    BTFSS GP2, GPIO      ; XOUT == 1 ?
    GOTO T3_OVFL_WAIT    ; Hayır, donguye devam
;
; Timer taşma yaptı
;
}
*/
while(GPIO.GP4);
T1CON.TMR1ON=0;          // TIMER1 Kapalı
T1X=(TMR1H*256)+TMR1L;   // Mikro saniye olarak değer alındı

}

////////////////////////////////////
//////////////////////////////////// SİNYAL //////////////////////////////////////

void signal(){
unsigned short k_signal;

for(k_signal=0;k_signal<tur;k_signal++){ // tur kadar
tedavi_led=1;          // led aç
puls();                // puls gönder

```

```

delay_us(8900);           // 9900 us bekle 9ms
tedavi_led=0;             // led kapat
delay_us(1000);           // 1000 us bekle 10ms
}                           //

}

```

```

////////////////////////////////////
//////////////// PULS //////////////////

```

```

void puls(){
/*fet=1;
delay_ms(100);
fet=0;
delay_ms(100);
*/
// TMR1
T1CON=0b00000000;        // timer 1 ps=1, fosc/4, T1 kapalı

// INTENSITY
TMR1H=Hi(i_PV);           // TMR1H değeri maksimum ( taşma )
TMR1L=Lo(i_PV);           // TMR1L sayma değeri yüklendi
PIR1.TMR1IF=0;            // TMR1 interrupt temizlendi

fet=1;                    // Fet Aktif
T1CON.TMR1ON=1;           // Timer1 Aktif

asm{
T2_OVFL_WAIT:
    BTFSS PIR1, TMR1IF     ; TMR1 int yaptı mı?

```

```

        GOTO  T2_OVFL_WAIT      ; Hayır, donguye devam
;
; Timer taşma yaptı
;
    }

fet=0;                // FET Kapat
T1CON.TMR1ON=0;       // TMR1 durdu
PIR1.TMR1IF=0;        // TMR1 interrupt temizlendi

// BEKLEME
TMR1H=Hi(b_PV);       // TMR1H değeri maksimum ( taşma )
TMR1L=Lo(b_PV);       // TMR1L sayma değeri yüklendi

T1CON.TMR1ON=1;       // timer1 aktif
asm{
T3_OVFL_WAIT:        ;
    BTFSS PIR1, TMR1IF    ; TMR1 int yaptı mı?
    GOTO  T3_OVFL_WAIT    ; Hayır, donguye devam
;
; Timer taşma yaptı
;
    }
T1CON.TMR1ON=0;       // TMR1 durdu
PIR1.TMR1IF=0;        // TMR1 interrupt temizlendi

}

////////////////////////////////////
//////////////////////////////////// HESAPLAMALAR //////////////////////////////////////

void hesap(){

```

```

unsigned int PV;
//intensity=EEPROM_Read(0x01); // 1 nolu adresten e2 oku intensity al
EEADR=0x01; // okunacak adres
EECON1.RD=1; // okuma başla
intensity=EEDAT; // intensity alındı
if(intensity>100) intensity=100; // intensity basit kontrol
PV=intensity; // Kademe ayarı 100 kademe..
i_PV=0xFFFF-PV; // hesaplar tamam
b_PV=0xFF9B+PV; // 256-(100-PV) == 156 + PV

EEADR=0x03; // okunacak adres
EECON1.RD=1; // okuma başla
tur=EEDAT; // tur değeri alındı
if(tur>40) tur=20;

EEADR=0x05; // bacak seçimi yapıldı
EECON1.RD=1;
bacak=EEDAT;

EEADR=0xFE; // delta ayarı
EECON1.RD=1; //
delta=EEDAT; //
if(delta>100) delta=45; //
delta*=10; // delta onkatı
}

//////////

////////////////////////////////////

//////////////////////////////////// SERİ PORT //////////////////////////////////////

```

```

void seri_haberlesme() {
    WDTCR=0b00010111;          // WDT 268 s kuruldu
    veri_al();                  // veri al
    WDTCR=0b00001000;          // WDT reset value
    tedavi_led=0;               //
    delay_ms(100);              // ufak animasyon
    tedavi_led=1;               //
    kk=0;                       // veri sayacını sıfırla
    switch(i_veri[0]) {
        case 210: pic_pc(); break;    // 210 ise okuma yapılacak
        case 208: pc_pic(); break;    // 208 ise yazma yapılacak
        case 212: delta_ayar(); break; // 212 ise delta ayarı yapılacak
        case 220: cikis(); break;     // 220 ise çıkış yapılacak
        default: break;
    }

}

void merhaba() {
    // Data gönderimi
    unsigned short k;
    ByteToStr(25, veri);        // txt is " 25." (one blank here)
    for(k=0;k<4;k++)            //
        Soft_UART_Write(veri[k]); // merhaba gönderimi tamamlandı
}

////////// PC to PIC
void pc_pic() {
    unsigned short sayac=0;
    pic_reset();                // e2 temizlendi
    son_yer=0;
    e2yaz_sayac=0;              // e2yazma sayacı sıfırlandı yeni yazım
}

```

```

do {
merhaba();           // ilk merhaba
veri_al();           // veri al
e2_yaz();            // e2ye yaz
sayac++;             // sayacı 1 artır
} while(sayac<6);
EEPROM_write(0xFF, (son_yer-1)); // köpüksüz son yer hafızalandı
}

void veri_al() {
kk=0;                // veri indeksi sıfırlandı
do {
    do {
        iveri=Soft_UART_Read(&err); // okuma işlemi
    } while (err);           // okuma yoksa bekle
    i_veri[kk]=iveri;        // veri atandı
    kk++;                    // Sıradaki eleman
    asm CLRWDT;              // WDT temizlendi yeni veri 268 sn sonra gelmezse
} while(iveri != 176);      // b0 gelmedei ise devam et
}

void e2_yaz() {
unsigned short kk_yedek;
kk_yedek=kk;         // kk yedeklendi
do {                  //
    EEPROM_write(kk + e2yaz_sayac, i_veri[kk-1]); // yazım işlemi yapıldı
    kk--;              //
} while(kk>0);        //
kk=0;                 // kk sıfırlandı
e2yaz_sayac+=kk_yedek; // son_yer hafızalandı
tedavi_led=0;         //
delay_ms(500);        // tur bitti animasyonu

```

```

tedavi_led=1;          //
}
void pic_reset() {
    unsigned short tk;          // yerel sayac
    for(tk=0;tk<255;tk++) {      // bütün e2
        EEPROM_write(tk, 0xFF);    // FF yaz
    }
    EEPROM_write(0,0xb0);        // 0.ncı adres b0
}
//////////
////////// PIC to PC

void pic_pc() {
    unsigned short sayac=0;      // sayac sıfırlandı
    unsigned short son_yer_yedek=0; // son_yer yedek
    e2oku_sayac=0;              // e2 okuma sayacı sıfırlandı
    son_yer_yedek = EEPROM_Read(0xff); // son_yer (veri sayısı)
    delay_ms(100);              // 100 ms bekle
    do {
        e2_oku();                // e2 oku
        veri_gonder();           // veriyi gönder gitsin
        sayac++;                 // veri sayacını bir artır
    } while(sayac<6);
    son_yer_yedek=80-son_yer_yedek; // buffer
    do {                          // 80
        Soft_UART_write(0xfe);    // oluncaya dek
        son_yer_yedek--;          // veri
    } while(son_yer_yedek>0);     // gonder

}

void veri_gonder() {

```

```

unsigned short sayac=0;          // FIFO
do {                             // veriyi
Soft_UART_write(i_veri[sayac]); // First In
sayac++;                         // First Out
}while (sayac<kk);              // olacak şekilde gönder
}

void e2_oku() {
kk=0;                           // veri indeksi sıfırlandı
do {                             //
iveri=EEPROM_read(e2oku_sayac+1); // okuma işlemi yapıldı
i_veri[kk]=iveri;               // veri diziye atandı
e2oku_sayac++;                  // e2okuma sayacını artı
kk++;                           // veri indeksini artır
} while(iveri!=176);            // bo gelmedi isie sıradaki veri
tedavi_led=0;                   //
delay_ms(500);                  // tur bitti animasyonu
tedavi_led=1;                   //
}

////////////////////////////////////

void cikis() {
while(GPIO.GP0) {               // Pin 1 ise bekle
tedavi_led=0;
delay_ms(500);
tedavi_led=1;
delay_ms(500);
}
WDTCN = 0x0b;                   // Pin 0 oldu ise WDT aç, WDT prescaler 2,3 sn
while(1){                       // wdt gelecek
tedavi_led=0;

```

```

delay_ms(50);
tedavi_led=1;
delay_ms(50);
}
}

```

```

////////////////////////////////////

```

```

void delta_ayar() {
merhaba();
veri_al();
delta=i_veri[0];
delay_ms(100);
EEADR=0xfe;
EEDATA=delta;
EECON1.WREN = 1;
asm {
MOVLW 0x55
MOVWF EECON2
MOVLW 0xAA
MOVWF EECON2
}
EECON1.WR = 1;
delay_ms(100);
}

```

```

////////////////////////////////////

```

```

/*

```

```

void sol_bacak() {

unsigned short sayac=0;
tedavi_led=1;

```

```

while(sayac<254) {
b_olcum();
EEADR=sayac;
EEDATA=Hi(toplam);
EECON1.WREN = 1;
asm {
MOVLW 0x55
MOVWF EECON2
MOVLW 0xAA
MOVWF EECON2
}
EECON1.WR = 1;
sayac+=1;
delay_ms(100);

```

```

EEADR=sayac;
EEDATA=Lo(toplam);
EECON1.WREN = 1;
asm {
MOVLW 0x55
MOVWF EECON2
MOVLW 0xAA
MOVWF EECON2
}
EECON1.WR = 1;
sayac+=1;

```

```

sayac+=1;
EEADR=sayac;
EEDATA=hi(toplam);

```

```

EECON1.WREN = 1;
asm {
    MOVLW 0x55
    MOVWF EECON2
    MOVLW 0xAA
    MOVWF EECON2
}
EECON1.WR = 1;

}
tedavi_led=0;
while(1);
}
*/
////////////////////////////////////

void b_olcum() {
    unsigned short k=0;
    toplam=0;
    for(k=0;k<3;k++){
        //delay_ms(10);
        olcum();
        toplam+=T1X;
    }
    toplam/=3;
}

```

EK- B

BİLGİSAYAR PROGRAMI

Form1 - 1

Private Sub Calibration_Click()

Form2.Show

End Sub

Private Sub Command1_Click()

Dim sayac As Integer 'intensity byte olarak tanımlandı

Dim intensity As Byte 'intensity string

Dim BytesReceived(50) As String 'Undimensioned (Dynamic) array

Dim yazi As String

Dim InBuffer 'Variant

Dim intCount As Integer

Dim intDecimal As Integer

Dim i As String

Dim k(80) As Byte

Dim t As Integer

Dim sayac2 As Byte

Dim Arr() As Byte

Command1.Enabled = False

Call Port_AC

MSComm1.PortOpen = True

MSComm1.InputLen = 1

MSComm1.InBufferCount = 0

MSComm1.Output = Chr(210) 'yazma başla gönder

MSComm1.Output = Chr(176) 'her verinin sonu

MSComm1.InBufferCount = 0 'clear the input buffer

Do 'merhaba

dummy = DoEvents() 'faslı

Loop Until MSComm1.InBufferCount >= 80 'muhabbeti

For t = 0 To 79

```

Arr = MSComm1.Input
k(t) = Arr(0)
Next t
'Intensity
t = 0
INTEN.Text = k(t)
'TUR
t = 2
DES.Text = k(t) * 10
'bacak
t = 4
If k(t) = 10 Then
LEFTO.Value = True
Else
RIGHTO.Value = True
End If
'DOKTOR ISMI
t = t + 2
Do
yazi = yazi + Chr(k(t))
t = t + 1
Loop Until k(t) = 176
PHNAME(0).Text = yazi
yazi = ""
'HASTA ISMI
t = t + 1
Do
yazi = yazi + Chr(k(t))
t = t + 1
Loop Until k(t) = 176
PTNAME.Text = yazi
yazi = ""
'HASTALIK ISMI
t = t + 1
Do
yazi = yazi + Chr(k(t))
t = t + 1
Form1 - 2
Loop Until k(t) = 176
DIS.Text = yazi
yazi = ""
Command1.Enabled = True
MSComm1.PortOpen = False
End Sub

Private Sub Command2_Click()
Call Port_AC
MSComm1.PortOpen = True

```

```

Command2.Enabled = False
MSComm1.InBufferCount = 0 ' clear the input buffer
MSComm1.Output = Chr(208) ' yazma başla gönder
MSComm1.Output = Chr(176) ' her verinin sonu
Do ' merhaba
dummy = DoEvents() ' faslı
Loop Until MSComm1.InBufferCount >= 4 ' muhabbeti
MSComm1.InBufferCount = 0 ' clear the input buffer
MSComm1.Output = Chr(Val(INTEN.Text)) ' intensity değeri gönder
MSComm1.Output = Chr(176)
Do ' merhaba
dummy = DoEvents() ' faslı
Loop Until MSComm1.InBufferCount >= 4 ' muhabbeti
MSComm1.InBufferCount = 0 ' clear the input buffer
MSComm1.Output = Chr(Val(DES.Text) / 10) ' Tur sayısı gönder
MSComm1.Output = Chr(176)
Do ' merhaba
dummy = DoEvents() ' faslı
Loop Until MSComm1.InBufferCount >= 4 ' muhabbeti
MSComm1.InBufferCount = 0 ' clear the input buffer
If LEFTO.Value = True Then ' sol bacak seçildi ise
MSComm1.Output = Chr(10) ' 10 gönder
End If '
If RIGHTO.Value = True Then ' sağ bacak seçildi ise
MSComm1.Output = Chr(20) ' 20 gönder
End If '
MSComm1.Output = Chr(176)
Do ' merhaba
dummy = DoEvents() ' faslı
Loop Until MSComm1.InBufferCount >= 4 ' muhabbeti
MSComm1.InBufferCount = 0 ' clear the input buffer
MSComm1.Output = PHNAME(0).Text ' doktor ismi gönder
MSComm1.Output = Chr(176)
Do ' merhaba
dummy = DoEvents() ' faslı
Loop Until MSComm1.InBufferCount >= 4 ' muhabbeti
MSComm1.InBufferCount = 0 ' clear the input buffer
MSComm1.Output = PTNAME.Text ' hasta ismi gönder
MSComm1.Output = Chr(176)
Do ' merhaba
dummy = DoEvents() ' faslı
Loop Until MSComm1.InBufferCount >= 4 ' muhabbeti
MSComm1.InBufferCount = 0 ' clear the input buffer
MSComm1.Output = DIS.Text ' hastalık ismi gönder
MSComm1.Output = Chr(176)
MSComm1.PortOpen = False
Command2.Enabled = True
End Sub

```

```

Private Sub Command3_Click()
INTEN.Text = INTEN.Text + 1
Form1 - 3
If INTEN.Text > 100 Then INTEN.Text = 0
End Sub
Private Sub Command4_Click()
INTEN.Text = INTEN.Text - 1
If INTEN.Text < 0 Then INTEN.Text = 100
End Sub
Private Sub Command5_Click()
DES.Text = DES.Text + 50
If DES.Text > 400 Then DES.Text = 200
End Sub
Private Sub Command6_Click()
DES.Text = DES.Text - 50
If DES.Text < 200 Then DES.Text = 400
End Sub

Private Sub Disconnect_Click()
Call Port_AC
MSComm1.PortOpen = True
MSComm1.Output = Chr(220)
MSComm1.Output = Chr(176)
MSComm1.PortOpen = False
End Sub
Private Sub Exit_Click()
End
End Sub
Private Sub Form_Load()
INTEN.Text = 100
End Sub
Private Sub Port_AC()
MSComm1.CommPort = 1
strDescription = "TI1"
If FT_OpenEx(strDescription, FT_OPEN_BY_DESCRIPTION, lngHandle) <> FT_OK
Then
MsgBox "Open Failed", vbOKOnly, "HATA!"
Exit Sub
End If
If FT_GetComPortNumber(lngHandle, ComPortNumber) <> FT_OK Then
MsgBox "Numara Alınamadı", vbOKOnly, "HATA!"
Exit Sub
End If
MSMsgBox ComPortNumber, vbOKOnly
MSComm1.CommPort = ComPortNumber
If FT_Close(lngHandle) <> FT_OK Then
MsgBox "Close Failed", vbOKOnly, "HATA!"

```

```
Exit Sub  
End If  
End Sub
```


ÖZGEÇMİŞ

Adı Soyadı : Mehmet ALINBAY
Doğum Yeri ve Tarihi : İzmir / 16.08.1988
Yabancı Dili : İngilizce
E-Posta : mehmet.alinbay@gmail.com

Öğrenim Durumu

Derece	Bölüm/Program	Üniversite/Lise	Mezuniyet Yılı
Lise	Elektronik Bölümü	Mithatpaşa Anadolu Meslek Lisesi	2006
Üniversite	Mekatronik Öğr.	Marmara Üniversitesi	2010