

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

**HAVA ARACI OTOPILOT DESTEK SİSTEMİ OLARAK BİR PİLOT SAĞLIK
TAKİP SİSTEMİ TASARIMI, PROTOTİPLENMESİ VE ENTEGRASYONU**

YÜKSEK LİSANS TEZİ

Efkan YILMAZ

Mekatronik Mühendisliği Anabilim Dalı

Mekatronik Mühendisliği Programı

TEMMUZ 2020

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

**HAVA ARACI OTOPILOT DESTEK SİSTEMİ OLARAK BİR PİLOT SAĞLIK
TAKİP SİSTEMİ TASARIMI, PROTOTİPLENMESİ VE ENTEGRASYONU**

YÜKSEK LİSANS TEZİ

**Efkan YILMAZ
(518171014)**

Mekatronik Mühendisliği Anabilim Dalı

Mekatronik Mühendisliği Programı

Tez Danışmanı: Dr. Öğr. Üyesi Ramazan YENİÇERİ

TEMMUZ 2020

İTÜ, Fen Bilimleri Enstitüsü'nün 518171014 numaralı Yüksek Lisans Öğrencisi Efkan YILMAZ, ilgili yönetmeliklerin belirlediği gerekli tüm şartları yerine getirdikten sonra hazırladığı “HAVA ARACI OTOPILOT DESTEK SİSTEMİ OLARAK BİR PİLOT SAĞLIK TAKİP SİSTEMİ TASARIMI, PROTOTİPLENMESİ VE ENTEGRASYONU” başlıklı tezini aşağıda imzaları olan jüri önünde başarı ile sunmuştur.

Tez Danışmanı : **Dr. Öğr. Üyesi Ramazan YENİÇERİ**
İstanbul Teknik Üniversitesi

Jüri Üyeleri : **Prof. Dr. Müştak Erhan YALÇIN**
İstanbul Teknik Üniversitesi

Dr. Öğr. Üyesi. Tuba AYHAN
MEF Üniversitesi

Teslim Tarihi : **01 Haziran 2020**
Savunma Tarihi : **01 Temmuz 2020**





Aileme,



ÖNSÖZ

Hayatımın her döneminde bana desteklerini sunan aileme, tez çalışmam için fikirlerini ve desteklerini esirgemeyen aynı şirkette çalıştığımız başta yöneticim Bengi OLCAY' a, ekip arkadaşım Burak BÜYÜKBASKIN' a ve arkadaşım Onur KARAGÜLLE' ye, tez için gerekli araç-gereç, kaynak ve malzeme kullanımı konusunda destek sunan ve aynı zamanda adı altında çalışmaktan gurur duyduğum şirketim TUSAŞ-Türk Havacılık ve Uzay Sanayii A.Ş.' ye, son olarak da çalışmalarım boyunca desteklerini esirgemeyen saygıdeğer Dr. Öğr. Üyesi Ramazan YENİÇERİ hocama teşekkürlerimi sunarım.

Temmuz 2020

Efkan YILMAZ
(Mekatronik Mühendisi)



İÇİNDEKİLER

Sayfa

ÖNSÖZ	ix
İÇİNDEKİLER	xi
KISALTMALAR	xiii
SEMBOLLER	xv
ÇİZELGE LİSTESİ	xvii
ŞEKİL LİSTESİ	xix
ÖZET	xxi
SUMMARY	xxiii
1. GİRİŞ	1
1.1 Tezin Amacı	3
1.2 Literatür Araştırması	4
1.3 Hipotez	11
2. SİSTEMİN TANIMI VE BİLEŞENLERİ	13
2.1 Donanımlar	13
2.1.1 Nabız sensörü	13
2.1.1.1 PPG ile nabız ölçümü	14
2.1.1.2 EKG ile nabız ölçümü	15
2.1.1.3 PPG ve EKG kıyaslaması	16
2.1.1.4 DFRobot nabız hızı sensörü	18
2.1.2 Ataletsel ölçüm sensörü	21
2.1.3 Göz kırpma algılama modülü	24
2.1.3.1 Göz kırpma algılama modülü tasarımı	26
2.1.4 Mikrokontrolcü	33
2.2 Yazılımlar	35
2.2.1 Gömülü yazılım geliştirme	35
2.2.1.1 EKG ve nabız hızı ölçümü	36
2.2.1.2 Baş pozisyonu ölçümü	37
2.2.1.3 Göz kırpma sayısı hesaplanması	40
2.2.1.4 SD karta veri kaydetme	41
2.2.1.5 Gerçek zaman saati (RTC)	42
2.2.2 Kullanıcı arayüzü geliştirme	43
3. ELEKTRONİK DEVRE KARTI GELİŞTİRİLMESİ	45
3.1 Kütüphane Dosyalarının Oluşturulması	45
3.2 Devre Şeması Tasarımı	47
3.2.1 Elektriksel güç haritası	47
3.2.2 Haberleşme protokolleri	49
3.2.2.1 UART haberleşme protokolü	49
3.2.2.2 SPI haberleşme protokolü	51
3.2.2.3 I ² C haberleşme protokolü	52
3.2.3 Devre şematik çizimleri	54

3.3 Devre PCB Tasarımları	56
3.4 Üç Boyutlu Model Dosyalarının Oluşturulması	57
3.5 Baskı Devre Üretimi İçin Gerber Dosyalarının Hazırlanması	60
3.6 Baskı Devre Kartı Fonksiyon Testleri	61
3.7 Baskı Devre Kartı Entegrasyon Süreci	62
4. UÇUŞ SİMULASYONU	65
4.1 FlightGear Flight Simulator	65
4.2 Uçuş Senaryoları	67
4.3 Pilot Sağlık Takip Sistemi – Pilot Entegrasyonu	68
4.4 Simulör Uçuşu Verilerinin İncelenmesi	70
5. SONUÇ VE ÖNERİLER	77
KAYNAKLAR	79
EKLER	85
ÖZGEÇMİŞ	99



KISALTMALAR

EKG	: Elektrokardiyogram
EEG	: Elektroensefalografi
PPG	: Fotopletismografi
IMU	: Inertial Measurement Unit (Ataletsel Ölçüm Birimi)
İHA	: İnsansız Hava Aracı
INS	: Inertial navigation system (Ataletli Seyir Sistemi)
LSB	: Last Significant Bit
mA	: Mili Amper
mm	: Milimetre
V	: Volt
IR	: Infrared (Kızılötesi)
PCB	: Printed Circuit Board (Baskı Devre Kartı)
USB	: Universal Serial Bus
SD Kart	: Secure Digital Hafıza Kartı
ADC	: Analog-Dijital Dönüştürücü
RTC	: Gerçek Zaman Saati
UART	: Universal Asynchronous Receiver Transmitter
SPI	: Serial Peripheral Interface
I²C	: Inter-Integrated Circuit
PC	: Kişisel Bilgisayar
MSB	: Most Significant Bit
3B	: 3 Boyutlu
GNU	: GNU's Not Unix



SEMBOLLER

ω_{Angle}	: Açısal Hız
τ	: Zaman Sabiti
G	: G Kuvveti
α	: Filtre Katsayısı
θ_{Angle}	: Yunuslama Açısı
α_{Acc}	: İvmeölçerden alınan değerlerden elde edilen açı değeri



ÇİZELGE LİSTESİ

Sayfa

Çizelge 1.1 : İki pilot grubu için ortalama nabız hızları.	7
Çizelge 2.1 : PPG ve EKG kıyaslama tablosu.	17
Çizelge 2.2 : EKG sensörleri ve özellikleri.	18
Çizelge 2.3 : DFRobot nabız hızı sensörü özellikleri.	19
Çizelge 2.4 : IMU modülleri ve özellikleri.	23
Çizelge 2.5 : MPU-9250 IMU sensor modülü özellikleri.	24
Çizelge 2.6 : Kızılötesi sensörler ve özellikleri.	25
Çizelge 2.7 : Teensy 3.5 özellikleri.	34
Çizelge 3.1 : Besleme gerilimi ve kaynaktan çekilen akım.	47
Çizelge 3.2 : Teensy 3.5 veri iletim hızı ve hata oranı çizelgesi [65].	50
Çizelge 4.1 : Uçuş rotası noktaları.	68



ŞEKİL LİSTESİ

Sayfa

Şekil 2.1 : PPG Sinyali [54].....	14
Şekil 2.2 : Nabız Oksimetresi [55].	15
Şekil 2.3 : İnsan kalbindeki normal sinüs ritminin şematik gösterimi [56].....	16
Şekil 2.4 : PPG ve EKG dalga formları kıyaslaması [57].	16
Şekil 2.5 : DFRobot nabız sensörü.	18
Şekil 2.6 : DFRobot nabız sensörü bağlantıları.	19
Şekil 2.7 : DFRobot nabız sensörü EKG grafiği.	20
Şekil 2.8 : Nabız hızı ölçümü.	20
Şekil 2.9 : MPU-9250 ataletsel ölçüm sensör kartı.	23
Şekil 2.10 : QRD1114 Sensörü.....	26
Şekil 2.11 : Göz kırpma algılama devresi tasarımı.....	27
Şekil 2.12 : QRD1114 PCB ve şematik çizimleri.....	28
Şekil 2.13 : 10k trimpot PCB ve şematik çizimleri.	28
Şekil 2.14 : Devrenin PCB çizimi.	29
Şekil 2.15 : Devrenin 3B görüntüsü.	29
Şekil 2.16 : Gerber dosyasının özellikleri.....	30
Şekil 2.17 : FlatCAM arayüzü ve çizim parametreleri.	30
Şekil 2.18 : GrblControl program arayüzü.	31
Şekil 2.19 : CNC kazıma makinesi ve devre kartı üretimi.	31
Şekil 2.20 : Üretilmiş devre kartı.....	32
Şekil 2.21 : Göz kırpma modülü entegrasyonu.	32
Şekil 2.22 : Göz kırpma modülü fonksiyon testleri.....	33
Şekil 2.23 : Teensy 3.5 geliştirme kartı.	34
Şekil 2.24 : Elektrot bölgeleri [61].	36
Şekil 2.25 : Tamamlayıcı filtre algoritması [64].	38
Şekil 2.26 : Tamamlayıcı filtre akış diyagramı [64].....	39
Şekil 2.27 : Teensy 3.5 RTC besleme pini.	42
Şekil 2.28 : Kullanıcı arayüzü.	44
Şekil 3.1 : Şematik (solda) ve PCB çizimleri (sağda): (a) Teensy 3.5. (b) DFRobot modülü. (c) MPU-9250 IMU.	46
Şekil 3.2 : Elektriksel güç haritası.	48
Şekil 3.3 : 5V regülatör devre şeması.....	48
Şekil 3.4 : P kanal mosfet ile ters gerilim koruma devresi.	48
Şekil 3.5 : UART protokol yapısı.	49
Şekil 3.6 : SPI master ve slave cihaz bağlantıları.	51
Şekil 3.7 : SPI master ve slave cihaz veri transfer mimarisi [66].	52
Şekil 3.8 : I ² C master ve slave cihaz bağlantıları [67].....	53
Şekil 3.9 : Devre şematik çizimi.....	55
Şekil 3.10 : Devre kartı PCB çizimi.	57
Şekil 3.11 : Devre kartının 3B modellenmiş hali: (a)Ön yüz. (b)Arka yüz.....	58

Şekil 3.12 : Devre kartının patlatılmış gösterimi.	59
Şekil 3.13 : Devre kartının kutulanmış hali.	60
Şekil 3.14 : Gerber çıktısı arayüzü.	61
Şekil 3.15 : Besleme gerilimi ölçümü.	62
Şekil 3.16 : Devre kartı entegrasyonu.	63
Şekil 3.17 : Devre kartı-baret entegrasyonu.	64
Şekil 4.1 : FlightGear örnek ekran görüntüsü [70].	67
Şekil 4.2 : Uçuş rotası.	67
Şekil 4.3 : Elektrot entegrasyonu.	69
Şekil 4.4 : Göz kırpma sensörü entegrasyonu.	69
Şekil 4.5 : Simulatör uçuşu.	70
Şekil 4.6 : 1 nolu uçuş için nabız hızı ve göz kırpma sayısı ortalamaları.	71
Şekil 4.7 : 2 nolu uçuş için nabız hızı ve göz kırpma sayısı ortalamaları.	71
Şekil 4.8 : 3 nolu uçuş için nabız hızı ve göz kırpma sayısı ortalamaları.	72
Şekil 4.9 : 4 nolu uçuş için nabız hızı ve göz kırpma sayısı ortalamaları.	72
Şekil 4.10 : 5 nolu uçuş için nabız hızı ve göz kırpma sayısı ortalamaları.	73
Şekil 4.11 : 6 nolu uçuş için nabız hızı ve göz kırpma sayısı ortalamaları.	73
Şekil 4.12 : 7 nolu uçuş için nabız hızı ve göz kırpma sayısı ortalamaları.	74
Şekil 4.13 : 8 nolu uçuş için nabız hızı ve göz kırpma sayısı ortalamaları.	74
Şekil 4.14 : 9 nolu uçuş için nabız hızı ve göz kırpma sayısı ortalamaları.	75
Şekil 4.15 : 10 nolu uçuş için nabız hızı ve göz kırpma sayısı ortalamaları.	75
Şekil 4.16 : 11 nolu uçuş için nabız hızı ve göz kırpma sayısı ortalamaları.	76
Şekil 4.17 : 12 nolu uçuş için nabız hızı ve göz kırpma sayısı ortalamaları.	76
Şekil A.1 : EKG görüntüleme ve Nabız hızı ölçüm kodu.	86
Şekil A.2 : IMU Complimentary filtre kodu.	89
Şekil A.3 : Göz kırpma algılama kodu.	90
Şekil A.4 : SD karta kayıt kodu.	91
Şekil A.5 : Göz kırpma algılama kodu.	92
Şekil B.1 : Kullanıcı arayüzü seri port tanımlama kodu.	94
Şekil B.2 : Kullanıcı arayüzü veri toplama ve anlamlandırma kodu.	97

HAVA ARACI OTOPILOT DESTEK SİSTEMİ OLARAK BİR PİLOT SAĞLIK TAKİP SİSTEMİ TASARIMI, PROTOTİPLENMESİ VE ENTEGRASYONU

ÖZET

Günümüzde insan faktörü, neredeyse her sektörde önemli bir parametre olarak değerlendirilmekte ve bu yönüyle bir çok araştırmacının dikkatini çekip, çeşitli araştırmalara konu olmaktadır. Özellikle havacılık, insan faktörü konusunun daha da önem kazandığı bir sektördür. Sivil ve askeri havacılık kazaları düşünüldüğünde, bu kazaların %70-80 oranında insan hatası kaynaklı ortaya çıktığı görülmüştür. Ancak bugüne kadar çoğu kaza raporlama sistemi insan faktörünün teorik çerçevesi etrafında tasarlanmamıştır. Bu sebeple pilotların hatalı ya da eksik davranışlar sergilemesindeki temel faktörlerin kaynağı tam olarak belirlenememiştir. Gerekli olan, etrafında yeni araştırma yöntemlerinin tasarlanabileceği ve bu kazalara sebebiyet veren genel bir insan hatası çerçevesi belirlenmesidir. Son yirmi yıl boyunca, uçuş sırasında pilot iş yükünü değerlendirme ihtiyacı, yeni uçakların, sistemlerin ve işletme prosedürlerinin geliştirilmesinde önemli bir faktör haline gelmiştir. Son zamanlarda bu ihtiyaçları karşılamak için pilotların sağlık durumu ve iş yoğunluğu arasında çeşitli bağlantılar kurulması üzerine çalışmalar yapılmaktadır. Bu çalışmalar incelendiğinde, pilotların bilişsel ve fiziksel aktivitelerinin uçuş esnasında önemli parametre haline geldiği gözlenmiştir. Bu aktivitelerin tespit edilmesi, askeri uçaklarda görev etkinliğini, sivil uçaklarda ise güvenliği artırmada önemli bir yapıtaş olacağı düşünülmektedir.

Bu tez kapsamında, hava aracı kontrolünde döngüdeki pilotun yerine otopilotun geçmesi gereken pilot sağlığı ile ilgili şartları yakalayan bir pilot iş yükü ve sağlık takibi sistemi geliştirilmiştir. Sistem, hava aracı kontrolünde pilot kaynaklı problemleri minimum seviyeye indirmeyi amaçlar. Tez çalışmasında, ataletsel ölçüm sensörü, nabız sensörü, kızılötesi sensör, ARM tabanlı mikrodenetleyici, batarya ve besleme katı barındıran bir donanım (elektronik kart) tasarlanmış ve prototiplenmiştir. Bu donanım üzerinde çalışan, pilotun baş oryantasyonunu, nabzını ve göz kırpması sıklığını ölçen, yorumlayan, otopilot ve uçuş görev bilgisayarı için çıkış üreten bir gömülü yazılım geliştirilmiştir. Kask, gözlük ve uçuş simülatörü ile entegrasyon tamamlanmış ve test uçuşları ile veri toplanmıştır. Elde edilen veriler ayıklanmış, otopilot geçişleri için kritik eşikler tespit edilmiştir. Prototipi ile test edilen sistem, otopilot entegrasyonu sürecine hazır hale getirilmiş ve bu süreç gelecek çalışma olarak önerilmiştir.



DESIGN, PROTOTYPING AND INTEGRATION OF A PILOT HEALTH MONITORING SYSTEM AS A SUPPORT SYSTEM FOR AIRCRAFT AUTOPILOT

SUMMARY

Nowadays, the human factor is considered as an important parameter in almost every sector, and in this respect, it attracts the attention of many researchers and is subject to various researches. In particular, aviation is a sector in which the human factor issue becomes more important. Considering civil and military aviation accidents, it has been observed that 70-80% of these accidents occurred due to human error. But so far, most accident reporting system is not designed around the theoretical framework of the human factor. For this reason, the source of the main factors in pilots' displaying erroneous or incomplete behaviors has not been determined exactly. The necessary solution is to identify a general human error framework which causes these accidents.

During the design, development, testing and evaluation of any aircraft system, the capabilities and limits of the crew on board should be considered. These limits must be correctly defined in order for the crew to process too much information in a short time without producing erroneous or incomplete behavior and to put it in logical behavior. The evaluation and definition of these limits is defined as the determination of the pilot's workload.

The progressive development of subsystems in the light of today's technologies has become a factor that alleviates the workload of the pilot. All these developments, redundancy criteria and safety procedures in aircraft make pilots the weakest link in the aircraft. For this reason, determining the workload or health status of the pilot constitutes an important place in terms of flight safety.

Although existing air force aircraft and civil aircraft have advanced technologies, they put cognitive or physical pressure on the pilot. For a more reliable completion of a flight mission, the operator's workload and health status must be monitored continuously and in a way that does not interfere with flight comfort.

When the literature on mental workload is reviewed, there are two results. This is that there is no single definition and universal measure of mental workload. The mental workload is a theoretical structure and can therefore be described as the best operational. Obviously, it concerns factors like operator stress and effort, but these concepts also require operational definitions.

Over the past two decades, the need to assess the pilot workload during flight has become an important factor in the development of new aircraft, systems and operating procedures. Recently, studies have been made to establish various connections between the health status and workload of pilots to meet these needs. When these studies were examined, it was observed that the cognitive and physical activities of the pilots became an important parameter during the flight. It is thought that determining these activities will be an important building block in increasing the task efficiency in

military aircraft and security in civil aircraft. For this purpose, the health and workload of the pilot can be determined using robust and reliable techniques. For example, controlling oxygen consumption or measuring the heart rate depending on the pilot's breathing rate through the pilot mask is a common way to estimate workload. In addition, there are various studies aimed at evaluating the workload according to the blink speed and head position of the pilot.

Operator workload and cognitive status may vary proportionally according to changing flight scenarios and conditions. During normal flight, pilot workload contains a small amount of physical and cognitive load. In addition, in complex situation scenarios such as landing, take-off, dropping ammunition and refueling, this workload and cognitive state increase relative to normal flight.

In the pilot workload and health monitoring system, the physiological parameters required to monitor the workload and health status of the operator were determined as pulse rate, blink rate and head position. For this purpose, various sensors were examined for each physiological parameter and the most suitable one was selected. Besides these choices, microcontroller selection, which is at the center of the system architecture, is also an important factor. After these selection stages, design, production and integration processes were completed, test stages were performed in flight simulation.

Sensors in the system to be designed are very diverse because they use different sensing methods. Appropriate hardware components to work with these sensors have been identified and a system design has been realized. Various parameters such as price suitability, size, communication buses, power consumption, market availability are effectively evaluated during the determination of the sensors, controllers and hardware components used in the most system.

Heart rate sensors are used to measure people's heart activities. Pulse rate is the rate of the heart's beat, measured by the number of contractions per minute of the heart. Pulse rate varies according to the changing physical needs of the body, including oxygen absorption and carbon dioxide secretion. Pulse rate can be measured from various parts of the body and their numbers are equal in each region that can be measured. Pulse rate may vary in situations such as physical exercise, sleep, anxiety, stress, illness or medication. The pressure on the pilot may vary depending on the changing demands during a flight mission. These changing job demands simultaneously create an equivalent stress on the pilot. Pulse rate is one of the most effective tracking tools to detect this stress level.

Inertial measurement sensors (IMU) are electronic devices capable of measuring the specific force, angular velocity, and direction of a system using a combination of an accelerometer, gyroscope, and sometimes magnetometers. These sensors are typically used in a variety of systems, including unmanned aerial vehicles, aircraft, satellites, vehicles, electronic devices, and spacecraft. In the pilot workload and health monitoring system project, it is planned to obtain head position information for the determination of activities that will compromise the task process of the operator, such as fainting and sleep. Although it is known that there will not be unconsciousness due to high G or oxygen deficiency in the flight simulator environment, this is normal in real flights.

In the pilot workload and health monitoring system, it is planned to use blink detection module to detect indirectly increased stress levels with the increase of the workload of the operators. However, these modules must be supplied from abroad since they are

not available in the domestic market. On the other hand, various difficulties can be encountered in terms of both material and time. In order to overcome all these difficulties and to progress the process as planned, this module has been developed to develop.

The software development works carried out within the scope of the project are divided into two. The first is the embedded software development, and the second is the user interface development. In the embedded software development process, there are studies on the selected development board. In this section, the relevant sensor data is processed so that the system can produce the specified outputs. The processed data are then recorded on the micro SD card located internally on the development board for analysis. In addition, the processed data is transferred to the user interface via micro USB in real time. The second stage, the user interface development section, consists of software development studies carried out to display the processed data transmitted via micro USB in the relevant sections of the user interface.

Since the development board used within the scope of the pilot workload and health monitoring system project is Teensy 3.5, the Teensyduino program integrated in the Arduino IDE compilation program is used as software. Since most programs written for Arduino can also work on Teensy, this development board offers a lot of advantages. In addition, Teensyduino is compatible with many Arduino libraries. In this way, there are ready-made libraries for many functions.

In the pilot workload and health monitoring system, it is aimed to display the pilot data received instantly on the operator screen. In this way, the pilot who performs the tasks in the flight simulator environment is instantly monitored through the user interface program. This user interface is designed on the Unity program. Unity is a game engine developed and published in Russia. It was first produced for a different purpose, and then it was used and developed to make games. But Unity 3D did not settle with this and developed itself further. Another reason for using Unity in the pilot workload and health monitoring system is that it is free.

This study consists of taking some physical changes of the operator as a parameter to determine the workload and health status of the operator during the flight, developing a suitable equipment in the light of these parameters and evaluating the workload and health status of the operator according to the changing flight scenarios. Finally, this system has been tested on the operator in a flight simulator environment and the changing health status and workload analysis of the operator has been made according to the flight scenarios determined.



1. GİRİŞ

Herhangi bir uçak sisteminin tasarımı, geliştirilmesi, test edilmesi ve değerlendirilmesi sırasında, uçaktaki mürettebatın yetenekleri ve sınırları göz önünde bulundurulmalıdır. Mürettebatın kısa sürede çok fazla bilgiyi, hatalı ya da eksik davranışlar meydana getirmeden işlemesi ve bunu mantıklı bir davranışa dökmesi için bu sınırların doğru bir şekilde tanımlanmış olması gerekmektedir. Bu sınırların değerlendirilip tanımlanması, pilotun iş yükünün belirlenmesi olarak nitelendirilir. Pilotun sınırları, yani kapasiteleri bir olay esnasında pilotun o olay ile ilgili verdiği tepki süresi olarak tanımlanabilir. Bir uçuşun başlangıcından sonuna doğru yaklaşıldıkça pilotun yorgunluk seviyesi artmakta ve buna bağlı olarak gelişen durumlara verdiği tepki süresi artmaktadır. Yorgun bir pilot, yorgun olmayan bir pilota göre uçuş esnasında gelişen olaylara karşı az duyarlı olabilmekte ve daha yavaş tepki verebilmektedir. Değişen uçuş fazları esnasında pilottan beklenen operasyonel görevler bu yavaş tepki süresinden dolayı birikebilmektedir. Bir süre sonra bu durum, pilotun kapasitesinin üzerine çıkmakta ve pilotun yapması gereken görevleri zamanında yapamayacak duruma gelmesiyle sonuçlanabilmektedir. Pilot iş yükü ise, bir pilotun değişen uçuş fazları, uçuş senaryoları, acil durum faaliyetleri gibi durumlarda yapması gereken operasyonel görevler ve prosedürler olarak tanımlanabilir. Her uçuş fazı ya da senaryosu kendine özgü yapılması gereken görevler ve prosedürlerden oluşmaktadır.

Uçak içerisindeki alt sistemlerin günümüz teknolojileri ışığında giderek gelişmesi, pilotun iş yükünü hafifleten bir etken haline gelmiştir. Tüm bu gelişmeler, uçaklardaki yedeklilik kriterleri ve güvenlik prosedürleri pilotları uçak içerisindeki en zayıf halka haline getirmektedir. Bu sebeple pilotun iş yükünün ya da sağlık durumunun belirlenmesi uçuş güvenliği açısından önemli bir yer teşkil etmektedir.

Mevcut hava kuvvetleri uçakları ve sivil uçaklar her ne kadar gelişmiş sistemler de olsa pilot üzerinde ister ister istemez bilişsel ve aynı zamanda fiziksel bazı baskılar uygulayabilmektedir. Bir uçuş görevinin daha güvenilir bir şekilde tamamlanması için sürekli ve uçuş konforuna müdahale etmeyecek bir şekilde pilotun iş yükünün ve sağlık durumunun takip edilmesi gerekmektedir [1].

Zihinsel iş yüküne ilişkin literatür gözden geçirildiğinde ortaya iki sonuç çıkmaktadır. Bu da, zihinsel iş yükünün tek bir tanımı ve evrensel ölçüsü olmadığıdır. Zihinsel iş yükü teorik bir yapıdır ve bu nedenle en iyi operasyonel olarak tanımlanabilir. Açıkçası, pilot stresi ve çabası gibi faktörlerle ilgilidir, ancak bu kavramlar aynı zamanda operasyonel tanımlar gerektirir.

Reising 1972 yılında yaptığı çalışmada iş yükünü tanımlamak ve ölçmekle ilgili zorluklara ve karmaşıklıklara mükemmel bir genel bakış sağlamaktadır [2]. Daha sonra 1979 yılında ise Moray deneysel psikoloji, kontrol mühendisliği, matematiksel modeller, fizyolojik psikoloji ve uygulamalı psikoloji tarafından bakıldığında zihinsel iş yükünün ölçülmesine ilişkin tanımların, kavramların ve yaklaşımların çeşitliliğini özetlemektedir. Bu özette tek bir tanım sunmak yerine, pilotun zihinsel iş yükünü ölçmede kullanılan çeşitli operasyonel tanımları göz önünde bulundurmak gerektiğini vurgulamaktadır. Örneğin sistem mühendisi, bir görevi yerine getirmek için mevcut zamana dayalı operasyonel tanımları belirtebilir. Psikologlar zihinsel iş yükünün bilgi işleme yönlerini vurgulama eğilimindedir. Bu sebeple kanal kapasitesi ve kalan dikkat ile ilişkilendirerek işlevsel olarak tanımlama eğilimindedir. Fizyologlar ise pilotların stresi ve uyarılma durumlarını vurgulamaktadır [3].

İş yükü ile ilgili araştırma literatürü çok çeşitlidir. Daha önce bunu tanımlamak için sınırlı girişimlerde bulunulmuştur. Bu incelemelerden ikisi özellikle dikkat çekicidir. 1973 yılında Jahns literatürü, girdi yükü, pilot gayreti ve pilot performansı olarak 3 farklı fonksiyonel gruba bölmüştür. Bu kavramsal yapı içerisinde zaman ve hareket çalışmaları, bilgi işleme çalışmaları, pilotun harekete geçme seviyesi çalışmaları ve ekipman geliştirme çalışmalarını gözden geçirmiştir [4,5]. Gartner ve Murphy 1976 yılında pilot iş yükü ve yorgunluğu ile ilgili çeşitli kavram ve değerlendirme tekniklerini incelemişlerdir. İş yükü göstergelerini, görev talep analizleri, görev performans ölçümleri, davranışsal önlemler, psikofizik önlemler ve öznel raporlar açısından gözden geçirmişlerdir. Ayrıca her ölçüm tekniği için önemli sınırlamalar olduğunu belgelemektedirler [6].

Bu aşırı değerlendirmelerin yanı sıra, daha yeni incelemelerde pilot iş yükü değerlendirmesinin çeşitli yönleri derinlemesine incelenmiştir. 1978 yılında Roscoe, metodolojik hususlara ve çeşitli fizyolojik ölçümlere genel bir bakış sağlamaktadır [7,8]. Ellis 1978 yılında öznel ölçümlerle ilgili bazı konuları gözden geçirmiştir [9]. Yine 1978 yılında Chiles, 1979 yılında ise Chiles ve Alluisi laboratuvar, analitik,

sentetik, simülatör ve uçak içi olarak sınıflandırılmış nesnel metodolojileri tartışmışlardır [10,11]. Schiflett ise bu iş yükü tespit metodlarının büyük bir çoğunluğunun pilot sistemlerinin tasarım aşamasında kullanıldığını ve daha sonraki operasyonel testlerde ve değerlendirme ortamlarında kullanımı zorlaştırdığı sonucuna varmıştır [12]. Wienville ve Williges 1978 yılında, özellikle uçuş testi ve değerlendirme ortamına yönelik olabilecek mevcut iş yükü değerlendirme metodolojilerinin kapsamlı bir araştırmasını ve analizini gerçekleştirmişlerdir. Bu kapsamda belirli 28 iş yükü ölçüm kategorisinde sınıflandırılan 400'den fazla referansı incelemişlerdir. Ayrıca uçuş esnasında uçak içi iş yükü değerlendirme tekniklerinin fizibilitesinin belirlenmesinde fiziksel alan gereksinimleri, taşınabilirlik, izinsiz giriş-güvenlik, veri iletimi ve kaydı, deneysel kontrol, uçak sistemine entegrasyon ve mürettebat onayı gibi kritik kriterlerin dikkate alınması gerektiğini belirtmişlerdir [13].

Bu tez süresince, pilot iş yükü ve sağlık durumunu takip edebilen bir sistem geliştirilmesi, geliştirilen sistemin simülasyon ortamında bulunan bir pilota entegrasyonu ve simülasyon ortamındaki değişen uçuş fazları ve senaryolarına göre pilotun iş yükü ve sağlık takibi tespiti konuları üzerine yoğunlaşmıştır.

Tasarım süreci, pilottan alınan nabız hızı, baş pozisyonu ve göz kırpması hızı parametrelerinin sisteme girdi olarak alınıp bunların değerlendirilmesi üzerine kurulmuştur. Bu bilgilerin anlamlandırılıp bir çıktı olarak değerlendirilebilmesini sağlamak amacıyla gerekli donanım ve yazılım tasarım faaliyetleri yürütülmüştür. Donanım tasarım faaliyetlerinin ardından baskı devre yöntemiyle ürettirilen devre kartının ve ilgili sensörlerin bir kask üzerine entegre edilme süreci ile devam edilmiştir. Bu kısımda baş pozisyonu ve göz kırpması hızı sensörlerinin düzgün yerleştirilmesi, fonksiyonlarını doğru bir şekilde icra edebilmesi için önemli bir parametredir. Nabız hızı ise doğrudan pilot üzerinden EKG ile ölçülmektedir. Entegrasyon süreci bittikten sonra daha önceden belirlenmiş uçuş senaryolarına göre pilot ile simülasyon uçuşları yapılarak parametreler değerlendirilmiştir.

1.1 Tezin Amacı

Pilot iş yükü ve sağlık takip sistemi tasarımıdaki asıl amaç, hava araçlarındaki insan faktörü kaynaklı problemlerin minimum seviyeye indirgenmesi ve bu sebeple kaynaklanabilecek olası kazaların engellenmesine yönelik bir alt yapı çalışması oluşturmaktır. 5. Nesil savaş uçakları ile birlikte kullanılması öngörülen bu ve buna

benzer iş yükü tespiti amaçlı sistemlerin yaygınlaşacak olması bu çalışmanın önemini daha da artırmaktadır.

1.2 Literatür Araştırması

1940' tan bu yana, periyodik olarak, dört uçak kazasından üçünün, insan faktörünün insan-makine arayüz sistemindeki yetersiz performansından kaynaklandığını gösteren veriler yayınlanmıştır. Bu oran, pilotların yıllarca daha tutarlı ve daha az hata ile performans göstermesine rağmen devam etmiştir. Bu hataların tespiti amacıyla çok fazla çalışma yapılmıştır. Bu çalışmaların büyük bir kısmını ise nabız hızı ölçümü oluşturmaktadır. Nabız hızının uçuşta diğer değişkenlerden daha sık ölçülmesinin temel sebebi entegrasyonu teknik olarak uğraştırıcı olmaması ve uçuş güvenliğinden ödün vermemesidir. Yapılan çalışmaların birçoğu laboratuvarlarda yapılan simülasyon testlerini içermesine rağmen, nabız hızı gerçek uçuşlar ile de birçok kez kaydedilmiştir. Özellikle, gerçek uçaklarda ve uçuş simülatörlerinde uçuş ile ilişkili olan çeşitli stresli görevlerin etkisini değerlendirmek için nabız hızının izlendiği çalışmalar ilgi çekici olmuştur.

1917' de Gemelli, pilotun duygusal tepkilerini belirlemek için uçuş sırasında kan basıncını, solunum hızını ve nabız hızı için ölçüm yöntemleri geliştirmiştir. Bu sayede değerlendirilen veriler, pilotların maruz kaldığı stresi açıkça göstermiştir [14]. Bu çalışma, pilotların kalp atış hızlarının uçuş sırasında kaydedildiği ilk çalışmalardan biri olmuştur. Bu araştırmada stres terimi ile ne kastedildiğine dair bir açıklama olmasa da, Gemelli' nin makalesi, artan nabız hızı ve uçuş stresi arasındaki ilişkiyi ilk kez tanımlayan makalelerdendir.

İkinci Dünya Savaşı sırasında ve sonrasında az sayıda yapılan deney uçuşları esnasında EKG ve nabız hız verileri kaydedilmiştir. Örneğin, 1940 yılında White, hipoksinin etkilerini belirlemek için 20000 ft' e kadar çeşitli yüksekliklerde uçan gönüllülerin EKG verilerini, modifiye edilmiş bir klinik kardiyografiyi kullanarak kaydetmiştir [15]. 1945 yılında Kirsch, dört motorlu bombardıman uçağı pilotlarının bilekten ölçülen nabız hızlarını, normal seyir esnasında dakikada ortalama 70 atım ölçerken, hedef üzerinde dakikada ortalama 90-95 atım olarak ölçmüştür. Ancak Kirsch, bu artış oranlarını için herhangi bir önerme yapmamıştır [16]. 1949 yılında Narsete tarafından uzun menzilli Kuzey Kutbu uçuşlarında kalp atış hızlarını kaydetmek için benzer bir teknik kullanılmıştır. Bu çalışma sonucunda yorgunluk ve

soğuk algınlığı, kalp atış hızında gözlenen artışın en olası nedenleri olarak kabul edilmiştir [17].

1960' lı yıllarda izleme ve kaydetme tekniklerinde kademeli bir iyileşme, uçuş içi çalışmaların sayısının artmasını sağlamıştır. Uzay programı tarafından teşvik edilen bu çalışmaların çoğu, uçaklarda ve uzay araçlarında kullanım için uygun enstrümantasyon geliştirme ve çeşitli stres türlerinin oluşturduğu fizyolojik yanıtları gözlemlene ile ilgilidir [18,19]. Bu ilk çalışmalar nabız hızını ve diğer fizyolojik değişkenleri öncelikle fiziksel taleplerin ve metabolik gereksinimlerin endeksleri olarak kaydetmiş olsa da, daha sonraki deneyler zihinsel veya psikolojik stresin etkilerini de incelemeye yönelmişlerdir. Uçak izleme ekipmanlarını değerlendirmek amacıyla bir dizi deney yapan Roman ve Lamb, süpersonik F-100 avcı uçaklarıyla uçan pilotlardan birkaç kez EKG verisi kaydetmiştir. Bu yazarlar nabız hızının faktörlere göre değiştiği sonucuna varmışlardır ve bu faktörleri halsizlik ve istem dışı durumlar olarak tanımlamışlardır [20].

Birleşik Devletler Donanması araştırmacıları, Vietnam Savaşı esnasında da, alandaki taşıyıcı pilotların kalp atış hızlarını izlemişlerdir. 1967 yılında yapılan başka bir çalışmada Roman, Older ve Jones, nabız hızlarını farklı görev koşullarında gözlemlemiştir. Bu esnada ortalama nabız hızları uçak gemisinden mancınık ile fırlatma esnasında için dakikada 111.3 atım, bombalama koşulları için dakikada 94.3 atım ve uçak gemisine geri iniş için dakikada 103.1 atım olarak bildirilmiştir [21,22]. İkinci bir çalışma Lewis, genel kalp atış hızlarının dakikada ortalama 94.4 atım olduğunu bulmuştur. Ancak pilotlar şiddetli manevralar yapmak zorunda kaldıklarında, şüphesiz fiziksel stresörlerin etkisinden dolayı oranlar dakikada 118 atıma yükselmiştir.

1977 yılında S'Jongere, Bertels ve Ego, planör pilotlarından EKG kaydetmiştir. Bu verilerin incelenmesinin ardından nabız hızlarının, özellikle zorluklar ile karşılaşıldığında ve uçuşun tehlikeli aşamalarında arttığını belirtmiştir [23].

Sivil havayolu operasyonları, yolcu taşıyan uçuşlar sırasında nabız hızının kaydedildiği bir dizi çalışmaya dikkat çekmiştir. Howitt, dakikada 130-150 atım arasındaki nabız hızlarını, çeşitli uluslararası havaalanlarındaki yaklaşımlar ve inişler sırasında pilotlarından kaydettiğini bildirmiştir [24,25]. Ruffell-Smith, hava trafik kontrolü ve havaalanı navigasyon tesislerinin standartlarını pilotların kalp atış hızı

seviyeleri ile ilişkilendirmiştir [26]. Eğitim kaptanları, planlı uçuş hizmetleri de dahil olmak üzere farklı aktiviteler sırasında, Bateman, Goldsmith, Jackson, Ruffel-Smith & Mattocks tarafından kokpit ortamı değişmeden, kalp atış hızındaki değişikliklerin zihinsel stres göstergesi olup olmadığı gözlenmiştir. Uçuş eğitiminin normal çizgi uçuşundan daha stresli olduğu ve bunun sonucunda simülatör çalışmasından da daha stresli olduğu sonucuna varılmıştır [27].

Bir burun fototransistörü Bruner ve Hohlweck tarafından Almanya'da kısa mesafeli planlanan hizmetlere uçan pilotların kalp atış hızlarını kaydetmek için kullanılmıştır. Birinci pilot için yaklaşım sırasında dakikada ortalama 113.6 atım ve iniş sırasında 127.6 atım, yardımcı pilot için sırasıyla dakikada ortalama 9.6 ve 101.1 atım oranları bildirmişlerdir [28].

1976 yılında Carruthers, Arguelles ve Mosovich, Buenos Aires'ten Londra'ya uçuş sırasında üç pilotun EKG' lerini ve EEG' lerini kaydetmiştir. Yapılan incelemeler sonucunda kalkış ve iniş bölümlerinde yüksek nabız hızları olduğunu bildirmiştir [29].

1977 yılında Sekiguchi, Hunda, Gotoh, Kurihara, Nagasawa ve Kuroda, sivil pilotların nabız hızlarının iniş aşamasında kalkıştan daha fazla olduğunu belirtmiştir [30]. Daha sonraki bir çalışmada, Sekiguchi, Yamaguchi, Kitajima ve Veda, riskli olmayan durumlarda nabız hızının artmayabileceğini öne sürmüştür [31].

Hart ve Hauser, NASA C-141 Kuiper Hava Gözlemevindeki uçuşlar sırasında nabız hızlarını ve sübjektif derecelendirmeleri incelemişlerdir. Pilotun nabız hızının, uçuş esnasında diğer durumlara göre önemli ölçüde değiştiği sonucuna varmışlardır [32].

Roscoe ve Grieve, Boeing 737 ve 767 uçaklarında, yolcu uçuşları sırasında pilotlarının nabız hızlarını, her bir uçuş türü ile ilişkili iş yükü seviyelerini tespit etmek amacıyla izlemiştir. Bu yazarlar, B767' deki iş yükünü azaltmada geliştirilmiş otomatik pilotu ve gelişmiş bir uçuş yönetim sisteminin önemini göstermişlerdir [33]. Yine bir sonraki çalışmalarında, Roscoe ve Grieve, bir B767 uçuş simülatöründe çeşitli uçak içi arızalar ve acil durumlar sebebiyle pilotlar üzerinde oluşan iş yükü seviyelerini, yine öznel derecelendirmeleri artırmak için nabız hızlarını kullanarak değerlendirmektedir [34].

1988 yılında Kakimoto, Nakamura, Tarui, Nagasawa ve Yagura, nabız hızları ve tükürük kortizol seviyelerini, C-I jet uçağı ile uçuş yapan pilotlarda, planlanan güzergahlarda iş yükünü değerlendirmek amacıyla izlemişlerdir. Bu yazarlar, hem nabız hızı değişikliklerinin hem de kortizol düzeylerinin iş yükü seviyelerindeki

farklılıkları yansıttığını gözlemlemişlerdir [35]. Yine 1988 yılında Van de Graaff, pilotların nabız hızı ve kontrol aktivitelerini subjektif derecelendirme tekniğini kullanarak ölçmüştür [36].

Pilotların gerçek uçuş sırasında izlendiği çalışmaların raporları önemlidir. Aynı zamanda ilgi çekici olan bir başka husus ise uçuş simülatörleri kullanılarak yapılan deneylerin açıklamalarıdır. 1973 yılında Opmeer ve Krol, zihinsel yükün nabız hızı üzerindeki etkisini incelemek amacıyla iki sivil pilotu dört motorlu bir uçak simülatöründe ve 14 askeri öğrenci pilotu ise, iki motorlu uçak simülatöründe uçurarak elde edilen verileri incelemişlerdir. İki pilot grubu için ortalama nabız hızları hakkında çizelge 1.1’ deki sonuca ulaşmışlardır. Fakat bu sonuçlar ile ilgili herhangi bir açıklamada bulunmamışlardır [37].

Çizelge 1.1 : İki pilot grubu için ortalama nabız hızları.

Pilot Grubu	Dinlenme	Düz Uçuş	Kalkış	İniş
Sivil Pilotlar	90	94.5	98	112
Askeri Öğrenci Pilotlar	80.5	82	86.5	84.5

1973 yılında Roscoe ve Goodman, düşük görünürlükte yaklaşma ve iniş deneyimi sağlamak amacıyla tasarlanmış bir uçuş simülatörü egzersizi sırasında 75 havayolu pilotunun kalp atış hızını ölçmüşlerdir. Her pilot, 12 tür değişen görünürlük ve sis sekansına göre farklı yaklaşım gerçekleştirmiştir. Bu çalışmada otopilot, önceden belirlenmiş bir karar yüksekliğine kadar simülatörü kontrol etmiştir. Ardından pilot, görsel ipuçlarının yeterliliğine bağlı olarak iniş yapmak için kontrolü alarak inişe başlamıştır. 90 saniye süren her çalışmada nabız hızı değerleri 10-40 saniye ve 65-85 saniye periyotları için hesaplanmıştır. Sonuç olarak 75 pilotun tümünün, yaklaşma esnasında nabız hızında en az % 5 ortalama artış gösterdiği dikkat çekmektedir. Genel ortalama nabız hızları ilk periyot için dakikada ortalama 81.4 atım ve ikinci periyot için 92.9 atım olarak kaydedilmiştir. Bu durum normal uçuştan yaklaştırmaya, yaklaşımdan da inişe geçilme aşamaları arasında gözle görülür bir değişim olduğunun göstergesi olarak belirtilmiştir [38].

1967 yılında Eliasch, Rosen ve Scott, beta-adrenerjik reseptör blokajından dolayı oluşan kalp tepkilerinin duygusal stres ile olan bağlantısını incelemek amacıyla temel bir eğitim simülatörü kullanan pilotların kalp atış hızlarını kaydetmiştir. Ortalama nabız hızı beta blokajından önce ortalama 20 atım / dakika, daha sonra sadece 6 atım / dakika arttığını bildirmişlerdir [39]. Lindholm ve Cheatham, simüle edilmiş bir uçak

gemisi iniş görevini öğrenen altı kişiden kalp atış hızı, HRV ve deri iletkenlik verilerini ölçmüş ve kaydetmişlerdir. Bu üç ölçümden kalp atış hızının iş yükünü ölçmek için en yararlı olduğu sonucuna varmışlardır [40].

Simülatör uçuşu sırasında deneyimli pilotların kalp atış hızlarındaki artışlar, neredeyse her zaman gerçek uçuştan daha azdır. Bu fark, kullanılan simülatörün gerçekçilik seviyesiyle doğrudan ilgilidir. Simülatör ne kadar gerçekçi ve sofistike ise, pilotun nabız hızında önemli bir artış görülmesi bir o kadar olasıdır. Belirli bir simülatör için yeni olan pilotların, bu simülatörü düzenli olarak uçuran ve dolayısıyla sınırlamalarına aşına olan pilotlardan daha yüksek bir oranla yanıt vermeleri daha olasıdır. Kaydedilen gerçek uçuş verilerine göre, simulator uçuşlarındaki verilerin daha düşük olması, simulator ortamında tehlike faktörünün bulunmamasından kaynaklanmaktadır.

Daha önce atıfta bulunulan çalışmada, Roscoe ve Grieve B767 uçak ve uçuş simülatöründeki normal kalkış, yaklaşma ve iniş görevleri sırasında pilotlardan neredeyse benzer seviyelerde kalp atış hızı kaydetmiş ve aynı iş yükü derecelendirmelerini elde etmişlerdir.

Nabız hızı çalışmalarının yanında simülatör veya gerçek uçuşlarda göz kırpması sayısı ile iş yükü arasındaki ilişki ile ilgili de bir çok çalışma bulunmaktadır. 1984 yılında Stern ve Skelly, göz kırpması hızı ve göz kırpması sürelerini iki farklı operasyonel senaryoda incelemiştir. İlk operasyon, görev bölümlerinde iş yükünün değerlendirildiği genişletilmiş bir savaş misyonuydu. İkinci çalışma, görev zorluğunun sistematik olarak manipüle edildiği daha kısa, farklı eğitim misyonlarını içeriyordu. Bu çalışma sonucunda göz kırpması hızı ve süresinin görev talepleri ve görev şekline önemli ölçüde etkilendiğini belirtmişlerdir [41].

1987 yılında Geratewohl, tüm uçuş bilgilerinin % 80' ini işlemesi gerektiği için gözün pilotun en önemli duyuşsal organı olduğunu belirtmiştir. Geratewohl' a göre hem askeri hem de ticari havacılıkta teknoloji, otomasyon ve sofistike algılamasının sürekli gelişmesiyle, insanın bilgi işlemesi değişmiştir [42]. Örneğin, pilotlar havacılığın ilk yıllarında olduğu gibi kokpitin dışında gelişen görsel bilgileri işlemezler, ancak kokpit içerisindeki birden fazla kaynaktan bilgi çıkarmalı ve uçuşu yönetmek için tutarlı bir resme entegre etmelidirler. Görsel verilerin insan tarafından işlenmesi, uçuş güvenliği ve etkinliğinin kilit unsurlarından biri olmaya devam etmektedir [43]. Bu açıdan bakıldığında, uçan performans düşüşlerini veya yorgunluk veya stres gibi performansı

etkileyen faktörleri anlamaya çalışırken görsel bilgi işlemenin analiz edilmesi gerektiği açıktır. Yine 1987 yılında Wilson, Purvis ve Skelly, sekiz deneyimli A-7 pilotlarından nabız hızı, göz kırpması ve EEG verilerini ölçümlemişlerdir. Bu çalışmada her pilot farklı görev senaryolarında farklı uçuş stilleri ile aynı görevi 3 kez gerçekleştirmişlerdir. Bu çalışma sonucunda simülasyon ve gerçek uçuşlar esnasında önemli bir derecede fark olduğunu belirtmiştir. Özellikle görsel iş yükünün yüksek olduğu durumlarda göz kırpması sayısı ve süresinin önemli derecede azaldığına dikkat çekmektedir [44].

1990 yılında Wilson ve Fisher pilot iş yükünün parametrelerini incelemek için, eğitim görevleri sırasında 10 pilottan oluşan F4 uçuş ekibinin fizyolojik verilerini kaydetmiştir. Değişken olarak tanımlanan göz kırpması ve nabız hızı verileri, toplamda sekiz farklı uçuş görevi ile sınıflandırılarak analiz edilmiştir. Bu analizler sonucunda nabız hızı ile birlikte göz kırpması hızının birlikte değerlendirilmesi her parametrenin tek başına değerlendirilmesinden daha iyi sınıflandırmalar ürettiğini belirtmişlerdir [45].

2002 yılında Wilson, bir pilotun bilişsel yeteneklerinin çeşitli yönlerine talep getiren karmaşık bir görev olduğunu ve uçuşun çok yönlü doğası nedeniyle, bu taleplerin pilot üzerindeki etkilerini tanımlamak için birkaç önlem gerektiğini savunmuştur. Bu taleplerin etkilerinin daha iyi anlaşılabilmesi için çeşitli psikofizyolojik verileri değerlendirmiştir. Bu çalışma kapsamında nabız hızı, nabız hızı değişkenliği, göz kırpması hızı, elektrodinamik aktivitesi ve topografik olarak kaydedilen elektriksel beyin aktiviteleri aletli ve görerek uçuş segmentlerinde sınıflandırarak kaydedilmiştir. Görerek uçuş koşullarında, pilotlar navigasyon, yükseklik belirleme ve aynı zamanda kokpit aletleri tarafından belirtilen konumlarını doğrulamak için kokpitin dışından görsel bilgileri kullanabilirler. Aletli uçuş segmenti sırasında ise pilotlar, konumları ve rakımları hakkında bilgi için kokpit enstrümanlarına tamamen bağımlıdır. Bu da daha zor bir bilişsel görevdir. Çalışmanın sonucu olarak aletli uçuş esnasında göz kırpması hızının görerek uçuşa göre azaldığı belirtilmiştir. Bunun sebebi olarak ise, pilotların gerçekleştirdiği görevin genel taleplerinin karmaşıklığına bağlı olarak göz kırpması hızının ters orantılı bir şekilde değiştiğini belirtmişlerdir [46].

2019 yılında Zheng' in yaşları 30 ila 50 arasında değişen dokuz çinli erkek pilot ile yaptığı çalışmada, seçilen pilotlar ya ticari havayolu ya da China Eastern Airlines'ın uçuş eğitmenleri olarak görev yapmaktaydılar. Tüm testler hem gerçek bir uçakta hem de simulator ortamında yapılmıştır. Uçuş simülatörü ve uçağın yanı sıra, örnekleme

hızı 30 Hz olan ve başa takılan bir göz izleyici, deney sırasında deneklerin göz kırpmaya oranını belirlemek için kullanılmıştır.

İş yükünü tespit amaçlı toplamda 3 uçuş senaryosu, hem simulator ortamında hem de gerçek uçuşta tamamlanmak üzere planlanmıştır. Bu senaryolar, aletli kalkış, elle kalkış ve aletli iniş olarak tasarlanmıştır. Bu çalışmanın sonucunda ise simulator ortamındaki sonuçların gerçek uçuş verilerine göre daha inandırıcı olduğunu belirtmiştir [47].

Nabız hızı ve göz kırpmaya algımanın yanında baş pozisyonu algılama ile ilgili de bir çok çalışma bulunmaktadır. Bu çalışmaların çoğu görüntü işleme tabanlı olduğundan dolayı maliyeti epey artırmaktadır. Bu çalışmada imu tabanlı baş pozisyonu algılama yapılacağından dolayı bu bölümde sadece imu tabanlı benzer çalışmalara değinilmiştir.

Atalet izleyicileri, üstün çözünürlükleri ve düşük gecikme süreleri nedeniyle yaygın kabul gördükleri sanal ortam eğitimi, VR oyunlar ve hatta sabit tabanlı araç simülasyonu dahil olmak üzere çok çeşitli uygulamalarda başarıyla uygulanmıştır. 1998 yılında Foxlin, kaska entegre gösterge sistemlerinin pozisyon takibi amacıyla, sensör füzyon tekniklerini kullanarak düşük bütçeli, IMU tabanlı bir takip sistemi geliştirmiştir [48].

2001 yılında Heitmann, Guttkuhn, Aguirre, Trutschel ve Moore-Ede sürüc yorgunluğunu tespit etmek amaçlı baş pozisyonu algılama ile ilgili çalışmalarda bulunmuşlardır. Bu kapsamda bir dizi sürüş simülasyon çalışması yapmışlardır. Sonuç olarak bu çalışma, yapılan algılama sistemlerinin, yorgunluğu tespit amaçlı kullanılabileceğine dair bulgular içermektedir [49].

2014 yılında Li, Meziane, Otis ve Ezzaidi, gerçekleştirdikleri deney ile, kasklara entegre bir imu kullanarak işçilerin yorgunluk ve uykusuzluktan kaynaklanan kafa hareketini tanımlayabildiğini göstermişlerdir [50]. Yine aynı yıl Llorach, Evans, Agenjo ve Blat, sanal gerçeklik ortamlarında kullanıcının hareket etmesini sağlamak amacıyla Oculus Rift sanal gerçeklik gözlüğüne entegre edilmiş IMU tabanlı bir baş pozisyonu takibi sistemi üzerinde çalışmışlardır. Bu çalışmada gerçek zamanlı veri takibi ve çeşitli kalibrasyon yöntemleri ile ilgili açıklayıcı bilgiler yer almaktadır [51]. LaValle, Yershova, Katsev ve Antonov da kullanıcı deneyimini optimize etmek ve

etkin gecikmeyi önemli ölçüde azaltmak amacıyla Oculus Rift üzerinde baş pozisyonu takip eden IMU tabanlı bir sistem üzerinde çalışmışlardır [52].

2016 yılında Windau ve Itti, baş pozisyonunu algılamak amacıyla IMU tabanlı bir sistem kullanmıştır [53]. Fakat bu çalışmada baş pozisyonu verileri herhangi bir sağlık takip sistemi amacıyla kullanılmamıştır.

Bu tezde ise, belirlenen fizyolojik parametreler ile iş yükü ve sağlık takibine yönelik donanımsal bir ürün geliştirme çalışması yapılacaktır. Birbirinden bağımsız olarak bulunan bu algılayıcı devreler, donanımsal olarak tek bir sisteme dönüştürülerek, hem entegrasyonu kolay hem de pilot durumunu daha iyi analiz edebilen kompakt bir ürün haline getirilecektir.

1.3 Hipotez

Pilotun iş yükü ve sağlık durumu takibi amacıyla kullanılan çoğu sistem tam olarak bir ürün haline dönüşmemiş olup, sadece deneysel olarak belirli fizyolojik parametreler arasında ilişki kurmaya yönelik testlerden oluşmaktadır. Bu testlerde de parametrelerin elde edilmesi için farklı donanımsal ürünlerden destek alınmaktadır. Bu parametreleri sağlayacak algılayıcı sistemlerinin iş yoğunluğunu ve sağlık kontrolünü takip etmek amacıyla bütünleşik bir donanım olarak tasarlanması halinde, birbirinden bağımsız parametrelerin tek bir amaç üzerinde ilişkilendirilmesi ve incelenmesi daha kolay bir hale dönüşecektir. Tasarlanacak olan bu sistem, uçuş güvenliğini artırıcı derecede bilgi sağlayacak olup, farklı uçuş fazları ya da görev simülasyonlarında gerçekleştirilerek değişen iş yükü ve sağlık durumu bilgilerini karşılaştırılabilir hale getirecektir.



2. SİSTEMİN TANIMI VE BİLEŞENLERİ

Pilot iş yükü ve sağlık takip sisteminde, pilotun iş yükünü ve sağlık durumunu takip etmek amaçlı fizyolojik parametreler, nabız hızı, göz kırpması hızı ve baş pozisyonu olarak belirlenmiştir. Bu amaçla her bir fizyolojik parametre için çeşitli algılayıcılar incelenmiş ve bunlar arasından en uygun olanı seçilmiştir. Bu seçimlerin yanında sistem mimarisinin merkezinde yer alan mikrokontrolcü seçimi de önemli bir etkidir. Bu seçim aşamalarının ardından tasarım , üretim, entegrasyon süreçleri de tamamlandıktan sonra uçuş simülasyonunda test aşamaları gerçekleştirilmiştir.

2.1 Donanımlar

Tasarlanması düşünülen sistemde bulunan algılayıcılar, farklı algılama metodları kullanmalarından dolayı çok çeşitlidir. Bu algılayıcılar ile birlikte çalışacak olan uygun donanımsal komponentler belirlenerek bir sistem tasarımı gerçekleştirilmiştir. En sistemde kullanılan algılayıcıların, kontrolcü ve donanımsal tüm komponentlerin belirlenmesi aşamasında fiyat uygunluğu, boyut, haberleşme veriyolları, güç tüketimi, piyasada bulunabilirlik gibi çeşitli parametreler etkin şekilde değerlendirilmiştir.

2.1.1 Nabız sensörü

Nabız sensörleri insanların kalp aktivitelerini ölçme amaçlı kullanılmaktadır. Nabız hızı kalbin bir dakikadaki kasılma sayısı ile ölçülen kalbin atışının hızıdır. Nabız hızı, oksijen emme ve karbondioksit salgılama dahil vücudun değişen fiziksel ihtiyaçlarına göre değişim göstermektedir. Nabız hızı vücudun çeşitli bölgelerinden ölçülebilmekle ve ölçülebilen her bölgede sayıları eşit olmaktadır. Nabız hızı fiziksel egzersiz, uyku, kaygı, stres, hastalık veya ilaç kullanma gibi durumlarda değişiklik gösterebilmektedir.

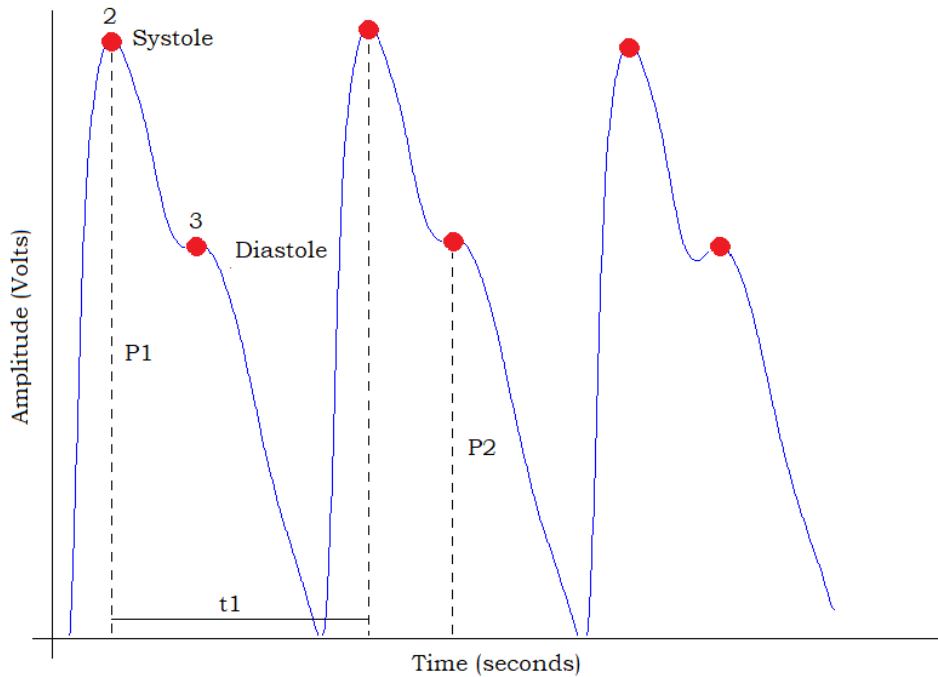
Bir uçuş görevi esnasında değişen taleplere göre pilot üzerindeki baskı değişiklik gösterebilmektedir. Değişen bu görev talepleri aynı anda pilot üzerinde eşdeğer bir stres yaratmaktadır. Nabız hızı bu stres seviyesinin tespiti amaçlı en etkili takip araçlarından bir tanesidir.

2.1.1.1 PPG ile nabız ölçümü

Giderek gelişen teknoloji ile birlikte eski yöntemlere kıyasla farklı ölçüm yöntemleri de türetilmiştir. Nabız hızındaki değişimi algılamak amacıyla kullanılan PPG yöntemi de bu metodlardan birisidir. Gündelik hayatta kullandığımız kol saati şeklinde olan nabız ölçerler bunların en bilindik örnekleridir. PPG, kalp hızını ölçmek için kullanılabilecek yaygın bir tıbbi uygulamadır.

PPG, bir organın hacminin optik ölçümü için uzun bir kelime olan fotopletizmografiyi ifade eder. Prensipte olarak PPG, cilt ve deri altı dokusunu belirli bir dalga boyunda ışıqla aydınlatarak ölçüm yapar. Bu ışıqla bir LED'den veya bir ışıqla yayan diyottan gelmektedir. Gelen bu ışıqla ya emilir, ya içinden geçer ya da geri yansıtılmaktadır. Bir fotodiyot sensörü, LED'e göre nereye yerleştirildiğine bağılı olarak iletilen veya yansıtılan ışıqla ölçer. Fotodiyot daha sonra ölçülen ışıqla bir elektrik sinyaline dönüştürür.

Her kardiyak döngü, kardiyovasküler sistemden bir basınç dalgası gönderir. Bu basınç dalgası, kan damarlarının genişlemesine ve büzülmesine neden olur, bu da PPG'ye bu şekil 2.1' dekiine benzer karakteristik bir dalga şekli verir. PPG dalga formu periyodu her kardiyak döngü ile tekrarlandığından, kişinin nabız hızını hesaplamak için de kullanılmaktadır.



Şekil 2.1 : PPG Sinyali [54].

Nabız oksimetreleri PPG' yi ölçmek için kullanılan en yaygın cihazlardandır. Şekil 2.2' de bunla ilgili bir örnek gösterilmektedir. Bu cihazlarda, 660 nanometre olan kırmızı ve 940 nanometre olan kızılötesi olmak üzere iki spesifik dalga boyu kullanılmaktadır. Her dalga boyunda ışığın emilmesi, doktorların arteriyel kan tarafından emilen ışığı venöz kandan ve diğer doku türlerinden izole etmesini sağlar. Bu cihaz bir kişinin kan dolaşımındaki oksijen konsantrasyonunun bir göstergesi olarak kullanılmaktadır.



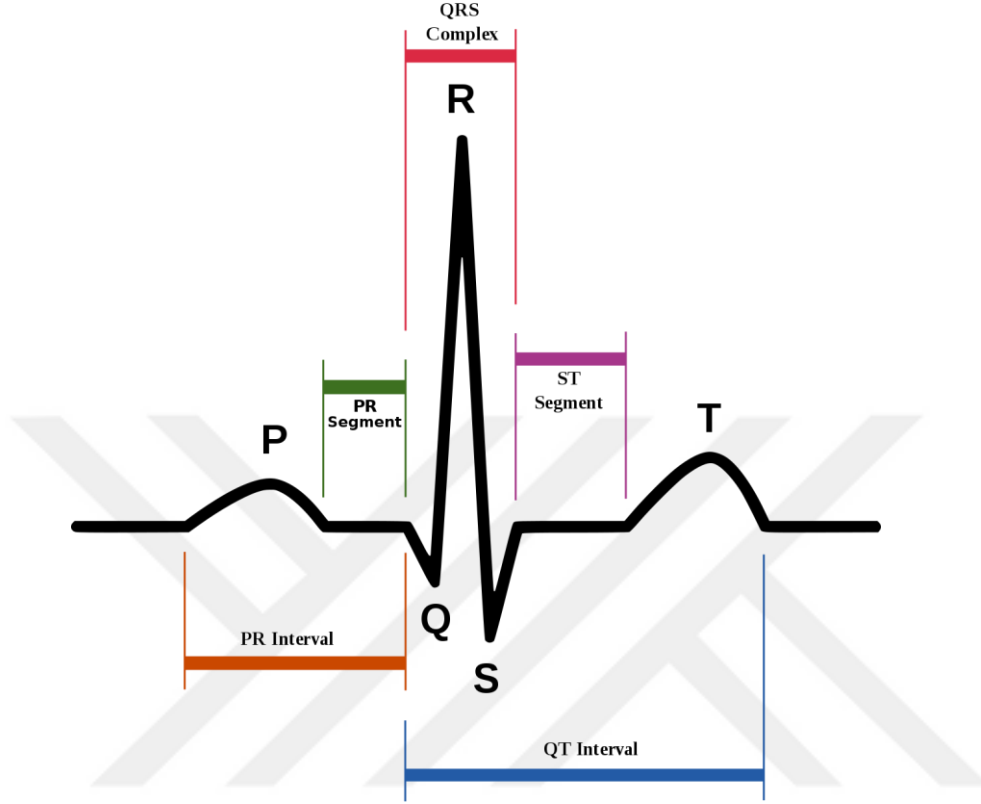
Şekil 2.2 : Nabız Oksimetresi [55].

2.1.1.2 EKG ile nabız ölçümü

EKG kalp kası dokusunun aktivitesini izlemek amacıyla kullanılan bir elektriksel ölçümlerdir. Kendine has özellikleri nedeniyle EKG, doktorlara ve kardiyologlara kalbin işlevi ve genel sağlık hakkında ayrıntılı bilgi sağlayan en tutarlı yöntemdir. EKG dalga formunun her bir kısmı, belirli bir kalp kası hücresi grubunun kasılmasına spesifiktir. Kanın doğru yönde akmasını sağlamak amacıyla bu hücreler doğru sırada kasılmaktadır.

Şekil 2.3' te insan kalbindeki normal sinüs ritmi şematik olarak gösterilmektedir. R pik tipik olarak EKG' deki en büyük sivri uçtur, çünkü sol ventrikülün depolarizasyonu ve büzülmesine karşılık gelir. Bu bölme en kalın kas duvarı ile çevrilidir ve büyük miktarda kanın aorttan ve sistemik devreye kalpten itilmesinden sorumludur. Bu büyük kan çıkışı, arterlerden geçen ve PPG kullanılarak ekstremitelerde ölçülebilen bir basınç dalgası oluşturur. PPG'nin arteriyel kan

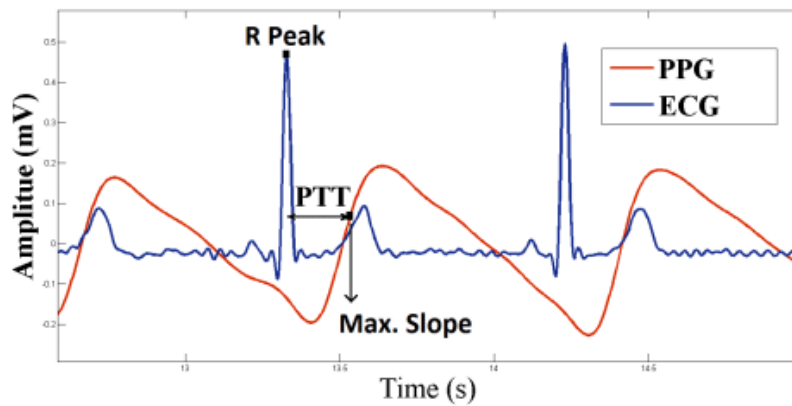
hacminin optik bir ölçümü olduğunu unutmayın. Basınç dalgası nihayetinde ekstremitelere ulaştığında, PPG dalga formu bir zirveye ulaşacak ve bir sonraki basınç dalgası gelene kadar kademeli olarak azalacaktır.



Şekil 2.3 : İnsan kalbindeki normal sinüs ritminin şematik gösterimi [56].

2.1.1.3 PPG ve EKG kıyaslaması

EKG ve PPG aynı anda ölçüldüğünde, iki sinyalin nasıl ilişkili olduğu şekil 2.4’ te gösterilmiştir. Her iki sinyaldeki tepe noktaları arasındaki süre incelenirse, iki yöntemle de nabız hızı tespit edilebildiği sonucuna varılmaktadır. Bununla birlikte, PPG dalga formu EKG dalga formuna göre her zaman gecikmeli görüntülenmektedir.



Şekil 2.4 : PPG ve EKG dalga formları kıyaslaması [57].

Çizelge 2.1’ de EKG ve PPG ölçümleri arasındaki farklar özetlenmiştir. Her şeyden önce EKG, birden fazla elektrot kullanılarak kalbin elektriksel aktivitesinin ölçülmesidir. PPG ise, sadece tek bir fotodiyot kullanılarak arteriyel hacmin optik bir ölçümüdür.

Çizelge 2.1 : PPG ve EKG kıyaslama tablosu.

Özellik Tanımı	EKG	PPG
Ölçüm Tipi	Elektriksel	Optik
Sensör Tipi	Elektrot	Fotodiyot
Nabız Ölçülebilirliği	Evet	Evet
Teşhis Bilgisi	Evet	Evet
Vücut ile Minimum Temas Yüzeyi	2	1
Gerekli ADC Kanal Sayısı	≥ 1	1

Her uygulama kendi teşhis bilgilerini sunmaktadır. EKG, kardiyologlar tarafından iyi bilindiği için kalp kası dokusunun elektriksel aktivitesine odaklanmaktadır. Bu sayede tıp uzmanları her türlü kalp hastalığı ve anormallikleri teşhis etmek için EKG kullanmaktadır.

PPG ise kan akışı ve kan basıncı hakkında daha fazla bilgi sağlamaktadır. Bu ölçüm, farklı bölgelerdeki kan akışını incelemek için vücudun çeşitli yerlerinde yapılabilir. Sol koldan yani kalbin aortuna en yakın yerlerden biri olarak bilinen yerden alınan ölçümden, kardiyak debi ve kalp kapakçığı işlevi hakkında bazı ek bilgiler elde edilebilmektedir.

PPG’ nin sahip olduğu bir avantaj, ölçmek için gerekli olan cilt temas sayısıdır. PPG’ yi yansıyan veya iletilen ışıktan belirlenebildiğinden, ölçmek için sadece tek bir temas noktası gereklidir. Bu avantaj, fitness ile ilgilenen kişiler için giyilebilir elektronik cihazların en cazip avantajı olma özelliğini beraberinde getirmektedir.

Buna karşılık EKG, potansiyelin kalpten ölçülmesini gerektirir. Bu, pozitif ve negatif bir elektroda en az iki temas noktasına ihtiyacımız olduğu anlamına gelir. Tipik olarak, bu bağlantı sadece sınırlı bir süre için yapılır. Gerekli EKG tipine bağlı olarak, ölçümü gerçekleştirmek için birkaç ADC kanalına ihtiyacınız olabilir.

PPG’ nin EKG’ ye göre en büyük dezavantajı yoğun hareket ve titreşimden kolay etkilenmesidir. PPG kullanılan bir ölçümde kişi ne kadar hareketsiz durursa ölçülen değer o kadar doğru okunabilmektedir. Uçuş manevraları, zor görevler ve kokpit ortamındaki iş yükü düşünüldüğünde pilotların titreşime maruz kalmadan hareketsiz

durumları gibi bir durum söz konusu olmadığından, EKG kullanılması daha uygun görülmektedir.

EKG ile ilgili modüller araştırıldığında Analog Devices' ın geliştirdiği AD8232 ve Texas Instruments' ın geliştirdiği ADS1293 yongasına sahip modüller olduğu görülmüştür. Bu yongalar ile ilgili bazı teknik veriler çizelge 2.2' de verilmiştir. AD8232 yongası hem besleme akımı hem de çalışma koşulları düşünüldüğünde Texas Instruments' ın ADS1293 modülüne göre ön plana çıkmaktadır.

Çizelge 2.2 : EKG sensörleri ve özellikleri.

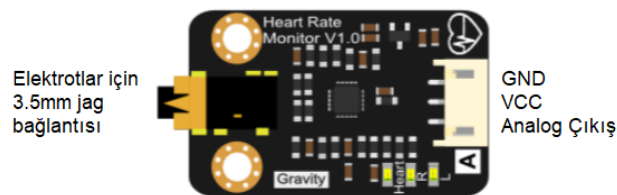
Özellik Tanımı	AD8232	ADS1293
Besleme Gerilimi	2 V - 3.5 V	2.7 V – 5.5 V
Boyut	4 mm x 4 mm	5 mm x 5 mm
Besleme Akımı	170 μ A	350 μ A
Çalışma Koşulları	(-40) $^{\circ}$ C – (+85) $^{\circ}$ C	(-20) $^{\circ}$ C – (85) $^{\circ}$ C
Haberleşme Arayüzü	Analog	SPI

Boyut olarak da daha küçük olmasıyla öne çıkan AD8232 yongası, özellikle çalışma koşulları düşünüldüğünde ADS1293 yongasına göre daha soğuk havalarda çalışabilme özelliğine sahiptir. Dolayısıyla hava aracı ekipmanları için yapılan MIL-STD-810G gibi çevresel sertifikasyon testlerini karşılayabilecek durumda olması bu sistemin ilerleyen aşamalarda hava aracında da kullanılabileceğini göstermektedir.

2.1.1.4 DFRobot nabız hızı sensörü

DFRobot Nabız Hızı Sensörü, kalbin elektriksel aktivitesini ölçmek için kullanılan bir algılayıcı çeşittir. DFRobot Nabız sensörü ile kalbin elektriksel aktivitesi EKG olarak çizelebilmektedir. EKG sinyali son derece gürültülü olabileceğinden dolayı, PR ve QT aralıklarından net bir sinyal üretecek bir AD8232 yongası sensör kartına entegre edilmiştir.

Bu sensör kartının en büyük avantajı kolaylıkla takılabilir ve çıkarılabilir bir yapıda olmasıdır. Şekil 2.5' te DFRobot nabız hızı sensörü ile ilgili bağlantı bilgileri verilmiştir.



Şekil 2.5 : DFRobot nabız sensörü.

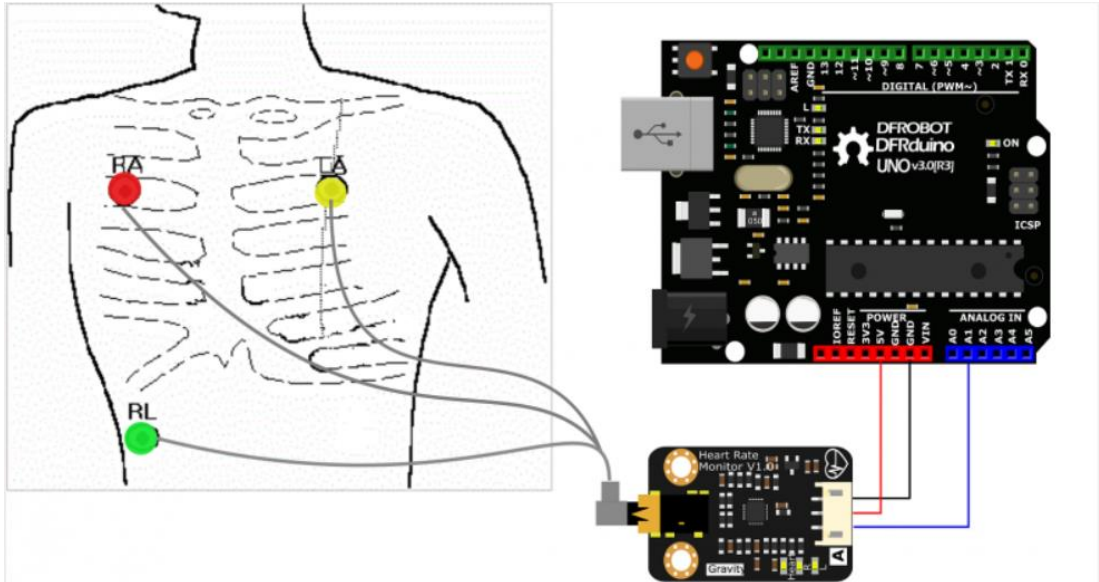
DFRobot nabız sensörü güç tüketimi ve boyut açısından değerlendirildiğinde piyasada kullanılmakta olan diğer muadil ürünlerin önüne geçmektedir. Ayrıca tak çalıştır yapısı sayesinde ölçüm yapmak için ekstra bir işlem gerektirmemektedir. Yapılan ölçüm, analog çıkış sayesinde EKG olarak görüntülenebilmektedir. DFRobot nabız sensörünün diğer özellikleri çizelge 2.3’ te verilmiştir.

Çizelge 2.3 : DFRobot nabız hızı sensörü özellikleri.

Özellik Tanımı	DFRobot Nabız Hızı Sensörü
Besleme Voltajı	3.3 - 6V (5V tavsiye edilen)
Çıkış Voltajı	0 - 3.3V
Çalışma Akımı	< 10mA
Boyutlar	35 x 22 (mm)
Bağlantı Tipi	Analog

DFRobot nabız sensörünün besleme voltaj aralığı 3.3 ile 6 volt arası olduğundan dolayı çoğu geliştirme kartı ile uyumlu bir şekilde çalışabilmektedir. Bu durum mikrokontrolcü seçimi aşamasında epey avantaj sağlamaktadır.

DFRobot nabız sensör kartı ile insan vücudu arasındaki bağlantı şeması şekil 2.6’ da gösterilmiştir. Elektrotların rengine göre vucüt üzerine yapıştırılma bölgeleri belirtilmiştir. Ayrıca örnek olarak kullanılan bir mikrokontrolcü ile olan arayüz bağlantılarına da yer verilmiştir.

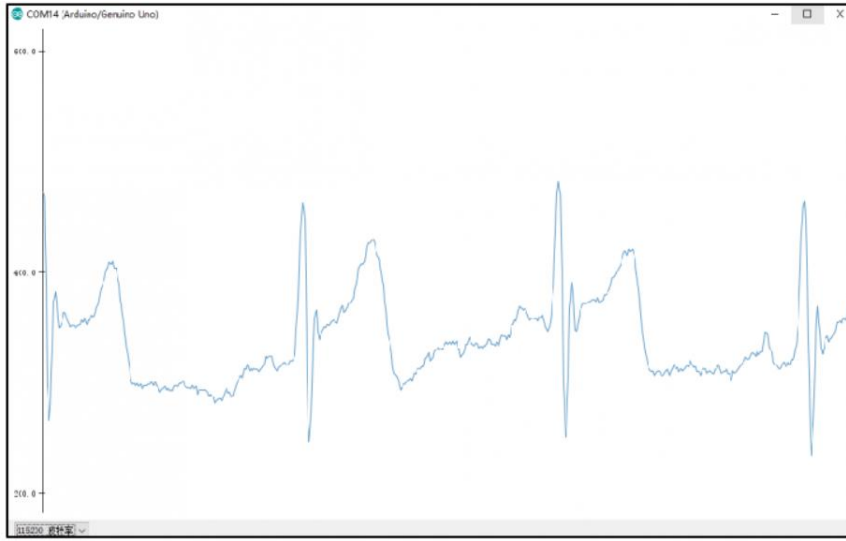


Şekil 2.6 : DFRobot nabız sensörü bağlantıları.

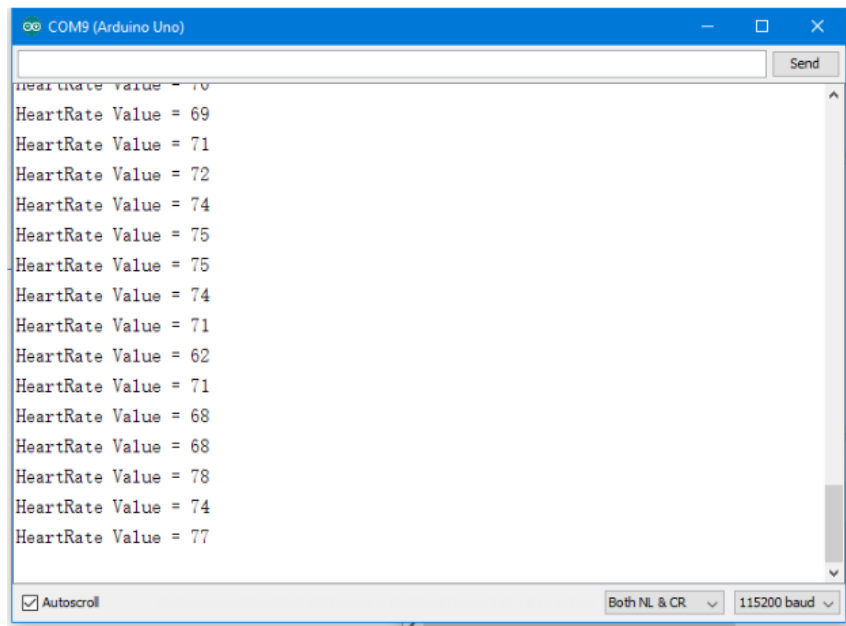
Bir diğer husus, DFRobot sensör kartı üzerinden EKG verisi okuyabilmek için halihazırda kartı geliştirici firma tarafından derlenmiş C kütüphaneleri mevcuttur.

İsteğe bağlı olarak bu kütüphanelerden faydalanılabilmekte ve doğrudan amaca yönelik tasarımlar yapılabilmektedir. EKG verisi kullanılarak grafik çizdirilmek istenirse, bu sensör kartının analog çıkış pini ile kullanılan mikroprosesörün herhangi bir analog pini bağlanılarak ilgili veriler kolaylıkla elde edilebilmektedir. Şekil 2.7’ de DFRobot sensör kartının analog çıkışından alınan ölçümlerin grafiği gösterilmiştir.

Üretici firmanın geliştirdiği kütüphane dosyası kullanılan mikrokontrolcü içerisine entegre edilerek kolaylıkla çalıştırılabilmektedir. Örnek olarak şekil 2.8’ de nabız hızı ile ilgili ölçüm verisi gösterilmektedir. Bu ölçüm verileri ile ilgili ilerleyen bölümlerde değerlendirmeler yapılmıştır.



Şekil 2.7 : DFRobot nabız sensörü EKG grafiği.



Şekil 2.8 : Nabız hızı ölçümü.

2.1.2 Ataletsel ölçüm sensörü

Ataletsel ölçüm sensörleri (IMU), bir ivmeölçer, jiroskop ve bazen manyetometrelerin bir kombinasyonunu kullanarak bir sistemin özgül kuvvetini, açısal hızını ve yönünü ölçebilme kabiliyetine sahip elektronik cihazlardır. Bu sensörler tipik olarak, insansız hava araçları, uçaklar, uydular, araçlar, elektronik cihazlar ve uzay araçlarını da dahil olmak üzere çeşitli sistemlerde kullanılmaktadır.

Bir atalet ölçüm sensörü, bir veya daha fazla ivmeölçer kullanarak doğrusal hızlanmayı ve bir veya daha fazla jiroskop kullanarak dönme hızını tespit ederek çalışmaktadır. Bazıları aynı zamanda yön referansı olarak yaygın olarak kullanılan bir manyetometre de içermektedir. Tipik konfigürasyonlarda, sistemin üç ekseninin her biri için eksen başına bir ivmeölçer, jiroskop ve manyetometre içermektedir.

IMU' lar eski nesil jiroskop ve manyometre gibi büyük ve ağır cihazlara kıyasla epey hafiftir. Bu sayede bir çok sistemde rahatlıkla kullanılabilir. IMU donanımlı navigasyon sistemler (INS), insanlı uçaklar, füzeler, gemiler, denizaltılar ve uydular gibi birçok ticari ve askeri aracın navigasyonu ve kontrolü için omurgayı oluşturmaktadır. IMU' lar ayrıca İHA' lar gibi insansız sistemlerin yönlendirilmesinde ve kontrolünde temel bileşenlerdir.

Navigasyon için IMU' ları kullanmanın en büyük dezavantajı, tipik olarak biriken hatalardan muzdarip olmalarıdır. Referans sistemi hız ve pozisyonu hesaplamak için zamana göre ivmeyi sürekli olarak entegre ettiği için, küçük de olsa her türlü ölçüm hatası zamanla birikerek kaymalara sebebiyet vermektedir. Bu tarz hataların önüne geçmek için literatürde bir çok çalışma mevcuttur. Bu çalışmalar içerisinde en çok kullanılan yöntemlerden birisi GPS düzeltmeli navigasyon sistemleridir. Bu sistemler sayesinde kümülatif olarak biriken hatalardan kaynaklı hesaplamalar, GPS sinyalinin gelen konum referansı ile bütünleştirilerek IMU' ların kayması engellenebilmektedir. Bu bütünleştirme esnasında kullanılan en bilindik metod kalman filtreleridir.

Pilot iş yükü ve sağlık takip sistemi projesinde pilotun baygınlık ve uyku gibi görev sürecini tehlikeye sürükleyecek aktivitelerin tespiti amaçlı baş pozisyonu bilgisi alınması planlanmaktadır. Uçuş simulatörü ortamında her ne kadar yüksek G' den veya oksijen yetmezliğinden kaynaklı baygınlıklar olmayacağı bilirse de gerçek uçuşlarda bu olağan bir durumdur.

Pilotların baş pozisyon bilgileri sadece sağlık takibi amaçlı kullanılmamaktadır. Literatürde kaska entegre kamera sistemleri yardımıyla IMU olmadan silah sistemlerinin yönlendirilebildiği çalışmalar mevcuttur. Fakat bu yöntemler IMU tabanlı sistemlere göre epey maliyet getireceğinden dolayı bu çalışma kapsamında değerlendirilmemiştir.

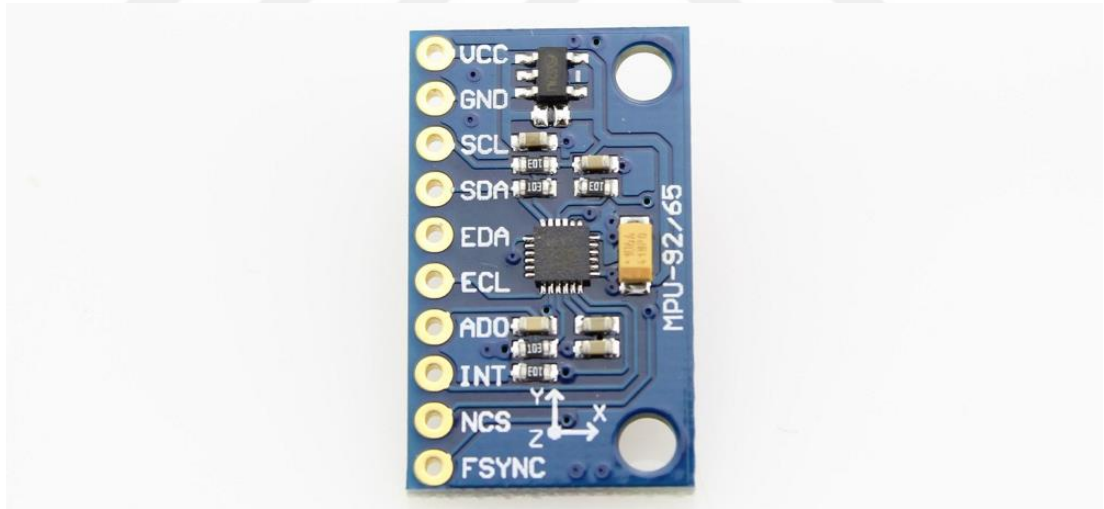
Amaçlanan projede ağır ve büyük mekanik jiroskop, ivme ölçer ya da magnometreler ile pilotun baş pozisyonu ölçülebilmesi mümkün değildir. Bu ve buna benzer hem entegrasyonu zor hem de maliyeti yüksek çözümlerin olumsuzlukları sebebiyle pilot iş yükü ve sağlık takip sistemi kapsamında IMU tabanlı bir çözüm geliştirilmesi daha uygun görülmektedir. Bu kapsamda düşük kalite, orta kalite ve yüksek kalitede IMU modülleri ve bunların bazı özellikleri çizelge 2.4' te verilmiştir. Proje çalışması kapsamında IMU sensörü, baş pozisyonunun belirli bir eşik değer yani açı değerinden fazla olduğu durumları tespit etmek amaçlı kullanıldığı için çok yüksek hassasiyetli IMU seçilmesine gerek görülmemiştir. Bu işlem en basit IMU modülünde bile çeşitli füzyon algoritmaları test edilerek bu eşik değerleri düzgün bir şekilde tespit edilebilmektedir. Bu değerlendirme, ilerleyen zamanlarda uçak üzerinde yapılacak sağlık takibi amacıyla baş pozisyonu ölçülmesinde uçuş sertifikasyon şartlarını karşılayabilen bir IMU seçilirken hassasiyetten bağımsız olarak daha çok boyut, güç tüketimi ve fiyat konularının daha ön planda olması gerektiğini göstermektedir. Dolayısıyla çizelge 2.4' te IMU modüllerinin boyut, güç tüketimi ve fiyatları incelendiği takdirde MPU-6500 isimli IMU diğer modüllere göre daha uygun görünmektedir.

Bu kapsamda hem piyasada bulunması kolay, ucuz ve boyut açısından fazla yer kaplamayan, üzerinde MPU-6500 jiroskop ve ivme sensörünü barındıran MPU-9250 isminde bir IMU kartı seçilmiştir. Şekil 2.9' da MPU-9250 ataletsel ölçüm sensör kartı gösterilmiştir.

MPU-9250 ataletsel ölçüm sensörü Invensense firmasının geliştirmiş olduğu akıllı telefonlar, tabletler, giyilebilir sensörler ve diğer tüketici pazarları için ürettiği ikinci nesil 9 eksenli ataletsel ölçüm sensörüdür.

Çizelge 2.4 : IMU modülleri ve özellikleri.

Özellik Tanımı	ADIS16489	ADIS16505	MPU-6500
Besleme Voltajı	3 – 3.6 V	3 – 3.6 V	1.8 – 3.3 V
Çalışma Akımı	< 186 mA	< 55 mA	< 3.4 mA
Boyutlar	44 x 47 mm	33 x 30 mm	3 x 3 mm
Arayüz Tipi	SPI	SPI	SPI / I ² C
Jiroskop Aralığı	±480 °/sec	±125 / ±500 / ±2000 (°/s)	±250 / ±500 / ±1000 / ±2000 (°/s)
Jiroskop Hassasiyet Faktörü	1310 (LSB / (°/s))	10 / 485 / 760 (LSB / (°/s))	131 / 65.5 / 32.8 / 16.4 (LSB / (°/s))
İvme Ölçer Aralığı	±18 g	±7.9 g	±2 / ±4 / ±8 / ±16 (g)
İvme ölçer Hassasiyet Faktörü	82500 LSB/g	26 / 756 / 268 (LSB/g)	16384 / 8192 / 4096 / 2048 (LSB/g)
Çalışma Aralığı	(-40)°C – (85)°C	(-40°C) – (105)°C	(-40°C) – (105)°C
Fiyat	1734 \$	270 \$	7.27 \$



Şekil 2.9 : MPU-9250 ataletsel ölçüm sensör kartı.

MPU-9250, iki farklı çipi birleştiren bir paket sistemdir. Üzerinde bulunan MPU-6500 isimli 3 eksenli jiroskop, 3 eksenli ivmeölçer ve dahili dijital hareket işlemcisi ve AK8963 isminde 3 eksen dijital pusulası sayesinde hareket füzyon algoritmaları için uygun bir altyapı sunmaktadır.

MPU-9250 sensör modülünün bazı temel özellikleri çizelge 2.5’ te verilmiştir.

Çizelge 2.5 : MPU-9250 IMU sensor modülü özellikleri.

Özellik Tanımı	MPU-9250 IMU Sensör Modülü
Besleme Voltajı	3.3 - 5V (5V tavsiye edilen)
Çalışma Akımı	< 3.5 (mA)
Boyutlar	26 x 15 (mm)
Arayüz Tipi	SPI / I ² C
Jiroskop Aralığı	±250 / ±500 / ±1000 / ±2000 (°/s)
Jiroskop Hassasiyet Faktörü	131 / 65.5 / 32.8 / 16.4 (LSB / (°/s))
İvme Ölçer Aralığı	±2 / ±4 / ±8 / ±16 (g)
İvme ölçer Hassasiyet Faktörü	16384 / 8192 / 4096 / 2048 (LSB/g)
Dijital Pusula Aralığı	±4800 (μT)
Dijital Pusula Hassasiyet Faktörü	0.6 (μT / LSB)

MPU-9250 IMU sensör modülü simülatör ortamında pilot kaskının ya da kulaklığının üst kısmına entegre edilmesi planlanmaktadır. Böylelikle pilotun baş hareketleri ile ilgili tüm ölçüm verileri toplanabilecektir.

2.1.3 Göz kırpma algılama modülü

Pilot iş yükü ve sağlık takibi sisteminde pilotların iş yükünün artması ile dolaylı olarak artan stres seviyelerinin tespiti amaçlı göz kırpma algılama modülü kullanılması planlanmıştır. Fakat bu modüllerin ülke içerisindeki piyasada bulunmamasından dolayı yurtdışından temin edilmesi gerekmektedir. Yurtdışından temin edilmesinde ise hem maddi hem de süre açısından çeşitli zorluklarla karşılaşılabilir. Tüm bu zorlukların aşılabilmesi ve sürecin planlanan şekilde ilerleyebilmesi açısından bu modülü geliştirilme çalışması yapılmıştır.

Bu çalışmada kızılötesi sensör kullanılarak dakikadaki göz kırpma sayısının algılanması hedeflenmiştir. Bu algılama, yayılan kızılötesi ışığın kornea ve göz kapağından yansması ve yansıyan bu ışığın şiddetinin ölçülmesiyle sağlanmıştır. Daha sonra geliştirilen sistemin test aşamalarında bir dakikada elde edilen göz kırpma verisi mikrodenetleyici yardımı ile sayılmıştır.

Algılayıcı sisteminde kullanılacak olan elemanların seçimi aşamasında, bir gözlük yardımıyla gözün önüne yerleştirilecek olan sensörün gözün görüş açısına müdahale etmemesi amacıyla devre elemanlarının küçük ölçülerde olmasına ve yine bu elemanlar tarafından kornea üzerine düşürülecek olan kızılötesi ışığın özelliklerinin gözün görme kabiliyetine negatif etki etmeyecek değerliklerde olmasına dikkat edilmiştir. Çizelge 2.6' da bu modüller ve özellikleri gösterilmiştir.

Çizelge 2.6 : Kızılötesi sensörler ve özellikleri.

Özellik Tanımı	TCRT5000	CNY70	QRD1114
Besleme Voltajı	3.3 – 5 V	3.3 – 5 V	3.3 – 5 V
Dalga Boyu	950 nm	950 nm	940 nm
Boyut	10 x 6 mm	7 x 7 mm	3 x 6 mm
Güç Tüketimi	200 mW	200 mW	100 mW
Çalışma Sıcaklık Aralığı	(-25) – (+85) °C	(-40) – (+85) °C	(-40) – (+85) °C
Etkin Ölçüm Mesafesi	15mm	20 mm	20 mm
Fiyat	1.26 \$	1.39 \$	0.5 \$

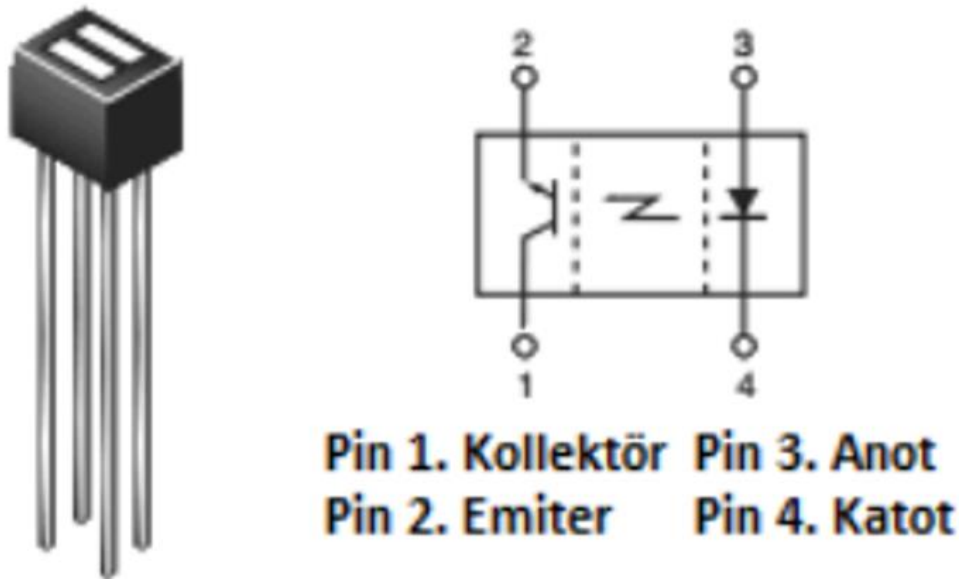
Geliştirilmesi planlanan göz kırpma algılama modülünün sensör seçimi yapılırken mümkün olduğu kadar küçük olmasına dikkat edilmiştir. Bu sensör gözlük üzerine entegre edileceği için pilotun görüş alanını mümkün olduğu kadar kapatmaması gerekmektedir. Çizelge 2.6’ da görüldüğü üzere hem boyut olarak hem de güç tüketimi yönünden en efektif sensörün QRD1114 olduğu görülmektedir. Bu nedenle QRD1114 sensörü projede kullanılmak üzere seçilmiştir [58]. Bu modülün IR LED ve IR Fototransistör bölümünden yayılan ışık emisyonunun maksimum dalga boyu değeri, 20mA olarak belirlenen test sonuçlarında, 940nm olarak belirtilmiştir [59]. Ayrıca bu değer bandındaki bir ışığın IR-A sınıfında olduğu ve herhangi bir retinal bozukluğa yol açmayacağı belirtilmiştir [60].

Tasalanacak olan modülün pilota entegrasyonunun nasıl yapıldığı, bu modülün planlandığı gibi çalışmasında önemli bir yer teşkil etmektedir. Her pilotun kafa yapısı ve göz bölgelerine göre sistemin kolay entegre edilebilir olması gerekmektedir.

Infrared LED yani diğer bir ismiyle kızılötesi vericiler, kızılötesi ışık yayabilen LED’ lerdir. Kızılötesi ışık tayfı, insan gözüyle algılanabilen dalga boyu aralığının dışında olması sebebiyle, ancak kızılötesi dalga boylarını algılayabilme kapasitesi olan fotodiyot ya da fototransistör gibi elektronik komponentler yardımı ile görüntülenebilmektedir.

Fototransistörler, elektrik akımını ışık ile kontrol edebilen elektronik komponentlerdir. Fototransistörler üzerlerinden geçen akım miktarı üzerine düşen ışık miktarı ile doğru orantılıdır. Pilot iş yükü ve sağlık takibi sisteminde, gözün kapalı ya da açık olduğu durumlarda ölçülen elektrik akımı, farklı seviyelerde olacaktır. Bu farklılıkların tespit edilmesi ile bir dakikadaki göz kırpma sayısı elde edilecektir. Elde edilen bu veri sayesinde pilotun iş yükü ve stres seviyesinin tespit edilmesi sağlanacaktır.

Şekil 2.10’ da gösterildiği gibi QRD1114 sensörü üzerinde IR LED ve IR fototransistör bütünleşik olarak bulunmaktadır.



Şekil 2.10 : QRD1114 Sensörü.

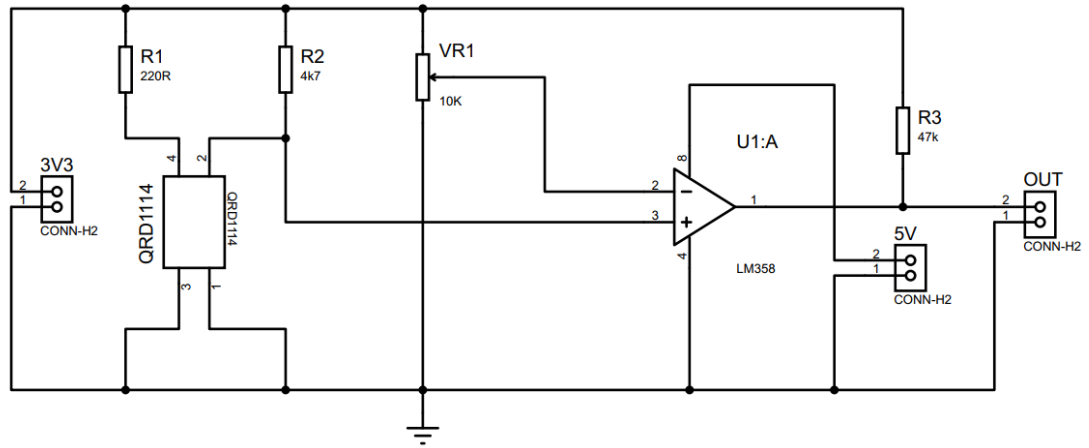
2.1.3.1 Göz kırpma algılama modülü tasarımı

Hareketin algılanma sürecince, kızılötesi LED’den gelen ışık gözü aydınlatır ve yansıyan kızılötesi ışık kızılötesi algılayıcıda bir elektrik akımı indükler. Göz kapağı kapandığında, daha fazla kızılötesi ışık yansır. Bunun nedeni göz kapağının üzerindeki epidermis tabakasının kızılötesi ışığı korneadan daha fazla yansıtmasıdır. Kornea ve göz kapağından yansıyan ışığın şiddetinin farklı seviyelerde olması, bu algılayıcı sisteminin çalışmasının temel prensibidir.

Piyasada hazır ürün şeklinde bulunan göz kırpma algılama devresi sayısı çok az, neredeyse hepsi yurtdışında ve ulaşılması zordur. Bu yüzden devrenin tasarlanması ve üretilmesinde yerli şartlar göz önünde bulundurulmuştur. Bu algılama devre kartının tasarımında, opamp ile karşılaştırma yönteminden yararlanılmaktadır.

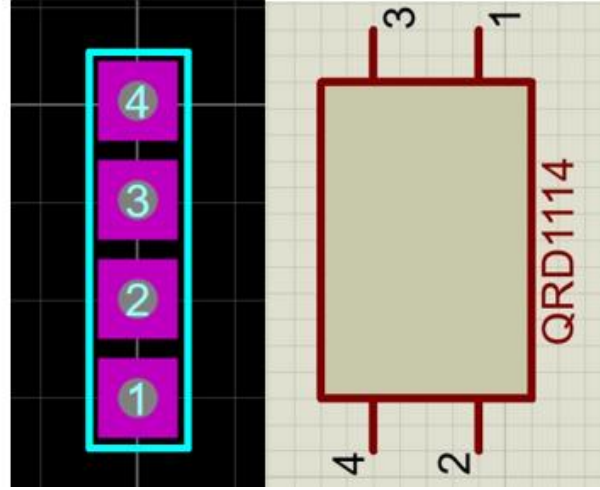
Algılama devresi temelde bir karşılaştırıcı devre sistemidir. Sensörden alınan gerilim değeri ile trimpot üzerindeki referans gerilim değeri karşılaştırılmaktadır. Sensördeki gerilim değeri trimpotun gerilim değerinden yüksek ise op-amp çıkışında lojik '1', düşük ise op-amp çıkışında lojik '0' değeri görülmektedir. Sistem 3.3V ile beslenirken, op-amp elemanı 5V ile beslenmektedir. Her ikisi de aynı voltaj değeri ile beslendiği zaman sistemin düzgün bir sonuç vermediği gözlemlenmiştir. Bu nedenle iki farklı besleme gerilimi kullanılmıştır. Algılama mesafesini ayarlamak için ise trimpot tercih edilmiştir. Yapılan ölçümler sonucu gözük ile göz mesafesi arasındaki en uygun algılama için trimpotun direnci 0.7k, gerilimi ise 3.13V olarak seçilmiştir.

Oluşturulan devre test edildikten sonra devre kartı tasarım programı ile şekil 2.11' de gösterilen devre şematik çizimi yapılmıştır.



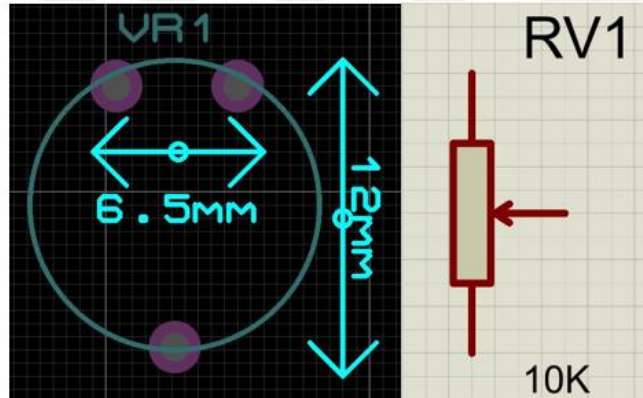
Şekil 2.11 : Göz kırpma algılama devresi tasarımı.

Devrenin oluşturulması sırasında QRD1114 ve üç bacaklı 10k trimpot için sistemde baskı devre çizimi bulunmamaktaydı. Bu nedenle program içinde boyut özellikleri dikkate alınarak çizimleri yapılmıştır. QRD1114 için 4 adet kablo çıkışı planlandığı için sensörün baskı devre çizimi için 4lü header kullanılmıştır. Sensör çizimleri şekil 2.12' de gösterilmektedir.



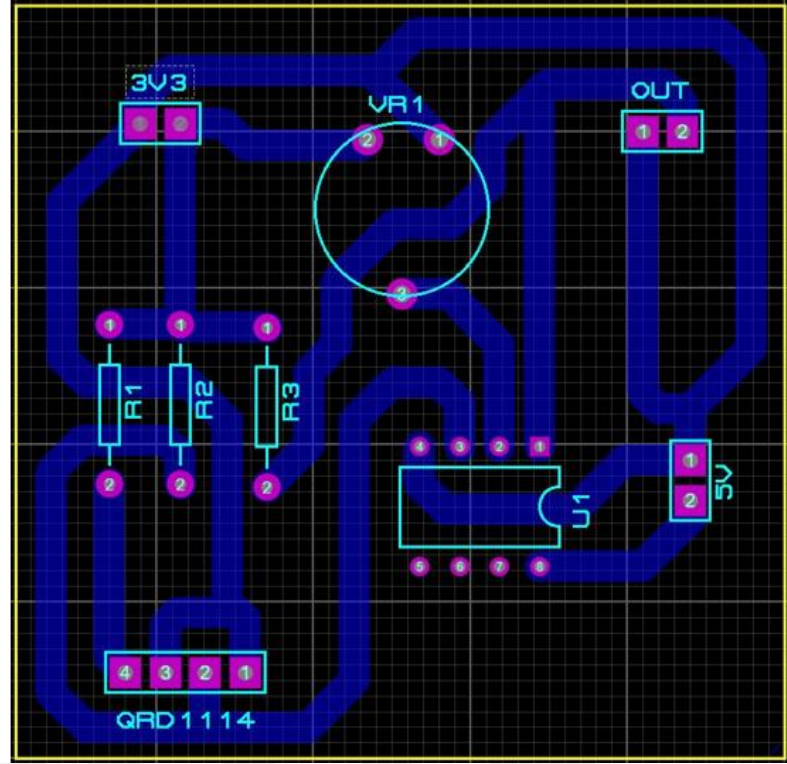
Şekil 2.12 : QRD1114 PCB ve şematik çizimleri.

Trimpot elemanının devre üzerine lehimlenmesi planlandığından footprint çizimi için bacakları arasındaki mesafe kumpas yardımıyla ölçüldükten sonra model çizimi yapılmıştır. Trimpot çizimi şekil 2.13’ te gösterilmiştir.



Şekil 2.13 : 10k trimpot PCB ve şematik çizimleri.

Test devresi tamamlandıktan sonra PCB çizimi için elemanların yerleri belirlenmiştir. PCB çizimi yapılırken giriş ve çıkışların devrenin kenarlarında olmasına, devre daha sonrasında CNC kazıma makinası ile çizileceğinden dolayı yolların kalın olmasına, ayrıca PCB çizimi tek taraflı olacağından herhangi bir kısa devre olmamasına dikkat edilmiştir. Devre boyutu 50x50mm olarak tasarlanmıştır. Devrenin PCB çizimi ve üç boyutlu gösterimleri sırasıyla şekil 2.14 ve 2.15 te gösterilmiştir.



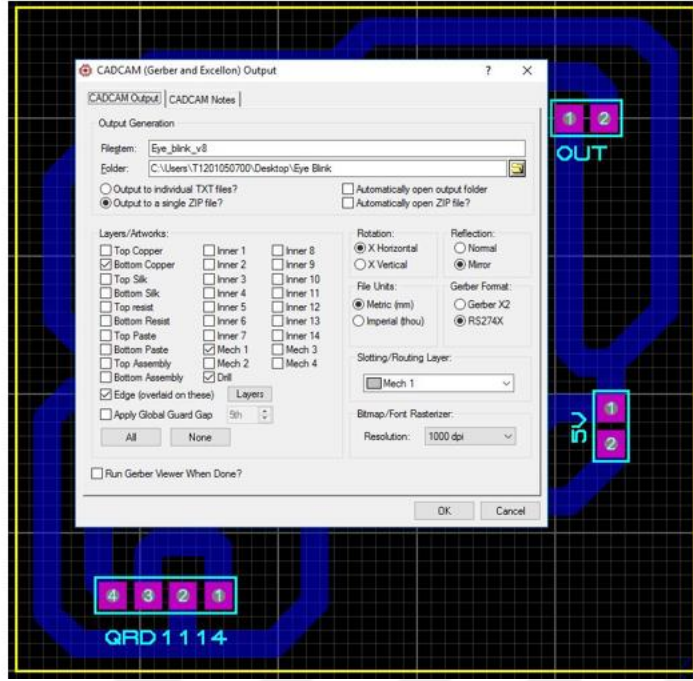
Şekil 2.14 : Devrenin PCB çizimi.



Şekil 2.15 : Devrenin 3B görüntüsü.

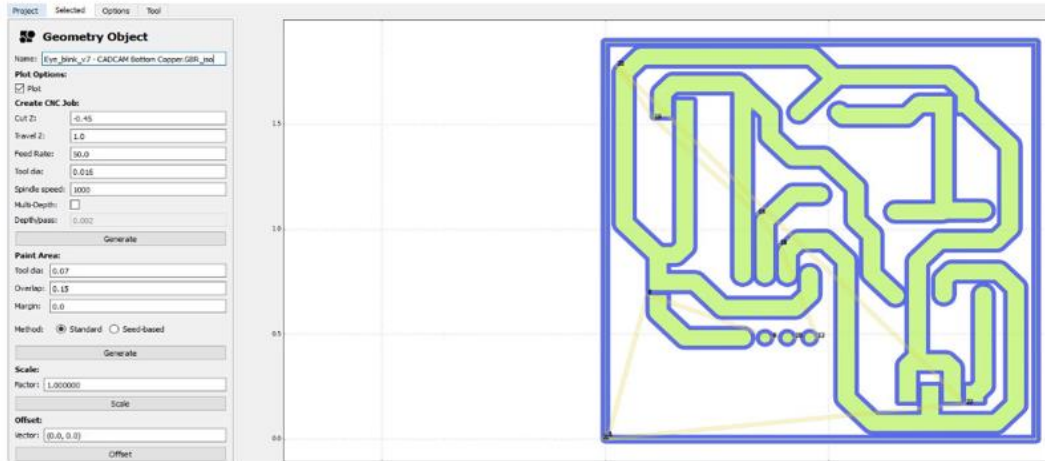
Daha sonrasında devreyi üretmek için kullanılacak olan gerber dosyası oluşturulmuştur. Test devresi için gerber dosyası oluşturulurken tek katlı bir devre üreteceğimizden dolayı sadece alt bakır yüzeyi dosyası yeterli olmuştur. Devrenin ayna görüntüsü alınmış ve formatı RS274X olarak seçilmiştir. Çözünürlük değeri ise

1000 dpi olarak belirlenmiştir. Şekil 2.16’ da oluşturulan gerber dosyasının özellikleri gösterilmiştir.



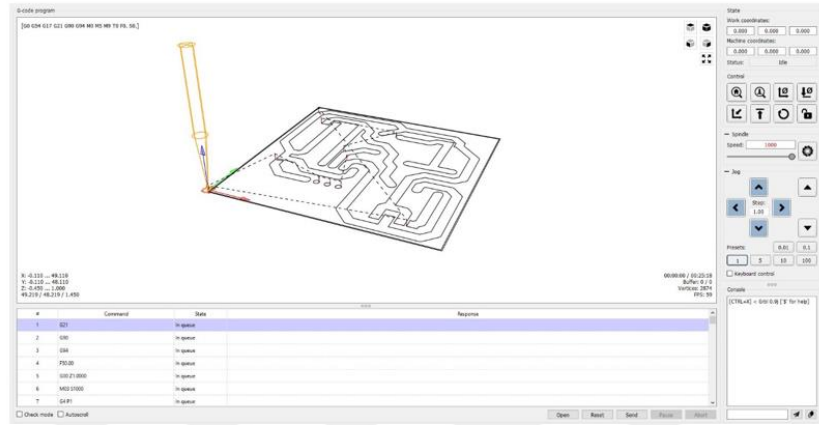
Şekil 2.16 : Gerber dosyasının özellikleri.

Daha sonra CNC makinesinde çizim yapılabilmesi için gerber dosyasındaki alt bakır yüzey dosyası g-code dosyasına dönüştürülmüştür. Bunun için FlatCAM programı kullanılmıştır. Bu program ile yolların derinliği, makinenin hızı, matkabın dönüş hızı ve delme işlemi yapmıyorken matkabın hangi mesafede duracağı parametreleri belirlenmiştir. Derinlik -0.45 mm, makine ilerleme hızı 50 mm/dk, matkabın dönüş hızı 1000 d/dk ve matkabın durma mesafesi 1 mm olarak ayarlanmıştır. Şekil 2.17’ de ayarlanan bu parametreler ve program arayüzü gösterilmektedir.



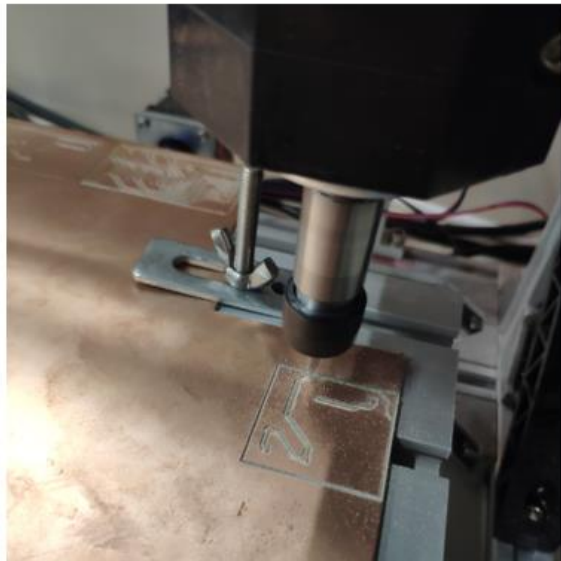
Şekil 2.17 : FlatCAM arayüzü ve çizim parametreleri.

Daha sonrasında G-code dosyasını CNC makinesine yüklemek için grblControl programı kullanılmıştır. Plaka makineye sabitlendikten sonra grblControl programı yardımıyla makinedeki matkap ucunun yeri plakaya göre ayarlanıp sıfırlanmıştır. FlatCAM programı ile grblControl programının boyutlarının mm olarak seçilmiş olmasına dikkat edilmiştir. Aksi takdirde çizim yapılırken mm ve inç kaynaklı boyut hataları oluşabilmektedir. Şekil 2.18’ de grblControl programına aktarılan çizim dosyası ve uç sıfırlama işlemi gösterilmiştir.

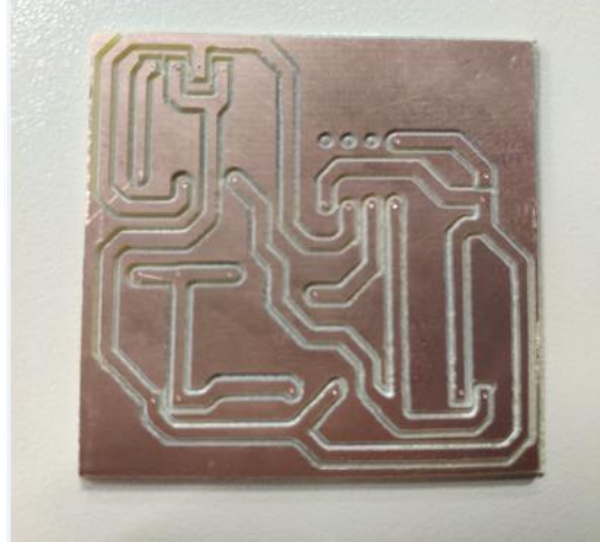


Şekil 2.18 : GrblControl program arayüzü.

Tüm ayarlamalar yapıldıktan sonra devre CNC kazıma makinası yardımı ile üretilmiştir. İşlem yaklaşık 25dk sürmüştür. Takım hızı artırılarak bu süre kısaltılabilmektedir. Fakat bu kısaltma işlemlerinde takım hızının zarar görebileceği dikkate alınmalıdır. Şekil 2.19 ve 2.20’ de üretim esnasında ve üretimin ardından çekilen devre kartının fotoğrafı gösterilmiştir.



Şekil 2.19 : CNC kazıma makinesi ve devre kartı üretimi.



Şekil 2.20 : Üretilmiş devre kartı.

Devre tamamlandıktan sonra yolların kısa devre kontrolü yapılmıştır. Daha sonrasında devre elamanlarının bacak uçları için delme işlemi yapılmıştır. Ardından elemanlar plaka üzerine lehimlenmiştir. Lehimleme işlemi bittikten sonra tekrardan her bir yol için kısa devre kontrolü yapılmıştır. QRD1114 sensörü kablo ile uzatılarak göz mesafesini kapatmayacak şekilde iş gözlüğü üzerine entegre edilmiştir. Böylelikle tüm devrenin gözlükte yer işgal etmemesi sağlanmıştır. Şekil 2.21’ de göz kırpma modülünün entegrasyonu tamamlanmış hali gösterilmiştir.



Şekil 2.21 : Göz kırpma modülü entegrasyonu.

Entegrasyon aşaması bittikten sonra devre osiloskop yardımı ile arduino üzerinde fonksiyon testine tabi tutulmuştur. Şekil 2.22’ de test aşamasından bir fotoğraf

gösterilmiştir. Göz kırpma algılama modülü tasarımı gereği gözler açık durumdayken lojik '1' seviyesinde bir değer üretmektedir. Bu fotoğrafta bulunan osiloskop ekranı incelendiğinde normal sinyal seviyesinden anlık düşüşler görülmektedir. Her göz kırpma sırasında op-amp çıkışında lojik '0' seviyesine anlık olarak düşüşler görüleceğinden, modülün tasarlandığı gibi çalıştığı sonucuna ulaşılmıştır.

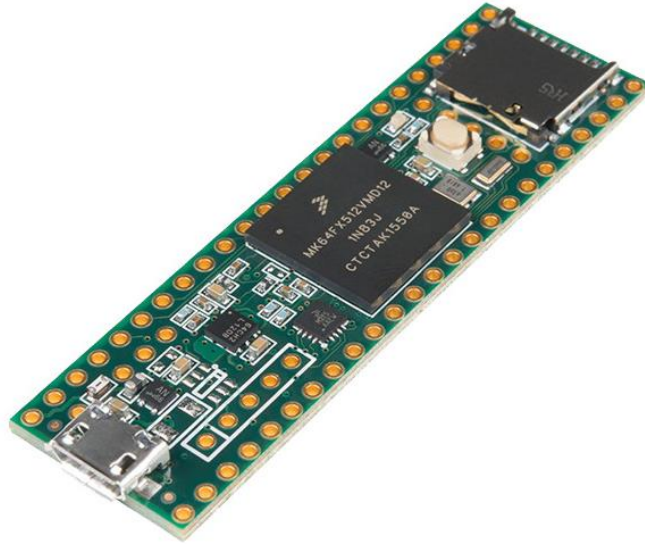


Şekil 2.22 : Göz kırpma modülü fonksiyon testleri.

2.1.4 Mikrokontrolcü

Pilot iş yükü ve sağlık takip sisteminin temel taşı olan mikrokontrolcü, performans, güç, hız, boyut, arayüz, giriş - çıkış ve fiyat gibi değişkenler ışığında değerlendirilmiştir. Bu mikrokontrolcü belirlenirken en önemli unsur sistemde kullanılan sensör modülleri ve bu modüllerin veriyolu bilgileridir. Sensörlerin veriyollarını destekleyeci şekilde giriş - çıkış bağlantısı bulundurması sistem entegrasyonu ve fonksiyonları açısından önemli bir yer teşkil etmektedir.

Teensy 3.5 geliştirme kartı hem boyut hem giriş - çıkış hem de hız / performans açısından istenilen tüm şartları fazlası ile yerine getirmektedir. Bu yüzden pilot iş yükü ve sağlık takip sisteminde mikrokontrolcü olarak belirlenmiştir. Şekil 2.23' te Teensy 3.5 geliştirme kartı gösterilmiştir.



Şekil 2.23 : Teensy 3.5 geliştirme kartı.

Teensy 3.5, küçük boyulu ve breadboard uyumlu bir geliştirme kartıdır. 32-bit ARM Cortex-M4 platformuna sahiptir. C ya da Arduino IDE' nin modifiye edilmiş bir versiyonu (Teensyduino) ile kolay bir şekilde programlama imkanı sunmaktadır. Tüm dijital ve analog girişler 5V uyumludur. Ayrıca tüm dijital pinleri kesme fonksiyonu yeteneğine sahiptir.

Çizelge 2.7' te Teensy 3.5 geliştirme kartının diğer özellikleri gösterilmektedir.

Çizelge 2.7 : Teensy 3.5 özellikleri.

Özellik Tanımı	Teensy 3.5
Besleme Voltajı	3.3 - 6V
İşlemci	MK64FX512VMD12 Cortex-M4F
Hız	120 MHz
Flash Hafıza	512 (Mbyte/s)
Ram	256 (kilobyte)
EEPROM	4096 (byte)
Dijital Giriş	58 Adet
Analog Giriş	27 Adet
Analog Çıkış	2 Adet
Zamanlayıcı	17 Adet
USB Haberleşme	1 Adet
Seri Haberleşme	6 Adet
SPI Haberleşme	3 Adet
I ² C	3 Adet
CAN Bus	1 Adet
Dijital Ses	1 Adet
SD Kart	1 Adet
Ethernet	1 Adet

Pilot iş yükü ve sağlık takip sisteminde kullanılan sensör modülleri analog, I²C ve dijital çıkışlara sahip olduğundan dolayı Teensy 3.5 geliştirme çoğu özelliği ile üstün görülmektedir. Bunun yanında tüm verilerin kaydedilmesi için gerekli SD kart yuvasını kendi üzerinde bulundurması sebebiyle ekstra bir fayda sağlamaktadır. Bu sayede mikrokontrolcüye uygun SD kart modülü arama ihtiyacına çözüm sağlamaktadır. Aynı zamanda üzerinde micro USB çıkışı bulundurması, anlık olarak alınan verilerin görüntülenmesinde epey kolaylık sağlamaktadır.

2.2 Yazılımlar

Proje kapsamında yapılan yazılım geliştirme çalışmaları ikiye ayrılmaktadır. Bu aşamalardan birincisi gömülü yazılım geliştirme, ikincisi ise kullanıcı arayüzü geliştirme bölümleridir. Gömülü yazılım geliştirme sürecinde, seçilen geliştirme kartı üzerinde yapılan çalışmalar bulunmaktadır. Bu kısımda ilgili sensör verileri , sistemin belirlenen çıktıları üretebilmesi için işlenmektedir. Daha sonra işlenen veriler analiz edilmek üzere geliştirme kartı üzerinde dahili şekilde bulunan mikro SD karta kaydedilmektedir. Ayrıca işlenen veriler gerçek zamanlı olarak mikro USB üzerinden kullanıcı arayüzüne aktarılmaktadır. İkinci aşama olan kullanıcı arayüzü geliştirme bölümünde ise mikro USB üzerinden aktarılan işlenmiş verilerin kullanıcı arayüzünün ilgili bölümlerinde görüntülenebilmesi için yapılan yazılım geliştirme çalışmalarından oluşmaktadır.

2.2.1 Gömülü yazılım geliştirme

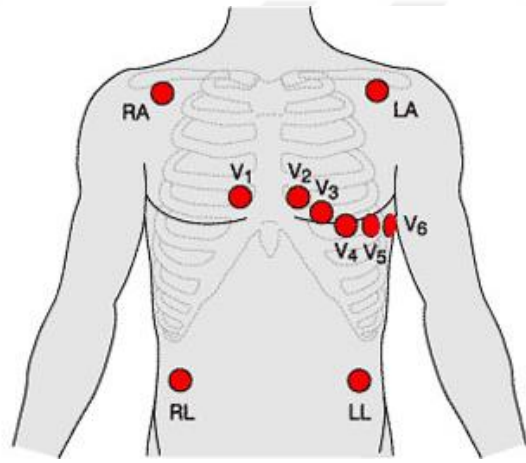
Pilot iş yükü ve sağlık takip sistemi projesi kapsamında kullanılan geliştirme kartı, Teensy 3.5 olduğundan dolayı Arduino IDE derleme programında entegre edilen Teensyduino programı yazılım olarak kullanılmaktadır. Arduino için yazılan çoğu program Teensy üzerinde de çalışabiliyor olmasından dolayı bu geliştirme kartı epey avantaj sağlamaktadır. Bunun yanında Teensyduino, birçok Arduino kütüphanesi ile de uyumludur. Bu sayede kullanılan bir çok fonksiyon için hazır kütüphaneler bulunmaktadır.

Proje çalışması kapsamında 3 farklı sensör kullanılmaktadır. Bu sensörlerden her biri için geliştirilen yazılımlar ayrı ayrı test edilmiştir. Bu testler sonucunda gerekli iyileştirmeler yapılarak tüm sistemin kod entegrasyonu başarılı bir şekilde tamamlanmıştır.

2.2.1.1 EKG ve nabız hızı ölçümü

EKG ve nabız hızı verileri için kullanmakta olduğumuz DFRobot nabız hızı algılama modülü sayesinde EKG verileri kolaylıkla elde edilmekte ve bu verilerden nabız hızı hesaplanabilmektedir. Bu modül Teensy 3.5' e doğrudan analog pininden bağlandığı için 0 - 3.3V arasında gelen EKG verileri programın seri portunda belirlenen frekans aralıklarında gösterilmektedir.

Elektrotların bağlanma şekline ve bağlanılan bölgeye göre EKG sinyali üzerinde bulunan R, P, Q, S ve T değerleri değişebilmektedir. DFRobot nabız modülü 3 elektrot içerdiğinden dolayı elektrot bağlantıları şekil 2.24' de gösterilen V1, V3 ve RL bölgelerinden yapılmıştır.



Şekil 2.24 : Elektrot bölgeleri [61].

Nabız hızı verisinin hesaplanabilmesi için DFRobot nabız hızı algılama modülü için üretici firma tarafından derlenmiş “heartspeed.h” kütüphanesi bulunmaktadır. Bu kütüphane dosyası incelenmek amacıyla geliştirme kartı üzerinde test edilmiş ve örnek olarak bazı ölçüm sonuçları alınmıştır. Bu sonuçlar incelendiğinde birbiri ile tutarsız bir şekilde bazı nabız hızı verileri olduğu gözlenmiştir. Örneğin nabız hızı 65 atım / dakika iken bir anda 85 atım / dakikalara yükseldikten sonra tekrardan 65 atım / dakika seviyelerine düştüğü görülmüştür. Bu değişimler yaklaşık 1 saniye içerisinde olduğu için sağlıklı bir bilgi olmadığı tespit edilmiştir. Bu sebeple hazır bulunan kütüphane dosyası yerine farklı bir algoritma geliştirilmiştir.

Sağlıklı bir bireyden alınan EKG değerleri geliştirme kartının analog girişinden okunduğu zaman ADC' nin bit sayısına göre maksimum bir değer almaktadır. Teensy 3.5 10 bit ADC' ye sahip olmasından dolayı analog pin üzerinden ölçülebilecek

maksimum deęer 1023 olarak bilinmektedir. Alınan EKG deęerleri incelendięinde R noktası yaklaşık olarak 1000 civarında olduęu gözlenmiştir. Tasarlanan algoritma temel olarak iki R tepe noktası arasında bulunan zaman farkından nabız hızı tespit etmek üzerine kurulmuştur.

Nabız hızı tespit edilirken öncelikle bir eşik deęeri belirlenmektedir. Bu eşik deęeri EKG sinyali üzerindeki R tepe noktalarını tespit etmek amacıyla 750 olarak belirlenmiştir. Tasarlanan algoritma belirli periyot aralıklarla sürekli olarak ölçülen EKG verisinin belirlenen eşik deęerinin üstüne olup olmadığını deęerlendirmektedir. EKG deęeri 750 eşik deęerinin üstüne çıktığı ilk anda o anki süre millis() fonksiyonu ile bir dizinin ilk sütununa kaydedilir [62]. Bu işlemin ardından 750 eşik deęerini ikinci kez geen ilk deęerin süresi de aynı dizinin ikinci sütununa kaydedilir. Bu işlemlerin hemen ardından diziye kaydedilen iki zaman deęeri arasındaki fark 60000 milisaniyeye bölünerek nabız hızı elde edilmektedir. Tasarlanan bu algoritmanın Teensy 3.5 için yazılmış kodu şekil A.1’de gösterilmektedir.

2.2.1.2 Baş pozisyonu ölçümü

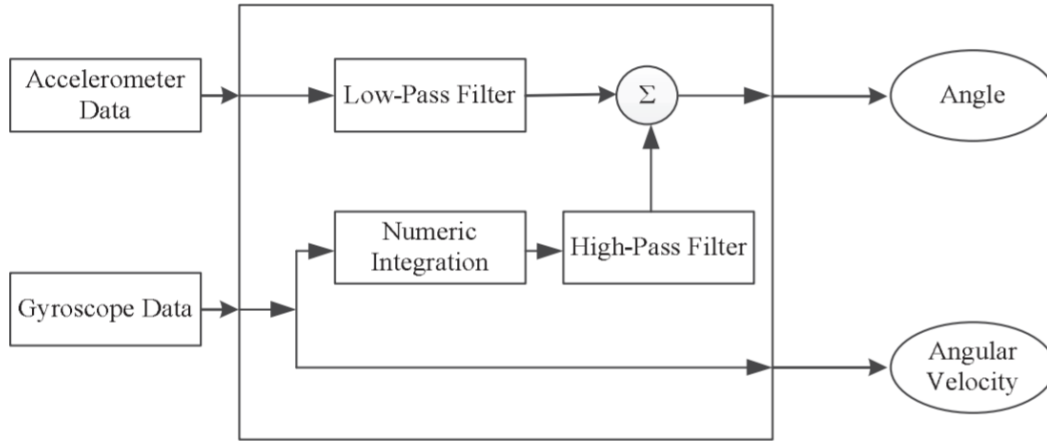
Pilot iş yükü ve saęlık takibi sisteminde, pilotun yorgunluk, uyuklama, yüksek g kuvveti ve oksijen yetmezlięi dolayısıyla baş pozisyonundaki anormal deęişiklikleri algılamak amacıyla iki eksenle baş pozisyonu takibi yapılmaktadır. Bu iki eksenle yunuslama ve yuvarlanma açılarındaki deęişiklikleri algılamak amacıyla IMU modülü kullanılmaktadır.

Jiroskop ve ivme ölçer sensörleri hassas açı ölçümlerinde tek başlarına kullanılmamaktadırlar. Bunun sebebi bias drift adı verilen bir kayma yapmalarıdır. Bu kaymalardan dolayı ölçümlerde hareketten kaynaklı hatalar oluşmaktadır. Bir dięer problem ise ivme ölçerler titreşime karşı çok duyarlılardır. Bu sebeple de çok küçük titreşimler bile büyük gürültüler oluşturarak sistem cevabını olumsuz bir şekilde etkilemektedir. Jiroskoplar ise bu gürültülerden ve titreşimlerden etkilenmemektedir. Bu gibi problemlerden dolayı, IMU modülü üzerinde bulunan jiroskop, ivme ölçer ve magnometreden alınan datalar belirli algoritmalar ile birleştirilerek daha doęru ve hassasiyeti yüksek sonuçlar elde edilebilmektedir. Bu algoritmalar arasında en yaygını olarak da bilinen tamamlayıcı filtredir.

Tamamlayıcı filtre 2007 yılında Shane Colton tarafından önerilmiştir [63]. Bu yöntemde alak geiren filtre ile düşük frekanslardaki eğim hesaplanması

yapılmaktadır. Yüksek geçiren filtre ile de yüksek frekanslardaki eğim hesaplaması yapılmaktadır. Daha sonra yapılan bu iki hesaplama toplanarak eğim açısı elde edilir.

Tamamlayıcı filtre , hem ivmeölçer hem de jiroskopun avantajını kullanmaktadır. Kısa vadede jiroskoptaki veriler kullanılmaktadır. Bu veriler dış kuvvetlere duyarlı değildir. Uzun vadede ise, veri kaymasını önlemek için ivmeölçerden gelen veriler ile düzeltme yapılmaktadır. Tamamlayıcı filtrenin prensibi şekil 2.25’ te gösterilmektedir.



Şekil 2.25 : Tamamlayıcı filtre algoritması [64].

Alçak geçiren filtrenin görevi, kısa vadeli dalgalanmaları filtreleyerek yalnızca uzun vadeli değişikliklerin geçmesine izin vermektir. Böyle bir hedefe ulaşmanın bir yolu, sonraki zaman aralıklarında dalgalanmaları yavaş yavaş arttırmaktır. Örneğin eğer eğim açısı sıfırdan başlarsa, ivmeölçerin okuması aniden belirli bir açı değerine atlar. Tamamlayıcı filtre kullanılarak açı tahmini, ani olmadan sorunsuz bir şekilde bu açı değerine yükselecektir. Bu açı değerine tam olarak ulaşmak için gereken süre, filtrenin parametrelerine ve veri toplamadaki örnekleme hızına bağlıdır.

Yüksek geçiş filtresi de, uzun vadeli dalgalanmalar için aynı işlemi yapmaktadır. Zaman içinde sabit olan sinyalleri filtrelerken kısa süreli sinyallerin geçmesine izin vermektedir. Böyle bir karakter, doğru bir tahmini açı değeri elde etmek yolunda kayan jiroskop verilerini engellemek için gereklidir.

Tamamlayıcı filtrenin matematiksel modeli denklem 2.1’ de ifade edilmiştir.

$$\theta_{Angle} = \alpha.(\theta_{Angle} + \omega_{Angle}.dt) + (1 - \alpha).a_{Acc} \quad (2.1)$$

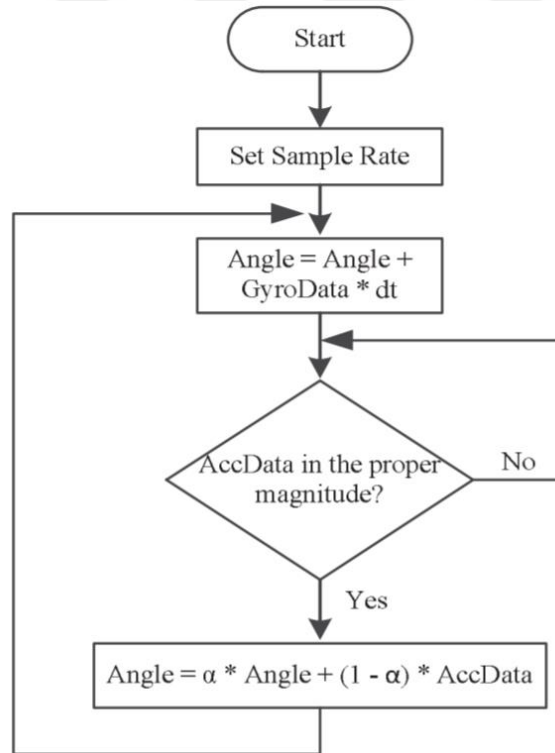
θ_{Angle} açısı yuvarlanma ya da yunuslama hareketinde oluşan eğim açılarıdır. α filtre katsayısıdır. ω_{Angle} jiroskoptan gelen açısal hızı temsil etmektedir. a_{Acc} ise

ivmeölçerden alınan verilerden elde edilen açı değeridir. Denklem 2.1 kullanılmadan önce jiroskop ve ivmeölçerden gelen veriler sıfırlanma ve ölçeklendirilmelidir. Filtre katsayısı α denklem 2.2’ de hesaplanmaktadır.

$$\alpha = \frac{\tau}{\tau + dt} \quad (2.2)$$

Denklem 2.2’ de τ filtrenin zaman sabitidir. Alçak geçiren bir filtre için, zaman sabitinden çok daha uzun olan sinyaller filtreden değiştirilmeden geçirebilirken, zaman sabitinden daha kısa olan sinyaller filtrelenebilir. Yüksek geçiren bir filtre için bu durumun tam tersi geçerlidir. Her zaman aralığı için, jiroskop verileri önce anlık açıyla entegre edilip, daha sonra alçak geçiren filtreden çıkan ivmeölçer verileri ile birleştirilmektedir. Böylelikle daha doğru ve mantıklı açısal tahminler yapılabilmektedir.

IMU veri füzyonunda tamamlayıcı filtrenin veri işleme prosedürü şekil 2.26’ da gösterilmektedir.



Şekil 2.26 : Tamamlayıcı filtre akış diyagramı [64].

IMU veri füzyonu işleminde uygulanan tamamlayıcı filtre bir döngü içinde uygulanmaktadır. Yunuslama veya yuvarlanma açı değerleri, zaman içinde entegre edilen jiroskop değerleri ile güncellenmektedir. Daha sonra filtre, ivmeölçer tarafından

ölçülen ivme büyüklüğünün, gerçek bir g-kuvvet vektörü olabilecek makul bir değere sahip olup olmadığını kontrol etmektedir. Değer çok küçük veya çok büyükse, bozucu etki olarak dikkate alınmaz. Bu veri füzyon yönteminde, mevcut yunuslama veya yuvarlanma açısı önceki yunuslama veya yuvarlanma açısı, anlık jiroskop çıkışı, filtre katsayısı α ve ivmeölçerden gelen mevcut verilerin yüzdesi ile belirlenmektedir.

Pilot iş yükü ve sağlık takibi çalışmasında MPU 9259 IMU modülünden alınan jiroskop ve ivmeölçer verileri tamamlayıcı filtre yardımıyla füzyon edilerek baş pozisyon açıları elde edilecektir. Pilotun öne-arkaya ve sağa-sola olan eğilme açıları tespit edilerek sağlık durumu ile ilgili çıkarımlar yapılabilecektir. MPU 9250 IMU modülü için hazırlanan kod şekil A.2 de gösterilmiştir.

2.2.1.3 Göz kırpma sayısı hesaplanması

Pilot iş yükü ve sağlık takip sistemi çalışması kapsamında bir dakikada oluşan göz kırpma verisinin elde edilmesi pilotun iş yükü büyüklüğünü ve stresini belirlemede kullanılan etkili yöntemlerden birisidir. Bu bölümde daha önceden üretilmiş olan göz kırpma algılama devresi için mikrokontrolcü üzerinde oluşturulacak olan algoritmanın tasarımı gerçekleştirilmiştir.

Tasarlanan algoritmaların diğer modüllerin çalışmalarına olumsuz etki etmeyecek şekilde olması planlanmıştır. Fakat göz kırpma süresi çok anlık bir durum olduğu için onu algılamak algoritmanın normal akışında oldukça zordur. Bu yüzden bu tip durumların algılanabilmesi için mikrokontrolcüler üzerinde kesme fonksiyonları kullanılmaktadır.

Mikrokontrolcülerde kesme işlemi derhal ilgilenilmesi gereken bir olayı ifade etmektedir. Bir kesme sinyali işlemciyi uyarır ve işlemcinin o anda yürütülen kodu kesmesi için bir istek oluşturur. Böylece olay anlık gelişme zamanında işlenebilir. Kesme sinyali geldiği anda işlemci geçerli etkinliklerini askıya alarak, durumunu kaydederek ve olayla başa çıkmak için kesme işleyicisi adı verilen bir işlevi yürüterek yanıt verir. Bu kesinti geçicidir ve işlemci kesinti işleyicisi bittikten sonra normal faaliyetlerine devam etmektedir.

Göz kırpma algılama devresinde giriş sinyalinin bir referans değer ile karşılaştırılması amacıyla opamp kullanılmıştır. Opampa giren IR sensörden gelen sinyal, referans sinyali ile karşılaştırıldıktan sonra, 0 veya 3.3 V değerlerine yakın bir sinyal üretmektedir. Üretilen bu sinyalin kesme fonksiyonu ile algılanabilmesi için

mikrokontrolcünün kesme fonksiyonunu kullanabildiği bir pinine bağlanmıştır. Böylelikle göz kırpma durumlarında mikrokontrolcü kesme fonksiyonu sayesinde anlık olarak göz kırpmaları tespit edebilmektedir. Ek olarak göz kırpma modülünün kullanıcı üzerine entegrasyonunu kolaylaştırmak için göz kırpma anında eş zamanlı olarak bir led yanmaktadır. Böylelikle kullanıcının göz kırpma fonksiyonun doğru bir şekilde algılanıp algılanmadığı kontrol edilmektedir. Şekil A.3' te göz kırpma modülü için hazırlanmış algoritma mevcuttur.

2.2.1.4 SD karta veri kaydetme

Pilot iş yükü ve sağlık takip sisteminde pilottan alınan verilerin daha sonra incelenebilmesi için bir hafıza birimine tüm verilerin kaydedilmesi gerekmektedir. Teensy 3.5 kendi üzerinde dahili olarak mikro SD kart yuvası bulundurmaktadır. Bu sayede dışarıdan ekstra bir SD kart modülüne gerek duyulmamıştır.

Mikrokontrolcüye gelen veriler SD karta kaydedilmeden önce .csv uzantılı bir dosya türü açılmakta ve gelen tüm dataların bu dosya içerisine kaydedilmesi sağlanmaktadır. Bu dosya oluşturma işlemi esnasında dosya isminden kaynaklı hatalı dosyaya yazma, üzerine yazma ve veri kaybedilmesi gibi problemlerin önüne geçilmesi amacıyla otomatik dosya numarası üreten bir fonksiyon oluşturulmuştur. Bu fonksiyon sayesinde her başlangıçta dosya ismine daha önceden EEPROM' a kaydedilmiş bir sayı verilmekte ve sonrasında bu sayı bir artırılarak EEPROM' da güncellenmektedir. Böylelikle bir sonraki yeniden başlatmada yeni dosya numarası bir önceki dosya numarasından bir fazla olmaktadır. Bu yöntem sayesinde kullanıcı her defasında yeniden dosya ismi oluşturmakla uğraşmamaktadır.

Kullanıcının veri analizini kolaylaştırmak ve gereksiz bilgi kalabalığından uzaklaştırılması amacıyla bir adet kayıt sonlandırma butonu fonksiyonu algoritmaya entegre edilmiştir. Böylelikle kullanıcı sadece veri kaydetmek istediği aralıkta önce sisteme reset atıp sonrasında da butona basarak kaydı sonlandırmaktadır. Bu kayıt sonlandırma tuşu da kesme fonksiyonu olarak görev yapmakta ve basıldığı anda otomatik olarak ilgili fonksiyonu çalıştırmaktadır. Şekil A.4' te SD kart ile ilgili yazılmış kod gösterilmektedir.

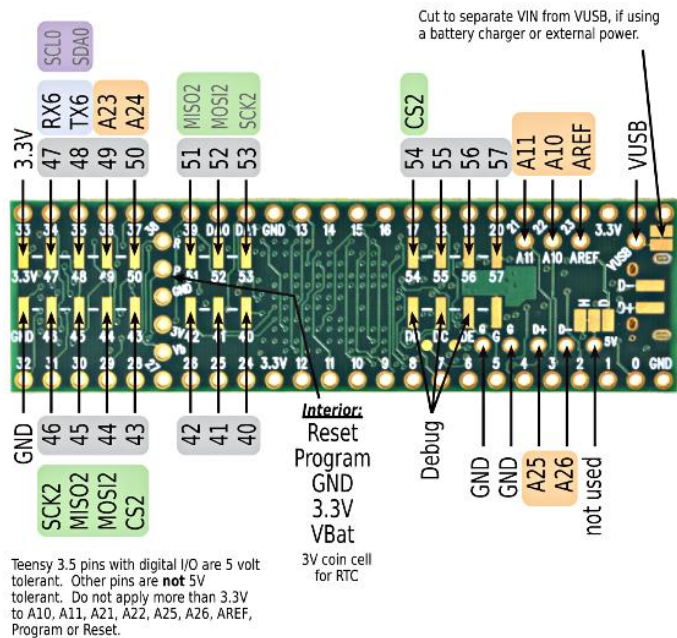
2.2.1.5 Gerçek zaman saati (RTC)

Mikrokontrolcülerde sıklıkla kullanılmakta olan gerçek zaman saati, çoğu data toplama sisteminin vazgeçilmez bir elemanıdır. Bazı RTC' ler modül şeklinde bazıları ise mikrokontrolcü üzerinde dahili bulunabilmektedir. Pilot iş yükü ve sağlık takibi projesi kapsamında RTC, daha sonradan analiz edilmek üzere kaydedilen verilerin süreye göre sınıflandırılmasında kullanılmaktadır.

RTC' ler genellikle alternatif bir güç kaynağına sahiptir. Bu sayede birincil güç kaynağı kapalıyken veya kullanılmıyorken zaman tutmaya devam edebilmektedirler. Bu alternatif güç kaynağı normalde eski sistemlerde bir lityum pildir, ancak bazı yeni sistemler bir süperkapasitör de kullanılabilmektedir. Teensy 3.5 için 3V' luk CR2032 lityum bir pil kullanılmaktadır.

Çoğu RTC kristal osilatör kullanır. Kristal frekansı genellikle 32.768 kHz' dir. Teensy 3.5 diğer sürümlerine kıyasla bu kristali kendi üzerinde bulundurmaktadır. Bu sayede dışarıdan ekstra bir kristal osilatöre gerek duymamaktadır.

Şekil 2.27' de Teensy 3.5' in alt yüzünde bulunan Vb olarak isimlendirilmiş pin RTC için besleme pinidir. 3V' luk CR2032 lityum pil bu pin üzerinden RTC için besleme olarak kullanılmaktadır.



Şekil 2.27 : Teensy 3.5 RTC besleme pini.

Şekil A.5' te RTC için hazırlanan kod gösterilmektedir.

2.2.2 Kullanıcı arayüzü geliştirme

Pilot iş yükü ve sağlık takip sisteminde anlık olarak gelen pilot verilerinin operatör ekranında gösterilmesi hedeflenmiştir. Bu sayede uçuş simülatörü ortamında görevleri icra eden pilotun anlık olarak durumu kullanıcı arayüz programı üzerinden takip edilmektedir.

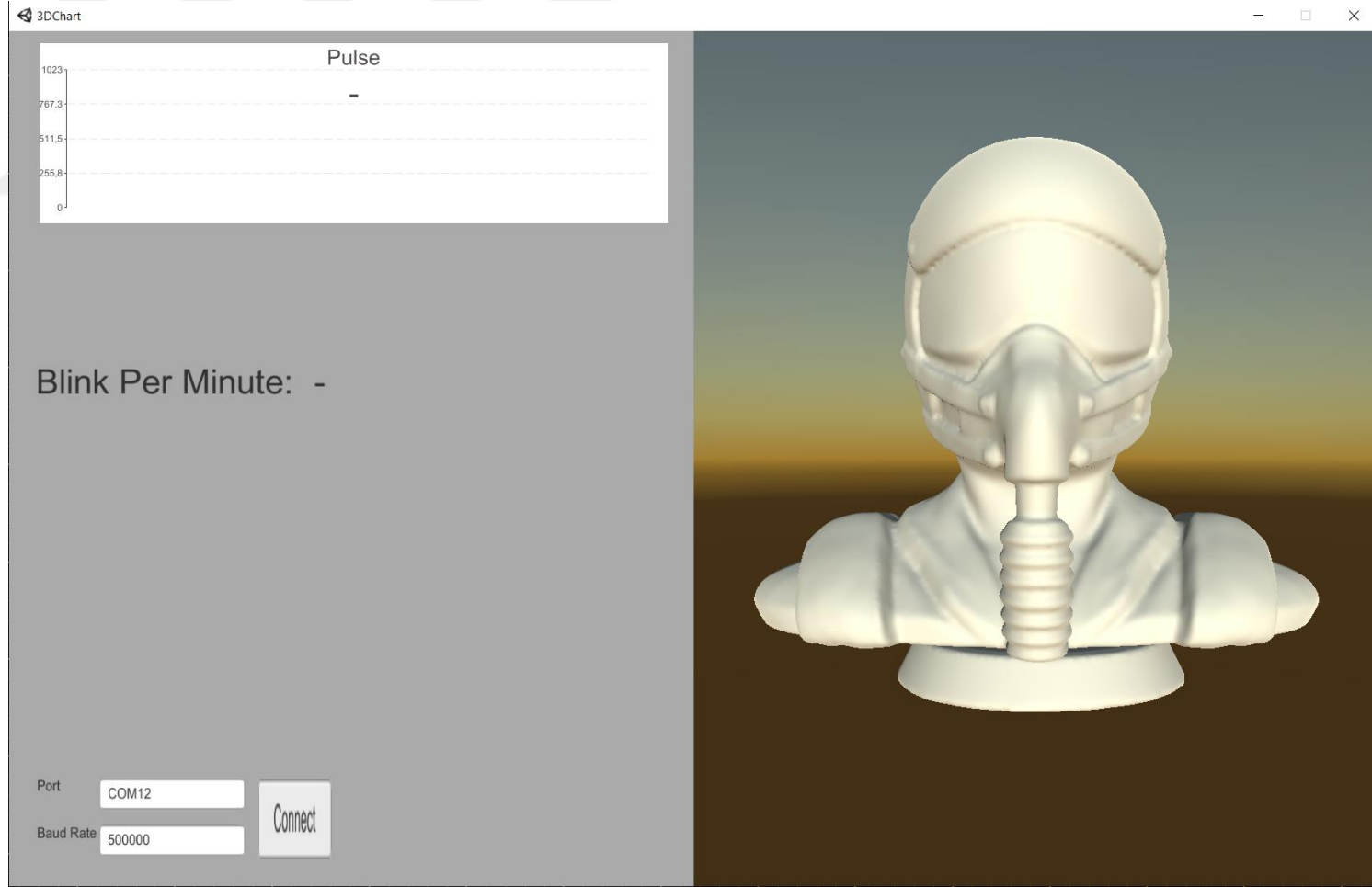
Bu kullanıcı arayüzü Unity programı üzerinde tasarlanmıştır. Unity Rusya' da geliştirilen ve yayınlanan bir oyun motorudur. Öncelikle farklı bir amaç için üretilmiş olup daha sonra oyun yapmak amacıyla kullanılmış ve geliştirilmiştir. Fakat Unity 3D bununla yetinmeyerek kendini daha da geliştirmiştir. Pilot iş yükü ve sağlık takip sisteminde Unity kullanılmasının bir diğer sebebi ücretsiz olmasıdır.

Unity üzerinde açılan C# dosyası ile C# dilinde gerekli görsel arayüzler tasarlanabilmektedir. Pilot iş yükü ve sağlık takip sisteminde C# dili ile ilgili algoritmalar oluşturulmuştur. Bu algoritmalar gelen verilerin seri porttan okunması, okunan verilen parçalanarak anlamlı veriler elde edilmesi ve bunların ilgili görsel ekranda gösterilmesi aşamalarından oluşmaktadır.

Program çalıştırılmadan önce pilot iş yükü ve sağlık takip sisteminin bilgisayara bağlı olduğu port ve ilgili portun baud rate' i seçilmektedir. Şekil 2.28' de pilot iş yükü ve sağlık takibi sistemi için geliştirilen kullanıcı arayüzü programı yer almaktadır.

Bu kullanıcı arayüzünde EKG verisi kayan bir grafik olarak gösterilmektedir. Bu kayan grafik pilotun anlık nabız sinyallerini ifade etmektedir. Hemen bu grafiğin üzerinde ise nabız hızı verisi yer almaktadır. Bir dakikada oluşan göz kırpması EKG grafiğinin hemen altında yer almaktadır. Ekranın sağ tarafında ise pilotun başını simüle eden bir 3 boyutlu kafa görüntüsü yer almaktadır. Bu bölümde ise pilotun başı üzerindeki IMU modülünden gelen baş pozisyon datasına göre pilotun baş pozisyonu gösterilmektedir.

Şekil B.1' de kullanıcı arayüzü ile Teensy 3.5 arasındaki haberleşmeyi sağlayan seri port oluşturma kodu gösterilmektedir. Şekil B.2' de ise tüm verilerin seri porttan alınıp anlamlı veriler haline dönüştürüldükten sonra ilgili bölümlere yönlendirildiği kod gösterilmektedir.



Şekil 2.28 : Kullanıcı arayüzü.

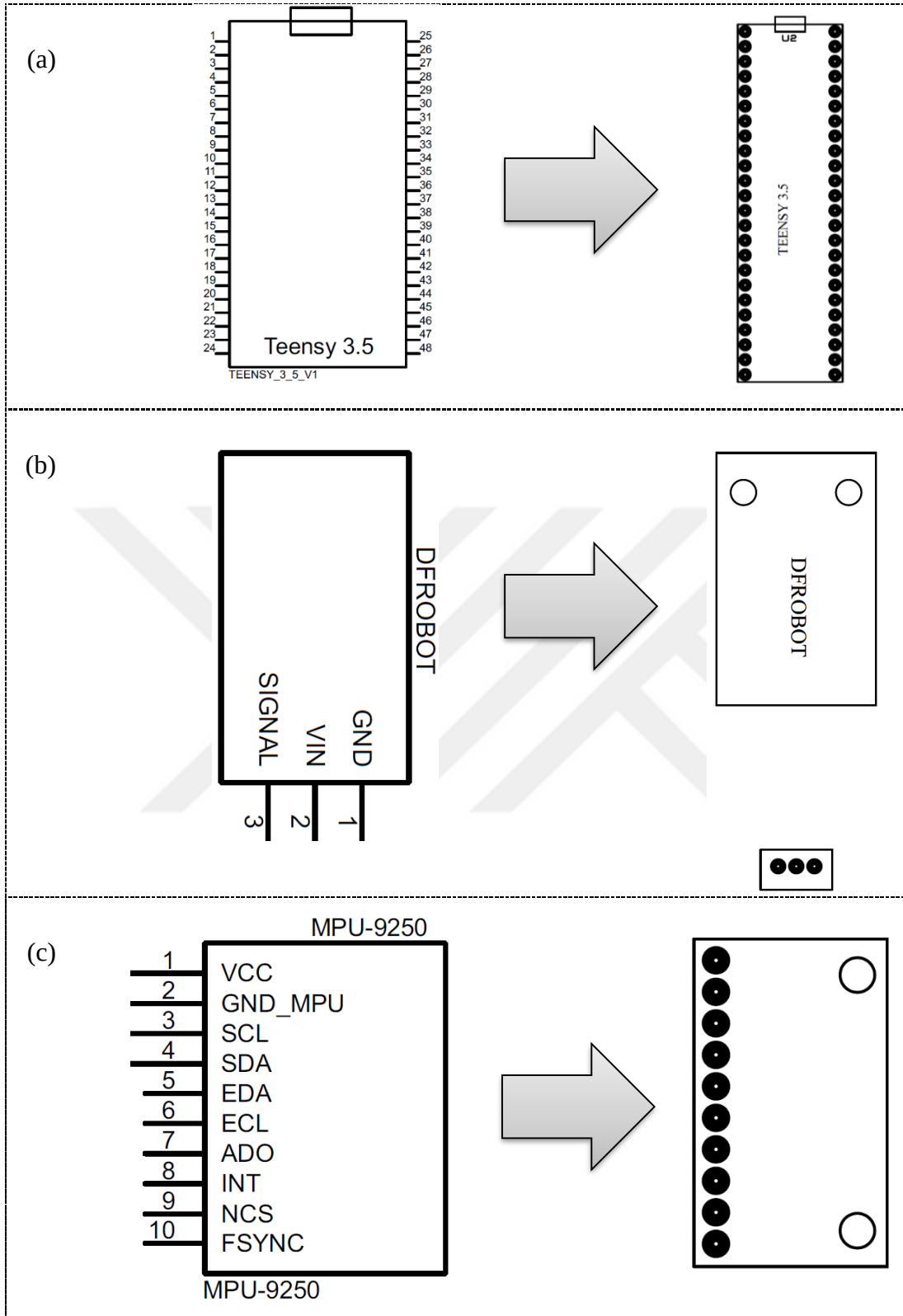
3. ELEKTRONİK DEVRE KARTI GELİŞTİRİLMESİ

Pilot iş yükü ve sağlık takip sisteminin kolaylıkla kullanılabilir bir yapıda olması ve pilot tarafından entegrasyon sürecinin minimum seviyede olabilmesi amacıyla uygun şartları sağlayabilecek bir baskı devre kartı geliştirilmesi uygun görülmektedir. Bu baskı devre kartının, sistemde kullanılan elektronik komponentler göz önünde bulundurularak minimum boyutlarda tasarlanması hedeflenmiştir. Boyut olarak küçük olması ürünün hem kompakt bir yapıda hem de ağırlık olarak oldukça düşük seviyede olmasını sağlamaktadır. Baskı devre kartının boyut itibari ile küçük olması düşünüldüğünden dolayı çift taraflı ve delik içi kaplamalı olacak şekilde üretilmesi planlanmıştır.

Baskı devre kartı geliştirilme süreci, komponent kütüphanesi oluşturma süreci, devre kartının şematik çizimleri, PCB çizimleri, 3 boyutlu model dosyalarının oluşturulması ve üretim süreçlerinden oluşmaktadır. Daha sonra üretilen baskı devre fonksiyon testlerine tabi tutulmakta ve çalışabilirliği test edilmektedir.

3.1 Kütüphane Dosyalarının Oluşturulması

Devre kartı tasarımına başlanılmadan önce pilot iş yükü ve sağlık takibi projesi kapsamında kullanılması planlanan elektronik komponentlerin ilgili çizim programı içerisindeki kütüphanede olup olmadığı incelenmiştir. Bu inceleme sonrasında, mikrokontrolcü olarak kullanılması planlanan Teensy 3.5 geliştirme kartı, DFRobot EKG ve nabız algılama modülü, ve ataletsel ölçüm modüllerinin kütüphanelerinin bulunmadığı tespit edilmiştir. Bu tespitin ardından ilgili komponentlerin ölçümleri ve çizimleri tamamlanarak çizim program kütüphanesine dahil edilmişlerdir. Bu çizimler tıpkı göz kırpma algılama modülünde de yapıldığı şekilde kumpasla ölçümler alınarak tamamlanmıştır. Baskı devre yöntemi ile üretilmesi planlanan devre kartı çift taraflı delik içi kaplamalı bir yapıda olması planlandığından dolayı komponent pinlerinin çapları bu delik içi kaplama toleransları göz önünde bulundurularak oluşturulmuştur. Şekil 3.1’ de oluşturulan kütüphane dosyalarını sırasıyla gösterilmektedir.



Şekil 3.1 : Şematik (solda) ve PCB çizimleri (sağda): (a) Teensy 3.5. (b) DFRobot modülü. (c) MPU-9250 IMU.

3.2 Devre Şeması Tasarımı

Devre şematik çizim bölümünün ilk aşamasında öncelikle devrede yer alan elektronik komponentlerin besleme gerilimleri sınıflandırılmıştır. Bu sınıflandırmanın amacı devre kartı çizilirken hem besleme voltaj seviyelerine uygun regülatörler kullanılması hem de devrede bulunan komponentlerin güç tüketimine göre de kullanılacak ana besleme kaynağının belirlenmesidir.

3.2.1 Elektriksel güç haritası

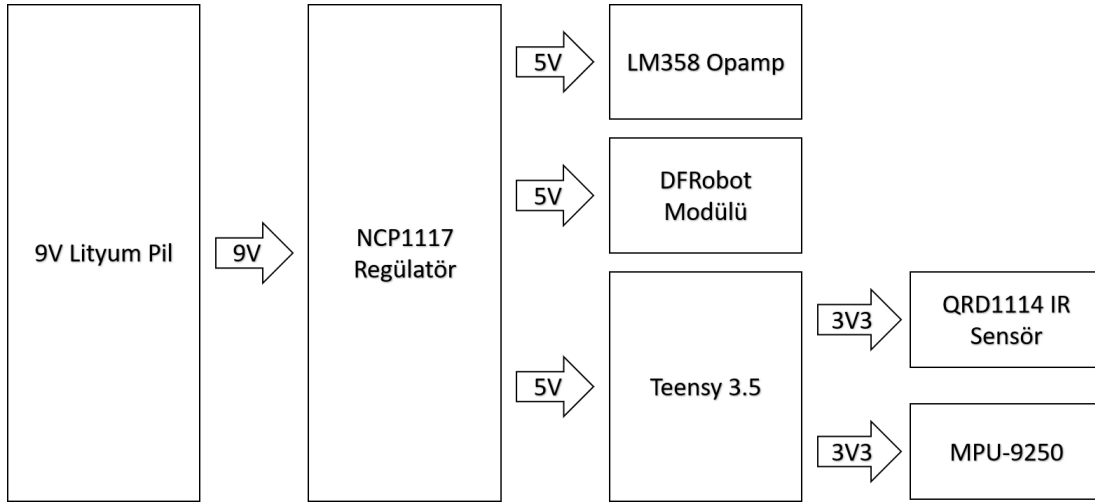
Devrede yer alan elektronik komponentler incelendiğinde 2 farklı besleme gerilimi olduğu belirlenmiştir. Bu besleme gerilimleri 3.3V ve 5V olarak sınıflandırılmıştır. Çizelge 3.1’ de devrede bulunan güç tüketimi yüksek ana komponentlerin besleme gerilimi ve kaynaktan çekilen akımlar gösterilmektedir.

Çizelge 3.1 : Besleme gerilimi ve kaynaktan çekilen akım.

Komponent	Besleme Gerilimi	Akım
Teensy 3.5	5V	< 45 mA
MPU-9250	3.3V	< 3.5 mA
DFRobot Nabız Modülü	5V	< 10 mA
QRD1114 Sensörü	3.3V	< 50 mA
LM358 Opamp	5V	< 31 mA

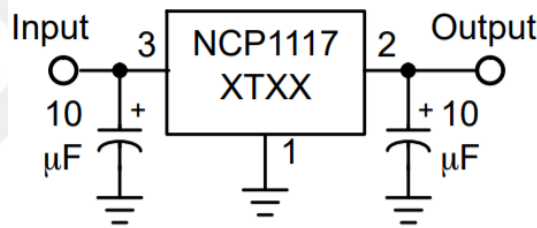
Çizelge 3.1 incelendiği taktirde toplam akım tüketiminin yaklaşık olarak 140 mA civarında olduğu görülmektedir. Besleme kaynağı olarak 9V 800 mAh bir lityum iyon pil seçildiği taktirde ilgili devre kartı yaklaşık 5 saat 45 dakika boyunca çalışabilmektedir. Bu da simülatör ortamındaki test uçuşlarında yeterli bir çalışma süresi sağlayabildiğini göstermektedir.

Besleme gerilim seviyeleri göz önünde bulundurulduğunda, 9V’ luk bir pilden 2 farklı gerilim seviyesi elde edilmesi gerekmektedir. 3.3V luk gerilim seviyesi Teensy 3.5 üzerinden sağlanabildiğinden dolayı 5V 1A’ lik bir regülatör kullanımı uygun bulunmuştur. Teensy 3.5 üzerinde bulunan 3.3V regülatör 250 mA’ e kadar besleme yapabildiğinden dolayı 3.3V besleme gerilimi ile çalışan komponentlerin toplam tüketiminin bu sınırın içerisinde olduğuna dikkat edilmiştir. Şekil 3.2’ de devre kartının güç haritası gösterilmektedir.



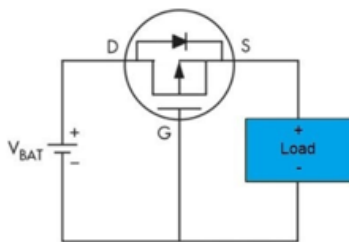
Şekil 3.2 : Elektriksel güç haritası.

Şekil 3.2 gösterildiği üzere 9V' luk pilden 5V gerilim seviyesi elde etmek amacıyla NCP1117 5V regülatörü kullanılmıştır. NCP1117 regülatöründen 5V seviyesinde bir gerilim elde etmek için şekil 3.3' de gösterilen devre şeması referans alınmıştır.



Şekil 3.3 : 5V regülatör devre şeması.

Devreye elektriksel güç verecek olan lityum iyon pil yönlü konnektör ile devre kartına bağlanmaktadır. Yönlü konnektörler günümüzde oldukça yaygın kullanılmaktadır. Böylelikle olası ters bağlamalardan kaynaklı komponentlerin bozulması engellenmektedir. Pilot iş yükü ve sağlık takip sisteminde p kanal mosfet kullanılarak ters bağlanmalara karşı ekstra bir güvenlik önlemi daha alınmıştır. Böylelikle geliştirme sürecindeki çalışmalarda oluşabilecek arızaların da önüne geçilmiştir. Şekil 3.4' te p kanal mosfet ile ters gerilimden kaynaklı arızaların önüne geçilmek amacıyla nasıl bir koruma devresi yapıldığı gösterilmektedir.



Şekil 3.4 : P kanal mosfet ile ters gerilim koruma devresi.

3.2.2 Haberleşme protokolleri

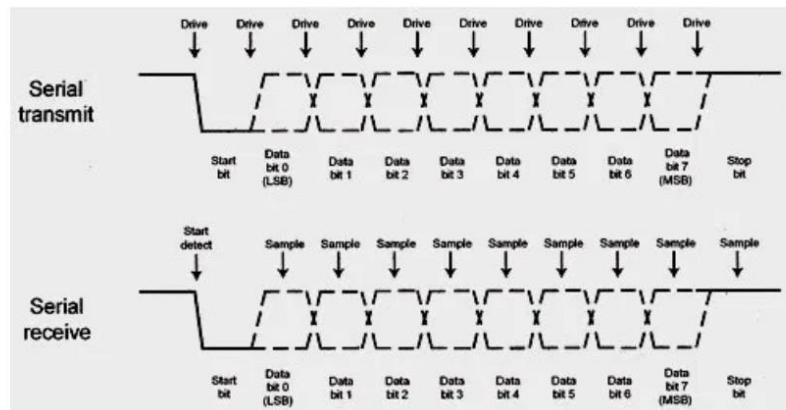
Gömülü sistemlerde mikrokontrolcü ile çevre birimler arasındaki veri transferini sağlamak amacıyla çeşitli iletişim protokolleri kullanılmaktadır. Bu protokoller arasında en bilindik olanları SPI, I²C, CAN bus ve UART' tır.

Pilot iş yükü ve sağlık takip sisteminde kullanılan sensör ve komponentler sistemin merkezi olan mikrokontrolcü ile belirli haberleşme protokolleri üzerinden haberleşme sağlanmaktadır. Pilot iş yükü ve sağlık takip sisteminde SD kart ile mikrokontrolcü arasında SPI, MPU-9250 ile mikrokontrolcü arasında I²C haberleşme protokolü kullanılmaktadır. Ayrıca mikrokontrolcü üzerinde işlenen veriler UART protokolü ile kullanıcı arayüzü programına aktarılmaktadır. Diğer sensör modülleri analog ve dijital sinyal çıkışı ürettiklerinden dolayı doğrudan mikrokontrolcü üzerindeki analog ve dijital pinlerden veri göndermektedir.

3.2.2.1 UART haberleşme protokolü

UART protokolü bilgisayar ve mikrokontrolcüler veya mikrokontrolcüler ve çevre birimleri arasındaki haberleşmeyi sağlayan haberleşme protokolüdür. UART asenkron olarak çalışmaktadır. Bu sebeple herhangi bir saat sinyaline ihtiyaç duymamaktadır.

UART haberleşmesindeki önemli parametlerden ilki veri taşıma hızıdır (baud rate). Haberleşme gerçekleştirilirken bu hızın ayarlanması gerekmektedir. Bu hız bir saniyede ne kadarlık byte taşınacağını belirtmektedir. UART protokolünden veri iletiminin gerçekleşmesi için şekil 3.5' teki yapı kullanılmaktadır.



Şekil 3.5 : UART protokol yapısı.

UART protokolü sırasıyla başlangıç biti, data bitleri, parity biti ve bitiş bitlerinden oluşmaktadır. İki sistem arasındaki haberleşmenin başlayabilmesi için veri iletim hızlarının aynı olması gerekmektedir. Şekil 3.5’ te görüldüğü üzere haberleşmenin gerçekleşebilmesi için öncelikle 1 seviyesinde bulunan iletişim hattı 0 seviyesine çekilmektedir. Bu durum başlangıç bitini temsil etmektedir. Sonrasında ise gönderilmek istenen veriler başlangıç bitinin arkasına eklenmektedir. Parity biti isteğe göre değişiklik göstermektedir. Son olarak ise 0 seviyesindeki iletişim hattı 1 seviyesine çekilir. Bu da bitiş bitini temsil etmektedir. Böylelikle iletişimin sonlandığı mesajı alıcıya iletilmektedir. Gönderici sistem bu işlemleri yaparken alıcı sistem de aynı şekilde işlemleri tamamlamakta ve gelen verileri de kendi üzerindeki UART data register’ ına yazmaktadır. Teensy 3.5 için seçilebilen veri iletim hızı ve hata oranı çizelgesi çizelge 3.1’ de verilmiştir.

Çizelge 3.2 : Teensy 3.5 veri iletim hızı ve hata oranı çizelgesi [65].

Veri İletim Hızı	Teensy 3.5 Hata oranı (120 MHz) Serial 1-2	Teensy 3.5 Hata oranı (120 MHz) Serial 3-6
4608000	0.16%	0.16%
2000000	0.00%	0.00%
1000000	0.00%	0.00%
921600	0.16%	0.16%
500000	0.00%	0.00%
460800	-0.03%	0.16%
250000 (DMX)	0.00%	0.00%
230400	-0.03%	-0.03%
115200	0.02%	-0.03%
57600	0.01%	0.02%
38400	0.00%	0.00%
31250 (MIDI)	0.00%	0.00%
19200	0.00%	0.00%
9600	0.00%	0.00%
4800	0.00%	0.00%
2400	0.00%	0.00%
1200	0.00%	0.00%
300	205.18%	52.59%

Teensy 3.5 üzerinde toplamda 6 adet UART haberleşme portu bulunmaktadır. Kullanıcı arayüzüne mikro usb üzerinden veri aktarımı sağlayacak olan 1 numaralı seri porttur. Teensy 3.5 üzerinde önceden tanımlı olan seri haberleşme veri iletim hızı 115200 olduğu gözlenmiştir. Çizelge 3.1 incelendiğinde veriiletim hızı 500000 iken

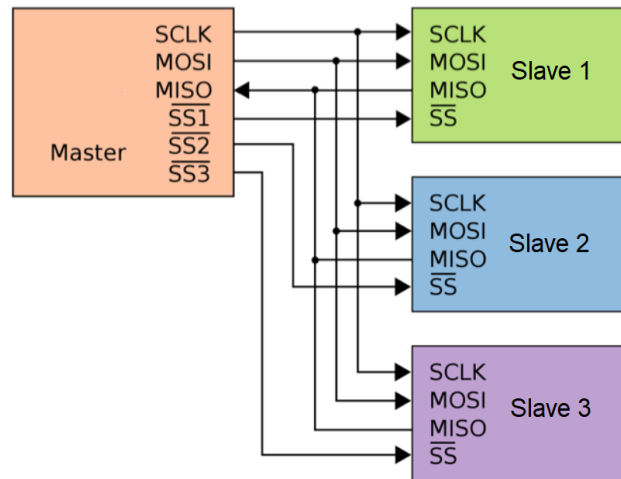
hata oranı 0.00% olarak görüntülenmektedir. Bu sebeple mikrokontrolcü ile kullanıcı arayüzü arasındaki veri iletim hızı 500000 olarak belirlenmiştir.

3.2.2.2 SPI haberleşme protokolü

Seri Çevresel Arabirim (SPI), başta gömülü sistemler olmak üzere kısa mesafeli iletişim için kullanılan senkronize ve seri haberleşme birimidir. Bu arayüz 1980lerde Motorola tarafından geliştirilmiş ve standart haline gelmiştir. SD kartlar ve likit kristal ekranların da içerisinde bulunduğu çok sayıda uygulama alanı mevcuttur.

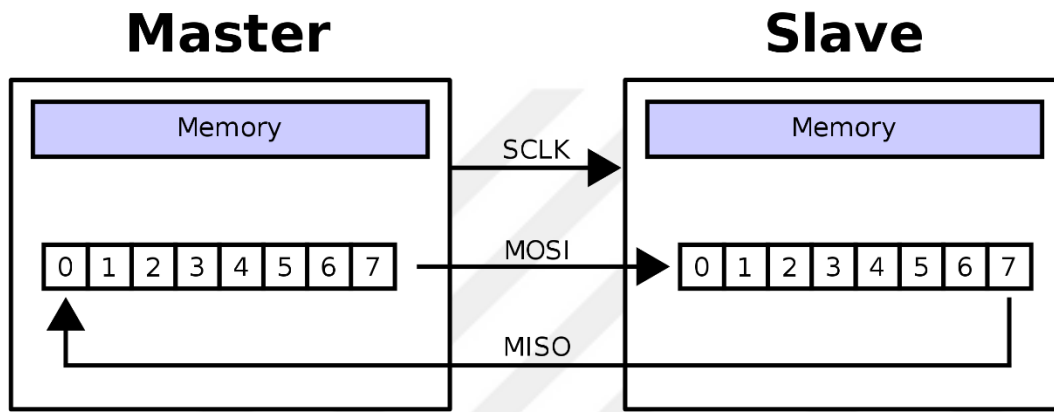
SPI cihazlar kendi aralarında master – slave mimari referansında full dublex modda iletişim kurmaktadır. Bu mimaride master cihaz okuma ve yazma için çerçeveyi oluşturur. Birden fazla slave cihaz CS ya da SS pinleri üzerinden seçilebilmektedir. Master ve çevresel cihazlara bağlanan SPI hattı MISO (Master In Slave Out), MOSI (Master Out Slave In) ve SCK (Serial Clock) olmak üzere üç hat bulunmaktadır. MISO, slave cihazlardan yollanan verilerin master cihaza aktarıldığı hattır. MOSI ise master cihazdan yollanan verilerin slave cihazlara aktarıldığı hattır. SCK da SPI haberleşmenin senkronize bir şekilde olmasını sağlayan saat sinyalinin bulunduğu hattır. Saat sinyalleri master cihazlar tarafından üretilmektedir.

MISO ve MOSI hatlarının olması bu haberleşme mimarisindeki yolların tek yönlü olduğunu göstermektedir. Bağlanan her slave cihazın SS ya da CS seçim ayağı sayısı kadar master cihazdan hat çıkmaktadır. Her cihaz için master cihazdan ayrı bir hat çıkmaktadır. Çıkan bu hatlardan hangisi logic 0 seviyesinde ise o slave cihaz master cihaz ile iletişime geçmektedir. Şekil 3.6’ da SPI master ve slave cihaz arasındaki bağlantı gösterilmiştir.



Şekil 3.6 : SPI master ve slave cihaz bağlantıları.

SPI haberleşme arayüzünde iletişim master cihazın saat sinyali ile başlamaktadır. Saat sinyali ile birlikte master cihaz tarafından seçilen slave cihaz ile iletişim başlamaktadır. İletişim esnasında master cihaz MOSI hattı üzerinden slave cihaza veri bitlerini iletmekte ve iletilen bitler slave cihaz tarafından okunmaktadır. Slave cihaz da MISO hattı üzerinden master cihaza bitleri iletmekte ve iletilen bitler master cihaz tarafından okunmaktadır. Haberleşme tamamlandığında ise master cihaz saat sinyalini durdurarak seçili slave cihazı seçimden çıkarır. Böylelikle iki cihaz arasında veri iletimi kesilmiş olur [66]. Şekil 3.7’ de master ve slave cihaz arasındaki veri transferine ait mimari gösterilmektedir.



Şekil 3.7 : SPI master ve slave cihaz veri transfer mimarisi [66].

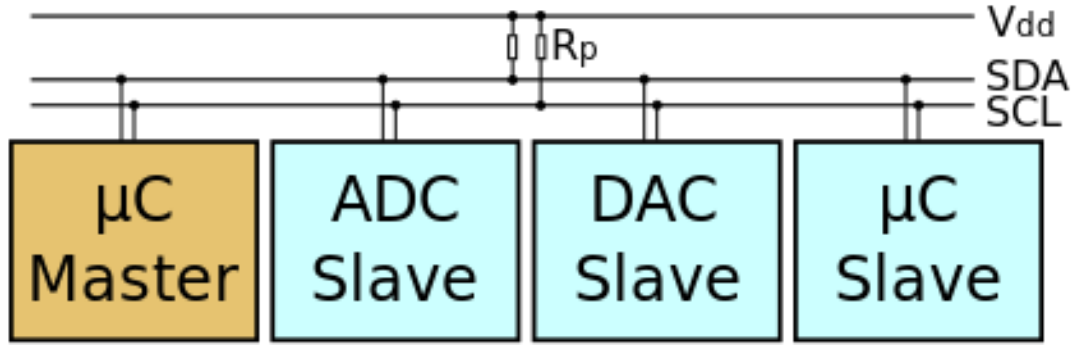
3.2.2.3 I²C haberleşme protokolü

I²C protokolü kısa mesafeli seri veri aktarımı için Philips tarafından 1982 yılında tasarlanmış bir haberleşme protokolüdür [67]. Bu haberleşme protokolü telefon, tablet, bilgisayar veya TV gibi gömülü sistem barındıran neredeyse tüm cihazlarda yaygın bir şekilde kullanılmaktadır.

I²C temelde iki hatlı bir mimariye sahiptir. Bu sebeple haberleşme için sadece iki hat kullanılmaktadır. Veri için SDA ve saat sinyali için de SCL hatları kullanılmaktadır. I²C mimarisinde bu iki hat da çift yönlü olarak görev yapmaktadır. Böylelikle master cihaz hem veri gönderip hem de alabilmektedir. Saat sinyali de master cihaz tarafından kontrol edilmektedir.

UART haberleşmeden farklı olarak I²C senkron haberleşme ile iletişimi sağlamaktadır. I²C’ nin mimarisinde birden fazla master cihaz bulunabilmektedir. I²C’ nin half duplex bir yapısı olmasından dolayı SPI’ a göre hızı biraz daha düşüktür. Fakat

SPI' a göre daha az hat gerektirdiğinden dolayı da avantajlı olmaktadır. I2C haberleşme arayüzünün mimari yapısı şekil 3.8' de gösterilmiştir.



Şekil 3.8 : I²C master ve slave cihaz bağlantıları [67].

Şekil 3.8' de gösterilen referans tasarımı, saatli (SCL) bir veri yolu ve 7bit adreslemeli veri (SDA) hatlarından oluşmaktadır. Bu hatta bağlı cihazlar için iki seçenek vardır. Bunlar ya master cihaz olarak ya da slave cihaz olarak görev yapmaktadırlar. I²C' de aşağıdaki çalışma modları kullanılmaktadır.

- **Master mod:** Saat sinyali üreten ve slave cihaz ile iletişimi başlatan düğümdür.
- **Slave mod:** Master cihaz tarafından üretilen sinyali alan ve adreslendiği zaman cevap veren cihazdır.
- **Multi-master mod:** Aynı anda birden fazla master cihaz olması istendiği koşullarda kullanılan moddur.

I²C haberleşmede adres ve veri baytları MSB olarak gönderilmektedir. I²C ek olarak 1 ve 0 başama ve bitiş bitlerinin yanında özel olarak başlama ve bitiş sinyallerine izin vermektedir. Haberleşme esnasından öncelikle master cihaz haberleşmek istediği slave cihazın 7bitlik adresine başla komutunu yollamaktadır. Bunun ardından slave cihaz tarafından haberleşmek istediğine dair 1 ya da 0 değerinde sinyal gönderilmektedir. Bu sinyalin ardından master cihaz veri yollama ya da alma görevlerine başlamaktadır. Slave cihaz da bu görevleri tamamlayıcı nitelikte davranış sergilemektedir.

I²C referans tasarımı, nadiren kullanılan 10bit uzantıya sahip 7bit adres alanına sahiptir [68]. Genel olarak I²C' nin 100 kbit / s standart modda ve 400 kbit / s hızlı modda veri yolu hızı mevcuttur. Ek olarak 10 kbit / s düşük veri yolu hızı da mevcuttur. Ancak bu hız sadece düşük saat sinyali barındıran uygulamalarda kullanılabilir. Son yıllarda

yapılan güncellemelerden sonra I²C' nin veri yolu hızları artırılmış ve daha fazla düğüm yani cihaz eklenebilir hale gelmiştir. Bu yükske hızlarda kullanım daha çok gömülü sistemlerde ve PC' lerde sıklıkla kullanılmaktadır.

3.2.3 Devre şematik çizimleri

Pilot iş yükü ve sağlık takip sistemi için tasarlanan devre şeması üç farklı fonksiyon için veri sağlayacak olan devre kartı nabız sensörü, göz kırpma sensörü ve baş pozisyonu bilgisi için ataletsel ölçüm sensörü barındırmaktadır. İlgili devre şeması geliştirilme aşaması öncelikle seçilen elektronik komponentler ve bu komponentler ile mikrokontrolcü arasındaki haberleşme protokelleri referans alınarak ilgili tasarım süreçleri başlamıştır.

Nabız sensörü için herhangi bir haberleşme protokolü kullanılmamaktadır. Nabız sensörü doğrudan filtrelenmiş ve işlenmiş EKG datasını mikrokontrolcü üzerindeki analog hatta vermektedir. Bu işlem için Teensy 3.5 üzerinde bulunan 27 adet analog pinden herhangi bir tanesine bağlanması yeterlidir. DFRobot nabız sensörü devre kartına doğrudan bağlanılabilecek bir modül olmadığından dolayı yükseltici ayaklar yardımı ile devre kartına takılacaktır. Üzerinde bulunan 5V besleme, sinyal ve GND pinleri de devre kartı üzerine yönlü konnektör ile aktarılacaktır.

Ataletsel ölçüm sensörleri halihazırda modül olarak bulunduğu için devre kartı üzerine soketli olacak şekilde takılması planlanmıştır. Böylelikle ilgili modülün değiştirilmesi gibi işlemler kolaylıkla yapılabilmektedir. Sistemde ataletsel ölçüm modülü olarak kullanılan MPU-9250 I²C haberleşme protokolüne sahip olduğu için devre şeması çizilirken ilgili bağlantı türü referans alınmıştır. Teensy 3.5 üzerinde bulunan üç adet I²C bağlantı pinleri kullanılarak MPU-9250 modülü ile Teensy 3.5 arasındaki bağlantı sağlanmış olur. 3.3V ile çalışan bu modül besleme gerilimini Teensy 3.5 üzerinde bulunan 3.3V çıkış pinlerinden almaktadır.

Göz kırpma algılaması için daha önceki bölümlerde tasarlanmış ve test devresi oluşturulmuş modül doğrudan kullanılmayacaktır. Bu modülün tasarlanıp ürettirilecek olan baskı devre kartına entegre bir şekilde olması planlanmıştır.

Şekil 3.9' da pilot iş yükü ve sağlık takip sistemi için tasarlanmış olan devre şematik çizimi gösterilmektedir.

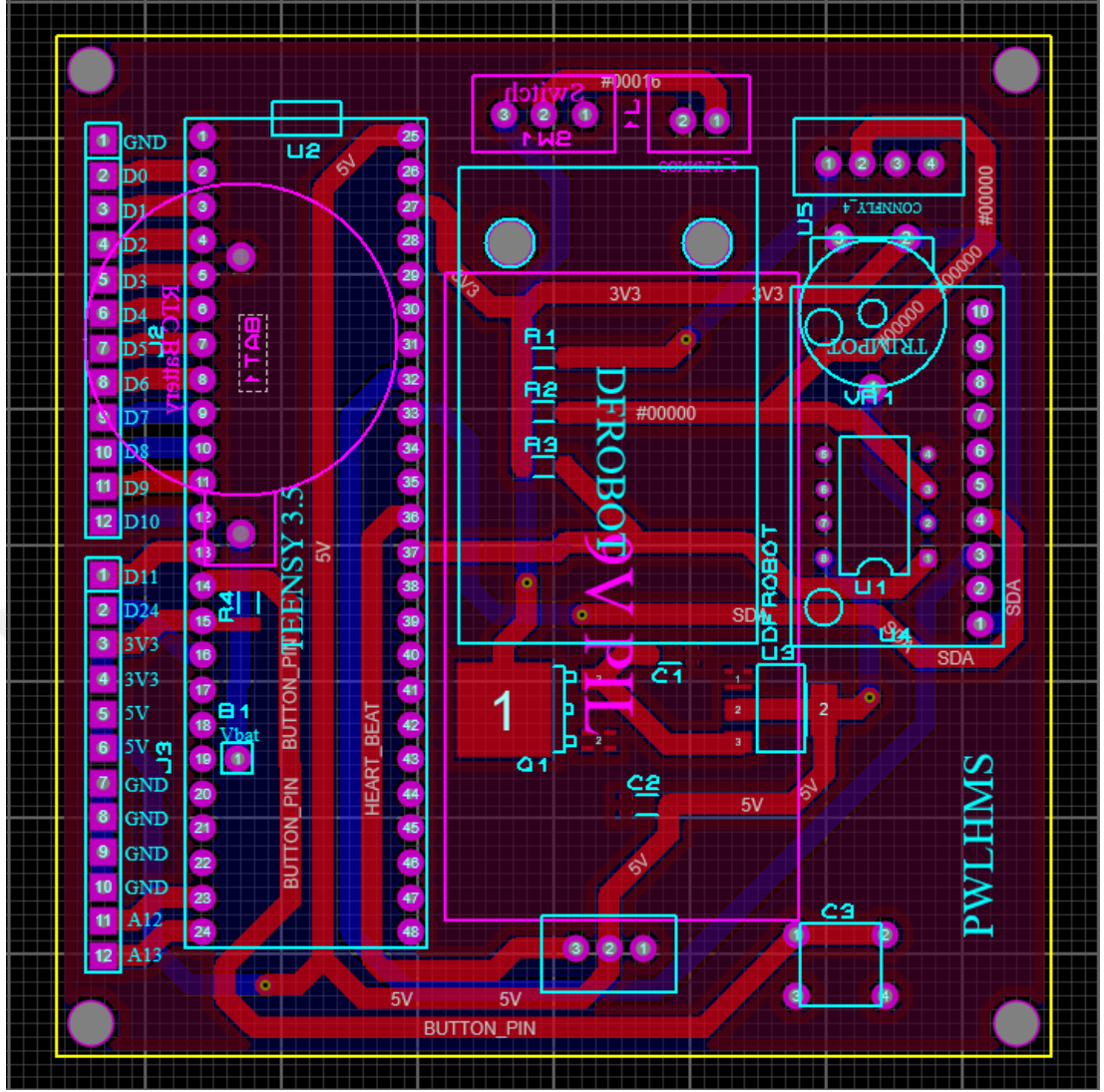
3.3 Devre PCB Tasarımları

Devre şematik çizimlerinin ardından PCB tasarımı için ilgili katmana geçilmiştir. Bu aşamada öncelikle devre kartının boyut bilgilerinin ayarlanması gerekmektedir. Bu ayarlamaların sağlıklı bir şekilde yapılabilmesi için devrede kullanılacak olan elektronik komponentlerin boyut bilgilerine ulaşılması ve devre üzerinde kaplayacağı alana göre bir varsayım yapılması gerekmektedir. Devre kartları ne kadar küçük ve kompakt bir şekilde tasarlanır ve üretilir ise kullanılacağı yerde o kadar az yer kaplamaktadır. Bununla birlikte seri üretimlerde devre kartının maliyeti de düşeceği için bu durum epey avantaj sağlamaktadır.

Pilot iş yükü ve sağlık takip sistemi çalışması kapsamında tasarlanan tüm devre kartının bir adet kask üzerine monte edilmesi planlanmıştır. Bundan dolayı şematik çizimleri tamamlanmış olan devre kartının PCB tasarımı kısmında boyutunun az yer kaplayacak şekilde olmasına özen gösterilmiştir. Bu çalışmada tasarlanan devre kartı daha çok iş geliştirme süreci içerdiği için devre kartının mümkün olan en küçük boyutlarda olması öncelikli hedef olarak gözetilmemiştir. Devre kartı üzerindeki elektronik komponentler, kask boyutları, fonksiyon test süreçleri ve simülasyon ortamında karşılaşılabilecek kısıtlar göz önünde bulundurularak devre kartı boyut hedefi 75x73mm olarak belirlenmiştir.

Devre kartının boyut olarak 75x73mm olması üzerinde bulunan elektronik komponentlerin devre kartının her iki yüzeyinde de bulunabilmesini gerektirmektedir. Böyle bir yapı ancak çift taraflı ve delik içi kaplamalı bir devre kartı ürettirilmesiyle mümkündür. Bu yöntemle üretilen devre kartı üzerindeki delikler sayesinde kartın ön yüzü ile arka yüzü arasında güç ya da veri iletimi için bir hat oluşturulmaktadır. Böylelikle taşınması istenilen besleme gerilimleri ya da sinyaller ön ve arka yüz arasında geçiş yapabilmektedir.

Şekil 3.10' da devre kartı için tasarlanan PCB gösterilmektedir. Bu gösterimde sarı renkli çerçeveye devre kartı sınırlarını, turkuaz renkli elektronik komponentler devre kartının ön yüzünde bulunan komponentleri, pembe renkli elektronik komponentler ise devre kartının arka yüzünde bulunan komponentleri temsil etmektedir.



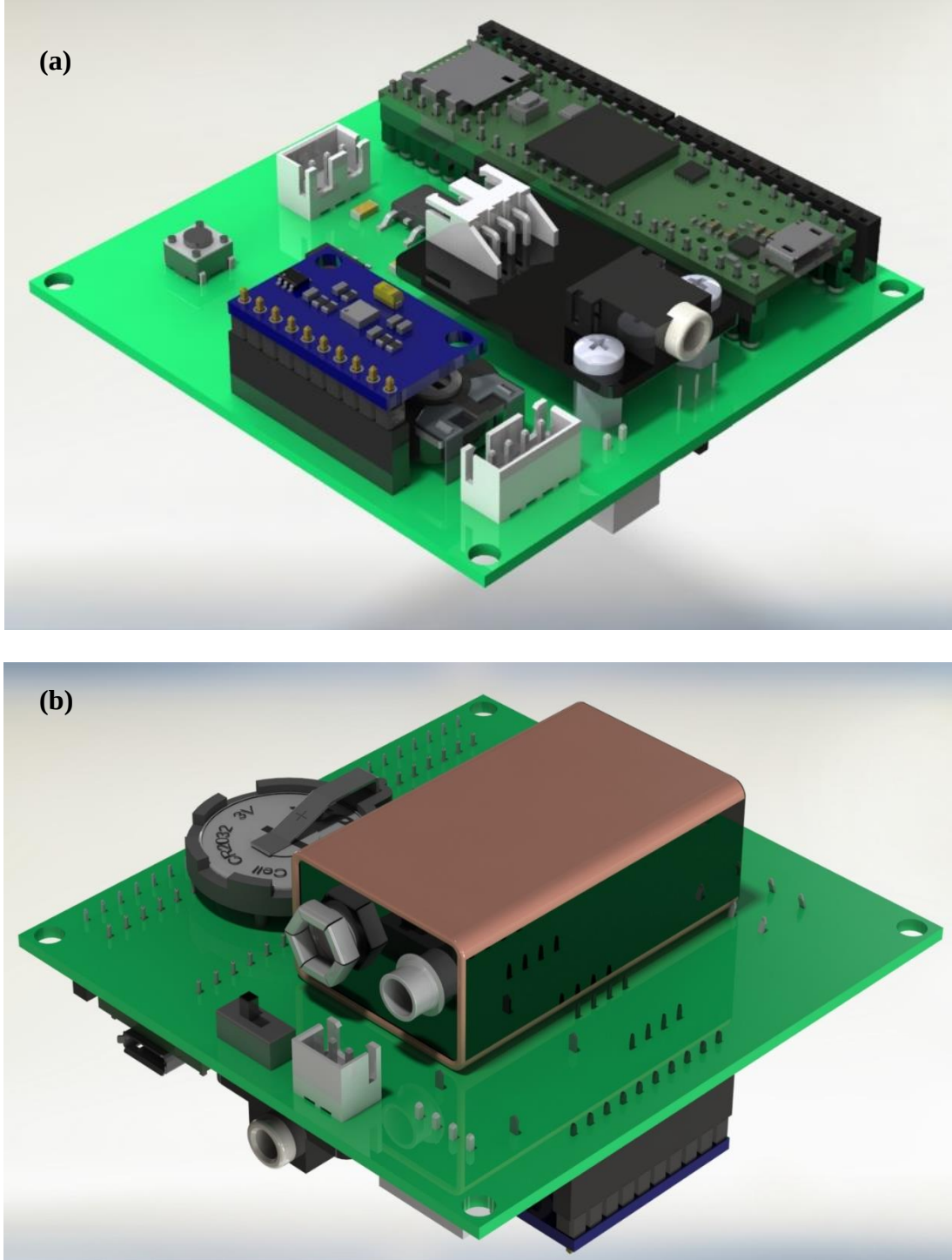
Şekil 3.10 : Devre kartı PCB çizimi.

Devre kartı üzerinde bulunan J2 ve J3 numaralı konnektörlere regülatör çıkışı (5V), mikrokontrolcü üzerinde bulunan besleme gerilimleri (3.3V), ve bazı giriş / çıkışlar aktarılmıştır. Böylelikle daha sonraki çalışmalar için ekstra sensör eklenebilir ve genişletilebilir bir alt yapı oluşturulmuştur.

3.4 Üç Boyutlu Model Dosyalarının Oluşturulması

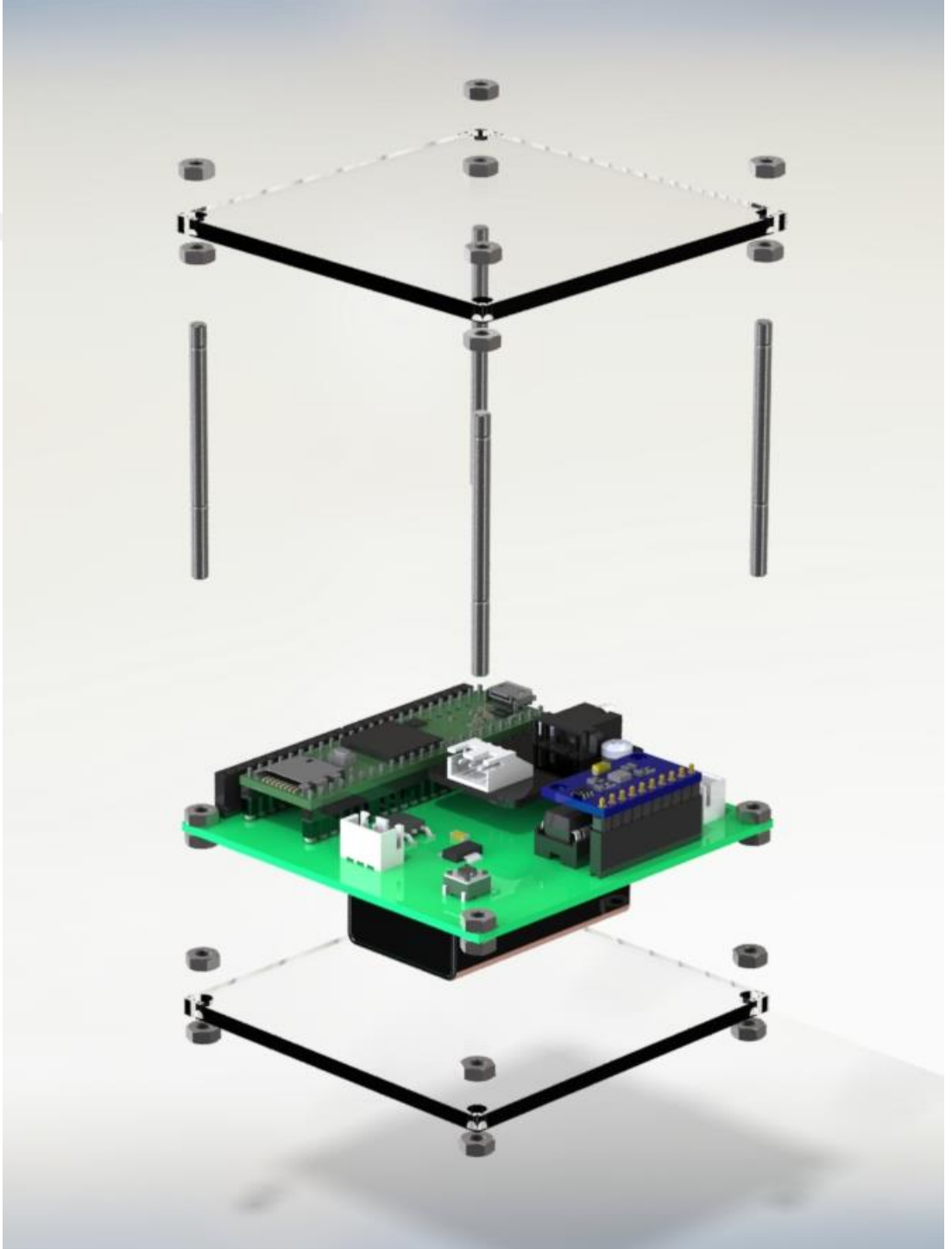
Pilot iş yükü ve sağlık takip sistemi için tasarlanan PCB dosyası üretim için hazırlanmadan önce üç boyutlu model dosyası oluşturulmaktadır. Bu yöntemdeki amaç devre kartının üretilmeden önce kontrol edilmesidir. Böylelikle daha sonraki süreçlerde karşılaşılabilecek sorunların önüne geçilmektedir. Örneğin bu model dosyasının oluşturulması sayesinde devre kartı PCB tasarımının ilk versiyonunda SD kart soketi ile ilgili bir entegrasyon hatası tespit edilmiş ve giderilmiştir. Bu ve bunun

gibi hataların daha önceden tespit edilmesi üretim sonrası entegrasyon süreçlerinde zaman ve maddi kaynak tasarrufu sağlamaktadır. Özellikle seri üretimi planlanan sistemler için yüksek miktarlarda oluşacak olan maddi kayıpların önüne geçilebilmektedir. Şekil 3.11’ de devre kartının ön ve arka yüzünün 3 boyutlu modellenmiş hali gösterilmektedir.



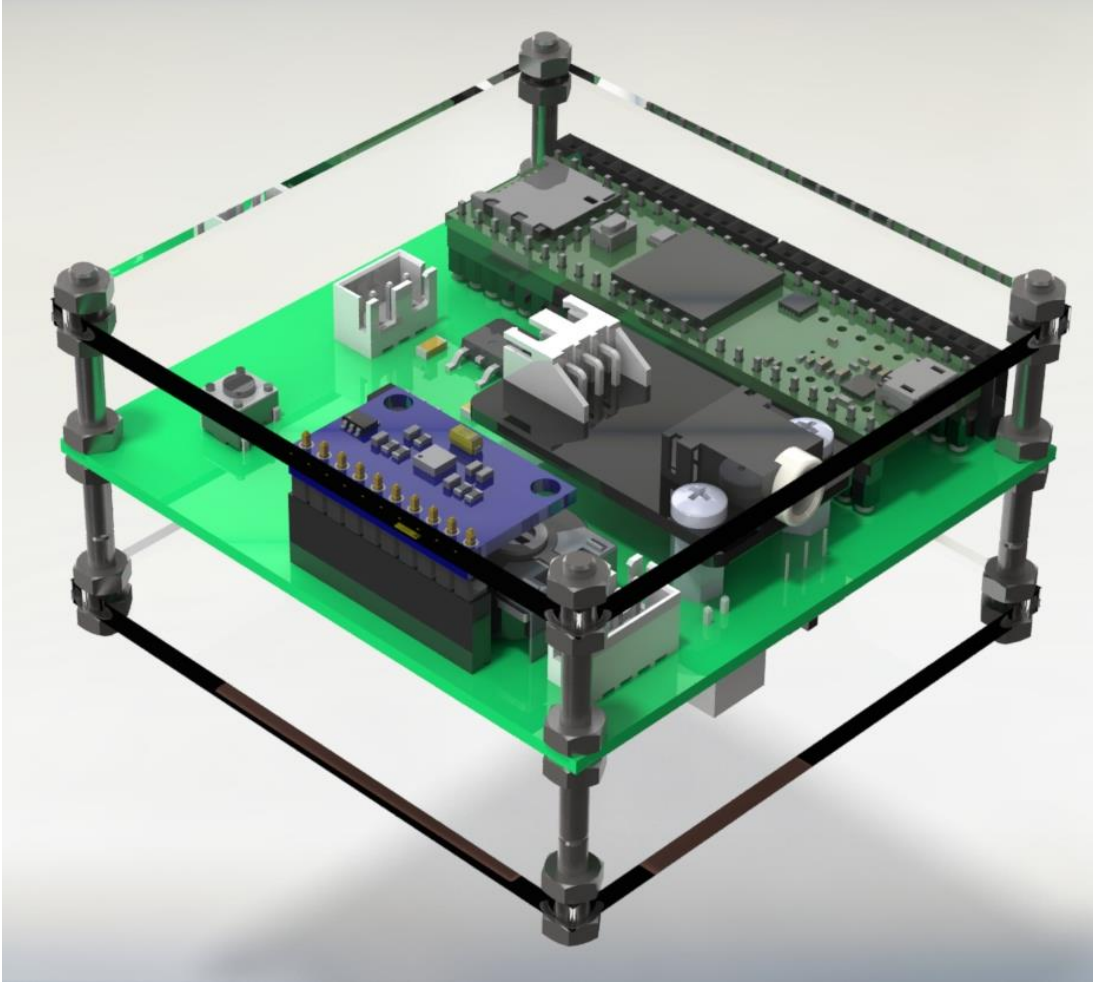
Şekil 3.11 : Devre kartının 3B modellenmiş hali: (a)Ön yüz. (b)Arka yüz.

Devre kartının 3 boyutlu modellemesi tamamlandıktan sonra kask yüzeyine monte edilebilmesini sağlayacak olan ve aynı zamanda devre kartını koruma görevi üstlenecek olan pleksi malzemeden üretilmiş parçalar lazer kesim yöntemi ile üretilmiştir. Bu levhalar devre kartının köşelerinde bulunan delikler yardımıyla hizalanmakta ve o bölgelerden geçen saplama vidalar ile monte edilmektedir. Şekil 3.12’ de bu montajın patlatılmış hali gösterilmiştir.



Şekil 3.12 : Devre kartının patlatılmış gösterimi.

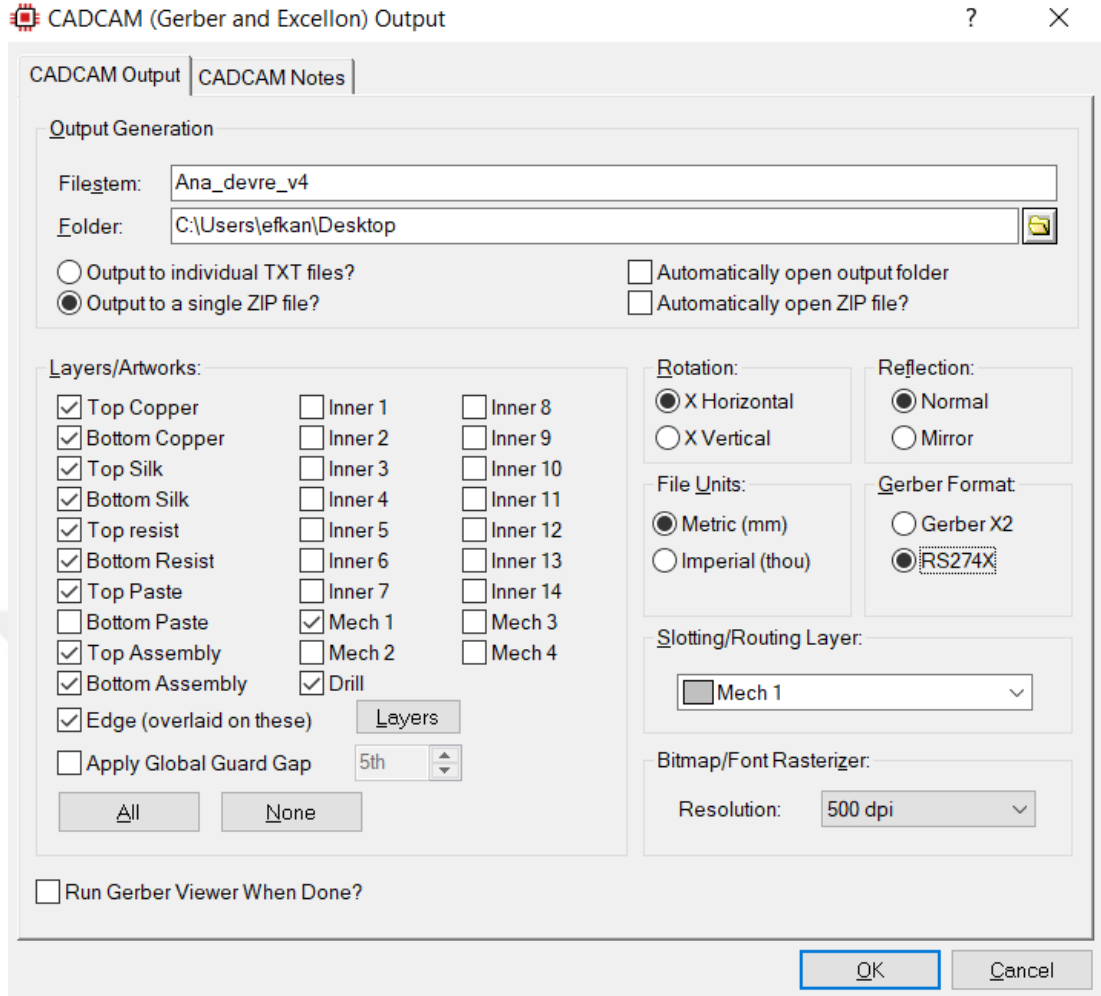
3B modelleme alışmasının sonucu olarak devre kartının tm entegrasyon srelerinin bitmesiyle Őekil 3.13’ te gsterilen hali alması ngrlmektedir.



Őekil 3.13 : Devre kartının kutulanmıŐ hali.

3.5 Baskı Devre retimi İin Gerber Dosyalarının Hazırlanması

Tasarlanan devre kartı retim iin hazırlanırken ncelikle gerber dosya trnde ıktılarının alınması gerekmektedir. Gerber dosya formatı olarak ise RS274X seilmelidir. Bu blmde alınan ıktının aynalanmamıŐ olmasına dikkat edilmelidir. Aksi halde hatalı retimden kaynaklı component lehimleyememe gibi problemlerle karŐılaŐılabilmektedir. Bu durum hem tasarım srecinin baŐtan kontrol edilmesine hem de maddi olarak olumsuzluklara yol amaktadır. Őekil 3.14’ de gerber dosyası ıktısı alınan tasarım programı arayz gsterilmiŐtir. Alınan bu ıktı retim iin anlaŐılan firmaya gnderilmiŐtir.

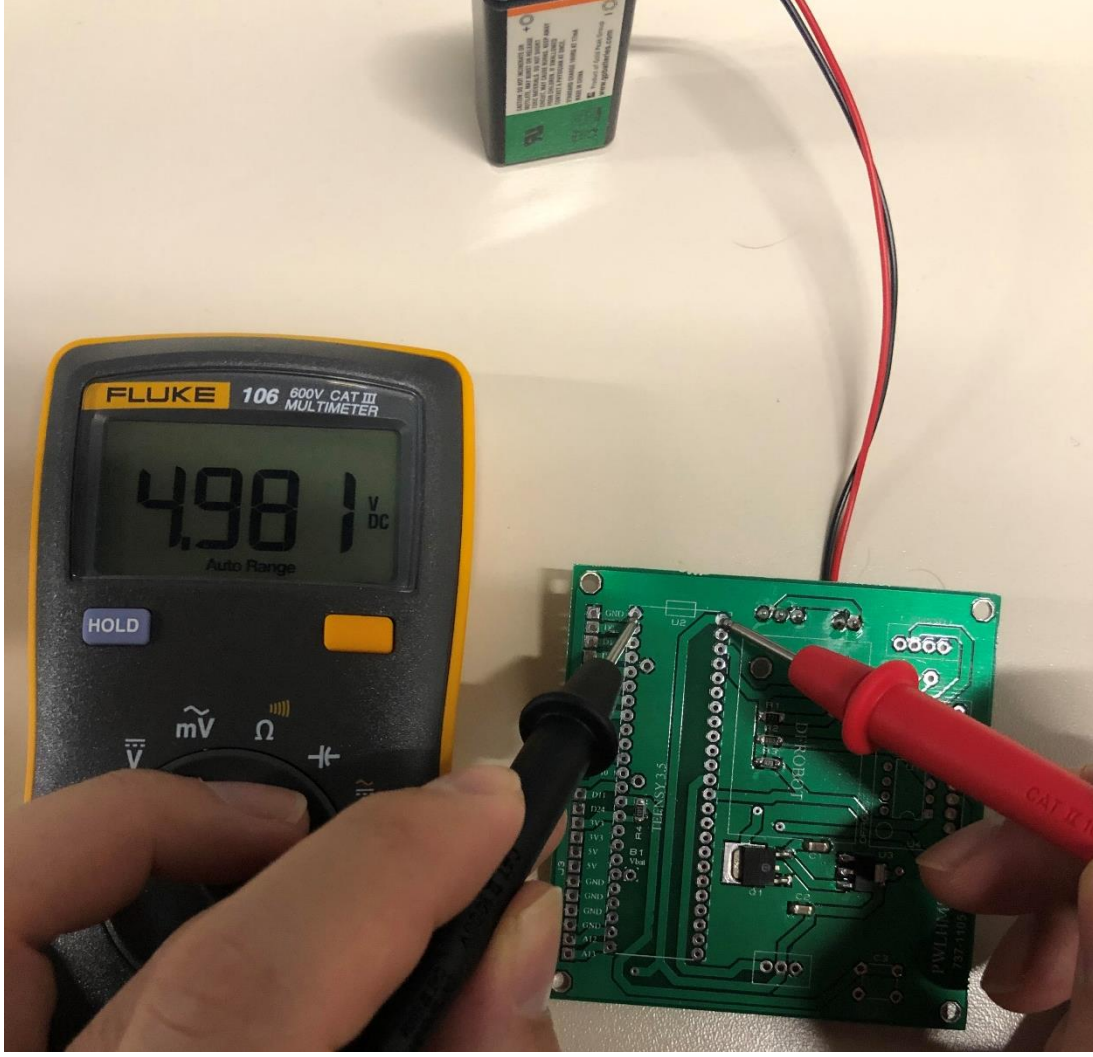


Şekil 3.14 : Gerber çıktısı arayüzü.

3.6 Baskı Devre Kartı Fonksiyon Testleri

Devre kartlarının üretim süreci firmaların iş yoğunluğuna göre değişmektedir. Normal zamanlarda 3-8 iş günü içerisinde üretilebilen devre kartları COVID-19 salgını sebebiyle toplamda 15 iş gününde üretilmiştir. Üretilen kart üzerinde herhangi bir kısa devre ya da hatalı üretimden kaynaklı sorun olup olmadığı gözle ve ölçüm aletleri ile kontrol edilmiştir.

Bu fiziksel kontrollerin ardından lehimleme aşamasına geçilmiştir. Bu aşamaada öncelikle voltaj regülatörleri lehimlenerek güç katmanı oluşturulmuştur. Oluşturulan güç katmanından diğer komponentlere giden voltaj ölçülerek ilgili komponentlerde doğru voltajın okunup okunmadığı kontrol edilmiştir. Şekil 3.15’ te örnek bir ölçüm gösterilmektedir.



Şekil 3.15 : Besleme gerilimi ölçümü.

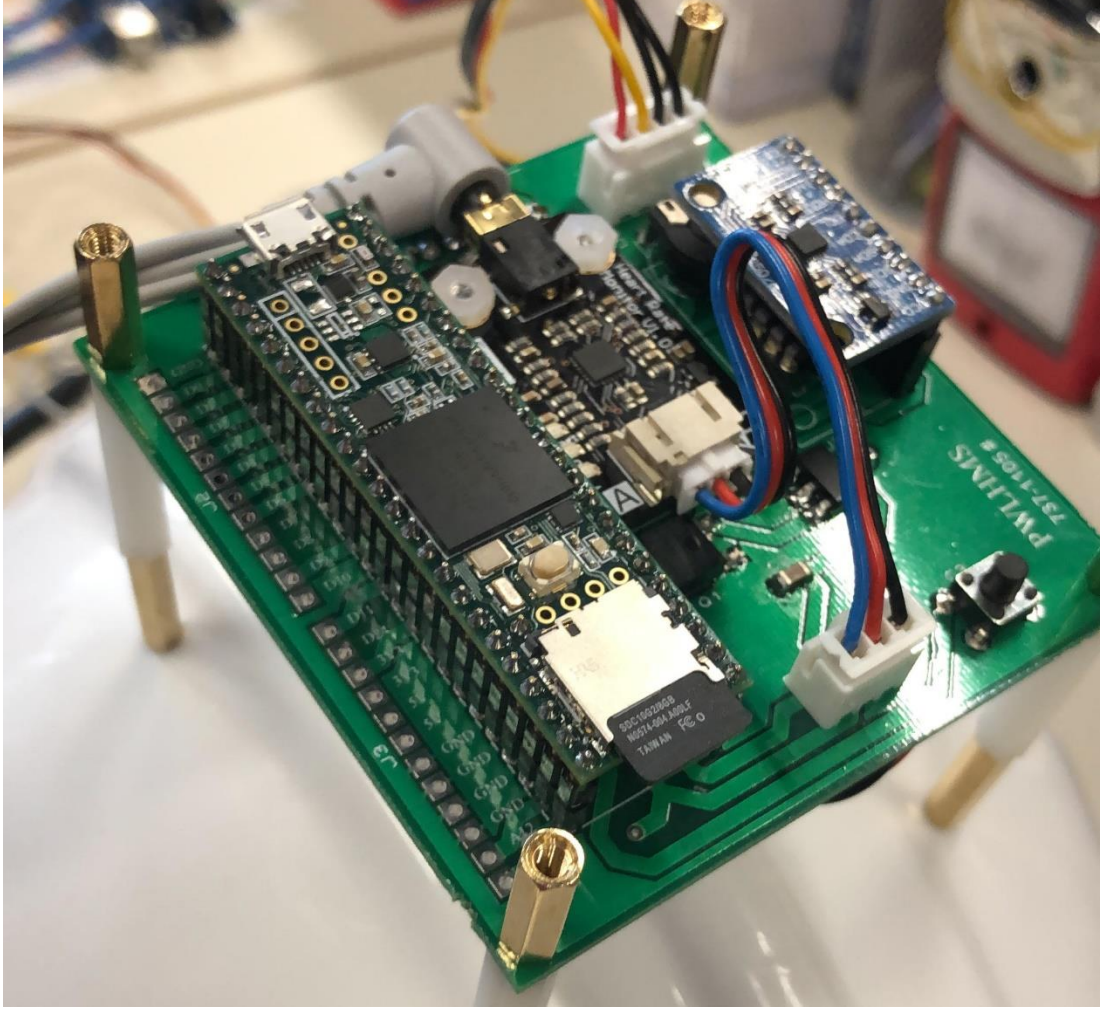
Teensy 3.5 geliştirme kartına gelen besleme gerilimi şekildeki gibi 4.981 V olarak ölçülmüştür. Bu ölçüm değeri de, Teensy 3.5 besleme gerilimi olarak uygundur.

Tüm komponentlerin lehimleme sürecinin ardından fonksiyon testleri yapılmıştır. Bu fonksiyon testi içerik olarak takılan tüm komponentlerin planlandığı gibi çalışıp çalışmadığını anlamak amacıyla yapılmaktadır.

3.7 Baskı Devre Kartı Entegrasyon Süreci

Baskı devrenin lehimleme aşaması ve fonksiyon testleri tamamlandıktan sonra kask üzerine entegrasyon çalışması yapılmıştır. Bu çalışma kapsamında pilot kaskı yerine işçi bareti kullanılmıştır. Bu baret simulasyon uçuşları esnasında pilot kaskı olarak kullanılmaktadır.

Baskı devre kartının baret üzerine montajı kart üzerinde bulunan deliklerden standoff geçirilerek yapılmıştır. Şekil 3.16’ da devre kartının komponentlerinin lehimlenmiş ve komponent entegrasyonu yapılmış hali gösterilmektedir.



Şekil 3.16 : Devre kartı entegrasyonu.

Baret üzerinden sadece nabız ve göz kırpma sensörü ile ilgili sinyal kabloları çıkmaktadır. Bu kabloların simülasyon pilotunun hareket alanını kısıtlamayacak şekilde olmasına özen gösterilmiştir.

Son olarak şekil 3.17’ de devre kartının baret üzerine montajlanmış hali gösterilmektedir. Bu görselde gözlük üzerine giden göz kırpma sensörü ve nabız ölçümü için ilgili kablolar bulunmaktadır.



Şekil 3.17 : Devre kartı-baret entegrasyonu.

4. UÇUŞ SİMULASYONU

Genellikle eğitim amaçlı kullanılan simülatörler, uçak veya diğer karmaşık sistemlerin kontrollerinin ve kullanımının gerçekçi bir şekilde taklit edilmesini sağlamak için tasarlanmış yazılımlardır. Uçuş simülatörü, pilot için uçan makinenin ortamını simüle edebilen sanal gerçeklik sistemidir. Uçuş simülatörleri, büyük ölçüde pilot eğitimi için kullanılmaktadır. Ayrıca uçak karakteristiklerini araştırmak, kullanım özelliklerini kontrol etmek, uçak tasarım ve geliştirmesini yapmak için de kullanılabilir [69].

Pilot iş yükü ve sağlık takip sistemi projesinin simulasyon uçuş fazları toplamda 3 aşamadan oluşmaktadır. Bu aşamalar sırasıyla kalkış, düz uçuş ve iniştir. Uçuş İzmir Adnan Menderes Havalimandan Sabiha Gökçen Havalimanına doğru yapılmaktadır. Bu uçuşlardan 10 tanesi farklı günlerde ve birbirine yakın öğle saatlerinde, 11. ve 12. uçuşlar ise yapılan ilk 10 uçuşa göre çok geç bir saatte ve yorgun bir durumda arka arkaya yapılmıştır. Tüm bu uçuş fazları ve senaryoları bir simulasyon programı olan FlightGear programında test edilmektedir.

Uçak modeli olarak ise F-16 Block 52 kullanılmaktadır. Bu model FlightGear programı için oluşturulmuş uçak modellerinin olduğu bir siteden ücretsiz olarak temin edilebilmektedir.

Pilot iş yükü ve sağlık takip sistemi projesinin simulasyon uçuşları aşamasında kullanılmak üzere bir adet uçuş ekranı, bir adet anlık veri takibinin yapıldığı monitor, ses sistemi ve uçuş kontrol kumandası temin edilmiştir.

Bu uçuş simulasyon testleri, değişen uçuş fazları arasında pilotta meydana gelen fiziksel ya da bilişsel bazı değişiklikleri algılamak ve bu değişiklikler arasında bağlantı kurmayı amaçlamaktadır.

4.1 FlightGear Flight Simulator

FlightGear Flight Simulator 1997'den beri FlightGear tarafından geliştirilen ücretsiz, açık kaynaklı çok platformlu bir uçuş simülatörüdür.

David Murr ve ekibi bu projeye 1996 yılında başlamış, ilk kez 1997’ de piyasaya sürmüş ve geliştirmeye devam etmiştir. Microsoft Windows, macOS, Linux, IRIX ve Solaris dahil olmak üzere çeşitli işletim sistemleri için özel yapılara sahiptir.

FlightGear kaynak kodu, GNU Genel Kamu Lisansı koşulları altında yayınlanmakta ve ücretsiz ve açık kaynaklı bir yazılımdır.

FlightGear kütle dengesi, yer reaksiyonları, tahrik, aerodinamik, kaldırma kuvvetleri, dış kuvvetler, atmosferik kuvvetler ve yerçekimi kuvvetleri tarafından desteklenen mevcut varsayılan uçuş modeli olan JSBSim tarafından uçuş özelliklerini belirlemek için kullanılabilir. FlightGear ayrıca LaRCsim, UIUC, YASim, MATLAB gibi harici bir kaynağın kullanımını ve sıcak hava balonları ve uzay araçları için özel uçuş modellerini desteklemektedir.

Çeşitli ağ seçenekleri FlightGear’ın diğer FlightGear örnekleriyle iletişim kurmasına izin vermektedir. FlightGear’ı çok uçaklı bir ortamda yerel bir ağda kullanmak için çok oyunculu bir protokol mevcuttur. Bu, formasyon uçuşu veya hava trafik kontrolü simülasyonu için de kullanılabilir. Orijinal çok oyunculu protokolü kullanıma sunulduktan kısa bir süre sonra, internet üzerinden oynamaya izin verecek şekilde genişletilmiştir. Aynı uçak modellerine sahip olmaları ve oyundaki çevrimiçi çok oyunculu harita ile uçuş yollarını görüntülemeleri sağlandığında, diğer oyuncuları simülörde görmek mümkündür.

Ayrıca birden çok monitör ortamına izin vermek için birkaç FlightGear örneği senkronize edilebilmektedir.

FlightGear gerçek zamanlı olarak canlı hava modelleri üretmek için metar verilerini kullanmaktadır. Ayrıntılı hava durumu ayarları, 3d bulutlara, çeşitli bulut türlerine ve yağışa izin vermektedir. Yüksek yol noktası ayarları, rüzgarın yüksek irtifa davranışlarının canlı hava durumu bilgisinden modellenmesine izin vermekte ve termaller de modellenmektedir.

Sadece geleneksel anlamda bir oyun olarak geliştirilmemesine veya tipik olarak analiz edilmemesine rağmen, FlightGear yine de bir dizi çevrimiçi ve çevrimdışı yayında gözden geçirilmiş ve bir uçuş simülatörü oyunu olarak olumlu eleştiriler almıştır [70]. Şekil 4.1’ de FlightGear programından örnek bir ekran görüntüsü gösterilmiştir.

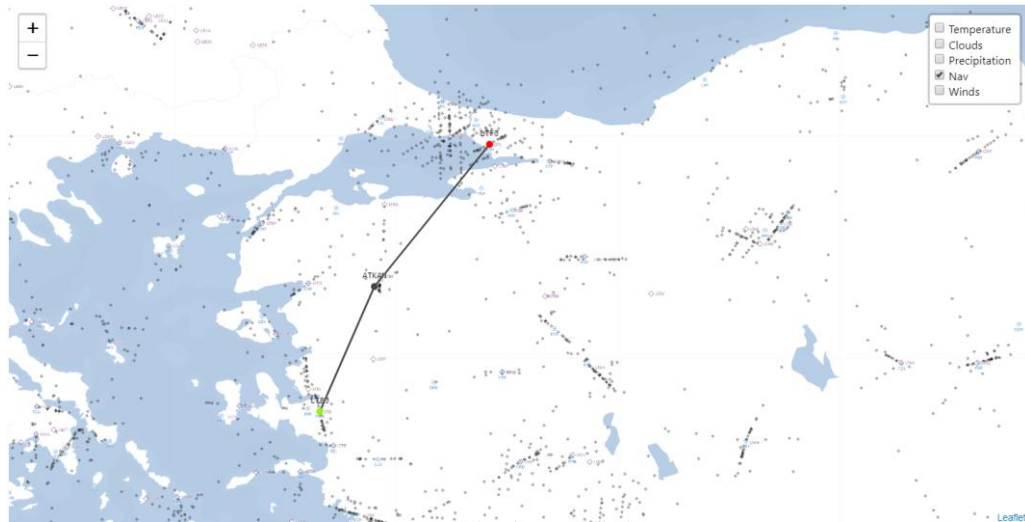


Şekil 4.1 : FlightGear örnek ekran görüntüsü [70].

4.2 Uçuş Senaryoları

Pilot iş yükü ve sağlık takip sistemi projesi kapsamında uçuş senaryosu 2 bölümden ve her bölümde 3 aşamalı uçuş fazından oluşmaktadır. Bu senaryolar daha önceki bölümlerde de belirtildiği üzere yorgun olunmayan 10 uçuş ve yorgun olunan 2 uçuştan oluşmaktadır.

Uçuş İzmir Adnan Menderes Havalimanından kalkış ile başlamaktadır. Bu kalkışın ardından 10000 km irtifaya ulaşıldıktan sonra düz uçuş fazına geçilmektedir. Düz uçuş fazının ardından Sabiha Gökçen havalimanına yaklaşılarak iniş yapılacaktır. Bu uçuş planı için oluşturulan rota şekil 4.2’ de gösterilmiştir.



Şekil 4.2 : Uçuş rotası.

Flightplandatabase.com üzerinden oluşturulan bu uçuş rotası .fgfp uzantılı bir dosya olarak indirilerek FlightGear programına import edilmektedir. Bu dosya içerisinde belirli yol noktalarına ait koordinatlar ve bu noktaların yükseklikleri verilmektedir. Adnan Menderes havalimanından Sabiha Gökçen Havalimanına doğru oluşturulan bu rotada toplamda 3 adet yol noktası bulunmaktadır. Bu noktalar ile ilgili detay bilgiler çizelge 4.1’ de gösterilmektedir.

Çizelge 4.1 : Uçuş rotası noktaları.

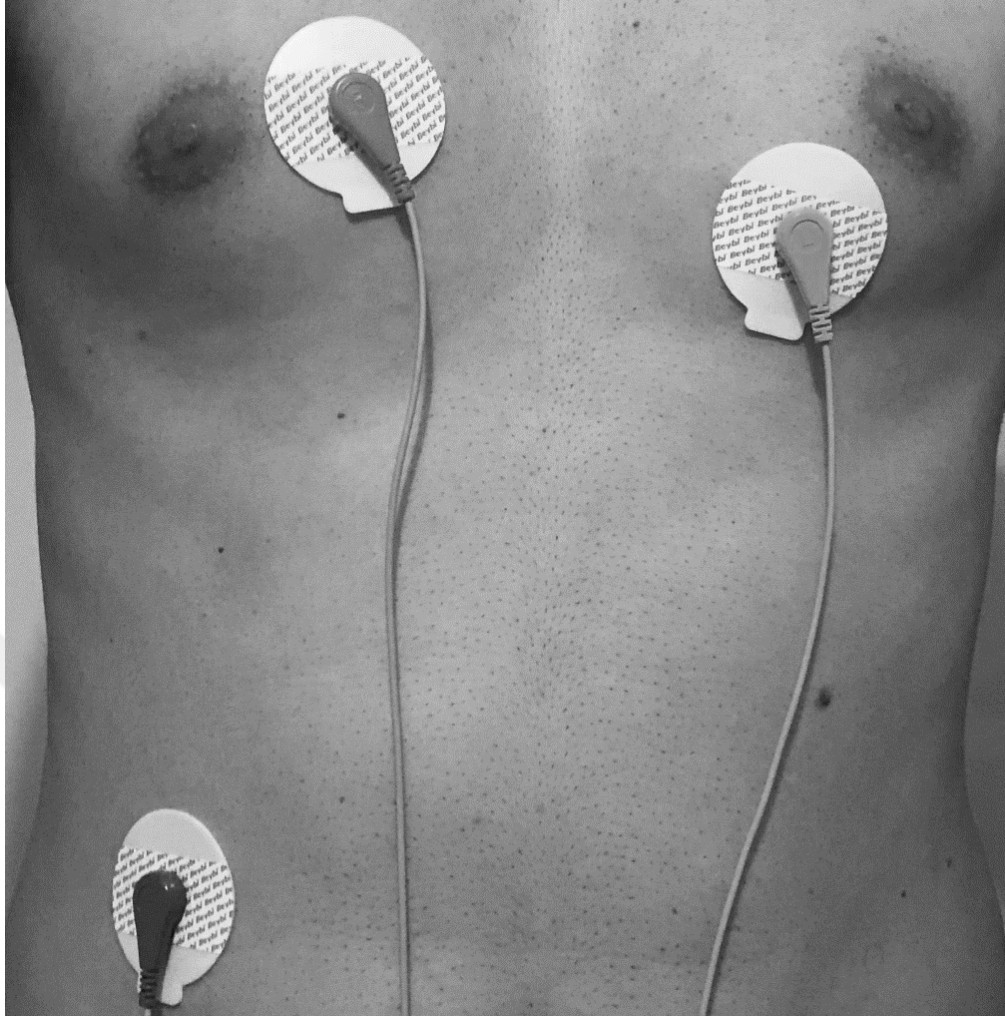
ID	Tip	Yükseklik (m)	Pozisyon	İsim
LTBJ	APT	0	38.29220 / 27.15590	Izmir Adnan Menderes Intl
ATKAN	FIX	7,285	39.52000 / 27.84940	-
LTFJ	APT	0	40.89860 / 29.30920	Sabiha Gokcen International Airport

Pilot iş yükü ve sağlık takip sistemi projesi kapsamında farklı zamanlarda ve son iki uçuş farklı şartlarda olmak üzere toplamda 12 simülör uçuş verisi daha sonra incelenmek üzere kaydedilmiştir.

4.3 Pilot Sağlık Takip Sistemi – Pilot Entegrasyonu

Simulator uçuşu aşamasına geçmeden önceki en önemli adım pilot iş yükü ve sağlık takip sistemi için tasarlanan sistemin pilota düzgün bir şekilde entegre edilmesidir. Bu entegrasyon sürecinde dikkat edilmesi gereken önemli noktalardan bir tanesi pilotun hareket alanını daraltmayacak ve hareket etmesini zorlamayacak şekilde sistemin entegre edilmesidir.

Öncelikle nabız ölçümü için gerekli elektrotlar simülör pilotu üzerine doğru bir şekilde yerleştirilmiştir. Bu elektrotların nasıl yerleştirileceği daha önceki bölümlerde incelenmiştir. Şekil 4.3’ de bu elektrot entegrasyonu gösterilmiştir.



Şekil 4.3 : Elektrot entegrasyonu.

Elektrot yerleşimi yapıldıktan sonra göz kırpma algılama sensörü de düzgün çalışabileceği bir mesafeye yerleştirilmektedir. Göz kırpma algılama sensörü çok hassas olduğu için üzerinde bulunduğu gözlüğün çok fazla hareket ettirilmemesi gerekmektedir. Çok fazla hareket olduğu takdirde sensör yanlış ölçümler oluşturmaktadır. Bu da çalışmanın doğruluğunu yitirtici sonuçlara sebebiyet vermektedir. Şekil 4.4’ de Göz kırpma algılama sensörünün simulatör pilotuna entegrasyonu gösterilmektedir.



Şekil 4.4 : Göz kırpma sensörü entegrasyonu.

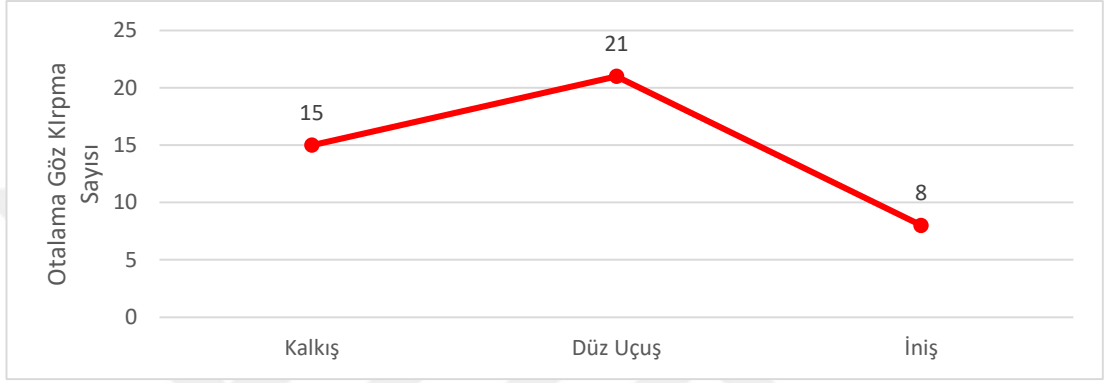
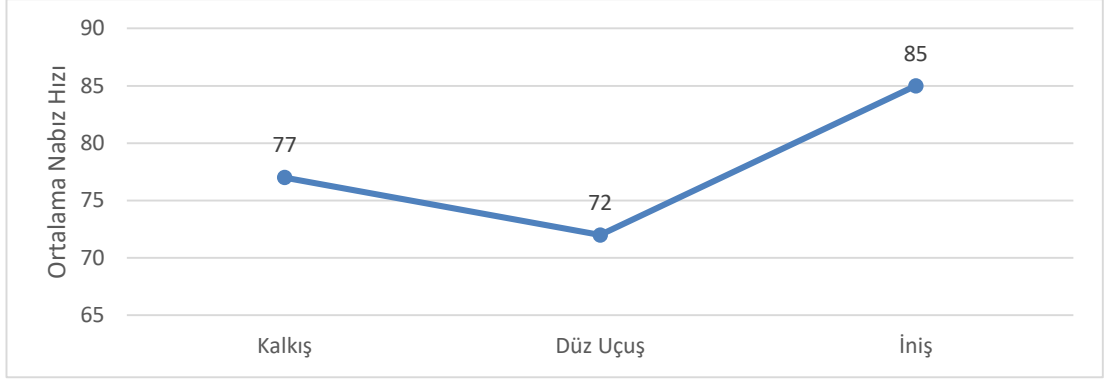
Göz kırpma sensörünün entegrasyonun ardından baret simlatör pilotuna takılarak uçuşa hazır hale getirilmektedir. Uçuş başlatılmadan önce simlatör pilotu üzerinde sistem çalıştırılarak son kontrol yapılmaktadır. Bu kontrollerde herhangi bir eksiklik ya da yanlışlık olup olmadığı göz ile kontrol edilmektedir. Pilotun hemen yanında bulunan sistem operatörü sistem üzerinden pilot verilerini gerçek zamanlı olarak görüntülemektedir. Bu aşamada pilotun göz kırpması sistem üzerinden test edilerek algılayıcının doğru çalışıp çalışmadığı tespit edilmektedir. Aynı zamanda baş hareketlerine göre kullanıcı arayüzündeki hareketlerin tutarlılığı da doğrulanmaktadır. Şekil 4.5’ de sistemin simlatör pilotuna tamamen entegrasyonu yapılmış ve pilotun uçuş esnasındaki görüntüsü gösterilmektedir.



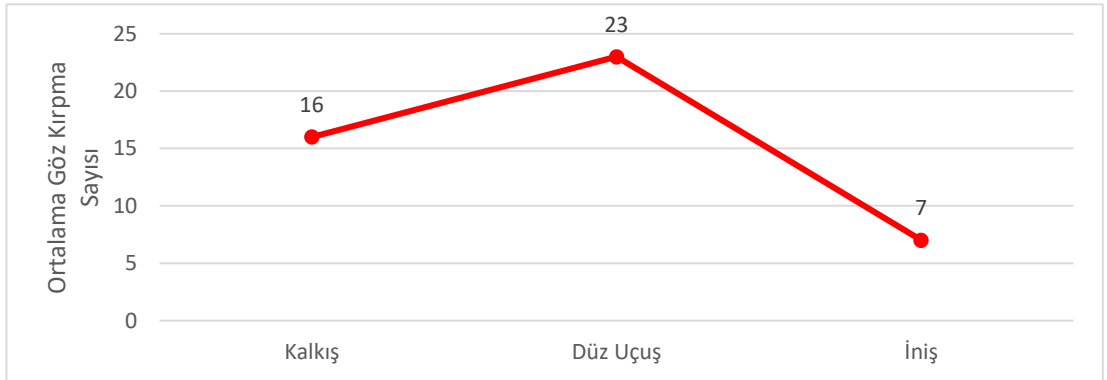
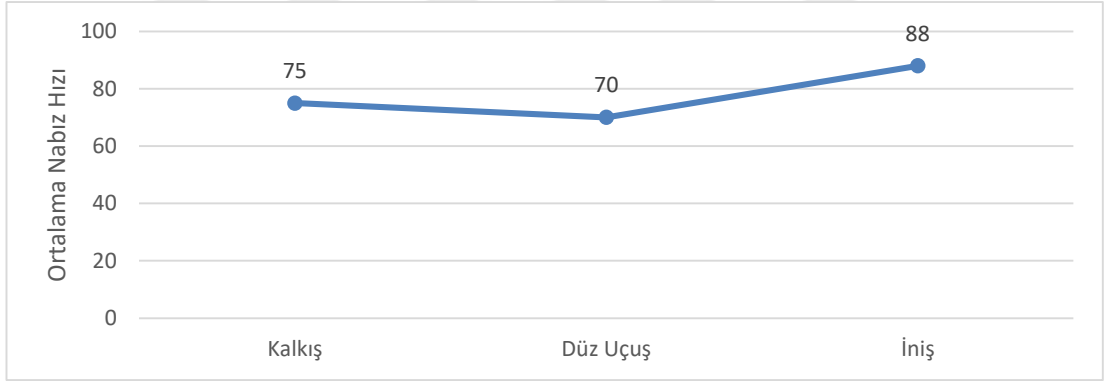
Şekil 4.5 : Simlatör uçuşu.

4.4 Simlatör Uçuşu Verilerinin İncelenmesi

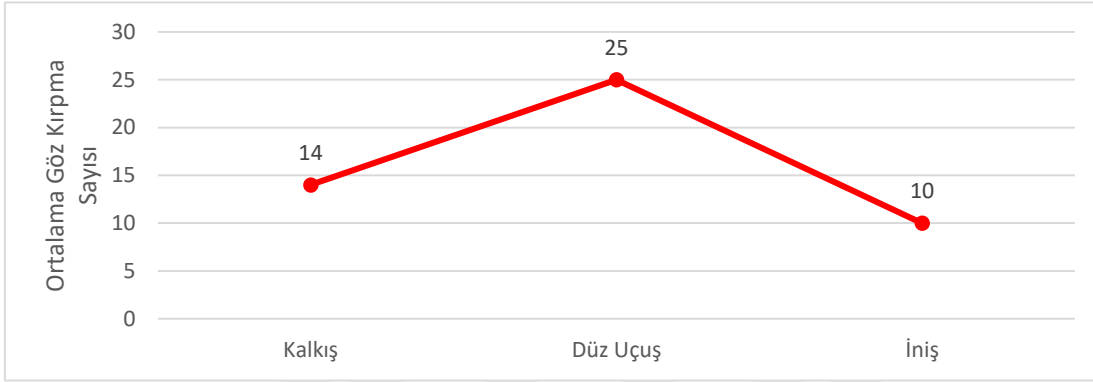
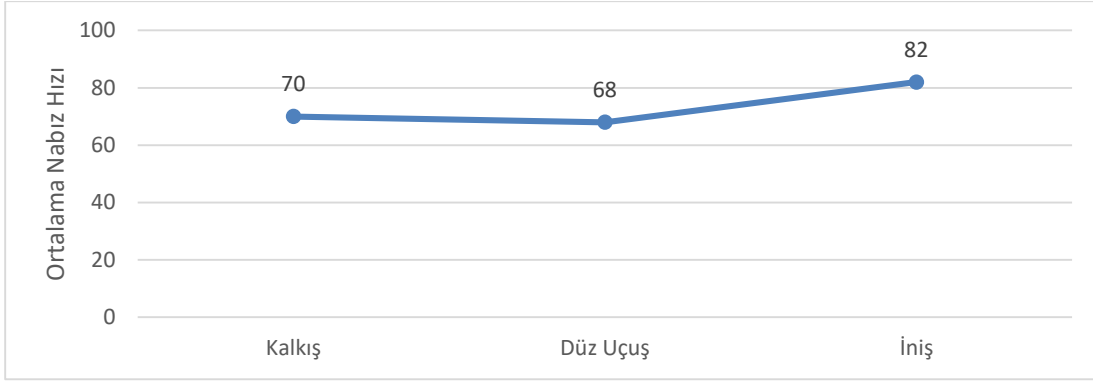
Pilot iş yükü ve sağlık takip sistemi projesi kapsamında yapılan simlatör uçuşları donanım üzerinde bulunan mikro SD karta kaydedilmiştir. Bu kayıt işlemi sonrasında toplanan veriler nabız hızı ve göz kırpma sayısı olmak üzere 2 farklı grafikte incelenmiştir. Şekil 4.6, 4.7, 4.8, 4.9, 4.10, 4.11, 4.12, 4.13, 4.14, 4.15, 4.16 ve 4.17’ de gösterilen grafiklerde her uçuş için kalkış, düz uçuş ve iniş esnasında bir dakikada gerçekleşen nabız atma ve göz kırpma sayılarının uçuş fazına göre değişen ağırlıklı ortalamaları verilmiştir.



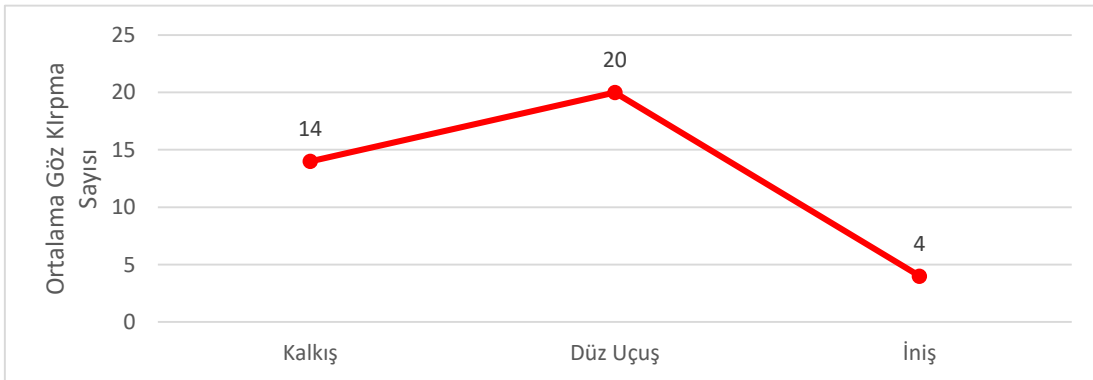
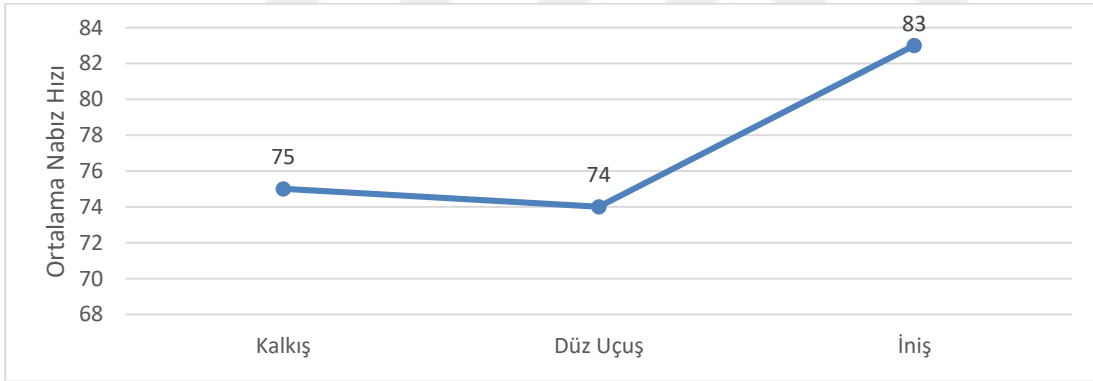
Şekil 4.6 : 1 nolu uçuş için nabız hızı ve göz kırpma sayısı ortalamaları.



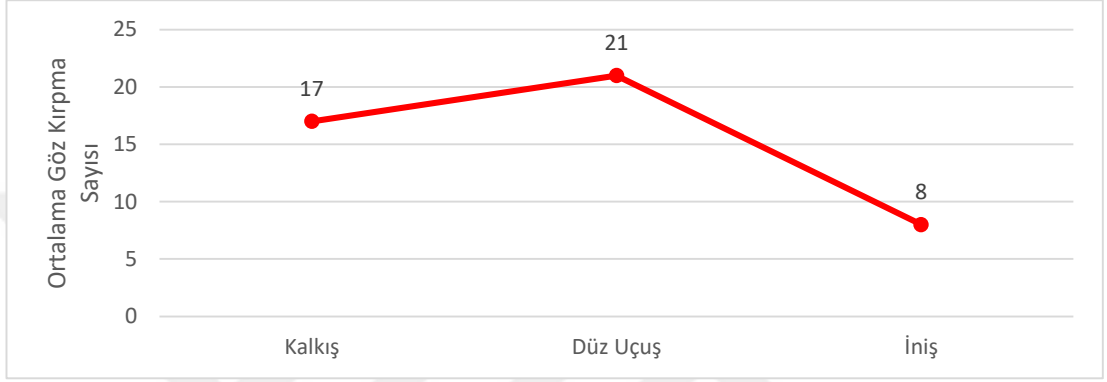
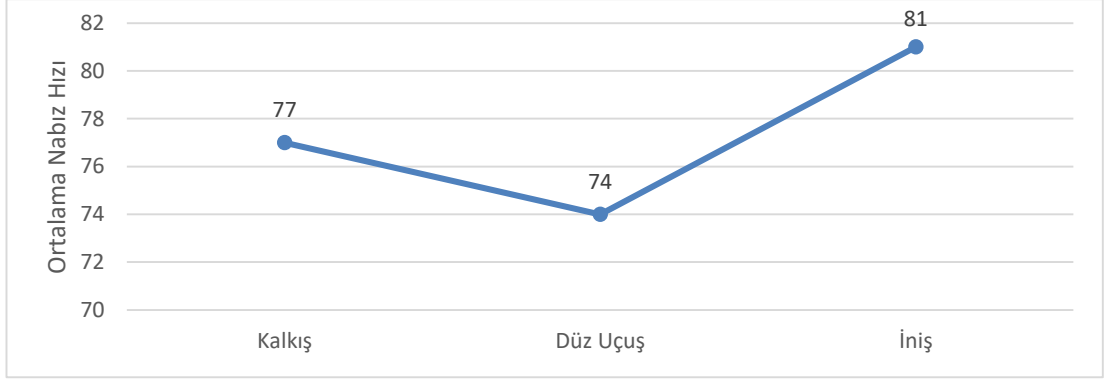
Şekil 4.7 : 2 nolu uçuş için nabız hızı ve göz kırpma sayısı ortalamaları.



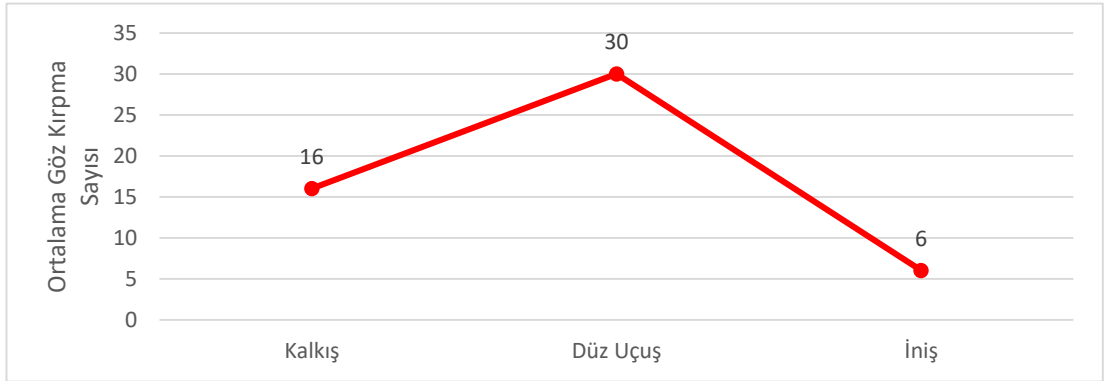
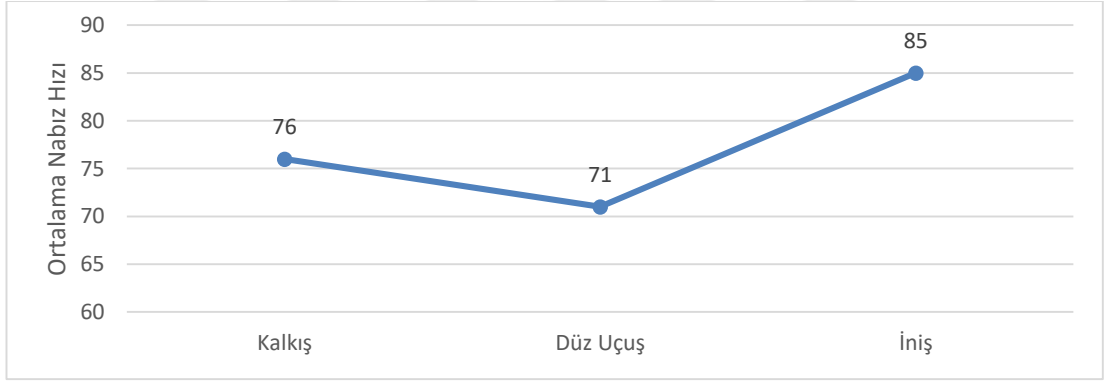
Şekil 4.8 : 3 nolu uçuş için nabız hızı ve göz kırpma sayısı ortalamaları.



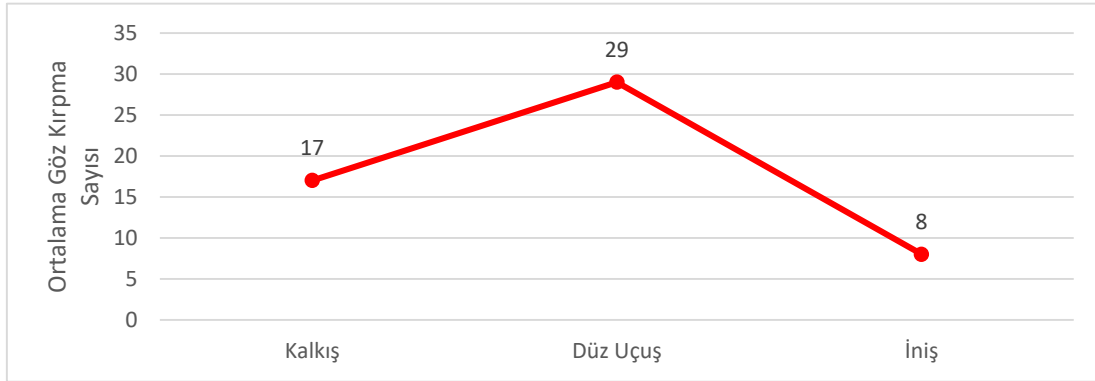
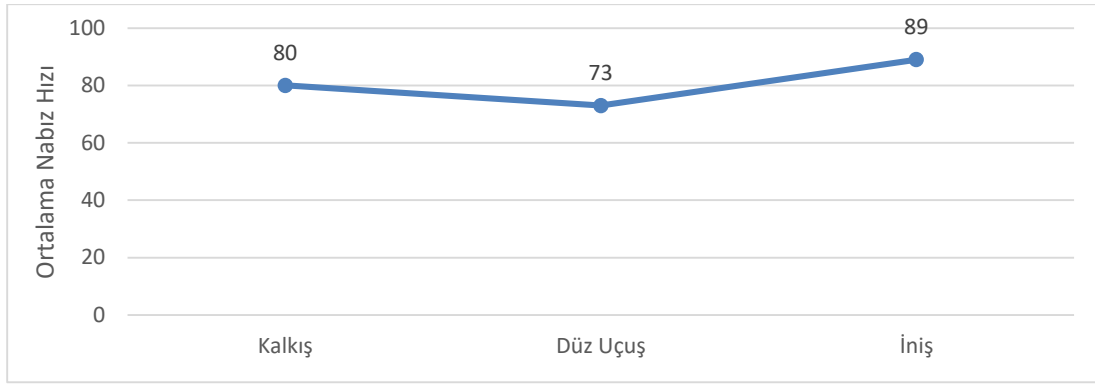
Şekil 4.9 : 4 nolu uçuş için nabız hızı ve göz kırpma sayısı ortalamaları.



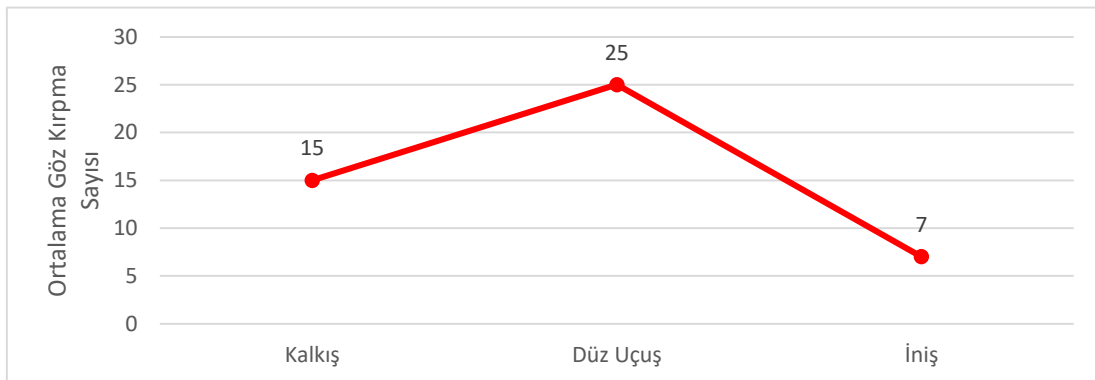
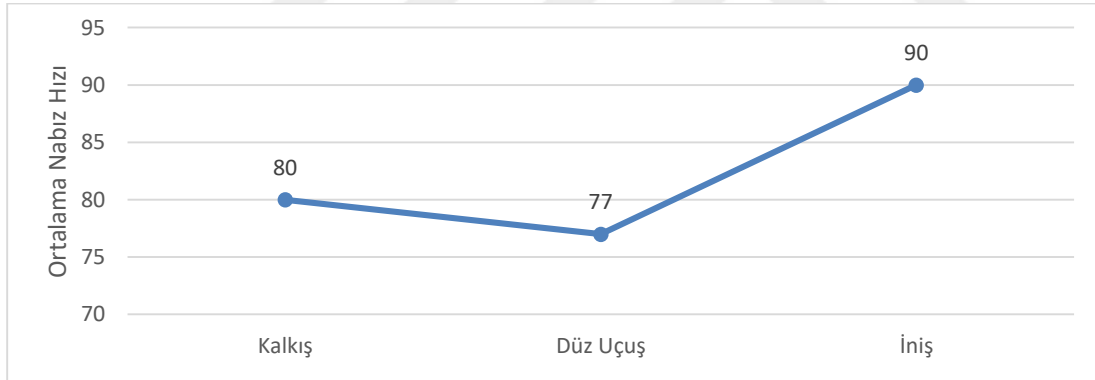
Şekil 4.10 : 5 nolu uçuş için nabız hızı ve göz kırpma sayısı ortalamaları.



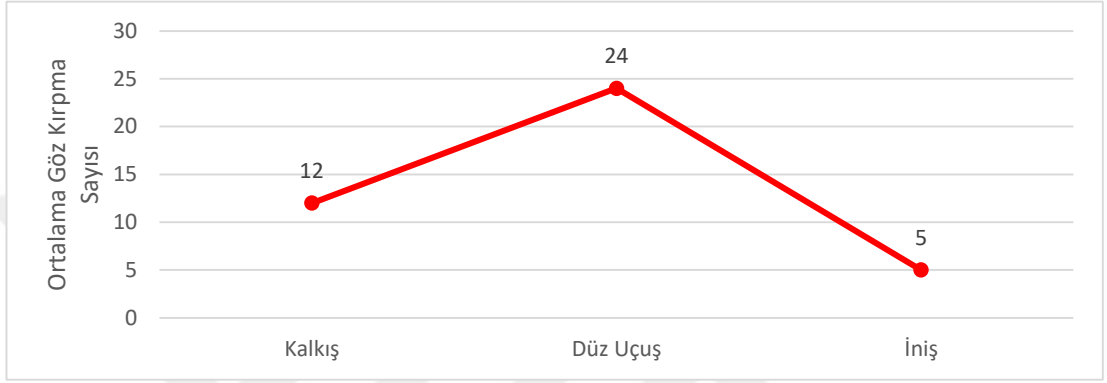
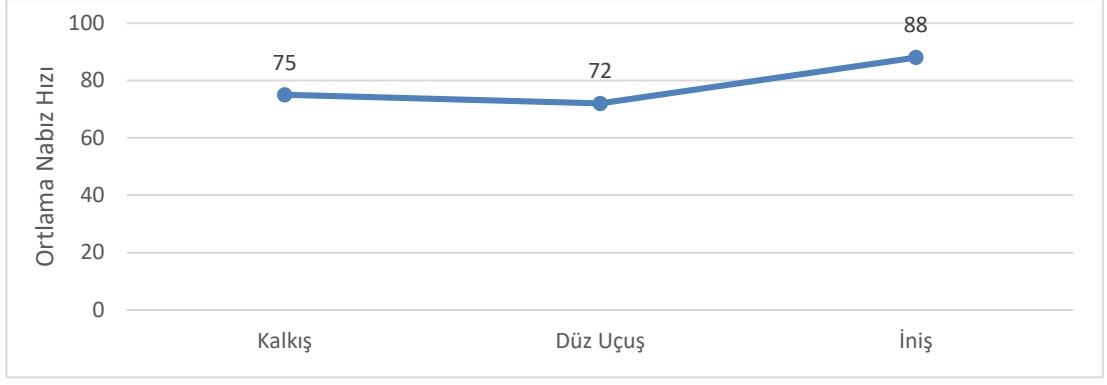
Şekil 4.11 : 6 nolu uçuş için nabız hızı ve göz kırpma sayısı ortalamaları.



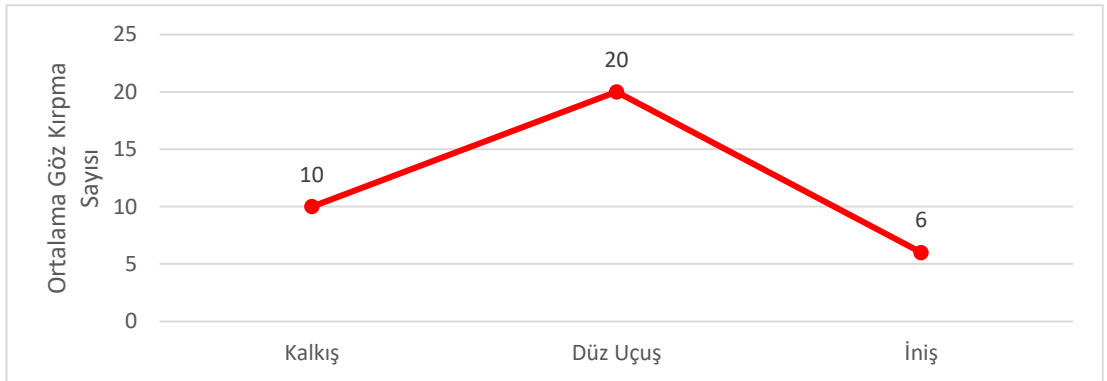
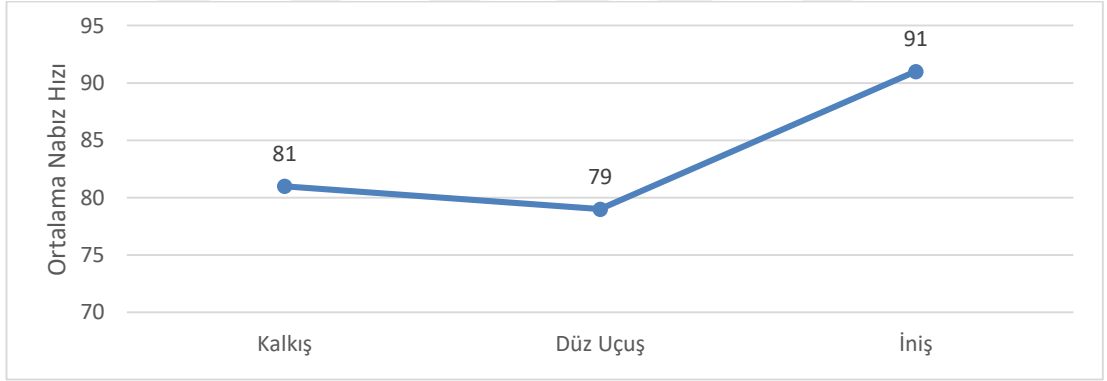
Şekil 4.12 : 7 nolu uçuş için nabız hızı ve göz kırpma sayısı ortalamaları.



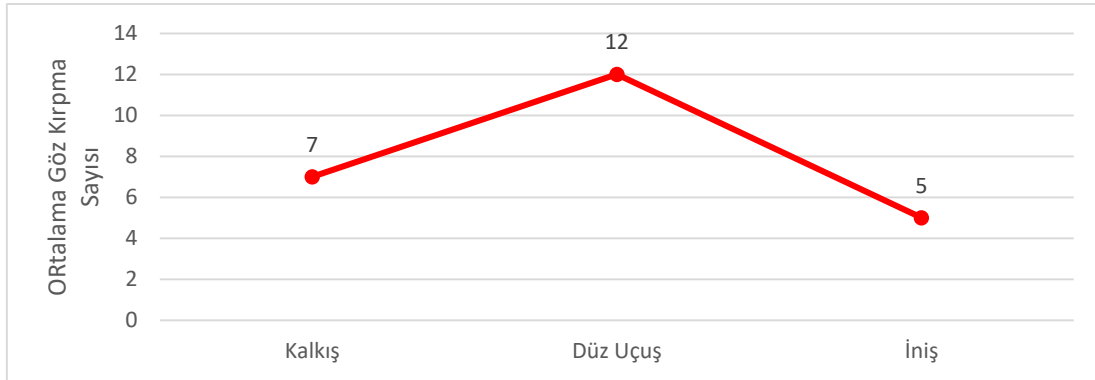
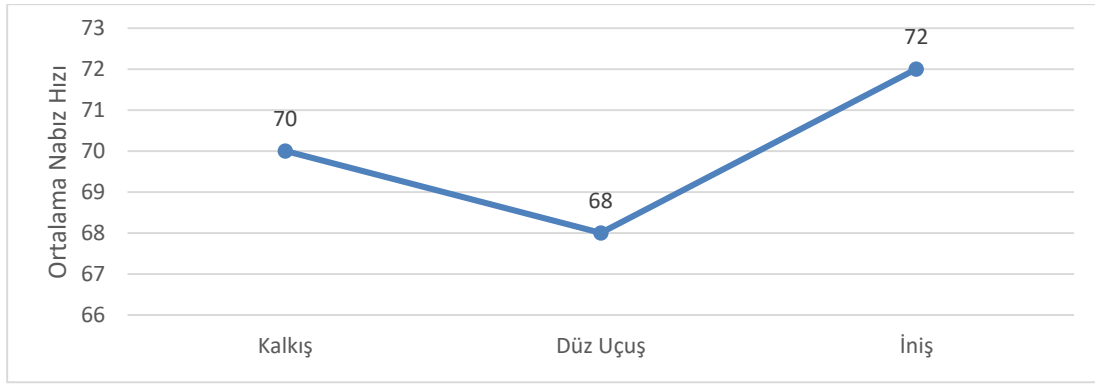
Şekil 4.13 : 8 nolu uçuş için nabız hızı ve göz kırpma sayısı ortalamaları.



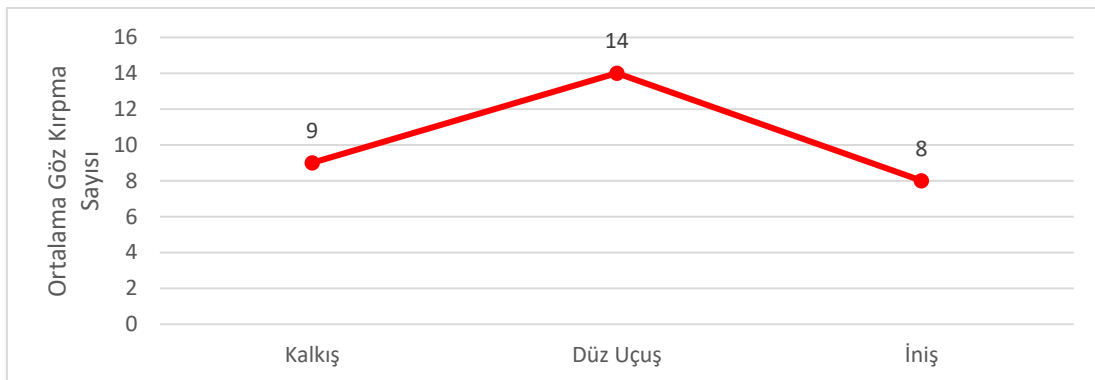
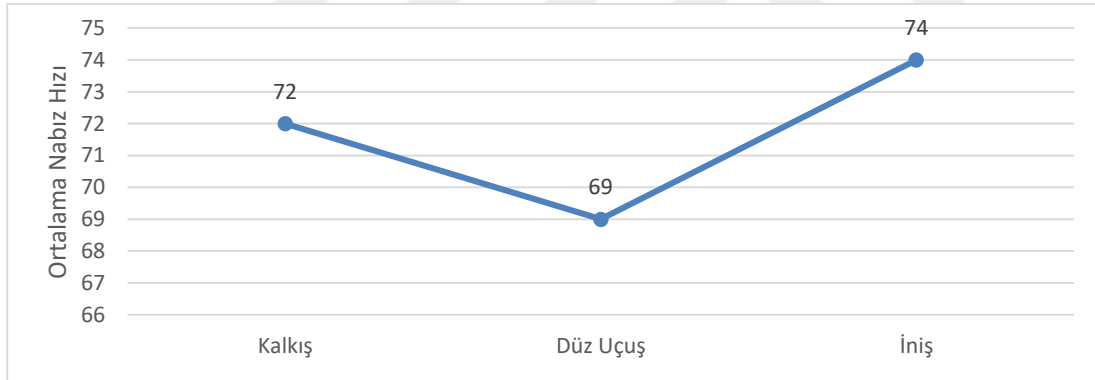
Şekil 4.14 : 9 nolu uçuş için nabız hızı ve göz kırpma sayısı ortalamaları.



Şekil 4.15 : 10 nolu uçuş için nabız hızı ve göz kırpma sayısı ortalamaları.



Şekil 4.16 : 11 nolu uçuş için nabız hızı ve göz kırpma sayısı ortalamaları.



Şekil 4.17 : 12 nolu uçuş için nabız hızı ve göz kırpma sayısı ortalamaları.

5. SONUÇ VE ÖNERİLER

Çalışma sürecince öncelikle literatürde bulunan çalışmaların çoğu incelenmiş, incelenen bu kaynaklar neticesinde düşük bütçeli olacak şekilde en etkin sistemin tasarlanması üzerine bir yaklaşım oluşturulmuştur.

Göz kırpma eylemi, yorgun olunmayan koşullarda yapılan ilk on uçuştaki değişen uçuş fazlarına göre incelendiğinde, dikkat seviyesinin maksimum noktaya çıktığı durumlarda minimum seviyeye geldiği gözlenmektedir. Bu da pilotun görev esnasında diğer görevlere göre daha zorlu bir durum ile başa çıkmaya çalıştığının göstergesidir. Bu da pilot üzerindeki iş yükünün arttığının bir göstergesi olarak kabul edilmektedir. Yorgun olunan son iki uçuşta bu durum incelendiği taktirde yorgun olunmayan senaryoya göre değişiklikler olduğu görünmektedir. Bu değişikliklerden özellikle nabız hızının yorgun olunmayan senaryoya göre daha düşük seviyelerde kaldığı görünmektedir. Göz kırpma sıklığı, iniş ve kalkış fazlarında çok belirgin değişiklikler göstermese de özellikle düz uçuş aşamasında yorgun olunmayan senaryoya göre belirgin bir düşüş göstermiştir. Dolayısıyla geliştirilen sistemin pilotun yorgun olduğu ve yorgun olmadığı senaryolar arasındaki değişiklikleri farkedebilecek kapasitede olduğunu göstermektedir. Bu değişikliklerin gerçek uçuşlarda simülör uçuşlarına göre daha da belirgin olacağı göz önünde bulundurulduğunda bu sistemin zamanla daha yaygın bir şekilde kullanılacağı öngörülmektedir.

Daha önceki çalışmalarda nabız hızı ile ilgili incelenen gerçek uçuş verilerinin değişen uçuş fazı ya da senaryolarına göre daha belirgin değişiklikler gösterdiği görülmüştür. Pilot iş yükü ve takip sistemi çalışmasında yapılan simülör uçuşlarının gerçek uçuşlara göre daha belirgin sonuçlar ortaya koymayacağı daha önceden varsayım olarak kabul edilmiştir. Bu kabulün en önemli parametresinin gerçek uçuş ile simülör uçuşu arasındaki en büyük fark yani simülör uçuşu esnasında hatadan kaynaklı kaza ya da ölüm durumlarının oluşmamasıdır. Simülör koltuğunda uçuş yapan pilot bu durumu bildiği için üzerinde herhangi bir psikolojik baskı oluşmamaktadır. Her ne kadar simülör koltuğu da olsa insan doğası gereği her zaman başarıya ya da başarılı

olma duygusuna sahip olmak istemiştir. Bu durumun simulator uçuşlarındaki nabız hızı değişimlerine küçük de olsa bir etkisi olduğu düşünülmektedir.

Ayrıca bir önceki bölümde sunulan grafiklerde görüldüğü üzere ortalama nabız sayısı ile göz kırpması sayısı arasında ters bir orantı olduğu gözlenmektedir. Bu da uçuş fazları arasındaki geçişe göre değişmektedir. Uçuş esnasında yapılan iş yoğunluğu ve dikkat gerektiren çalışmalar düşünüldüğünde uçuş fazlarını zordan kolayca doğru sıralamak gerekirse bunlar sırasıyla iniş, kalkış ve düz uçuş olduğu düşünülmektedir. Bir dakikada gerçekleşen göz kırpması hızının en az olduğu ve nabız hızının en yüksek olduğu aşama iniş aşaması, bu durumun tam tersi yani göz kırpması hızının en yüksek, nabız hızının en düşük olduğu aşama ise düz uçuş olduğu gözlenmektedir.

Pilot iş yükü ve takip sistemi çalışmasında bulunan pilot baş pozisyonu verisi doğrudan iş yükü kapsamında değerlendirilmemiştir. Bu daha çok pilotun başının baygınlık ya da uykulama gibi durumlarda öne ya da yanlara doğru devrilmesini gözlemlemek amaçlı kullanılmıştır. Bu çalışmada daha çok sağlık durumu takibi amaçla kullanılan bu durum anlık olarak operator ekranından görüntülenebilmektedir.

Yapılan bu çalışmanın genişletilebilmesi için çok fazla alan bulunmaktadır. Özellikle sadece hava araçları değil kara ve deniz ulaşımında da iş yükü ve sağlık takibinin daha kolay bir şekilde yapılarak acil durumlarda kazayı önleyebilici senaryolar çalışılabilir. Ayrıca bu iş yoğunluğu ölçümü için farklı senaryolar farklı hava koşulları, gece uçuşları, acil inişler ve aletli uçuş gibi farklı senaryolar ile simulator uçuşları yapılarak bu durumlardaki fizyolojik değişiklikler takip edilebilir. Böylelikle değişen uçuş şartlarına göre pilot üzerindeki iş yoğunluğu takip edilerek bu zorlu görevlerden bazılarını otomatik olarak uçuş bilgisayarının yapması sağlanabilir. Böylelikle pilot üzerindeki iş yoğunluğu azaltılarak uçuşun daha güvenli hale gelmesi sağlanabilir.

Son olarak pilot iş yükü ve sağlık takip sistemi için geliştirilen elektronik devre tasarımı prototip çalışması olduğu için kapladığı alan ve bulunduğu konum açısından ergonomik görünmemektedir. İleride yapılacak çalışmalarda daha az yer kaplayan hatta kask içerisinde entegre bir elektronik devre geliştirme çalışması yapılabilir.

KAYNAKLAR

- [1] **Human Factors Society Human-Computer Interaction Standards Committee.** (1987). *Proceedings Of The Human Factors Society Annual Meeting*, 31(8), 871-873. doi: 10.1177/154193128703100805.
- [2] **Reising, J. M.** (1972). The definition and measurement of pilot workload. Wright-Patterson AFB. Ohio: USAF Flight Dynamics Laboratory, AFFDL-TM-72-4-FGR.
- [3] **Moray, N.** (1979). Mental workload: Its theory and measurement. *New York: Plenum Press.*
- [4] **Jahns, D. W.** (1973). Operator workload: What is it and how should it be measured? In K. D. Cross and J. J. hlcGrath (Eds.) *Crew system design*. Santa Barbara. California: Anacapa Sciences.
- [5] **Jahns, D. W.** (1973). A concept of operator workload in manual vehicle operations meckenheim, Germany: *Forschungsinstitut fur Anthropotechnik*, Report No. 14.
- [6] **Gartner, W.B. and Murphy, M.R.** (1976). Pilot workload and fatigue: A critical survey of concepts and assessment techniques. *Moffett Field, California: National Aeronautical and Space Administration Ames Research Center*, NASA TN D-8365.
- [7] **Roscoe, A. H.** (1978). Assessing pilot workload. AGARD-AG. 233.
- [8] **Roscoe, A. H.** (1978). Physiological methods. In A. H. Roscoe (Ed.) *Assessing pilot workload*. AGARD-AG-233.
- [9] **Ellis, G. A.** (1978). Subjective assessment. In A. H. Roscoe (Ed.) *Assessing pilot workload*. AGARD-AG-233.
- [10] **Chiles, W. D.** (1978). Objective methods. In A. 11. Roscoe (Ed.) *Assessing pilot workload*, AGARD-AG-233.
- [11] **Chiles, W. D. and Alluisi, E. A.** (1979). On the specification of operator or occupational workload with performance measurement methods. *Human Factors*, 2f. 515-528.
- [12] **Schiflett, S. G.** (1976). Operator workload: An annotated bibliography. *Patuxent River, Maryland: US. Naval Air Test Center*, SY-257R-76.
- [13] **Wienville. W. W. and Williges. R. C.** (1978). Survey and analysis of operator workload assessment techniques. *Blacksburg. Virginia: Systemetrics*, Report S-78-101.
- [14] **Gemelli, A.** (1917). *Sull' applicazione dei metodi psico-fisici all' esame dei candidati all' aviazione militare.* (On the application of psycho-physical methods of examining candidates as military pilots). Milan: Vita epensiero.

- [15] **White, M.S.** (1940). The effect of anoxia in high altitude flight on the electrocardiogram. *Journal of Aviation Medicine*, II, 166-180.
- [16] **Kirsch, R.E.** (1945). A physiological study of aviators during combat flying. *Journal of Aviation Medicine*, 16, 376-384.
- [17] **Narsete, E.M.** (1949). Some physiologic responses of airmen on long-range flights in the Arctic. *Journal of Aviation Medicine*, 20, 336-342.
- [18] **Holden, G.R., Smith, J.R., & Smedal, H.A.** (1962). Physiological instrumentation systems for monitoring pilot response to stress at zero and high G. *Aerospace Medicine*, 33, 420-427.
- [19] **Helvey, W.M., Albright, G.A., & Axelrod, I.** (1964). A review of biomedical monitoring activities and report on studies made on F-105 pilots. *Aerospace Medicine*, 35, 23-27.
- [20] **Roman, J.A., & Lamb, L.E.** (1962). Electrocardiography in flight. *Aerospace Medicine*, 33, 527-544.
- [21] **Roman, J.A., Older, H., & Jones, W.L.** (1967). Flight research programme VI: Heart rate and landing error in restricted fields of vision landings. *Aerospace Medicine*, 38, 128-132.
- [22] **Roman, J.A., Older, H., & Jones, W.L.** (1967). Flight research programme VII: Medical monitoring of navy carrier pilots in combat. *Aerospace Medicine*, 38, 133-139.
- [23] **S'Jongere, J.J., Bertels, A.M., & Ego, T.** (1977). Some psychosocial and physiologic aspects of Belgian glider pilots (in French). *Brueles Medicine*, 57, 309-320.
- [24] **Howitt, J.S.** (1965). A preliminary study of flight deck workloads in civil air transport aircraft. *London: Flying Personnel Research Committee*, Report No 1240.
- [25] **Howitt, J.S.** (1969). Flight-deck workload studies in civil air transport aircraft. Conference Proceedings. *Measurement of Aircrew Performance. Paris: AGARD*, No: 56.
- [26] **Ruffell-Smith, H.P.** (1967). Heart rate of pilots flying aircraft on scheduled airline routes. *Aerospace Medicine*, 38, 1117- 1119.
- [27] **Bateman, S.C., Goldsmith, R., Jackson, K.F., Ruffell-Smith, H.P., & Mattocks, V.S.** (1970). Heart rate of training captains engaged indifferent activities. *Aerospace Medicine*, 41, 425-429.
- [28] **Bruner, H., & Hohlweck, H.** (1973). A New method for recording the heart and respiratory rates of cockpit crews in flight. In: *Current research work at the Institute for Aerospace Medicine*. (Bonn-Bad Godesberg Technical Translation of DLR-FB 73-15), 19-29.
- [29] **Carruthers, M., Arguelles, A.E., & Mosovich, A.** (1976). Man in transition: Biochemical and physiological changes during intercontinental flights. *Lancet*, 1, 977-981.
- [30] **Sekiguchi, C., Hunda, Y., Gotoh, M., Kurihara, Y., Nagasawa, Y., & Kuroda, I.** (1977). Continuous ECG monitoring on civil air crews during flight operations. *Aviation Space Environmental Medicine*, 48, 872-876.

- [31] **Sekiguchi, C., Yamaguchi, O., Kitajima, T., Veda, Y.** (1979). Frequency analysis of heart rate variability under flight conditions. *Aviation Space Environmental Medicine*, 50, 625-634.
- [32] **Hart, S.G., & Hauser, J.R.** (1988). Inflight application of three pilot workload measurement techniques. *Aviation Space Environmental Medicine*, 59, 511-516.
- [33] **Roscoe, A.H., & Grieve, B.S.** (1986). The impact of new technology on pilot workload. Warrendale, PA: Society of Automotive Engineers. *SAE Technical Paper Report 861773*, pp. 1-8
- [34] **Roscoe, A.H., & Grieve, B.S.** (1988). Assessment of pilot workload during Boeing 767 normal and abnormal operating conditions. *SAE Technical Paper Report 881382*.
- [35] **Kakimoto, Y., Nakamura, A., Tarui, H., Nagasawa, Y., & Yagura, S.** (1988). Crew workload in JASDF C-1 transport flight. In: Change in heart rate and salivary cortisol. *Aviation Space Environmental Medicine*, 59, 5 1 1-6.
- [36] **Van de Graaff, R.C.** (1987). An in-flight investigation of workload assessment techniques for civil aircraft operations. *National Aerospace Laboratory TR 87119*.
- [37] **Opmeer, C.H.J.M., & Krol, J.P.** (1973). Towards an objective assessment of cockpit workload: Physiological variables during different flight phases. *Aerospace Medicine*, 44, 527-532.
- [38] **Roscoe, A.H., & Goodman, E.A.** (1973). An investigation of heart rate changes during a flight simulator approach and landing task. *RAE Technical Memorandum Avionic*, p. 155.
- [39] **Eliasch, H., Rosen, A., & Scott, H.M.** (1967). Systemic circulatory response to stress of simulated flight and to physical exercise before and after propranolol blockage. *British Heart Journal*, 29, 671-683.
- [40] **Lindholm, E., & Cheatham, C.M.** (1983). Autonomic activity and workload during learning of a simulated aircraft carrier landing task. *Aviation Space Environmental Medicine*, 54, 435-439.
- [41] **Stern, J. A., & Skelly, J. J.** (1984). The Eye Blink and Workload Considerations. *Proceedings of the Human Factors Society Annual Meeting*, 28(11), 942-944.
- [42] **Geratewohl, S.** (1987). Leitfaden der militärischen Flugpsychologie. München, Germany: Verlag für Wehrwissenschaften.
- [43] **Mosier, K. L.** (2010). The human in flight. From kinesthetic sense to cognitive sensibility. In E. Salas & D. Maurino (Eds.), *Human factors in aviation* (pp. 175-207).
- [44] **Wilson, G. F., Purvis, B., Skelly, J., Fullenkamp, P., & Davis, I.** (1987). Physiological Data Used to Measure Pilot Workload in Actual Flight and Simulator Conditions. *Proceedings of the Human Factors Society Annual Meeting*, 31(7), 779-783.

- [45] **Wilson, G. F., & Fisher, F. R. A. N. K.** (1991). The use of cardiac and eye blink measures to determine flight segment in F4 crews. *Aviation, space, and environmental medicine*, 62(10), 959-962.
- [46] **Wilson, G.** (2002). An Analysis of Mental Workload in Pilots During Flight Using Multiple Psychophysiological Measures. *The International Journal of Aviation Psychology*, 12:1, 3-18, DOI: 10.1207/S15327108IJAP1201_2
- [47] **Zheng, Y., & Jie, Y.** (2019). Study of NASA-TLX and Eye Blink Rates Both in Flight Simulator and Flight Test. In: *Harris D. (eds) Engineering Psychology and Cognitive Ergonomics*, HCII.
- [48] **Foxlin, E. M., Harrington, M., & Altshuler, Y.** (1998). Miniature six-DOF inertial system for tracking HMDs. In *Helmet-and Head-Mounted Displays III, International Society for Optics and Photonics*. (Vol. 3362, pp. 214-228).
- [49] **Heitmann, A., Guttkuhn, R., Aguirre, A., Trutschel, U., & Moore-Ede, M.** (2001). *Technologies for the monitoring and prevention of driver fatigue*.
- [50] **Li, P., Meziane, R., Otis, M. J. D., Ezzaidi, H., & Cardou, P.** (2014). A Smart Safety Helmet using IMU and EEG sensors for worker fatigue detection. In *2014 IEEE International Symposium on Robotice and Sensors Environments (ROSE) Proceedings* (pp. 55-60). IEEE.
- [51] **Llorach, G., Evans, A., Agenjo, J., & Blat, J.** (2014, June). Position estimation with a low-cost inertial measurement unit. In *2014 9th Iberian Conference on Information Systems and Technologies (CISTI)* (pp. 1-4). IEEE.
- [52] **LaValle, S. M., Yershova, A., Katsev, M., & Antonov, M.** (2014). Head tracking for the Oculus Rift. In *2014 IEEE International Conference on Robotics and Automation (ICRA)* (pp. 187-194). IEEE.
- [53] **Windau, J., & Itti, L.** (2016). Walking compass with head-mounted IMU sensor. In *2016 IEEE International Conference on Robotics and Automation (ICRA)* (pp. 5542-5547). IEEE.
- [54] **Moraes, J., Rocha, M., Vasconcelos, G., Vasconcelos Filho, J., de Albuquerque, V., & Alexandria, A.** (2018). Advances in Photoplethysmography Signal Analysis for Biomedical Applications. *Sensors*, 18(6), 1894. doi: 10.3390/s18061894
- [55] **Adres-1** *Pulse oximetry* (2020). *En.wikipedia.org*, Available from: https://en.wikipedia.org/wiki/Pulse_oximetry (accessed 20 January 2020).
- [56] **Adres-2** *SinusRhythmLabels.svg* - *Wikimedia Commons* (2020). *Commons.wikimedia.org*, Available from: <https://commons.wikimedia.org/wiki/File:SinusRhythmLabels.svg> (accessed 20 January 2020).
- [57] **Ibrahim B., Nathan V., & Jafari R.** (2017). Exploration and validation of alternate sensing methods for wearable continuous pulse transit time measurement using optical and bioimpedance modalities. *2017 39th*

- [58] **Adres-3** *Semiconductor and Integrated Circuit Devices* (2020). Fairchildsemi.com, Available from: <https://www.fairchildsemi.com/datasheets/QR/QRD1114.pdf> (accessed 25 January 2020).
- [59] **Królak A. & Strumillo P.** (2011). Eye-blink detection system for human–computer interaction. *Universal Access in the Information Society*, 11(4), 409-419.
- [60] **Sliney D.H.** (1994). Ocular Hazards of Light. *International Lighting in Controlled Environments Workshop*, NASACP-95-3309.
- [61] **Adres-4** *MEFANET, C.* (2020). Unipolar and bipolar connection - WikiLectures. https://www.wikilectures.eu/w/Unipolar_and_bipolar_connection. Retrieved 29 January 2020.
- [62] **Adres-5** *Arduino Reference.* (2020). Retrieved 14 May 2020, from <https://www.arduino.cc/reference/en/language/functions/time/millis/>.
- [63] **Colton S. & Mentor F.R.C.** (2007). The balance filter. *Presentation, Massachusetts Institute of Technology.*
- [64] **Gui, P., Tang, L., & Mukhopadhyay, S.** (2015). MEMS based IMU for tilting measurement: Comparison of complementary and kalman filter based data fusion. *2015 IEEE 10Th Conference on Industrial Electronics and Applications (ICIEA)*. doi: 10.1109/iciea.2015.7334442.
- [65] **Adres-6** *Teensyduino: Using the UART (real serial) with Teensy on the Arduino IDE.* (2020). Retrieved 3 February 2020, from https://www.pjrc.com/teensy/td_uart.html.
- [66] **Adres-7** *Serial Peripheral Interface.* (2020). Retrieved 5 February 2020, from https://en.wikipedia.org/wiki/Serial_Peripheral_Interface.
- [67] **Adres-8** *I²C.* (2020). Retrieved 9 February 2020, from <https://en.wikipedia.org/wiki/I%C2%B2C>.
- [68] **Adres-9** *7-bit, 8-bit, and 10-bit I2C Slave Addressing.* (2020). Retrieved 9 February 2020, from <https://www.totalphase.com/support/articles/200349176>
- [69] **Adres-10** *Simülatör nedir? Simülasyon ne demek?.* (2020). Retrieved 9 May 2020, from <https://www.hurriyet.com.tr/gundem/simulator-nedir-simulasyon-ne-demek-41388050>
- [70] **Adres-11** *FlightGear.* (2020). Retrieved 9 May 2020, from <https://en.wikipedia.org/wiki/FlightGear>



EKLER

EK A: Gömülü Yazılım Kodları

EK B: Kullanıcı Arayüzü Yazılım Kodu



EK A

```
#include <Wire.h>

/* ECG variables */
unsigned long time_array[2] = {0};
int ecg_value = 0;
unsigned long int_time = 0;
int count = 0;
int ecg_analog = 0;

int analogValue = 0;

void setup() {
  Serial.begin(115200);
  pinMode(A1, INPUT);
}

void loop() {

  //Read ECG value
  analogValue = analogRead(A1);
  Serial.print("ECG= ");
  Serial.println(analogValue);

  //Heart Beat Count
  ECG_Count(analogValue);
}

void ECG_Count(int ecg_analog) {
  if (ecg_analog > 750) {
    count++;
    time_array[count - 1] = millis();
    if (count == 2) {
      int_time = time_array[1] - time_array[0];
      ecg_value = 60000 / int_time;
      if (ecg_value < 200) {
        Serial.print("\t\t\t\t\tHeartRateValue= ");
        Serial.println(ecg_value);
      }
      count = 0;
    }
  }
}
```

Şekil A.1 : EKG görüntüleme ve Nabız hızı ölçüm kodu.

```
#include <Wire.h>
#include <SparkFunMPU9250-DMP.h>

/* IMU variables */
```



```

MPU9250_DMP imu;
double roll , pitch, yaw;
long int pre_ts = 0;
float phi_quat;
float Q[4] = {1, 0, 0, 0} ;
float Q_dot [4];
float Q_pre [4] = {1, 0, 0, 0};

void setup() {
  Serial.begin(115200);

  // IMU initiation
  imu.begin();
  imu.setSensors(INV_XYZ_GYRO | INV_XYZ_ACCEL |
INV_XYZ_COMPASS);
  imu.setGyroFSR(250); // Set gyro to 2000 dps

  // Accel options are +/- 2, 4, 8, or 16 g
  imu.setAccelFSR(2); // Set accel to +/-2g
  imu.setLPF(10); // Set LPF corner frequency to 5Hz
  imu.setSampleRate(10); // Set sample rate to 10Hz
  imu.setCompassSampleRate(50); // Set mag rate to 10Hz

  // begin millis function
  pre_ts = millis();
}

void loop() {

  //IMU Update
  if ( imu.dataReady() ) {
    imu.update(UPDATE_ACCEL | UPDATE_GYRO | UPDATE_COMPASS);
    printIMUData(millis() - pre_ts);
    pre_ts = millis();
  }
}

void printIMUData(long int dt) {
  float accelX = imu.calcAccel(imu.ax);
  float accelY = imu.calcAccel(imu.ay);
  float accelZ = imu.calcAccel(imu.az);
  float gyroX = imu.calcGyro(imu.gx) / 57.3;
  float gyroY = imu.calcGyro(imu.gy) / 57.3;
  float gyroZ = imu.calcGyro(imu.gz) / 57.3;
  float magX = imu.calcMag(imu.mx);
  float magY = imu.calcMag(imu.my);
  float magZ = imu.calcMag(imu.mz);

  // NORMALIZE ACCELE VALUES
  float naccel = sqrt(accelX * accelX + accelY * accelY + accelZ * accelZ);

```

```

accelX = accelX / naccel;
accelY = accelY / naccel;
accelZ = accelZ / naccel;

//Normailze mag values
float nmag = sqrt(magX * magX + magY * magY + magZ * magZ);

magX = magX / nmag;
magY = magY / nmag;
magZ = magZ / nmag;

//Euler angle from accel
pitch = atan2 (accelY , ( sqrt ((accelX * accelX) + (accelZ * accelZ))));
roll = atan2(-accelX , ( sqrt((accelY * accelY) + (accelZ * accelZ))));

// yaw from mag
float Yh = (magY * cos(roll)) + (magZ * sin(roll));
float Xh = (magX * cos(pitch)) + (magY * sin(roll) * sin(pitch)) + (magZ *
cos(roll) * sin(pitch));

//yaw = atan2(Yh, Xh);

roll = (0.98 * (roll + (gyroX) * dt / 1000.0f) + 0.02 * (accelX)) * 57.3;
pitch = (0.98 * (pitch + gyroY * dt / 1000.0f) + 0.02 * (accelY)) * 57.3;
//yaw = ((yaw + gyroZ * dt / 1000.0f) + 0.02 * (magZ)) * 57.3;

//Quaternions
Q_dot[0] = -0.5 * ( (gyroX * Q_pre[1]) + (gyroY * Q_pre[2]) + (Q_pre[3] *
gyroZ));
Q_dot[1] = 0.5 * ( (gyroX * Q_pre[0]) + (gyroZ * Q_pre[2]) - (Q_pre[3] *
gyroY));
Q_dot[2] = 0.5 * ( (gyroY * Q_pre[0]) - (gyroZ * Q_pre[1]) + (Q_pre[3] *
gyroX));
Q_dot[3] = 0.5 * ( (gyroZ * Q_pre[0]) + (gyroY * Q_pre[1]) - (Q_pre[2] *
gyroX));

Q[0] = Q_pre[0] + (Q_dot[0] * dt / 1000.0);
Q_pre[0] = Q[0];
Q[1] = Q_pre[1] + (Q_dot[1] * dt / 1000.0);
Q_pre[1] = Q[1];
Q[2] = Q_pre[2] + (Q_dot[2] * dt / 1000.0);
Q_pre[2] = Q[2];
Q[3] = Q_pre[3] + (Q_dot[3] * dt / 1000.0);
Q_pre[3] = Q[3];

double n = (sqrt((Q[0] * Q[0]) + (Q[1] * Q[1]) + (Q[2] * Q[2]) + (Q[3] * Q[3])));
float Q0 = Q[0] / n;
float Q1 = Q[1] / n;

```

```

float Q2 = Q[2] / n;
float Q3 = Q[3] / n;

phi_quat = atan2 (2 * ((Q0 * Q1) + (Q2 * Q3)), ((0.5f - Q1 * Q1 - Q2 * Q2)));
float theta_quat = asin( 2 * ((Q0 * Q2) - (Q1 * Q3)));
float psi_quat = atan2(2 * ((Q0 * Q3) + (Q1 * Q2)), ((0.5f - Q2 * Q2 + Q3 *
Q3)));

phi_quat = (0.98 * (phi_quat + gyroX * dt / 1000.0f) + 0.02 * (accelX)) * 57.3;
theta_quat = (0.98 * (theta_quat + gyroY * dt / 1000.0f) + 0.02 * (accelY)) *
57.3;
psi_quat = (0.98 * (psi_quat + gyroZ * dt / 1000.0f) + 0.02 * (magZ)) * 57.3;

int filtered_roll = 0.99 * (roll + roll * dt / 1000.0f) + 0.01 * (phi_quat);

int filtered_pitch = 0.99 * (pitch + pitch * dt / 1000.0f) + 0.01 * (theta_quat);

//int filtered_yaw = 0.5 * (yaw + yaw * dt / 1000.0f) + 0.5 * (psi_quat);

Serial.print("\t\t\tIMU= ");
Serial.println(String(filtered_roll) + ' ' + String(filtered_pitch));
//,'+String(filtered_yaw));
Values[4] = filtered_roll;
Values[5] = filtered_pitch;
}

```

Şekil A.2 : IMU Complimentary filtre kodu.

```

#include <Wire.h>

/* Eye Blink variables */
volatile unsigned int eye_count = 0;
const byte ledPin = 13;
const byte interruptPin = 14;
volatile byte state = LOW;
IntervalTimer BPMTimer;

void setup() {
  Serial.begin(115200);

  // Eye blink interrupt routine
  pinMode(ledPin, OUTPUT);
  pinMode(interruptPin, INPUT_PULLUP);
  attachInterrupt(digitalPinToInterrupt(interruptPin), BPM_count, FALLING);
  //Count blink when falling edge
  BPMTimer.begin(BPM_print, 60000000); // Blink per 1min
}

void loop() {

  //Blinking led when blinking eye

```

```

    digitalWrite(ledPin, state);
}
/*
    Counting blink per min.
*/
void BPM_count() {
    state = HIGH;
    digitalWrite(ledPin, state);
    delay(25);
    state = LOW;
    eye_count++;
}
/*
    Print and write blink per min value
*/
void BPM_print() {
    Serial.print("BlinkPerMin= ");
    Serial.println(eye_count);
    eye_count = 0;
}

```

Şekil A.3 : Göz kırpma algılama kodu.

```

#include <SD.h>
#include <SPI.h>
#include <EEPROM.h>

/* SD card variables */
File myFile;
int Values[8] = {0}; //[HOUR, MIN, SEC, ECG, IMU_ROLL, IMU_PITCH,
BLINK_PER_MIN, HEART_RATE_VALUE];
const int chipSelect = BUILTIN_SDCARD;
unsigned int address = 0;
byte file_value = 123;
String data_number;

void setup() {
    Serial.begin(115200);
    SD.begin(chipSelect);

    // SD File name update
    file_value = EEPROM.read(address);
    data_number += String("data");
    data_number += String(file_value);
    data_number += String(".csv");
    char file_name[data_number.length() + 1]; //strlen kullanılabilir mi?
    data_number.toCharArray(file_name, data_number.length() + 1);
    myFile = SD.open(file_name, FILE_WRITE);
    file_value++;
    EEPROM.update(address, file_value);
}

```

```

// Button interrupt routine
pinMode(buttonPin, INPUT);
attachInterrupt(digitalPinToInterrupt(buttonPin), CloseFile, FALLING); //Close
file when button falling edge

}

void loop() {

    //Write all values to SD Card
    myFile.println(String(Values[0]) + ':' + String(Values[1]) + ':' + String(Values[2])
    + ',' + String(Values[3]) + ',' + String(Values[4]) + ',' + String(Values[5]) + ',' +
    String(Values[6]) + ',' + String(Values[7]));
    delay(10); //Wait for write process
}
/*
    Close .csv file
*/
void CloseFile() {
    //Serial.println("DOSYA KAPATILDI");
    myFile.close();
    delay(10);
}

```

Şekil A.4 : SD karta kayıt kodu.

```

#include <TimeLib.h>
#define TIME_HEADER "T" // Header tag for serial time sync message

void setup() {
    Serial.begin(115200);
    // Set the Time library to use Teensy 3.0's RTC to keep time
    setSyncProvider(getTeensy3Time);
    while (timeStatus() != timeSet);
    if (Serial.available()) {
        time_t t = processSyncMessage();
        if (t != 0) {
            Teensy3Clock.set(t); // set the RTC
            setTime(t);
        }
    }
}

void loop() {

    //Write real time to Values array
    digitalClockDisplay(Values);
}
/*
    Printing current time and date
*/

```

```

void digitalClockDisplay(int Values[]) {
    Serial.print(hour());
    Values[0] = hour();
    printDigits(minute());
    Values[1] = minute();
    printDigits(second());
    Values[2] = second();
    Serial.print(" ");
    Serial.print(" ");
    Serial.print(day());
    Serial.print(" ");
    Serial.print(month());
    Serial.print(" ");
    Serial.print(year());
    Serial.println();
}

time_t getTeensy3Time()
{
    return Teensy3Clock.get();
}
/*
    Process time sync messages from the serial port
*/
unsigned long processSyncMessage() {
    unsigned long pctime = 0L;
    const unsigned long DEFAULT_TIME = 1357041600; // Jan 1 2013

    if (Serial.find(TIME_HEADER)) {
        pctime = Serial.parseInt();
        return pctime;
        if ( pctime < DEFAULT_TIME) { // check the value is a valid time (greater than
Jan 1 2013)
            pctime = 0L; // return 0 to indicate that the time is not valid
        }
    }
    return pctime;
}
/*
    utility function for digital clock display:
    prints preceding colon and leading 0
*/
void printDigits(int digits) {
    Serial.print(":");
    if (digits < 10)
        Serial.print('0');
    Serial.print(digits);
}

```

Şekil A.5 : Göz kırpma algılama kodu.

EK B

```
using System.IO.Ports;
using System;
using UnityEngine;
public class Communication
{
    private string portName { get; set; }
    private int baudRate { get; set; }
    public SerialPort port;

    //delegates
    public delegate void QuaternionDataHandler(Quaternion quaternion);
    public event QuaternionDataHandler OnQuaternionDataReceived;

    public Communication(){}

    public Communication(string portName, int baudRate){
        this.portName = portName;
        this.baudRate = baudRate;
    }

    public void Connect(string portName, int baudRate){
        port = new SerialPort(portName);
        try
        {
            port.BaudRate = baudRate;
            port.NewLine = "\r\n";
            port.Handshake = Handshake.None;
            port.DtrEnable = true;

            port.DataReceived += new
            SerialDataReceivedEventHandler(GenericDataReceived);
            port.ErrorReceived += new
            SerialErrorReceivedEventHandler(ErrorReceived);
            port.Open();
        }
        catch(Exception e){
            Debug.LogError(e.Message);
            throw e;
        }
    }

    public void Connect(){
        try{
            Connect(this.portName, this.baudRate);
        }
        catch(Exception e){
            throw e;
        }
    }
}
```

```

private void GenericDataReceived(object sender, SerialDataReceivedEventArgs
args){
    SerialPort sp = (SerialPort)sender;
    string received = sp.ReadExisting();
    Debug.Log("Received:" + received);
    QuaternionDataReceived(received);
}

private void ErrorReceived(object sender, SerialErrorReceivedEventArgs args)
{
    Debug.Log(args);
}

private void QuaternionDataReceived(string received){
    //replace . to , for float conversion
    received = received.Replace('.', ',');
    string[] quaternion_str = received.Split();
    float [] quaternion = new float[4];
    for(int i= 0; i<quaternion_str.Length && i<4; i++){
        quaternion[i] = float.Parse(quaternion_str[i]);
    }
    Quaternion q = new Quaternion(quaternion[0], quaternion[1], quaternion[2],
quaternion[3]);
    OnQuaternionDataReceived(q);
}

public void ClearBuffers()
{
    port.DiscardInBuffer();
    port.BaseStream.Flush();
}
}

```

Şekil B.1 : Kullanıcı arayüzü seri port tanımlama kodu.

```

using System;
using System.Collections;
using UnityEngine;
using UnityEngine.UI;
using XCharts;

public class HeadMovement : MonoBehaviour
{
    public InputField port;
    public InputField baudRate;
    public Button connect;
    private Communication communication;

    //CHART
    public int maxCacheDataNumber;
}

```



```

public float initDataTime = 2;

//BLINK
public Text blinkValue;

private CoordinateChart chart;
private float updateTime;
private float initTime;
private int initCount;
private int count;
private bool isInitd;
private DateTime timeNow;
GameObject chartGameObject;
void Awake()
{
    chartGameObject = GameObject.Find("chart_auto");
    chart = chartGameObject.GetComponentInChildren<CoordinateChart>();
    chart.RemoveData();
    chart.xAxis0.maxCache = maxCacheDataNumber;
    var serie = chart.AddSerie(SerieType.Line);
    serie.symbol.type = SerieSymbolType.None;
    serie.maxCache = maxCacheDataNumber;

    blinkValue.text = "-";
    chart.title.subText = "-";
    chart.title.subTextFontSize = 60;
}

void Start()
{
    connect.onClick.AddListener(Connect);
    port.text = "COM12";
    baudRate.text = "500000";
}

void Update()
{
    if (communication != null && communication.port.IsOpen &&
communication.port.BytesToRead > 15)
    {
        string received = communication.port.ReadLine();
        received = received.Replace('.', ',');
        string[] received_str = received.Split(' ');
        if (received.Contains("HeartRateValue="))
        {
            try
            {
                float pulse = float.Parse(received_str[1]);
                chart.title.subText = pulse.ToString();
                chart.RefreshChart();
            }
            catch { }
        }
    }
}

```

```

    }
    catch(Exception e)
    {
        Debug.LogError("Cannot parse HR data");
    }
}
else if (received.Contains("IMU="))
{
    try
    {
        float[] imuData = new float[2];

        for (int i = 1; i < 3; i++)
        {
            imuData[i-1] = float.Parse(received_str[i]);
        }
        transform.rotation = Quaternion.Euler(-imuData[1], 0, -imuData[0]);
    }
    catch(Exception e)
    {
        Debug.LogError("Cannot parse IMU data");
    }
}
else if (received.Contains("ECG="))
{
    try
    {
        float ecg =float.Parse(received_str[1]);
        chart.AddData(0, ecg);
        chart.RefreshChart();
    }
    catch(Exception e)
    {
        Debug.LogError("Cannot parse ECG data");
    }
}
else if (received.Contains("BlinkPerMin="))
{
    try
    {
        float bpm = float.Parse(received_str[1]);
        blinkValue.text = bpm.ToString();
    }
    catch (Exception e)
    {
        Debug.LogError("Cannot parse ECG data");
    }
}
}

```

```
        //communication.ClearBuffers();
    }
}

void Connect()
{
    try
    {
        string portStr = port.text;
        int baudRateInt = int.Parse(baudRate.text);
        communication = new Communication(portStr, baudRateInt);
        communication.OnQuaternionDataReceived += RotateHead;
        communication.Connect();
        //connect.enabled = false;
    }
    catch (Exception e)
    {
        Debug.LogError(e.Message);
    }
}
}
```

Şekil B.2 : Kullanıcı arayüzü veri toplama ve anlamlandırma kodu.



ÖZGEÇMİŞ

Ad-Soyad : Efkan Yılmaz
Doğum Tarihi ve Yeri : 06.10.1993 Fethiye
E-posta : efkanyilmaz93@gmail.com

ÖĞRENİM DURUMU:

- **Lisans** : 2016, Yıldız Teknik Üniversitesi, Makine Fakültesi, Mekatronik Mühendisliği

DİĞER YAYINLAR, SUNUMLAR VE PATENTLER:

- [1] Patan, M.G., İbişhükçü, R., Yılmaz, E., Çetin, C., Kas, A. ve Sezer, V. (2019). İki Tekerlekli Kendini Dengeleyen Robot Tasarımı, Üretimi ve Kontrolü. *TOK 2019*, Muğla, Türkiye, 242-247.