

**T.C.  
İSTANBUL GELİŞİM ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ**

**ARDUINO VE ANDROID TABANLI  
KAZA BİLDİRİM VE UYARI SİSTEMİ**

**ABDULLAH ÇETİN**

**YÜKSEK LİSANS TEZİ  
MEKATRONİK MÜHENDİSLİĞİ ANABİLİM DALI**

**DANIŞMAN  
DR. ÖĞR. ÜYESİ ÜMİT ALKAN**

**İSTANBUL, 2019**

Abdullah ÇETİN tarafından hazırlanan “Arduino ve Android Tabanlı Kaza Bildirim ve Uyarı Sistemi” adlı tez çalışması aşağıdaki jüri tarafından OY BİRLİĞİ / OY ÇOKLUĞU ile İstanbul Gelişim Üniversitesi Mekatronik Anabilim Dalında YÜKSEK LİSANS TEZİ olarak kabul edilmiştir.

**Danışman:** Dr. Öğr. Üyesi Ümit ALKAN

Bilgisayar Mühendisliği, İstanbul Gelişim Üniversitesi

Bu tezin, kapsam ve kalite olarak Yüksek Lisans Tezi olduğunu onaylıyorum/onaylamıyorum

.....  
...

**Başkan:** Prof.Dr. BEDRİ YÜKSEL

Mekatronik Mühendisliği, İstanbul Gelişim Üniversitesi

Bu tezin, kapsam ve kalite olarak Yüksek Lisans Tezi olduğunu onaylıyorum/onaylamıyorum

.....

Tez Savunma Tarihi:

Jüri tarafından kabul edilen bu tezin Yüksek Lisans Tezi olması için gerekli şartları yerine getirdiğini onaylıyorum.

.....

Prof. Dr. Mustafa Bayram

Fen Bilimleri Enstitüsü Müdürü

## ETİK BEYAN

İstanbul Gelişim Üniversitesi Fen Bilimleri Enstitüsü Tez Yazım Kurallarına uygun olarak hazırladığım bu proje çalışmada;

- Tez içinde sunduğum verileri, bilgileri ve dokümanları akademik ve etik kurallar çerçevesinde elde ettiğimi,
- Tüm bilgi, belge, değerlendirme ve sonuçları bilimsel etik ve ahlak kurallarına uygun olarak sunduğumu,
- Tez çalışmada yararlandığım eserlerin tümüne uygun atıfta bulunarak kaynak gösterdiğimi,
- Kullanılan verilerde herhangi bir değişiklik yapmadığımı,
- Bu tezde sunduğum çalışmanın özgün olduğunu bildirir, aksi bir durumda aleyhime doğabilecek tüm hak kayıplarını kabullendiğimi beyan ederim.

Abdullah ÇETİN

10/06/2019

# ARDUINO VE ANDROID TABANLI KAZA TESPİT VE UYARI SİSTEMİ

(Yüksek Lisans Dönem Projesi)

Abdullah ÇETİN

GELİŞİM ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ  
Haziran 2019

## ÖZET

Trafik kazalarında ilk yardım hayat kurtarmada saniye ve dakikalarla sınırlıdır. Bu bağlamda ilk yardım ve sağlık ekibinin olaydan hemen haberdar olup olay yerine bir an önce intikal etmesi gerekmektedir.

Kaza gerçekleştiğinde, kazanın acil durum ekiplerine bildirilmesi kazaya karışan şahıslar ya da kazayı gören kişiler tarafından yapılmaktadır. Bazı durumlarda ise bu bildirim yapılamamaktadır. Ayrıca bu bildirimler esnasında kazanın şiddeti ve hasar miktarı doğru olarak bildirilememektedir. Kazanın bildirimini yapılamaması, gecikmesi veya hatalı bildirilmesi acil müdahale ekiplerinin etkin müdahalede bulunmasını zorlaştırmaktadır.

Bu çalışmada aracın trafik kazasına karışması durumunda, kazaya ait, çarpışma hızı, hasar sınıfı (hafif, orta, ağır ve ölümcül), türü (önden çarpışma, devrilme, yanma vb.), kaza zamanı, konumu (GPS koordinatları) gibi bilgiler tespit edilerek; kaza, anında acil durum ekiplerine bildirilebilecek, elektronik kaza bildirim ve uyarı sistemi devreye girecektir.

Bu sistem altı DOF eksenli (üç eksen jiroskop, üç eksen ivmeölçer) üzerinden alınan verileri mikrodenetleyicide işleyerek model olarak tasarlanan aracın ivmesel hareketini takip eder.

Sistem; kaza olması durumunda ivme şiddetini g-kuvveti üzerinden ölçerek kazayı kategorilendirir, kaza şiddetine göre, GPS üzerinden aldığı konum bilgilerini ve kaza şiddetini gerekli en yakın birimleri GPRS&GSM sistemi üzerinden mesajla bilgilendirir. Ayrıca trafik yoğunluğunu ve en kısa mesafeyi kat etmesi için konum bilgilerini harita sistemi şeklinde link olarak göndererek ilgili birimin olay mahalline hızlı bir şekilde ulaşmasını kolaylık sağlamaktadır.

Ayrıca ülkemizde artan sahte kazalardan dolayı sigorta şirketleri yüklü miktarlarda harcama yapmaktadır. Bu sistem kaza anından itibaren aracın üç boyutlu uzaysal alanda her hareketini ve her yönden aldığı darbeyi g kuvvetinden oluşan basınç ve ivmesel hareketlerinin kaydını araç hareketi sabitleninceye kadar kayıt altına alır. Böylece kazanın gerçekleştiği konum, saat, tarih ve kaza hakkında tam bir bilgi sahibi olmaya imkân sağlamaktadır.

Anahtar Kelimeler : Kaza bildirim, kaza bildirim, kaza uyarı sistemi, GPS Navigasyon,  
Sayfa Adedi : 90  
Danışman : Dr. Öğr. Üyesi Ümit ALKAN

# ARDUINO AND ANDROID BASED ACCIDENT DETECTION AND WARNING SYSTEM

(M. Sc. Thesis)

Abdullah ÇETİN

GELİŞİM UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

June 2019

## ABSTRACT

First aid in traffic accidents is limited to seconds and minutes in life-saving. In this context, first aid and medical team should be aware of the incident immediately and should be delivered to the scene as soon as possible.

When the accident occurs, the accident is reported to the emergency teams by the persons who may involve or the person who saw the accident. In some cases, this notification cannot be done. In addition, the severity of the accident and the amount of damage cannot be reported as.

In this study, if the vehicle is involved in a traffic accident, information such as crash rate, damage class (light, medium, heavy and fatal), type (frontal collision, tipping, burning, etc.), accident time, location (GPS coordinates) are determined. by; accident, electronic emergency detection and warning system will be activated. This system monitors the accelerated movement of the vehicle designed as a model by processing the data received on six DOF axis (three axis gyroscopes, three axis accelerometer) on the microcontroller.

System; In case of an accident, it measures the acceleration intensity over g force and categorizes the accident. According to the seriousness of the accident, the location information received via GPS and the seriousness of the accident to the nearest units with GPRS and GSM system. You can also observe traffic information and location information to cover the shortest distance. Map & navigation system in the form of a link to send the relevant unit to the event site provides quick access.

In addition, insurance companies spend large amounts due to increasing fake accidents in our country. This system records every movement of the vehicle in the three-dimensional spatial area from the moment of the accident and the impact it receives from all directions until the vehicle movement is stabilized. Consequently, this system gains the location, time, date, accident information to the people.

Keywords : Accident detection, accident warning system, GPS Navigation

Page Number : 90

Supervisor : Assist. Prof. Dr. Ümit ALKAN

## TEŞEKKÜR

Çalışmalarım esnasında bilgi, öneri ve yardımını esirgemeyen tez danışmanım Dr. Öğr. Üyesi Ümit ALKAN' a teşekkür ederim.

Çalışmalarımda beni destekleyen ASFA Eğitim Kurumları Genel Müdürü Bayram KUM' a ve kendimi geliştirmemde destek olan zümre arkadaşım Mehmet DUDU' ya teşekkür ederim. Ayrıca yaşantım boyunca çalışmalarımı manevi desteği ile yanımda bulunan eşim Müsavvet ÇETİN' e, ablam Elif KÖROĞLU' na ve her daim maddi manevi desteğini benden esirgemeyip bu günlere gelmemde katkısı olan annem Naide KÖSE' ye özellikle teşekkürü bir borç bilirim.



## İÇİNDEKİLER

ÖZET .....	iv
ABSTRACT.....	v
TEŞEKKÜR.....	vi
İÇİNDEKİLER .....	vii
ŞEKİLLERİN LİSTESİ.....	ix
RESİMLERİN LİSTESİ.....	x
TABLOLAR LİSTESİ.....	xi
1. GİRİŞ.....	1
1.1. Problem Durumu/ Konunun Tanımı .....	3
1.2. Araştırmanın Amacı .....	4
1.3. Araştırmanın Önemi.....	4
1.4. Sınırlılıklar .....	5
2. GENEL BİLGİLER.....	6
2.1. Trafik Kazası .....	6
2.1.1. Tanım .....	6
2.1.2. Trafik Kazası Çeşitleri .....	6
2.2. Yerçekimi Kuvveti .....	7
2.3. Dönme (Rotasyonel) Matrisi .....	8
2.4. GPS (Global Position System-Küresel Yer Belirleme Sistemi) Çalışma Prensipleri ...	11
3. SİSTEM TASARIM VE ÇALIŞMA PRENSİBİ.....	14
3.1. Kaza Bildirim ve Uyarı Sisteminin Çalışması .....	14
3.2. Sistemin Elektronik Tasarımı ve Birimlerin Tanıtılması .....	16
3.2.1. Mikrodenetleyici .....	16
3.2.2. GPS Genişletme Kartı.....	19
3.2.3. Bluetooth HC-06.....	25
3.2.4. I2C Haberleşme Protokolü.....	29
3.2.5. MPU 6050 İvme Ölçer ve Jiroskop .....	32
3.3. Yazılım Birimlerinin Tanıtılması .....	34
3.3.1. Arduino IDE.....	34
3.3.2. Processing 2.2.1. Simülasyon Platformu .....	36
3.3.3. Appinventor Android Yazılım Geliştirme Platformu .....	40
3.4. Sistem Algoritmaları .....	40
3.4.1. Mikrodenetleyici Sistem Algoritması .....	41
3.4.2. Processing 2.2.1 Simülasyon Sistem Algoritması .....	42
4. SONUÇLAR.....	43
KAYNAKLAR .....	46

EKLER.....	50
EK-1 Arduino IDE Yazılımı .....	51
EK-2 Processing Yazılımı .....	61
ÖZGEÇMİŞ .....	72





**ŞEKİLLERİN LİSTESİ**

<b>Şekil</b>	<b>Sayfa</b>
Şekil 1. Döndürme Matrisi.....	8
Şekil 2. Döndürme Matrisi.....	8
Şekil 3. Aracın üç boyutlu uzayda hareketi .....	10
Şekil 4. Sistem blok diyagramı .....	15
Şekil 5. Arduino Mega 2560 pin dağılımı diyagramı .....	18
Şekil 6. Bluetooth 3.3v için gerilim bölücü devre diyagramı .....	28
Şekil 7. Bluetooth voltaj bölücü devre diyagramı .....	28
Şekil 8. I2C Haberleşme start biti gönderimi.....	29
Şekil 9. I2C veri akışı için gerekli kare dalga durumu .....	30
Şekil 10. I2C Haberleşme veri döngüsü .....	30
Şekil 11. I2C Veri gönderme durum bitleri .....	31
Şekil 12. I2C Veri alma bit durumu.....	31
Şekil 13. I2C Haberleşme sonlandırma biti .....	31
Şekil 14. Mikro denetleyicinin kaza bildirim ve uyarı sistemine ait akış diyagramı.....	41
Şekil 15. Processing simulasyon ortamına ait java programının akış diyagramı .....	42
Şekil 16. Aracın ani durmada üç eksene göre ivmelenmesi .....	45

## RESİMLERİN LİSTESİ

<b>Resim</b>	<b>Sayfa</b>
Resim 1.1. Uydu konumunu bulma .....	12
Resim 2.2.Uydu konum bulma .....	13
Resim 3.3.Uydu konum bulma .....	13
Resim 4.Mikrodenetleyici kart.....	16
Resim 5.GPS Modülü pin dağılımı diyagramı.....	20
Resim 6.GPS Modülü jumper ayarı .....	20
Resim 7.GSM Modülü jumper ayarlamaları.....	21
Resim 8.GSM Modülü üst görünüş .....	22
Resim 9.Kaza sonrası acil müdahale birimlerine gönderilen mesaj .....	25
Resim 10.HC-06 Bluetooth modülü önden ve arkadan görünüş .....	26
Resim 11.HC-06 Bluetooth modülü bileşenleri.....	26
Resim 12.MPU605 ivme ölçer sensör .....	32
Resim 13.Arduino IDE görünüm.....	35
Resim 14.Arduino IDE'den üç eksene ait ölçüm alınması.....	36
Resim 15.Aracın üç boyutlu uzayda açısal konumunu temsil eden prizma .....	37
Resim 16.Processing 2.2.1 IDE görünümü.....	38
Resim 17.Aracın üç eksene göre hareketlerinin grafiksel görünümü .....	39
Resim 18.Kaza olayının gerçekleştiği konumun uydu görüntüsü .....	39

**TABLolar LİSTESİ**

<b>Tablo</b>	<b>Sayfa</b>
Tablo 1.TÜİK yıllara göre ölen ve yaralanan kaza sayısı	2
Tablo 2.Mikrodenetleyici kart çalışma gerilimi	17
Tablo 3.GPS Modülü elektriksel karakteristik durumu	19
Tablo 4.Mesaj gönderimi için AT komut seti	24
Tablo 5.Bluetooth giriş voltaj dağılımı	27
Tablo 6.Bluetooth çıkış voltaj dağılımı	27
Tablo 7.Çalışmanın diğer çalışmalar ile karşılaştırılması	44

## SİMGELER VE KISALTMALAR

Bu çalışmada kullanılmış simgeler ve kısaltmalar, açıklamaları ile birlikte aşağıda sunulmuştur.

<b>Simgeler</b>	<b>Açıklamalar</b>
<b>A</b>	Amper
<b>Hz</b>	Hertz
<b>I</b>	Akım
<b>KBps</b>	Kilo Bit per second
<b>Km</b>	Kilometre
<b>mA</b>	Mili Amper
<b>MBps</b>	Mega Bit per second
<b>MHz</b>	Mega Hertz
<b>mm</b>	Milimetre
<b>mV</b>	mili Volt
<b>R</b>	Ohm
<b>sn</b>	Saniye
<b>V</b>	Volt

<b>Kısaltmalar</b>	<b>Açıklamalar</b>
<b>ACC</b>	Accelerometer
<b>ACK</b>	Acknowledgement
<b>AT</b>	Attention
<b>BT</b>	Bluetooth
<b>DOF</b>	Degrees of freedom
<b>EN1317</b>	European standard
<b>GPS</b>	Global Position System
<b>GPRS</b>	General Packet Radio Service
<b>GSM</b>	Global System for Mobile Communications
<b>GYRO</b>	Gyroscope
<b>INT</b>	Interrupt
<b>I2C</b>	Inter Integrated Circuit

<b>I/O</b>	Input / Output
<b>KTK</b>	Karayolları Trafik Kanunu
<b>KGM</b>	KaraYolları Genel Müdürlüğü
<b>MCU</b>	Micro Controller Unit
<b>OBD-II</b>	On-board diagnostics
<b>PWM</b>	Pulse Width Modulation
<b>SCL</b>	Serial Clock Line
<b>SDA</b>	Serial Data Line
<b>SMS</b>	Short Message Service
<b>TÜİK</b>	Türkiye İstatistik Kurumu
<b>UART</b>	Universal Asynchronous Receiver Transmitter
<b>VCC</b>	Positive Supply Voltage

## 1. GİRİŞ

Dünya genelinde karayolu ulaşımı ulaşım türlerinden biri olarak en ön sırada yer almaktadır. Türkiye’de de karayolu ulaşımı hayatın zaruri bir parçasını oluşturmaktadır. Bundan dolayı, ülkemizde bir yerden bir yere yolcu ve yük transferinde halen en yüksek sırada yer almaktadır. Bu talep yüzünden karayollarında trafik akışı artmakta ve çoğunlukla büyükşehirlerde trafik daha vahim hale gelmektedir. Bunun sonucunda trafik kazalarında ciddi can ve mal kaybı oluşmaktadır [1].

Dünya Sağlık Örgütü'ne göre, trafik kazaları 2016 yılında dünya çapında yaklaşık 1.35 milyon insanın ölümüne neden olmuştur [2]. Karayolları Genel Müdürlüğü 2018 raporuna göre en çok ölüm ve yaralanmanın olduğu kaza oluş türleri sırasıyla;

- **Yoldan çıkma sonucu kaza sayısı:** Otoyol 533, Devlet Yolu 10 682, İl Yolu 237, Bağlantı Yolu 114 olmak üzere toplam kaza sayısı 14 910’dur.
- **Yandan Çarpışma Sonucu Kaza Sayısı:** Otoyol 588, Devlet Yolu 7 639, İl Yolu 1560, Bağlantı Yolu 130 olmak üzere 9 917’dir.
- **Arkadan Çarpma Sonucu Kaza Sayısı:** Otoyol 1 605, Devlet Yolu 6 222, İl yolu 721, Bağlantı Yolu 125 olmak üzere toplam 8 673’tür.
- **Devrilme/Savrulma/Takla Kaza Sayısı:** Otoyol 427, Devlet Yolu 4 774, İl Yolu 1 115, Bağlantı Yolu 101 olmak üzere 6 417’dir.
- **Engel/Cisim ile Çarpışma:** Otoyol 788, Devlet Yolu 3 044, İl yolu 361, Bağlantı Yolu 129 olmak üzere toplam 4 322’dir.

2000’den sonraki yıllarda, karayollarında ki kaza oranlarını azaltmak amacıyla, yeni yolların yapılmasına ve mevcut olan yolların iyileştirme çalışmalarına hız verilmiştir [3].

Ne yazık ki bu çalışmalar kazaların önemli derecede azalmasına katkı sağlayamamıştır. Dahası otomotiv sektörü güvenilir araçlar üretmek konusunda ilerlemesine rağmen karayolları TÜİK (Türkiye İstatistik Kurumu) verileri trafik kaza oranlarının hala yüksek olduğunu göstermektedir. Bu oran trafiğe kayıtlı araç sayısının artması ile günden güne artmaktadır. Türkiye’de karayollarındaki bu araçların ortalama olarak %67’si otomobil, %5’i otobüs ve %28’ini kamyon oluşturmaktadır [4].

Ulaşımının ana unsurlarını oluşturan bu araçlar çeşitli kazalara karışmaktadır TÜİK 2017 yılı verilerine göre 1 202 716 kazanın 182 669’u ölümlü ve yaralanmalı olup bunun 7 427’si ölümlü sonuçlanmıştır. Ayrıca bu ölümlerin 3 534’ü olay mahallinde, 3 893’ü de kaza sonrası gerçekleşmektedir [5].

Tablo 1.TÜİK yıllara göre ölen ve yaralanan kaza sayısı

Yıl	Toplam kaza sayısı	Maddi hasarlı kaza sayısı	Ölümlü, yaralanmalı kaza sayısı	Ölü sayısı			Yaralı sayısı
				Toplam	Kaza yerinde	Kaza sonrası	
2013	1 207 354	1 046 048	161 306	3 685	3 685	-	274 829
2014	1 199 010	1 030 498	168 512	3 524	3 524	-	285 059
2015	1 313 359	1 130 348	183 011	7 530	3 831	3 699	304 421
2016	1 182 491	997 363	185 128	7 300	3 493	3 807	303 812
2017	1 202 716	1 020 047	182 669	7 427	3 534	3 893	300 383

Tablo-1’de kaza sonrası ölüm oranının, kaza yerinde gerçekleşen ölüm oranından yüksek olduğu görülmektedir. Acil servise yapılan çağrılar içerisinde tıbbi olmayan gereksiz çağrılarının oranı %94 olarak saptandı. Ambulansların vakalara ortalama ulaşım süresinin 2012 yılında 8,6 dk., 2013 yılında ise 9,1 dk. olduğu belirlendi [6]. Bu verilerden de anlaşılacağı üzere başta sürücüler olmak üzere trafik ve yol güvenliğinin artırılabilmesi ve ölü sayılarının en aza indirgenmesi amacıyla kaza bildirim ve uyarı çalışmalarına verilen önem gitgide artmaktadır.

Gelişen teknoloji ile araçların uydu üzerinden eş zamanlı takip, bildirim sistemi ile ilgili pek çok çalışmalar yapılmakta ve bu çalışmalar Küresel Konumlandırma Sistemi (Global

Positioning System- GPS) üzerinden gerçekleştirilmektedir. Bu sistem sayesinde eş zamanlı takip, bildirim ve müdahale olanağı sağlanmaktadır. Bu konuda yapılan çalışmalar incelendiğinde 2009 yılında Küresel Konumlandırma Sistemi teknolojisi temel alınarak web tabanlı konum bildirim sistemi geliştirilmiştir. Bu sistemler aracılığıyla Ölüm ve yaralanma sayılarının düşebilmesi karayolu trafik kazasının gerçekleştiği konuma gerekli birimlerin hızlı bir şekilde ulaşması ile mümkündür. Bu çalışmada %94 oranında gerçekleştirilen gereksiz çağrılar önüne geçilerek otomatik bir sistem ile gerekli birimlerin olay mahalline en kısa yoldan sevk edilmesi amaçlanmaktadır. Gerek gerçek testlerin maliyetlerinin gerek temin edilen yüksek dayanıklılığa sahip olması gerekçesi ile aynı sistem bir prototip üzerinde modellenerek tasarlanmıştır. Aracın merkez bölgesinde bulunan bir yere sabitlenen bu sistem ile aracın 4 boyutlu uzayda kaza olması ihtimaline karşı sürekli olarak kazadan 5 dakika öncesinden kazanın sonlanmasına kadar olan her hareketi kayıt edilmektedir. Bu kayıt ile aracın kaza sonrası simülasyonu modellenmektedir. Bu çalışmanın diğer bir amacı ise ölçülen G-Kuvvetine bağlı olarak kazanın şiddetini belirlemek, bu şiddete göre kazayı kategorilendirilmesi ve kazanın yapıldığı konumu olay mahalline intikal edecek yetkililere bildirilerek öncesinde kaza şiddeti hakkında bilgi sahibi olmalarını sağlamaktadır.

### **1.1.Problem Durumu/ Konunun Tanımı**

Kaza bildirim ve uyarı sistemleri üzerine birçok ülkede farklı şekillerde cihazlar tasarlanarak farklı açılardan kaza algılama üzerine çalışmalar yapılmıştır. Bu alana yönelik yapılan araştırmaların birinde aracın kritik bölgelerine yerleştirilen basınç sensörlerinden gelen verilere göre kaza bildirim yapılmaktadır. Ayrıca araç içi kamera ile üç günlük kayıt tutulmaktadır [7]. Kaza öncesi uyarı sistemi tasarlanan bu çalışmada aracın hızı sürekli kontrol edilerek bölgelere göre sürücüyü uyarmakta ve bu bilgileri gerekli birimlere iletmektedir. Ayrıca kazaya karışan bu aracın konumunu acil servis birimlerine bildirmektedir [8]. Başka bir çalışmada ise sistemin çalışması için üç farklı sisteme ihtiyacı vardır. Bunlar araç ünitesi, kontrol ünitesi ve ambulansın geçişini kontrol etmek amacı ile trafik lambalarına yerleştirilerek bütün bir sistem oluşturulmuştur. Araç ünitesinde bulunan titreşim sensörlerinden gelen veri ile kazanın algılanması durumunda GPS ile bulunan koordinatlar, ambulansa bir GSM ile iletilerek ambulansın kaza yerine ulaşması sağlanmaktadır. Trafik lambası kontrol ünitesi ile ambulanda bulunan RF sistemi ile trafik ışık sistemi kontrol edilerek kaza yerine hızlı ulaşması amaçlanmıştır [9]. EN1317 Avrupa Standartlarına göre eşik değeri olarak 4g kuvveti belirlenmiştir. Android işletim sistemi ile OBD-II arasında gerçekleşen haberleşme protokolüne dayanılarak yapılan bu çalışmada



telefonun ivme ölçer sensöründen gelen veriler analiz edilerek kaza şiddetleri kategorilendirilmiştir. Yine ülkemizde yapılan başka bir android işletim sistemi ile aynı sistem oluşturulmuş olup analiz için K-star, J48 ve Navie Bayes algoritmaları kullanılmış olup weka programı üzerinde analiz edilmiştir [10]. 2007 yılında BMW firmasının yapmış olduğu bir yazılım ile 41.000 örnekleme 2.460 kişinin yani yüzde 6'sının hayatını kurtardığı anlatılan bu çalışma, sadece akıllı telefonda alınan veriler ile gerçekleştirilmiştir. Bu sistemin çalışma prensibi kurulan bir sunucu ile iletişime geçen telefonların sunucu tarafından kaza gerçekleşen yere acil müdahale birimlerini yönlendirmesi üzerine çalışmaktadır [11]. Doğrusal olan ve olmayan düşme hızına göre kaza algılama sistemi yapılan bu çalışmada doğrusal düşme ivme sadece z ekseninde gerçekleşirken, doğrusal olmayan düşme ise x, y ve z tüm eksenlerde gerçekleşmektedir. Doğrusal düşme olmadığı durumda ise bir önceki veri ile arasındaki ivme farkına bakılarak kaza olup olmadığına karar verilmiştir. Bu iki değer arasındaki ivme farkı 15,5 m/s<sup>2</sup> olarak alınmıştır. Bu sistemin sınırlılığı olarak ani fren ivmelenmesinin de kaza olarak bildirilmektedir [12]. Sosyal medyanın günden güne artan bir şekilde kullanımı ile sosyal medya üzerinden kaza konumunun paylaşımı üzerine yapılan bu çalışmada sadece kaza üzerine olmayıp deprem, sel ve suç işleme gibi olaylarda da yapılan android yazılımı ile facebook ve twitter üzerinden sosyal yayın yapılmaktadır [13].

### **1.2.Araştırmanın Amacı**

Kaza yerine ulaşım sürelerinin uzun olmasından yola çıkılarak hazırlanan bu çalışmada kaza sonrası ölüm oranlarını azaltmayı ve olay yerine acil müdahale ekiplerinin hızlı bir şekilde ulaşmasını mümkün kılacak bir sistemin tasarlanması amaçlanmaktadır.

Kazanın gerçekleştiğinin anlaşılması için ivmeölçer sensöründen filtrelenmiş verinin 4g kuvveti üzerinde olması beklenmektedir. Bu durumda kazanın gerçekleştiği varsayılmaktadır. Bu sistem ile kaza bölgesinde kimsenin bulunmaması durumunda dahi acil müdahale ekiplerinin olay yerine yönlendirilmesi ve daha varmadan kazanın durumu hakkında bilgi edinmelerini sağlamaktadır.

### **1.3.Araştırmanın Önemi**

Dünya Sağlık Örgütü'nün Global trafik kazaları araştırma raporuna göre dünyada 1,2 milyondan fazla insan her yıl trafik kazalarından hayatlarını kaybetmektedir [14]. Örneğin

2017 yılına baktığımızda KGM'nin yayınladığı rapora göre bu yıl içerisinde gerçekleşen toplam trafik kaza sayısı 1,202,716'dır. Ölü sayısı 7,427. Bu sayının 3,534'ü kaza yerinde gerçekleşmektedir. [15]. Aynı rapora göre; Almanya'da gerçekleşen Ölümlü ve Yaralanmalı Kaza Sayısı 305, 659 Ölü Sayısı 3.459 iken Türkiye'de ise Ölümlü ve Yaralanmalı Kaza Sayısı 182.669, ölü Sayısı 7,427'dir. Bu rakamlardan da anlaşılacağı üzere ülkemizde çok yüksek sayıda kaza oranları gerçekleşmekte ve daha kötüsü acil müdahale birimlerinin gecikmesi vb. gibi durumlardan dolayı ölü sayısı maalesef diğer Avrupa ülkelerine göre daha fazla olmaktadır. Bundan dolayı kaza yerine acil müdahalenin en hızlı ve etkili bir biçimde yapılmasının önemi artmaktadır. Diğer bir yandan da acil müdahale birimleri gereksiz yere kullanıldığı da görülebilmektedir. Bu alanda yapılan çalışmalar göstermektedir ki yapılan çağrılarının %94 ü sahte çağrı olmakla birlikte sistem gereksiz meşgul edilmekte ve böylece kazanın gerçekleştiği bölgeye gerekli birimler geç ulaşmaktadır. Bu süre ortalama 9.1dk olarak ölçülmüştür [6]. Yapılan bu araştırmalar sonucunda çalışmaların birçoğu akıllı telefonlar üzerinden yazılım ile gerçekleştirilmiş olup telefonun düşmesi sonucunda veya farklı durumlarda, ivme ölçerden sağlıklı veriler alınamayacaktır. Ayrıca bazı sistemlerde trafik lambalarına, acil müdahale birimleri araçlarına ve şahıs aracı ile birlikte bir server kurulması gerekmektedir. Bu sistemlerin hepsinin birlikte kurulması da ayrı bir maliyet gerektirmektedir. İçinde GPS, ivme ölçer ve GPRS, bulunan sistem ile maliyet düşmektedir. Bu sisteme uygulanan filtreleme ile parazit ivme verileri bertaraf edilerek sağlıklı bir sonuç elde edilmektedir. Daha sonrasında trafik kazasının nasıl gerçekleştiğine dair veriler simüle edilerek kaza hakkında daha detaylı bilgi sahibi olunabilmektedir.

#### **1.4.Sınırlılıklar**

Çalışmada koordinatlar uydulardan alınan veriler ile konum tespit edilmektedir. Fakat tünel ya da doğal engeller gibi yerlerde uydulardan veri alınamayacağı için bu gibi yerlerde kazanın gerçekleşmesi durumunda acil müdahale birimlerine ulaşamayabilecek ve vahim sonuçlara neden olabilecektir. Can kaybının çok olduğu kadar maddi zarar da çok büyük oranda olabilmektedir. Tünellerde kazalardan dolayı çıkabilecek herhangi bir yangının, onarımı ile birlikte Avrupa'da yılda 210 milyon Euro harcanmaktadır. Ayrıca kaza sonucu tünellerin kapanması yüzünden de ulaşım bedelleri yükselmekte, transfer süreleri uzamakta, rekabet gücü zayıflamakta ve maliyetler artmakta dahası çevreye kirliliğinde artışa neden olmaktadır. Bölge itfaiyesinin kaza mahalline varma süresi ortalama süresi 30 dakikayı aşabilmektedir [16]. Bunun sebebi tünellerin çoğunlukla dağlık ve kırsal bölgelerde inşa

edilmeleridir. Arduino ve Android Tabanlı Kaza Bildirim ve Uyarı Sistemi gibi sistemlerin bütün araçlarda bulunması ve araçların birbirleri otonom olarak haberleştiği bir sistem barındırması ile bu sınırlılıklar ortadan kaldırabilir.

## 2. GENEL BİLGİLER

Bu bölümde kaza bildirim ve uyarı sisteminde kullanılan kavramları detayları ile açıklanmıştır.

### 2.1.Trafik Kazası

Bir olayın trafik kazası sayılabilmesi için aşağıda bahsi geçen hükümler söz konusudur.

#### 2.1.1. Tanım

Karayolları Trafik Kanununun 2918 sayılı kanununun 3. maddesinde belirtildiği üzere bir kazanın trafik kazası sayılabilmesi için “Karayolu üzerinde hareket halinde olan bir veya birden fazla aracın karıştığı ölüm, yaralanma ve/veya zararlı sonuçlanmış olan olay” şeklinde tanımlanmıştır.

#### 2.1.2. Trafik Kazası Çeşitleri

Trafik kazaları oluş türlerine aşağıdaki gibi gruplara ayrılmaktadır

- a) Bir veya birden daha fazla aracın çarpışması neticesinde;
- b) Kaza sonucunda çarpışan araçların peş peşe farklı diğer araçların da dâhil olması ile zincirleme kaza;
- c) Şoför hatasından meydana gelen tek yanlı kaza;
- d) Tek yanlı, yalnız yol kusurlarının etken olması ile;
- e) Motorlu aracın yayalara çarpması biçiminde;
- f) Motorlu aracın, taşıdığı yolculara zarar vermesi biçiminde;
- g) Motorlu aracın hayvanlara, nesnelere ve tesislere çarpması biçiminde
- h) Aracın düzenli bakımının yapılmamasından kaynaklanan teknik arıza biçiminde;
- i) Aracın imalât hatasından kaynaklanan teknik arıza biçiminde;
- ı) Demiryolu geçidinde tren ile kara motorlu aracının çarpışması biçiminde;
- k) Karşı konulmaz doğa olayı (mücbir sebep) nedeniyle gerçekleşebilir.

3- Uygulanan yasalara göre trafik kazası türleri

Olay, 2918 sayılı Karayolları Trafik Kanunu'nda tanımlanan “trafik kazası” biçiminde gerçekleşmekle birlikte, aynı zamanda iş kazası sayılmakta veya yolcu ve

yük taşıyanın sorumluluğunu gerektirmekte ise, uygulanacak yasa hükümlerine göre trafik kazaları üçe ayrılır. Bunlar:

- a) Olağan trafik kazaları (2918 sayılı KTK)
- b) Trafik-iş kazaları (2918 sa. KTK, 4857 sa. İş Kanunu, Sosyal Güvenlik Yasaları)
- c) Yolcu ve yük taşıma kazaları (2918 sa. KTK., 4925 sa. Karayolu Taşıma Kanunu ve Türk Ticaret Kanunu'nun taşıma ile ilgili hükümleri)

Ülkemizde Trafik kazalarını ilk yardım talebinde bulunma ve olayı resmî kurumlara bildirmede 112 Acil çağrı numarası kullanılmaktadır. Yapılabilecek bu çağrılara en kısa sürede acil yardım ulaştırabilmesi kazazedenin hayatta kalma şansını kuşkusuz önemli ölçüde artırmaktadır. Olay mahalline yardımların en kısa zamanda ulaşımı için olay yerinin konumunun en kısa ve en doğru şekilde tespit edilebilmesinde kullanılan GPS gibi sistemleri kullanmakta olan bu gibi kurumların en iyi şekilde hizmet etmesini sağlamaktadır [17]. Kullanılan çağrı numaralarının çokluğu karışıklığa yol açabilecek ve her anın önemli olduğu bir kaza anında zaman kaybına yol açabilecektir. Bu nedenle Avrupa Birliği ilk yardımın talebinin basit hale getirilmesi ve bu talebi artırmak amacıyla her türlü yardım talebi için 112 numaralı kullanımını standart olmasını zorunlu kılmaktadır. Buna ek olarak, ülkemizde acil sağlık hizmetleri yönetmeliğine göre, 112 çağrı merkezi ve diğer destek ekipleri çağrı alma ve değerlendirme süresi 4 dakika olarak açıklanmaktadır [18].

## 2.2. Yerçekimi Kuvveti

Kütle çekimi manasına gelen G kuvveti “gravitational-yer çekimi” kelimesinden gelmektedir. İvmölçer ile ölçülebilen hızlanma ya da ivmelenme değerlerine G kuvveti denir. Şöyle ki bir cismin rastgele bir yönde, bu cismin kendisine uygulanan bir güç yardımıyla hızlanarak ve yavaşlayarak ağırlık ‘weight’ değeri meydana getirmesi G kuvveti sayesinde. Ağırlığın oluşabilmesini sağlamak için maruz kalınan hızlanmanın bir dirençle karşı karşıya gelmesi gerekmektedir. Direnç kaynağı hava, katı veya sıvı yüzeylerden herhangi biri olabilir. Söz konusu bir uçak düşünülüğünde, uçak için havanın kendisine uyguladığı direnç sonucunda G kuvvetlerine maruz kalabilmektedir. Herhangi bir arabayı ele aldığımızda ise, havanın ve yerin uyguladığı direnç kuvvetlerinin bileşeni yönünde G kuvveti üretmektedir.

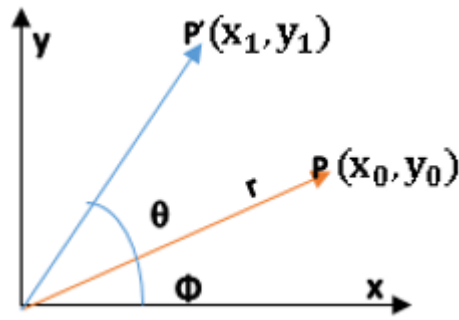
Kütle ve maruz kalınan G kuvveti ile birlikte ağırlık olarak geri döner

Bir otomobil kazası esnasında sürücüyü ön camdan fırlatan faktör, G kuvvetinden meydana gelmektedir. Kaza esnasında oluşan G kuvveti, çok güçlü ya da güçsüz olmak fark

etmeksizin, sürücüyü ve beraberindeki yolcuyla bir yere tutunarak kurtulmalarına imkân vermez.

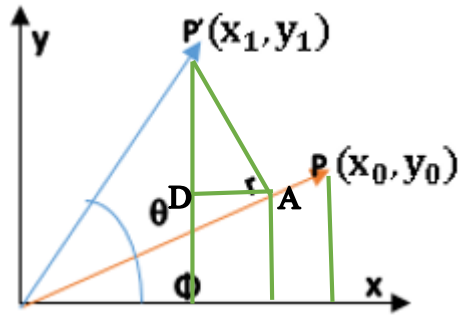
### 2.3.Dönme (Rotasyonel) Matrisi

Katı bir cismin Öklid uzayında dönme etkisini görmek için kullanılan denklem sistemi için bu tanımlama yapılmaktadır. Şekil 1 de görüldüğü gibi  $\Phi$  açısına sahip noktalar kümesinden oluşan bir  $r$  doğrusu  $P(x_0, y_0)$  noktasından  $P'(x_1, y_1)$  noktasına  $\theta$  açısı ile saat yönünün tersine döndürülmesi istenmektedir.



Şekil 1. Döndürme Matrisi

Bu döndürme işlemini gerçekleştirmek için aşağıda ki matematiksel denklemlere gereksinim duyulmaktadır.



Şekil 2. Döndürme Matrisi

$OP = r$ ,  $EP = y_0$ ,  $PE = x_0$  eşitliklerini denklemlerde yerine koyarsak

$\sin \phi = EP/r$  Burada  $EP=y_0$ ,  $OE=x_0$  için yerlerine koyduğunda

$$y_0 = r * \sin \phi \quad (1)$$

$$x_0 = r * \cos \phi \quad (2)$$

Eşitlikleri elde edilmektedir.

Diğer  $OP'A$  dik üçgeni için aynı denklemler uygulandığında aşağıda ki eşitlikleri elde edilmektedir.

$$OA = r * \cos \theta \quad (3)$$

$$AP' = r * \sin \theta \quad (4)$$

$$x_1 = OB - AD \quad (5)$$

$$\cos \phi = \frac{OB}{OA} \quad (6)$$

(3) nolu eşitliği (6) nolu eşitlikte yerine koyulduğunda (7) nolu eşitlik elde edilmektedir.

$$OB = r * \cos \phi * \cos \theta \quad (7)$$

(2) nolu eşitliği (7) nolu eşitlikte yerine koyulup düzenlendiğinde (8) nolu eşitlik elde edilmektedir.

$$OB = x * \cos \theta \quad (8)$$

(5), (6) ve (7) eşitlikleri AD uzunluğu için uygulandığında (9) denklemi elde edilmektedir.

$$AD = y_0 * \sin \theta \quad (9)$$

(5) numaralı eşitliği bulduğumuz (8) ve (9) denklemlerine göre yeniden düzenlendiğinde (10) denklemi elde edilmektedir.

$$x_1 = x_0 * \cos \theta - y_0 * \sin \theta \quad (10)$$

Yukarıda ki işlemleri  $y_1$  için uyguladığımızda (11) nolu denklem elde edilmektedir.

$$y_1 = x_0 * \sin \theta + y_0 * \cos \theta \quad (11)$$

Bu denklemleri matris formunda yazdığımızda (12) denklemleri elde edilmiş olur.

$$\begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} * \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} \quad (12)$$

Yukarıda ki matris formu iki boyutlu x-y kartezyen düzleminde bulunan noktalar kümesinin istenilen noktaya gelmesi için gerekli olan döndürme matrisidir [19].

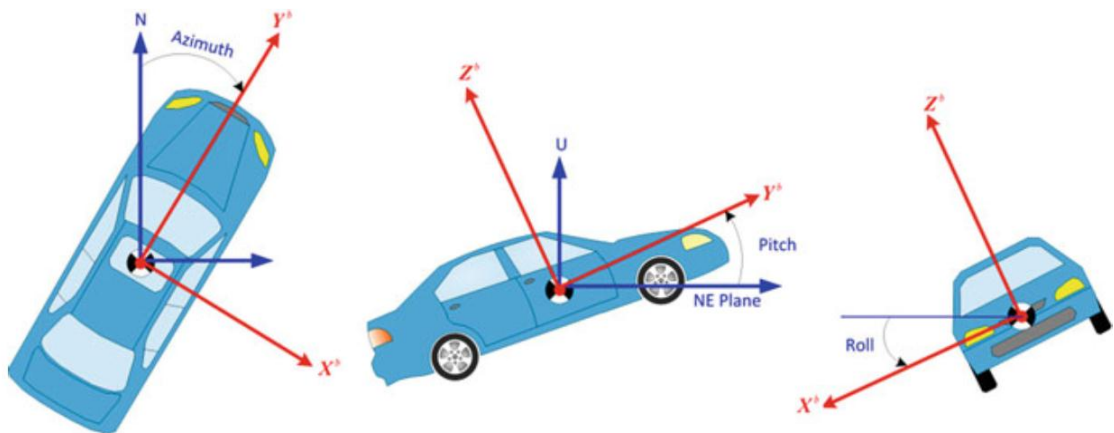
Her nesnenin uzayda yunuslama (pitch), yuvarlanma (roll) ve yönelme (yaw-azimuth) olmak üzere üç farklı hareketi bulunmaktadır [20].

Bu döndürme matrisi için üçüncü bir boyut sisteme dahil edildiğinde ayrı ayrı yunuslama (pitch) açısını hesaplamak için (13) numaralı denklemi, yuvarlanma (roll) açısını hesaplamak için (14) numaralı denklemi ve yönelme (yaw) açısını hesaplamak için (15) numaralı Euler denklemleri kullanılmaktadır [21].

$$R_x(\Phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\Phi & \sin\Phi \\ 0 & -\sin\Phi & \cos\Phi \end{bmatrix} * \begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} \quad (13)$$

$$R_y(\theta) = \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix} * \begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} \quad (14)$$

$$R_z(\Psi) = \begin{bmatrix} \cos\Psi & -\sin\Psi & 0 \\ -\sin\Psi & \cos\Psi & 0 \\ 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} \quad (15)$$



Şekil 3. Aracın üç boyutlu uzayda hareketi

Aracın yere göre X, Y ve Z eksenlerinde ki açısal konumu için Euler dönüşüm matrisinden elde edilen formül kullanılmaktadır [22]. Araçta oluşan toplam dönme açısının yere göre yani koordinat sistemine göre çevrimi ise bu üç yönde oluşan açısal konum matrislerinin (13), (14) ve (15) çarpımı ile elde edilmektedir [23]. Bu matrisleri (16) nolu eşitlikte ki gibi çarpıldığında nihai olarak Öklid Uzayında rijit bir cismin Euler açısal döndürme matrislerine göre (17) nolu denklem elde edilmektedir [24].

$$R = R_x(\Phi) * R_y(\theta) * R_z(\Psi) \quad (16)$$

$$R = \begin{bmatrix} \cos\theta\cos\Psi & \cos\theta\sin\Psi & -\sin\theta \\ \sin\Phi\sin\theta\cos\Psi - \cos\Phi\sin\Psi & \cos\Phi\cos\Psi + \sin\Phi\sin\theta & \sin\Phi\cos\theta \\ \sin\Phi\sin\Psi + \cos\Phi\cos\Psi & \cos\Phi\sin\theta\sin\Psi - \sin\Phi\cos\Psi & \cos\Phi\cos\theta \end{bmatrix} \quad (17)$$

#### 2.4. GPS (Global Position System-Küresel Yer Belirleme Sistemi) Çalışma Prensibi

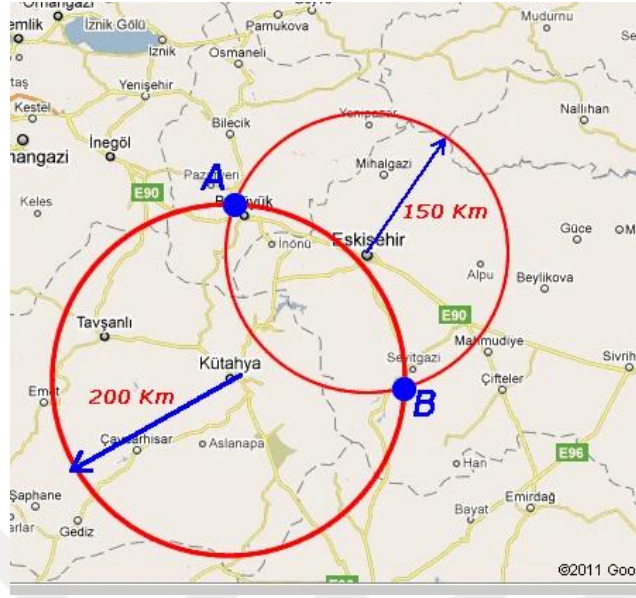
1980’li yıllara kadar askeri amaçlar için kullanılan küresel yer belirleme sistemi 1980’li yıllardan sonra ise özel amaçlar ile kullanımı sayesinde günlük yaşamı kolaylaştıran cep telefonlarımıza kadar birçok projeye dahil edilmiştir. Bu proje kapsamında da GPS modülü ile konum belirleme işlemi gerçekleştirilmiştir. Bu yüzden öncelikle GPS sisteminin nasıl çalıştığından bahsedilmektedir. GPS sistemleri dünya etrafında 7/24 dönüşünü sağlayan uydular vasıtasıyla GPS sistemine dahil olan herkesin tarafından istifade edilebilmektedir. Her bir uydunun dünya etrafında dönüş süresi ortalama günde iki kezdir. Bir kimsenin GPS sistemi ile enlem ve boylam bilgisinin bulunabilmesi için en az üç uyduya ihtiyaç vardır. Şu anda dünya etrafında herhangi bir konum bilgisine sahip olabilecek en az dört uydu bulunmaktadır. Konum bilgisinden başka yükseklik bilgisi için de en az bir uyduya daha ihtiyaç vardır. Bu durumda enlem, boylam ve yükseklik için en az 4 uydu tarafından bilgi alınmalıdır. Uydulardan gelen bilgiler iki çeşittir. Bunlardan biri askeri bir sinyal olan 1227.60 Mhz sinyal bandına sahip iken telefonlarımızdan araç navigasyon sistemlerine kadar birçok alanda kullanılan sivil haberleşme bandı ise 1575.42 Mhz bandında çalışmaktadır. Her biri bir uydudan alınan veriyi temsil etmekte olan aşağıdaki görsellerde konum bilgisinin nasıl bulunduğu görülmektedir [25]. GPS sisteminin üç uydu ile konumun nasıl bulunduğu şöyle örneklenebilir. Örneğin bir GPS modülünün Eskişehir’e 150 km



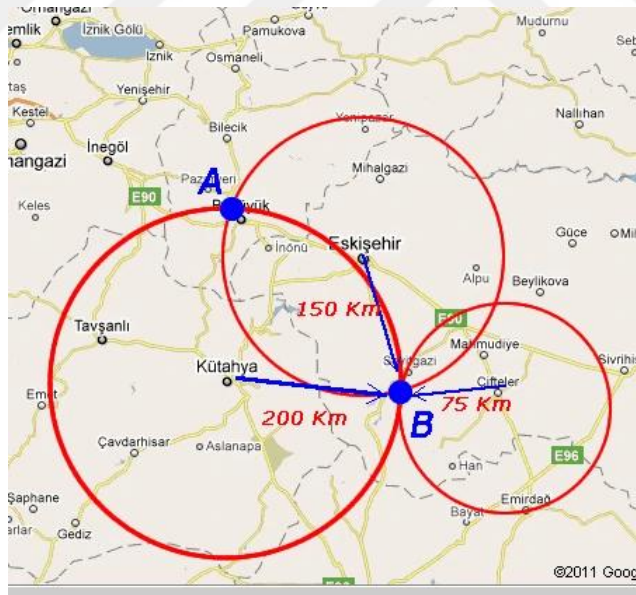
uzaklıkta olduğu bilindiğine göre Eskişehir'i merkezine alan 150 km yarıçapında bir çember çizilir. Konumu bulunmak istenen GPS cihazının çember üzerinde bir noktada olduğu bilinmektedir. Fakat bu veri konumunun GPS cihazının konumunun tam tespiti için yeterli değildir. 1 numaralı resimde bir uydudan alınan veriye göre bu çember oluşturulmuştur. Yine bu GPS modülünün Kütahya'ya 200 km mesafede olduğu bilinmektedir. Bunun içinde merkezi Kütahya olan bir çember çizilir. İlk çizilen çemberi iki noktadan kesen bu ikinci çember ile iki nokta oluşur. GPS modülünün bu iki konumdan birinde olduğu bilinmektedir. Bu kesin konum için yeterli değildir. Bu ikinci uydu verisini temsil etmektedir. Çifteler konumuna 75 km mesafede olduğu bilinen üçüncü bir çember çizilerek kesin nokta olan B noktası tespit edilir. Uydular konum bulmada da bu şekilde çalışmaktadır. Yükseklik konumu için ise dördüncü bir uydu verisi gerekmektedir.



Resim 1.1. Uydu konumunu bulma [25]



Resim 2.2.Uydu konum bulma [25]

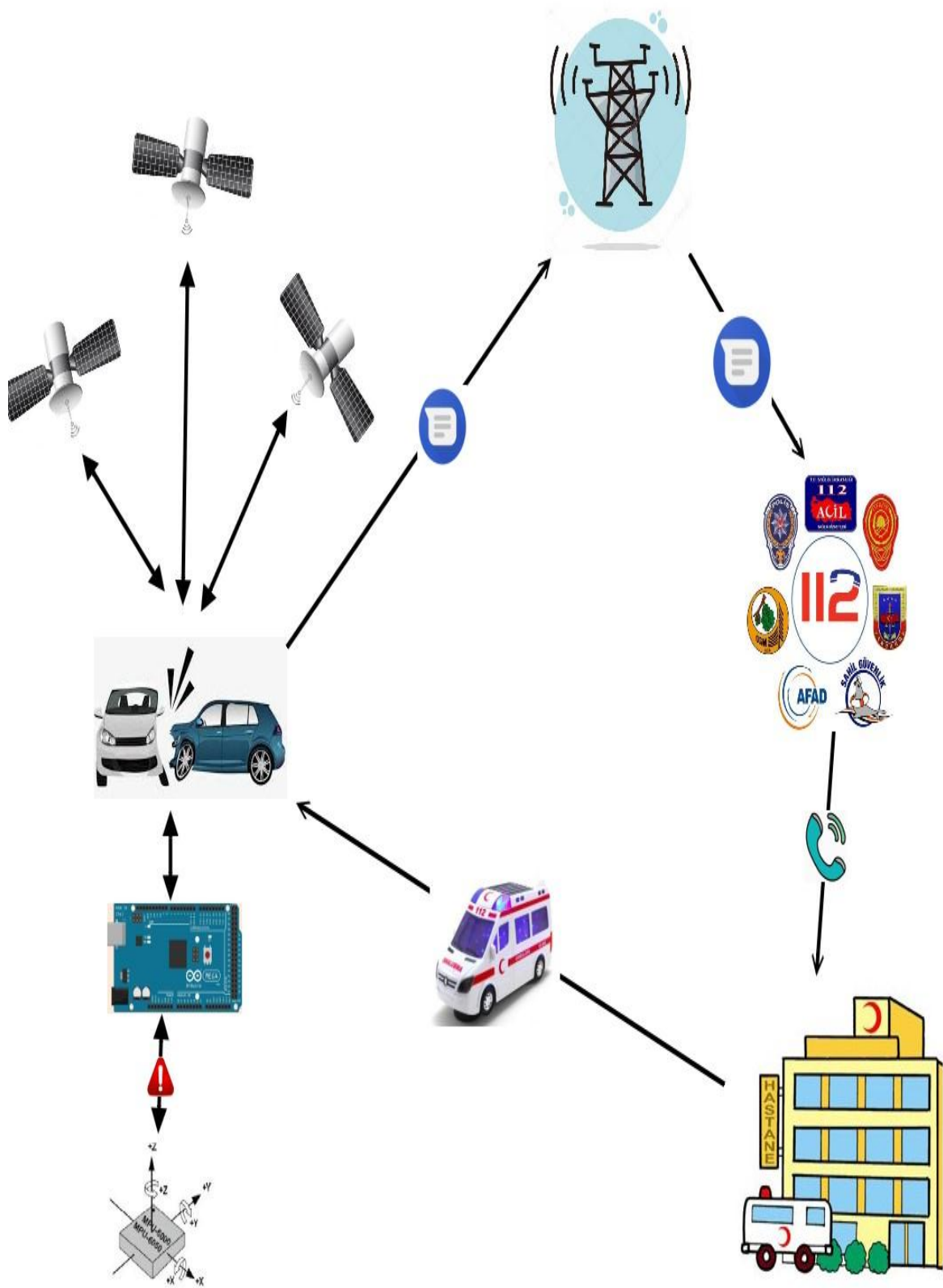


Resim 3.3.Uydu konum bulma [25]

### 3. SİSTEM TASARIM VE ÇALIŞMA PRENSİBİ

#### 3.1.Kaza Bildirim ve Uyarı Sisteminin Çalışması

Sistem Blok diyagramında ivme ve jiroskop olmak üzere iki hareket bileşenine sahip olan sistem x, y ve z düzleminde aracın her hareketini kontrol etmektedir. Otomobilin Eğim açısından 45°'den fazla dönüş gerçekleştiğinde veya 4g kuvvetinden fazla g kuvvetine maruz kaldığında kaza gerçekleşmiş sayılmaktadır. Bu esnada sürücünün, daha önceden kurulmuş olan android yazılımına telefonunun bluetooth vasıtası ile bağlanması halinde kaza olup olmadığını soran bir bildirim gelmektedir. Bu bildirim 1 dakika içerisinde cevap verilmezse en az 3 uydudan alınan veri ile dünya üzerinde konumu tespit edilen araç, sistem içerisinde barındırılan GPRS (General Packet Radio Service-Radyo Paketi Genel Servisi) sayesinde en yakın acil müdahale birimine konumunu ve kaza şiddetini mesaj ile bildirmektedir. Bu mesaj içeriğinde resimde görüldüğü üzere kazanın konumu, türü ve şiddeti belirtilmektedir. Ayrıca android sistemi üzerinde yazılmış olan yazılım ile de kaydedilmiş olan kazadenin yakınlarına kazanın gerçekleştiği konum hakkında bildirim gitmektedir.



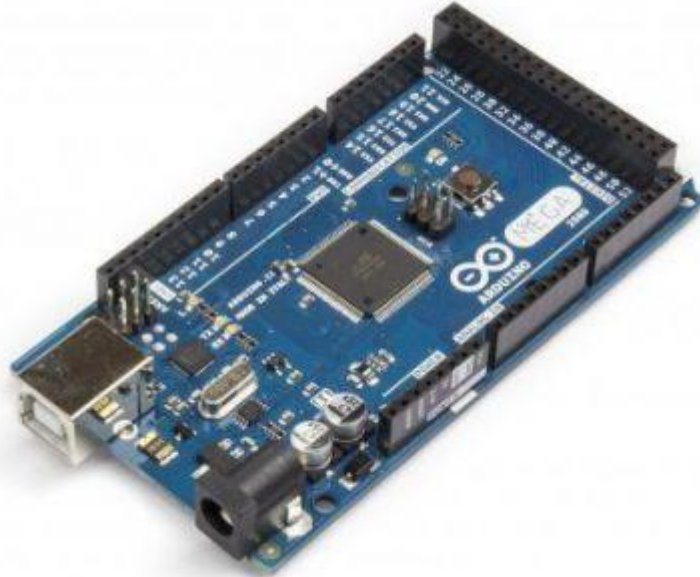
Şekil 4. Sistem blok diyagramı

### 3.2.Sistemin Elektronik Tasarımı ve Birimlerin Tanıtılması

Sistemin çalışabilmesi için kullanılan elektronik birimlerin tercih edilme sebepleri ve çalışma prensipleri aşağıda data sheetlerinden alınan bilgiler doğrultusunda açıklanmaktadır.

#### 3.2.1. Mikrodenetleyici

Çalışmada Arduino Mega Mikrodenetleyici kart tercih edilmiştir. Saniyelerin bile çok önemli olduğu bu çalışmada veri işleme hızı ve portlarının çokluğu sebebi ile bu açık kaynak kodlu kart tercih edilmiştir. Sistemin içerisinde barındırdığı birden fazla seri haberleşme portları da diğer cihazlar ile kolayca haberleşmesine imkân sağlamaktadır. Bu haberleşme portları bilgisayara veri aktarmada Seri1 haberleşme portu kullanılmıştır. GPRS ile haberleşmede ise Seri3 portu kullanılırken GPS sistemi ile haberleşmede Seri2 portu kullanılmıştır. Bunlara ek olarak 2 adet bluetooth sistemde bulunmaktadır. Bu bluetooth'ların bir tanesi kullanıcının android ara yüzü ile iletişim kurarken diğer sistemin simülasyonunu rahatça görebilmek için bilgisayar ile bağlantı kurmuştur. Haberleşmede tüm cihazlar için karışıklığa sebep olmaması için standart 9600 baud rate haberleşme frekansı tercih edilmiştir.



Resim 4. Mikrodenetleyici kart

Karta ait teknik detaylar [26];

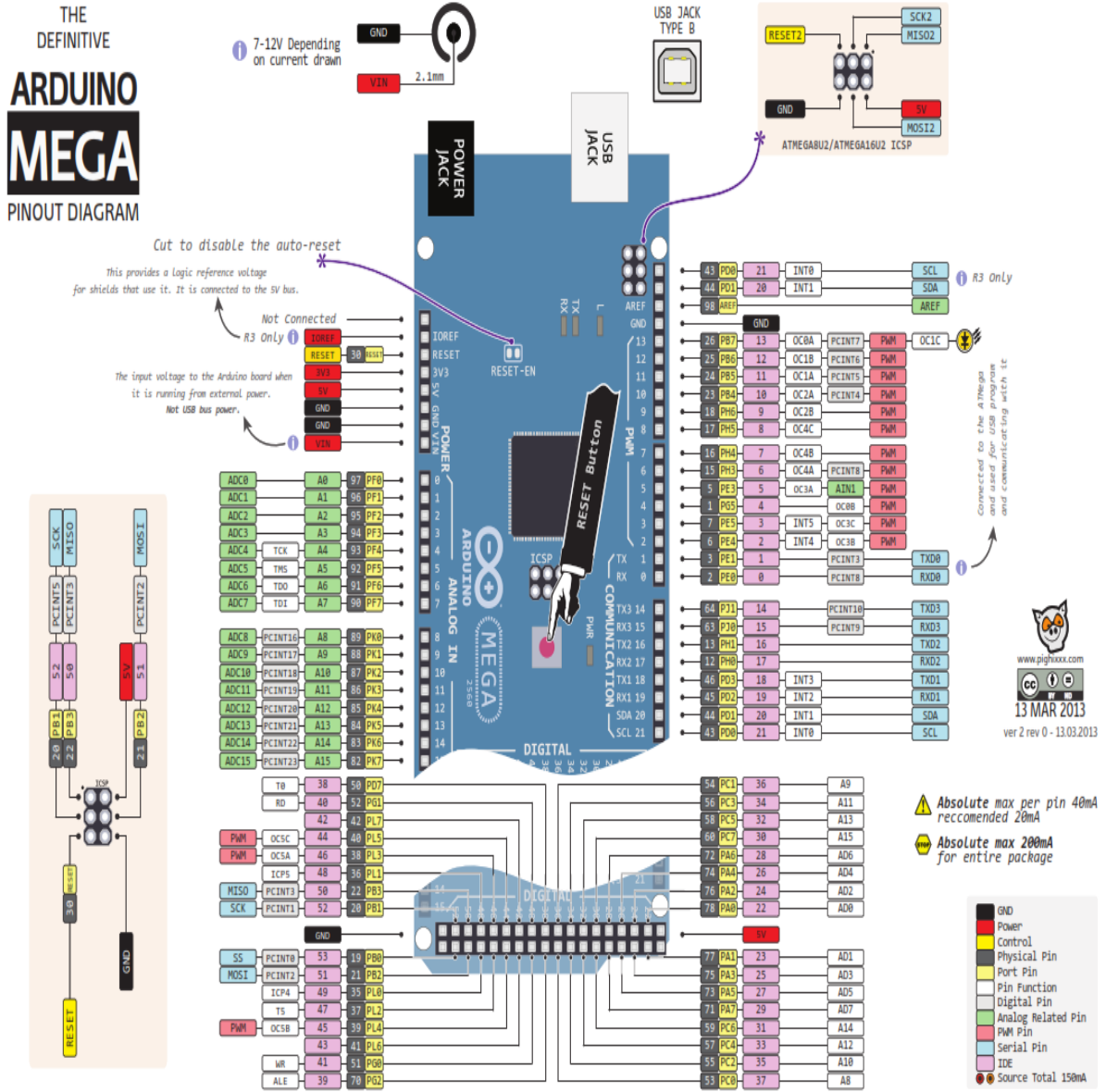
- Mikrodenetleyici Atmega2560
- 5V Çalışma Gerilimi,
- 7-12V Giriş Gerilimi
- Giriş Gerilimi (limit) 6-20V
- 54 Adet Dijital I/O Pini
- 15 Adet PWM Pini
- 16 Adet Analog Giriş Pini
- Flash Hafıza 256 KB
- 8 KB kadarı bootloader
- 4 KB EEPROM
- 16 MHz Kristal Osilatör
- Uzunluk 101.6 mm
- Genişlik 53.4 mm
- Ağırlık 36 g

Tablo 2. Mikrodenetleyici kart çalışma gerilimi

Device	Flash	EEPROM	RAM	General Purpose I/O pins	16 bits resolution PWM channels	Serial USARTs	ADC Channels
ATmega640	64KB	4KB	8KB	86	12	4	16
ATmega1280	128KB	4KB	8KB	86	12	4	16
ATmega1281	128KB	4KB	8KB	54	6	2	8
ATmega2560	256KB	4KB	8KB	86	12	4	16
ATmega2561	256KB	4KB	8KB	54	6	2	8



Arduino Mega 2560 Mikrodenetleyici kartın pin dağılımı nolu diyagramda gösterilmektedir [27].



Şekil 5. Arduino Mega 2560 pin dağılımı diyagramı

### 3.2.2. GPS Genişletme Kartı

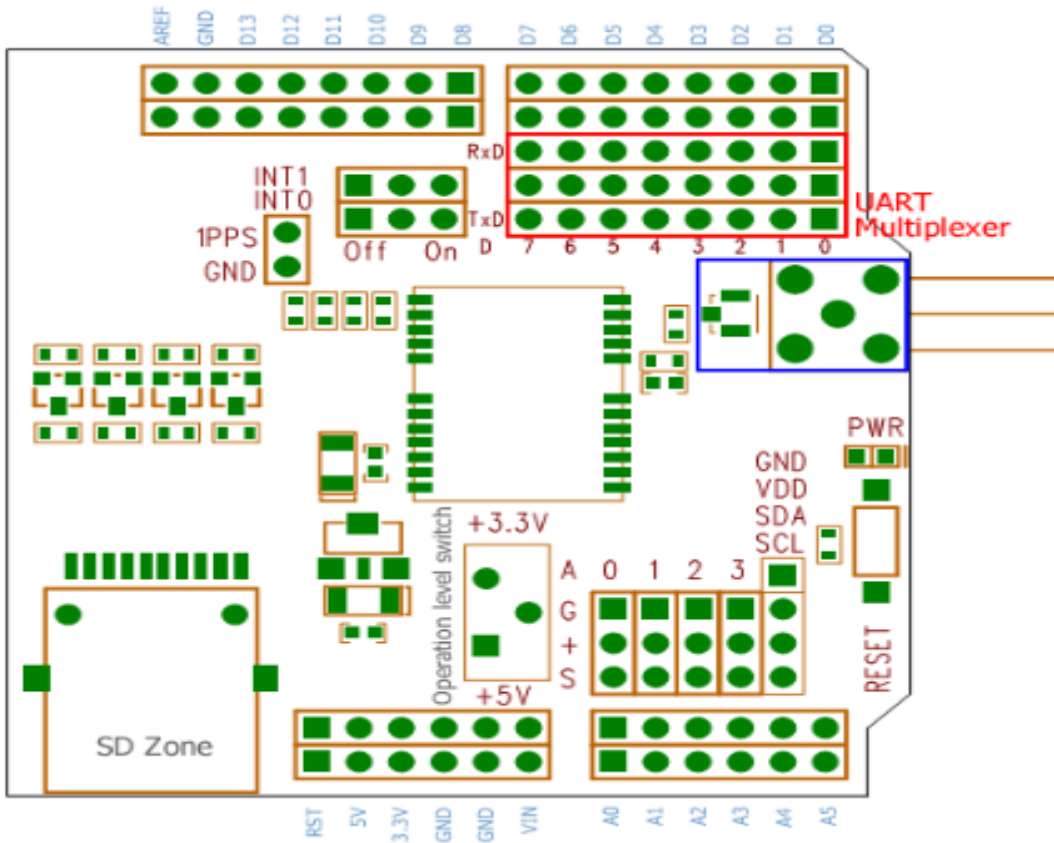
GPS Genişletme Kartından, GPS haberleşme protokollerine sahip bir GPS modülü üzerinden konum bilgisi alınmaktadır. Sistemde kullanılan GPS modülü Arduino Mega 2560 ve Arduino Uno kartları ile uyumludur. GPS modülü içerisinde mikro sd kart birimi barındırmaktadır. Uydudan alınan veriler mikro sd karta kayıt edilebilmektedir. Yüksek hassasiyetli aktif bir anten de bulundurmaktadır. Bu anten ile diğer cihazlara göre metrik değişimler dahi algılanmaktadır. Haberleşme için UART arabirime sahiptir. Düşük sinyal seviyesi ile hızlı konum sabitleme özelliğine sahiptir. 38400 haberleşme frekansına sahiptir. Sanal haberleşme portu oluşturularak 9600 baud rate ile haberleşebilmektedir. Sistemde tüm arabirimlerin haberleşme frekansı 9600 olarak tercih edilmiştir. Kartın elektriksel karakteristik tablosu tablo ile gösterilmektedir [28].

Tablo 3. GPS Modülü elektriksel karakteristik durumu

Specification	Min	Type	Max	Unit
Power Voltage	4.5	5	5.5	VDC
Input Voltage VH:	3	-	5.5	V
Input Voltage VL:	-0.3	0	0.5	V
Baud rate		38400		bps

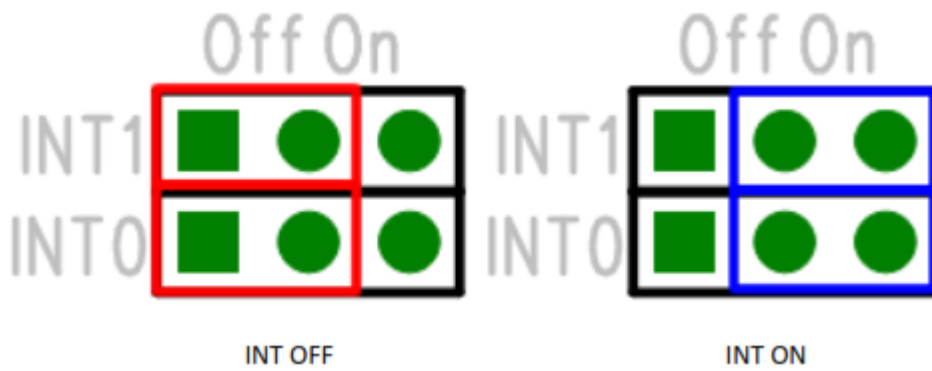
Seri haberleşme için kart üzerindeki UART Multiplexer bölümünden jumperlar ile Rx Tx haberleşme portları ayarlanabilmektedir.





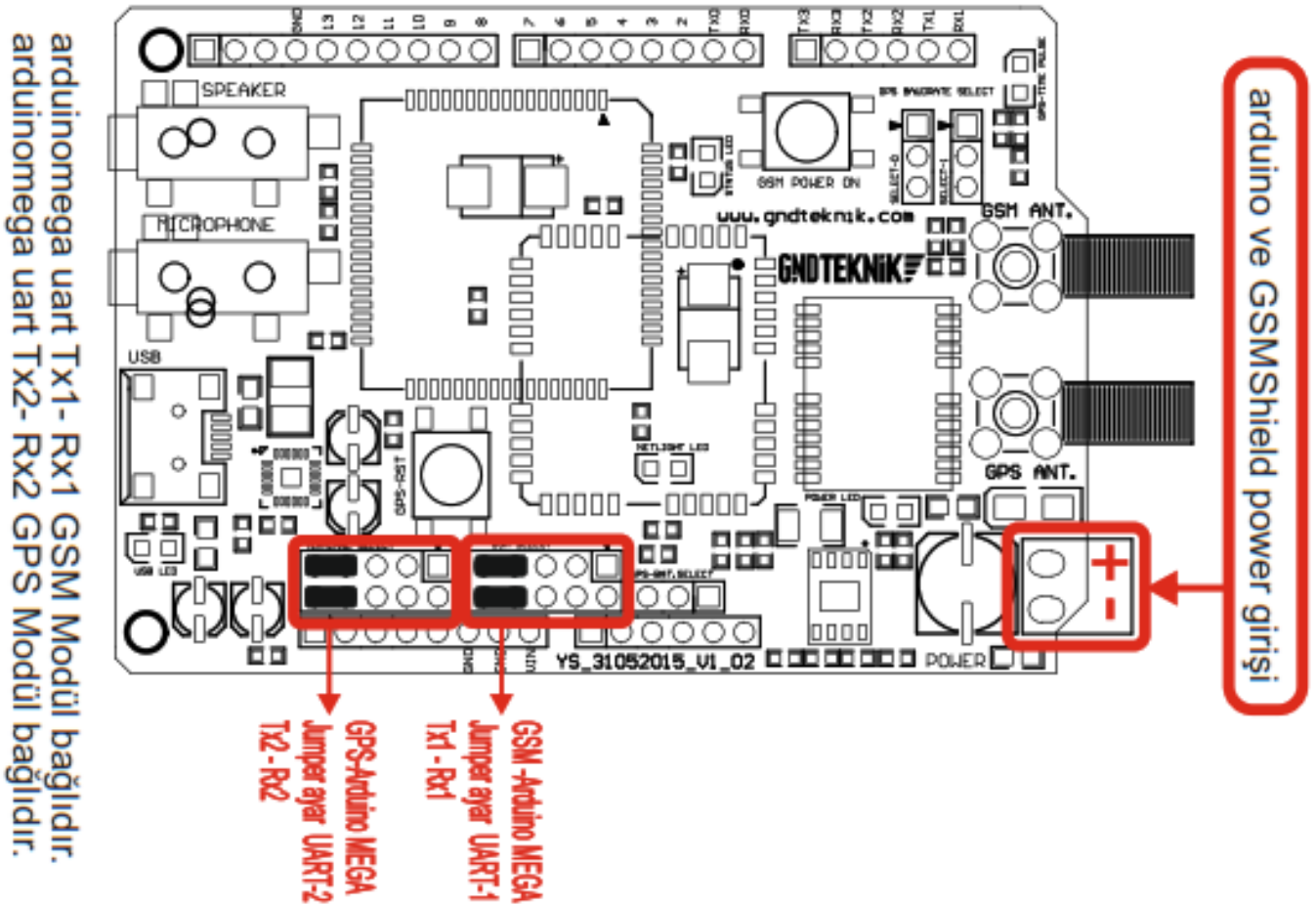
Resim 5. GPS Modülü pin dağılımı diyagramı

Harici GPS modülünden de veri alınmak istenirse bunun için de kesme pinleri (INT0,INT1) OFF konumundan ON konumuna alınmalıdır [28].



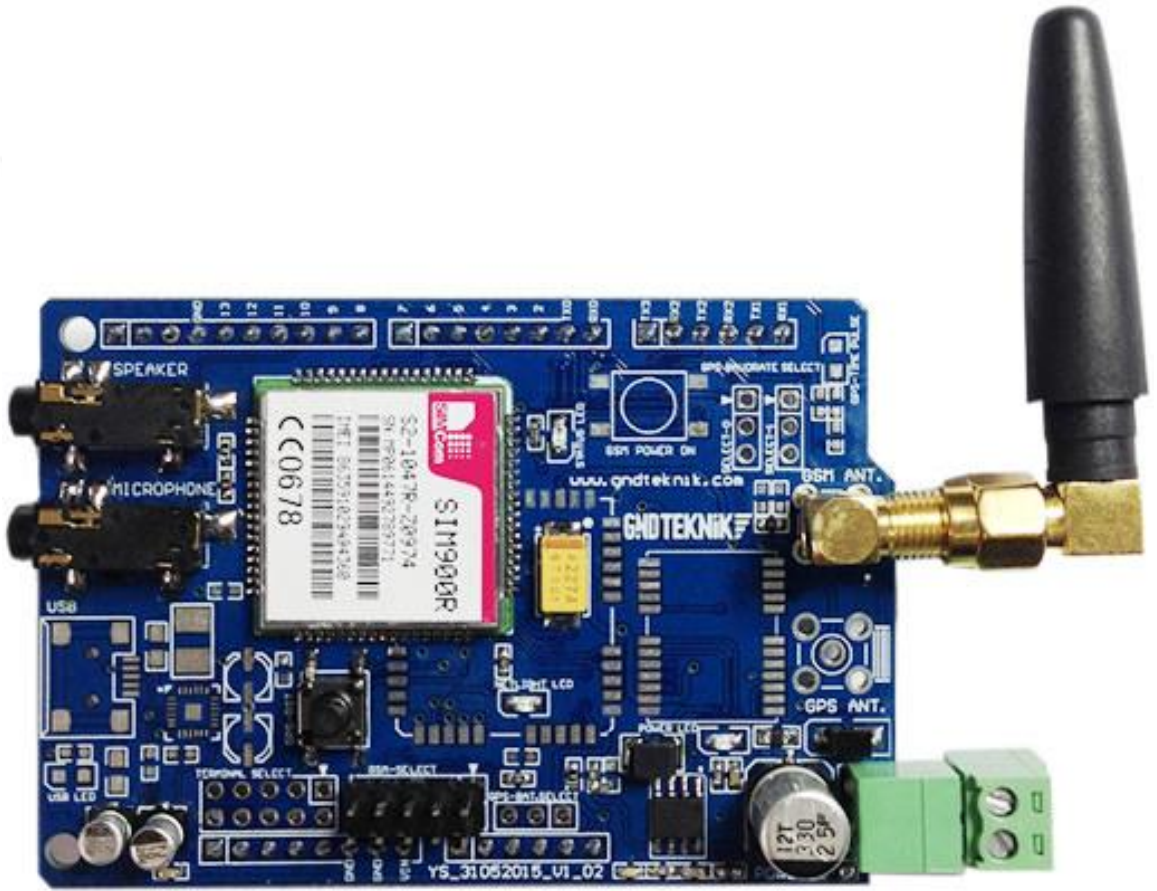
Resim 6. GPS Modülü jumper ayarı

Simcom Sim900 gsm modül özellikleri; SIMCOM firmasının üretmiş olduğu SIMxxx serisinin SIM900 modülü olan bu entegre sayesinde herhangi bir telefona gerek kalmadan teknik detayları verilen bu kart ile çağrı gönderme-alma ve mesaj gönderme-alma gibi işlemler AT (Attention Command-Dikkat Komutu) Komut setleri ile mümkün olmaktadır. Kart sadece GSM (Global System for Mobile Communication-Mobil İletişim için Küresel Sistem) modülü içermekte olup GSM ve GPS modüllerinin birlikte olduğu bu çalışmada ki tasarımda mevcuttur.



Resim 7. GSM Modülü jumper ayarlamaları

GPS'in daha hassas veriler sađlayan farklı bir kart tercih edilmiştir. Bu karta ait jumper ayarları aşağıda gösterilmektedir.



Resim 8. GSM Modülü üst görünüş

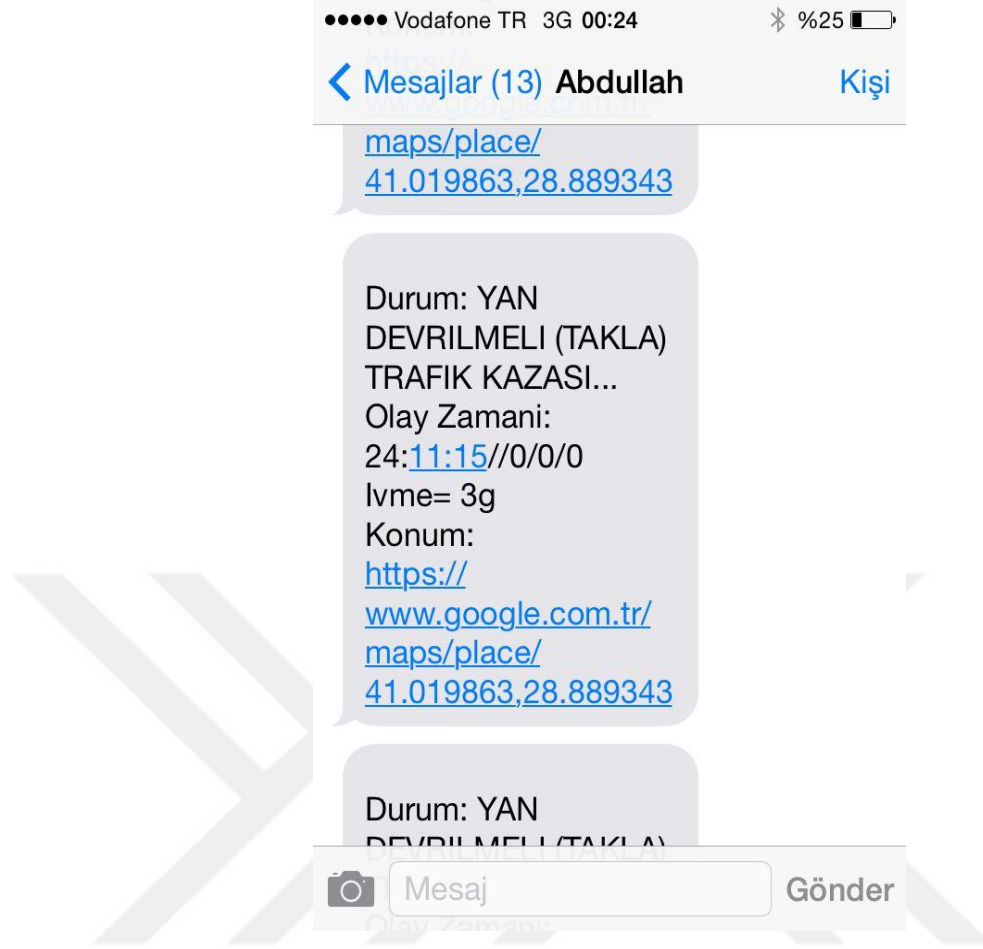
SIM900 GSM Shield Teknik Detaylar [29];

- “Quad-Band 850 / 900/ 1800 / 1900 MHz
- GPRS multi-slot class 10/8
- GPRS mobile station class B
- Compliant to GSM phase 2/2+
- Class 4 (2 W @ 850 / 900 MHz)
- Class 1 (1 W @ 1800 / 1900MHz)
- Control via AT commands – Standard Commands: GSM 07.07 & 07.05 | Enhanced Commands: SIMCOM AT Commands.
- Short Message Service – so that you can send small amounts of data over the network (ASCII or raw hexadecimal).
- Embedded TCP/UDP stack – allows you to upload data to a web server.
- RTC supported.
- Selectable serial port.
- Speaker and Headphone jacks
- Low power consumption – 1.5mA (sleep mode)
- Industrial Temperature Range – -40°C to +85 °

Mesaj gönderme ve alma, arama yapmak için kullanılan AT-Komut seti konfigürasyonu [30]; Mesaj gönderme ve alma için AT+CMGF=1 komutu kullanılarak sistem mesaj gönderme için aktif edilmiştir. Gönderilmesi istenilen numara String veri tipi tanımlı "+905xxxxxxxx" formatında olup Seri1 portuna bluetoothtan gelen String veriler iletilecek mesaj halinde düzenlendikten sonra ctrl+z komutu ile mesaj gönderilmektedir. ctrl+z ascii kodu karşılığı olan char tipinde 26 verisi ile mesaj istenilen adrese iletilir [30].

Tablo 4. Mesaj gönderimi için AT komut seti

SIM300	SIM900
AT+CMGF=1	AT+CMGF=1
OK	OK
AT+CSCS="GSM"	AT+CSCS="GSM"
OK	OK
AT+CSMP=17,255,0,241	AT+CSMP=17,255,0,241
OK	OK
AT+CMGS="13621682959"	AT+CMGS="13621682959"
>123456789012345678901234567890123456	>123456789012345678901234567890123456789
7890123456789012345678901234567890123	0123456789012345678901234567890123456789
4567890123456789012345678901234567890	0123456789012345678901234567890123456789
1234567890123456789012345678901234567	0123456789012345678901234567890123456789
8901234567890A	0A
+CMGS: 91	ERROR
OK	
Difference	SIM900 supports sending SMS of 160 bytes maximum for GSM code and 70 characters maximum for UCS2 code.

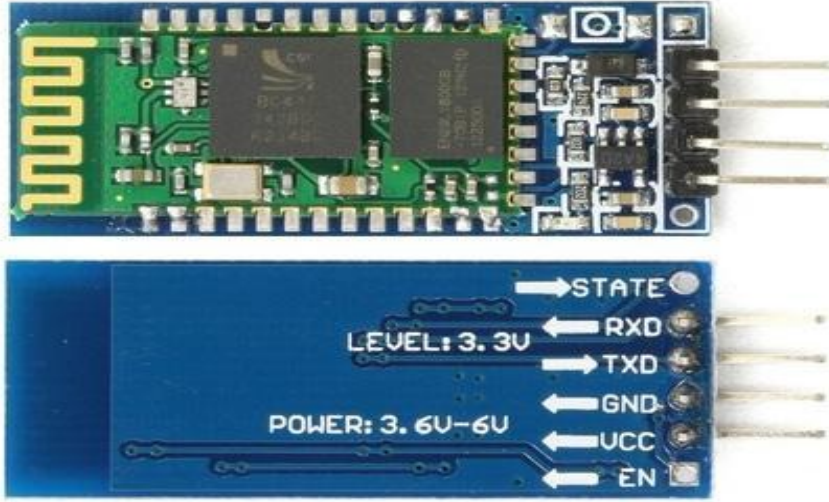


Resim 9. Kaza sonrası acil müdahale birimlerine gönderilen mesaj

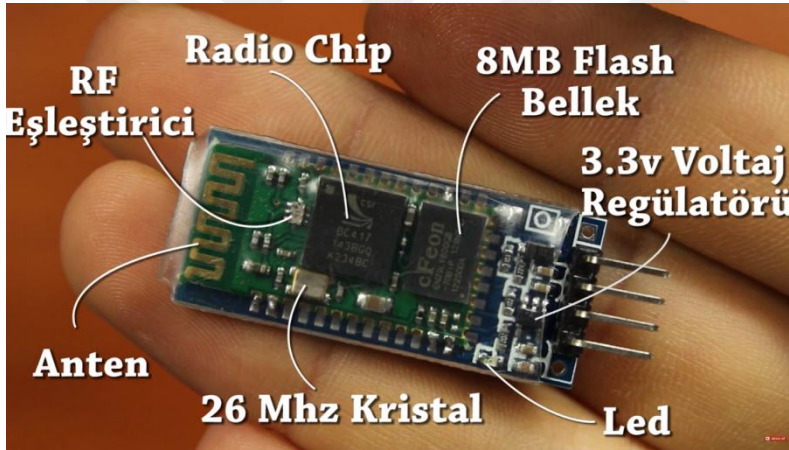
### 3.2.3. Bluetooth HC-06

İki master iki slave olmak üzere 4 adet bluetooth kullanılmıştır. Slave olan bluetoothlar akıllı telefon ve bilgisayarın kendi içerisinde barındırdığı bluetooth cihazlardır. Master olan bluetoothlar ise HC-06 tipine sahip iki yönlü veri alıp göndermede kullanılan bluetooth modülü tercih edilmiştir. Bu bluetooth modülünün konfigürasyon ayarları için varsayılan fabrika çıkışı ayarları tercih edilmiştir. Ayarların değiştirilmesi için ayrıca bir UART (Universal Asynchronous Receiver Transmitter-Evrensel Asenkron Alıcı Verici) modüle ihtiyaç vardır. Bunun yerine android işletim sistemine sahip telefon ve bilgisayardan bu ayarlamalar gerçekleştirilmiştir. Burada önemli olan kısım bilgisayarda bulunan bluetooth modülünün saniyede gönderilen bit sayısı ile programlanan mikrodenetleyici ve bluetooth modülünde saniyede işlenen bit sayısının aynı frekansta olması gerekmektedir. Aksi takdirde doğru veri iletişimi mümkün olmayacaktır. Diğer bir husus ise COM bağlantı portunun da doğru bir şekilde mikrodenetleyicide ayarlanmış olması gerekmektedir.





Resim 10. HC-06 Bluetooth modülü önden ve arkadan görünüş



Resim 11. HC-06 Bluetooth modülü bileşenleri

Bu kart için teknik detaylar;

- “Bluetooth Protokolü: Bluetooth 2.0+EDR (Gelişmiş Veri Hızı)
- 2.4GHz haberleşme frekansı
- Hassasiyet:  $\leq -80$  dBm
- Çıkış Gücü:  $\leq +4$  dBm
- Asenkron Hız: 2.1 MBps/160 KBps
- Senkron Hız: 1 MBps/1 MBps
- Güvenlik: Kimlik Doğrulama ve Şifreleme
- Çalışma Gerilimi: 1.8-5V (Önerilen 3.3V)
- Akım: 50 mA
- Boyutları: 43x16x7mm”

Çıkış gerilimi 3.3v olmasına rağmen arduino mega 2560 kartının işletme gerilimi 5v olarak belirlenmiştir [26].

Tablo 5. Bluetooth giriş voltaj dağılımı

$V_{IH}$	Input High Voltage, Except XTAL1 and RESET pins	$V_{CC} = 1.8V - 2.4V$ $V_{CC} = 2.4V - 5.5V$	$0.7V_{CC}^{(2)}$ $0.6V_{CC}^{(2)}$		$V_{CC} + 0.5$ $V_{CC} + 0.5$	V
----------	---	--	--	--	----------------------------------	---

Burada ki tabloya göre hangi giriş değerinden sonra lojik 1 değeri alınacağı gösterilmektedir. Condition sütunda belirtilen şartlara göre 2.4v-5.5v arasında yapılan beslemelerde sinyalin lojik 1 algılanması için minimum besleme geriliminin 0.6 kadar olması gerektiği belirtilmiştir.

$V_{CC}$  (işletme gerilimi) =5v

Input High Voltage =  $V_{CC} * 0.6$

$3v = 5v * 0.6$  denklemde görüldüğü üzere 3v-5.5v arası değer için lojik 1 kabul edilmektedir.

Bunun içindir ki bluetooth modülünün Tx pininden Arduino Mega 2560'ın Rx pinine gelen giriş voltajı içerisinde barındırdığı 3.3v regülatörü sayesinde gerilim bölücüye ihtiyaç duymadan doğrudan bağlanabilmektedir [26].

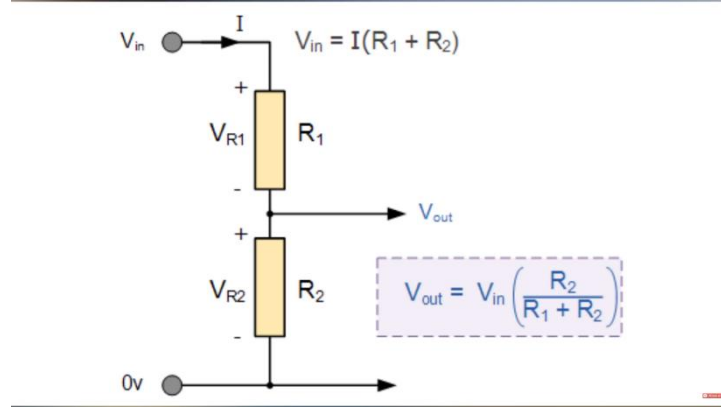
Tablo 6. Bluetooth çıkış voltaj dağılımı

$V_{OH}$	Output High Voltage <sup>(4)</sup> ,	$I_{OH} = -20mA, V_{CC} = 5V$ $I_{OH} = -10mA, V_{CC} = 3V$	4.2 2.3			V
----------	--------------------------------------	--	------------	--	--	---

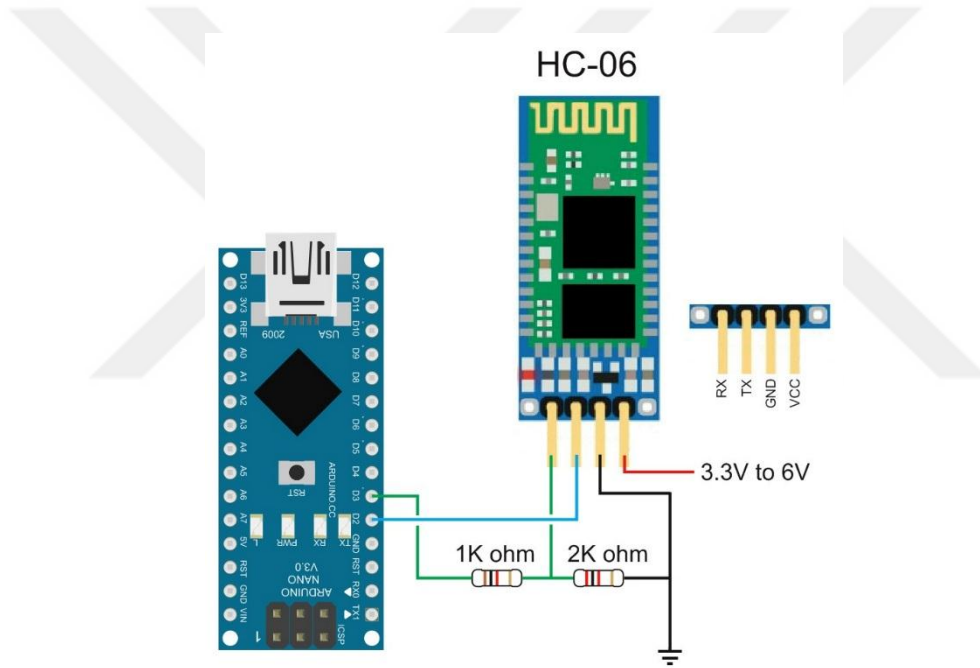
Arduino Mega 2560 kartının lojik 1 değeri için besleme voltajı 5v iken ve ortam sıcaklığı Max 85C derece olduğu durumda minimum 4.2v değer vereceği belirtilmektedir.

Bluetooth modülü için giriş voltaj değeri 3.3v geçmemesi gerekmektedir. Bu durumda HC-06 bluetooth modülünün doğru çalışabilmesi için voltaj bölücü bir devreye ihtiyaç vardır.





Şekil 6. Bluetooth 3.3v için gerilim bölücü devre diyagramı



Şekil 7. Bluetooth voltaj bölücü devre diyagramı

Diyagramı verilmiş bir voltaj bölücü devresinin değerleri hesaplanması sonucunda

$$R_2 = 2k\Omega$$

$R_1 = 1k\Omega$  direnç değerlerine sahip olup

$$V_{out} = 5v * 0.6666$$

$V_{out} = 3.3v$  olarak bulunmuştur.

### 3.2.4. I2C Haberleşme Protokolü

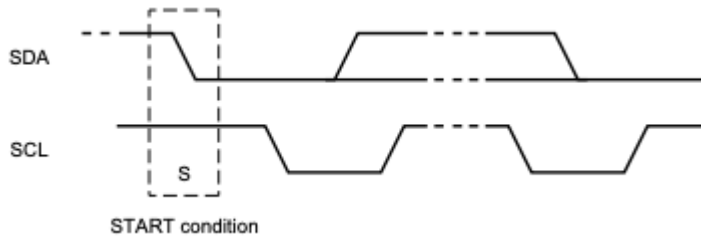
Inter-Integrated-Circuit diğer bir ifadeyle I2C protokolü; Philips firması tarafından 1982 yılında kısa mesafede master cihazın, en az bir slave cihazla adresleme sistemine göre çift yönlü olarak haberleştiği bir protokol türüdür. Bu haberleşme protokolünün avantajı hat sayısı değişmeden tercih edildiği kadar cihaz bağlanabilmesidir. Bu protokolün bir diğer avantajı da denetlemiş olduğumuz birim sayımız artmış olsa da mikro denetleyici üzerinde port sayımız değişmemekte ve istediğimiz kadar birimi slave olarak kontrol edebilmekteyiz. I2C protkolü haberleşmesi iki hat üzerinden gerçekleşmektedir. Veri hattı olan SDA (Serial Data Line-Seri Veri Hattı) ve SCL (Serial Clock Line-Seri Frekans Hattı) hatları ile senkronizasyon haberleşme yapılmaktadır. I2C haberleşme protokolünde seri clock hattından gelen frekanslara göre 3 farklı haberleşme hızı vardır.

Bu hızlar:

- 100kbit/s (yavaş)
- 400kbit/s (hızlı)
- 3.4Mbit/s (yüksek hızlı) şeklindedir.

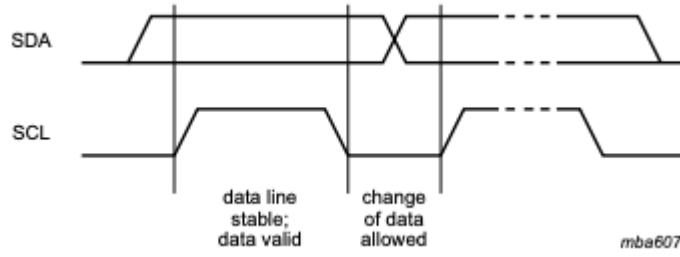
I2C haberleşme de SDA hattına yüklenecek veriler SCL sinyalinin tepe noktada olduğu durumlarda gönderilir. SDA hattında veri değişimleri SCL sinyalinin düşük olduğu noktalarda değiştirilir. Verinin sağlıklı gönderiminde yukarıda bahsedilen noktalara dikkat edilerek I2C Haberleşme Protokolü 4 adımda gerçekleşir. Bu adımlar açıklamaları ile aşağıda anlatılmıştır.

#### i. Start biti ile I2C haberleşme başlangıç işareti.



Şekil 8. I2C Haberleşme start biti gönderimi

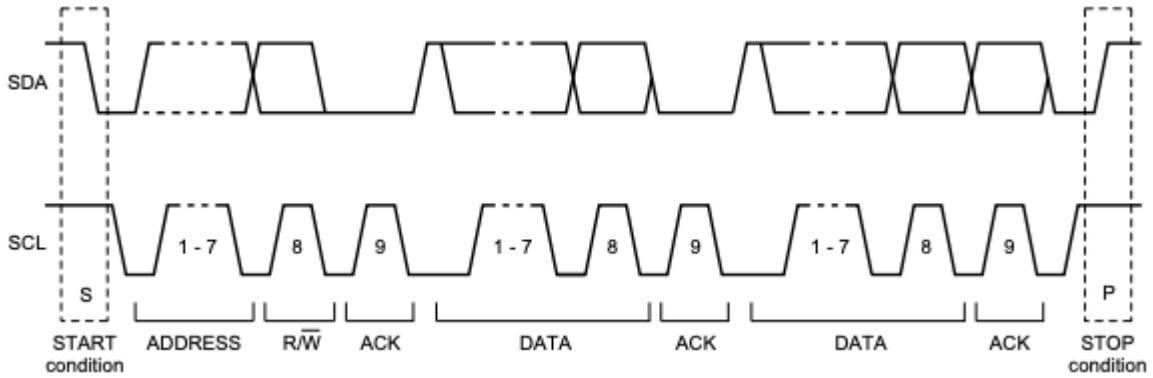
Yukarıdaki resimde görüldüğü üzere SDA=1 SCL=1 olduğu durumdan, SDA=0 SCL=1 durumuna geçtiği anda START biti verilmiş olup I2C haberleşmesi başlamaktadır. Daha sonrasında veri iletiminin devam edebilmesi için SDA=0 olduğu durumda önce SCL=0'a çekilmelidir [31].



Şekil 9. I2C veri akışı için gerekli kare dalga durumu

Bundan sonra veri aktarımları için yukarıda bahsedildiği gibi SCL hattının değiştiği ve yüksek olduğu (lojik bir) durumlarda veri gönderimi yapılırken SCL hattının düşük olduğu (lojik sıfır) durumlarda SDA veri değişimi yapılmaktadır [31].

- ii. **Adres ve işlem verisi gönderimi:** Her I2C haberleşmesine sahip cihazın kendine ait bir 7bitlik adresi vardır. Start bitinin gönderilmesinin ardından iletişime geçilecek cihaz için



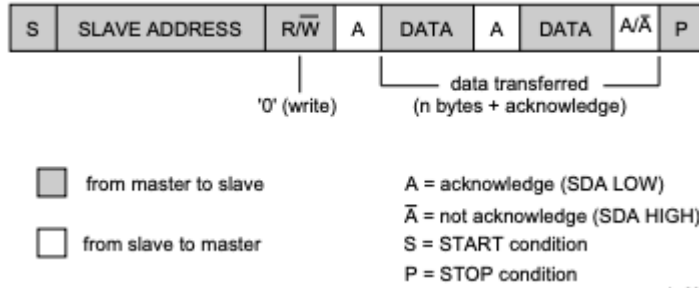
Şekil 10. I2C Haberleşme veri döngüsü

bu 7 bitlik adres verisi SDA hattı üzerinden gönderilir. Adres verisinden sonra iletişime geçilecek cihazdan veri okuma veya yazma işlemi için 8. bit işlem verisi gönderilmektedir. Slave cihazdan ACK (el sıkışma) biti geldikten sonra 8 bit 1 byte paketler halinde veri transferi başlamaktadır [31].

- iii. **Veri transferi**

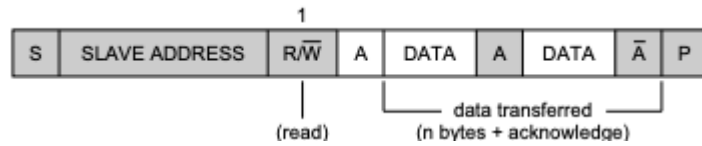
Veri transferi 7 adres bitinden sonra gönderilen 8. bit verisine göre okuma (lojik sıfır) ve yazma (lojik bir) olmak üzere şeklinde iki şekilde olmaktadır.

**Veri gönderme transferi:** Master cihazdan Slave cihaza gönderilen 1-7 adres bitinden sonra 8. bit lojik sıfır olması durumunda veri gönderimi yapılacağı bildirilir. SDA=0 olması durumunda ACK (el sıkışma) durumu gerçekleştirildikten sonra veri almaya hazır olduğunu bildirir. Sonrasında 8 bitlik paketler halinde veriler gönderilir. Slave olan cihazdan SDA=1 verisinden sonra NACK durumu gerçekleşerek STOP biti gönderilerek veri aktarımı durdurulur.



Şekil 11. I2C Veri gönderme durum bitleri

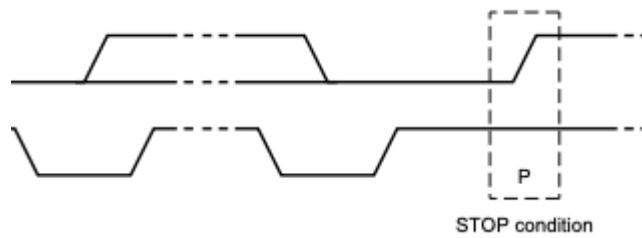
**Veri alma transferi:** Slave cihazın adres verisi alındıktan sonra 8. lojik bir yapılarak veri okuma yapılacağı bildirilmektedir. Haberleşmenin başlayacağı verisi ACK (el sıkışma) verisi alındıktan sonra SDA hattından gelen veriler okunduktan sonra doğru bir şekilde okunduğunu teyit etmek ve yeni veri alımına hazır olduğunu belirtmek için SLAVE cihaza 9. bit olarak ACK verisi gönderilmektedir.



Şekil 12. I2C Veri alma bit durumu

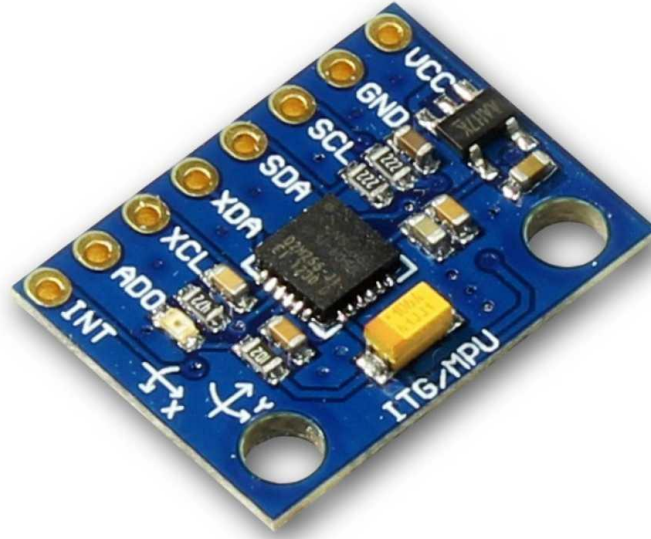
#### iv. Stop biti ile I2C haberleşme sonlandırma

SCL=1 SDA=0 durumundan SCL=1 SDA=1 durumuna geçilmesi I2C protokolünde STOP biti gönderimi anlamına gelmektedir. Bu durumda tüm veri aktarımı durdurulmaktadır.



Şekil 13. I2C Haberleşme sonlandırma biti

### 3.2.5. MPU 6050 İvme Ölçer ve Jiroskop



Resim 14. MPU605 ivme ölçer sensor

MPU6050 altı eksen DOF (Degrees of freedom-Serbestlik derecesi)'ne sahip ivme ve jiroskop ölçümü yapabilen bir IMU (çoklu sensor) kartıdır. Bu dokuz serbestlik derecesi ise üç eksen ivme, üç eksen jiroskop ve üç eksen g kuvvetinin ölçülebilmesi için üretilmiştir. Projede bu çoklu sensor kartı iki sebepten dolayı kullanılmaktadır. Birinci sebebi aracın üç boyutlu uzay düzleminde aracın her ivmelenme hareketini takip edebilmektir. Aracın açılal konumu hakkında, ivme ve jiroskop sensörlerinden gelen verilerin bütünleyici (complementary) filtreden geçirilerek matematiksel hesaplamalar sonucunda bilgi sahibi olunmaktadır. Bu veriler kayıt altına alınarak aracın kazadan sonra kazanın nasıl gerçekleştiği hakkında bilgiye erişilebilmektedir. Ayrıca aracın 45 dereceden fazla bir açıya ulaşmasını da kaza kabul ederek araç devrilme senaryosu sonucu gerekli birimlere ve ilgili kişilere konum ve kaza türü şiddeti mesaj olarak iletilmektedir. İkinci sebep ise yukarıda bahsedilen g kuvvetinin insan üzerinde ki etkisinden dolayı sistem içerisinde barındırılan g kuvvet ölçüm sensörü 4g G kuvvetinden sonra kazanın şiddetine göre kaza türü ve şiddeti hakkında bilgilendirme yaparak olay mahalline intikal edecek birimlerin kaza hakkında önceden bilgi sahibi olması sağlanmaktadır.

MPU6050 sensörü daha önce bahsedilmiş olan I2C haberleşme protokolüne göre veri aktarımında bulunmaktadır. 16bit ADC (analog dijital dönüştürücü)ye sahip olup yüksek çözünürlüklü veri seti ile hassas ölçüm yapabilmektedir. Sensor içerisinde gerekli ayarlamalar yapılarak sistemin kullanım alanına göre ivme ve jiroskop değerlerinin skalaları

önceden ayarlanabilmektedir. Bu proje maliyet açısından prototip olarak tasarlandığından g kuvveti testi için  $\frac{1}{2}$  oranı tercih edilmiştir. Kaza uyarı şiddeti 2g ve üzeri kuvvetler için kategorilendirilmiştir.

İvme ölçer sensör anlık olarak parazitlerden etkilenmektedir. Bu etkiden korunmak için alçak geçiren bir filtre olan bütünleyici filtreden yararlanılmaktadır. Uzun sürede ivme sensöründe aşırı değişim olmadığı için zamanın %90 lık bölümünde ivme sensöründen veri alınırken kısa vadede jiroskop sensöründen alınan veriler daha güvenilir olduğu için zamanın %10 luk bölümünde de bu sensörden istifade edilerek daha stabil veri elde edilmiştir. X ve Y eksenine bağlı açılar bütünleyici filtrede göz önüne alınarak yeniden düzenlendiğinde (18) ve (19) numaralı denklemler ile elde edilmektedir [32].

$$\text{Toplam X Açısı} = TX$$

$$\text{Toplam Y Açısı} = TY$$

$$\text{Jiroskop X Verisi} = JXV$$

$$\text{Jiroskop Y Verisi} = JYV$$

$$\text{Geçen Süre} = t$$

$$\text{X ivme verisi} = accX$$

$$\text{Y ivme verisi} = accY$$

$$\text{Z ivme verisi} = accZ$$

$$TX = 0.98 * (TX + JXV * t) + 0.02 * \tan^{-1}\left(\frac{accY}{\sqrt{accX^2 + accZ^2}}\right) \quad 18$$

$$TY = 0.98 * (TY + JYV * t) + 0.02 * \tan^{-1}\left(\frac{accX}{\sqrt{accY^2 + accZ^2}}\right) \quad 19$$

### 3.3.Yazılım Birimlerinin Tanıtılması

Çalışmada verilerin analiz edilebilmesi, görselleştirilebilmesi ve mikrodenetleyici ile çevre birimlerinin haberleşmesi için birçok farklı platformlardan istifade edilmiştir.

#### 3.3.1. Arduino IDE

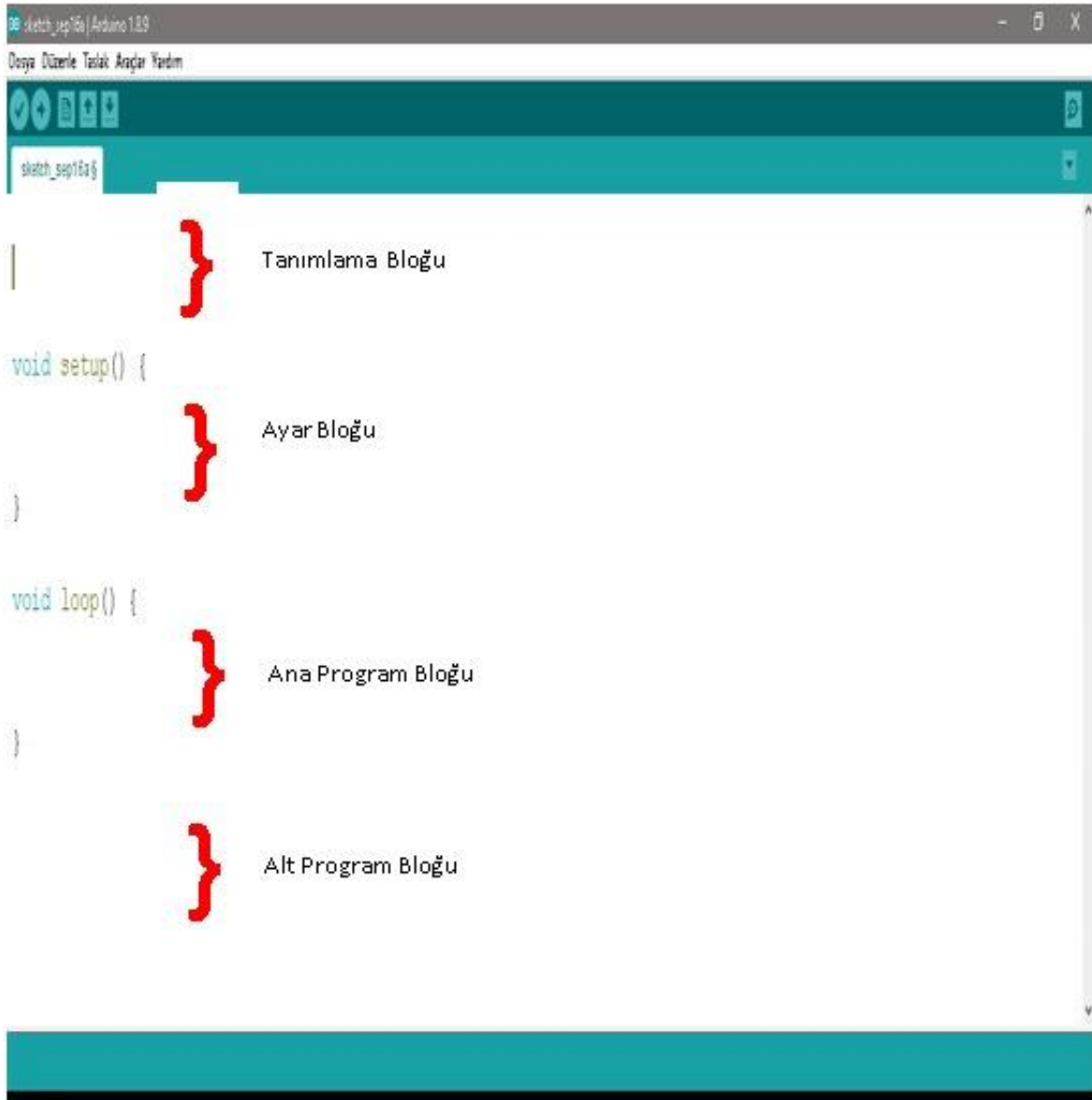
Arduino ismi geliştiricilerinin İtalyan olması sebebiyle italyanca da Sıkı Arkadaş anlamına gelmektedir. Arduino kartlarının çok tercih edilmesinin sebepleri açık kaynak kodlu olarak isteyen herkesin ücretsiz olarak üretip yazılımını kullanabilmesi. C/C++/Java gibi yaygın dillerle programlanabilmesinin yanında tinkercad, scratch for arduino ve ardublock gibi blok tabanlı yazılımlar ile de kodlama bilgi sahibi olmadan birçok proje geliştirilebilmesine kolaylık sağlamasıdır. Ayrıca çok bir elektronik bilgisine sahip olmadan arduino kartlarına uygun olarak üretilen shield diye ifade edilen çevrebirimlerinin karta headerlar vasıtasıyla kolayca monte edilmesiyle kullanılabilir. Bu shieldler, üreticilerinin kütüphaneleri ile birlikte piyasaya sürerek kullanıcıların üretilen cihazın teknik bilgisi hakkında bilgi sahibi olmadan hazır kodlar ile kolayca programlamasına imkân vermektedir. Arduino IDE içinde birçok örnek kod barındırarak kullanıcının proje geliştirmesine de kolaylık sağlamaktadır. Arduino IDE programlama da dört bölümden oluşmaktadır.

**Tanımlama bloğu:** Gerekli kütüphanelerin eklenmesi ve değişkenlerin tanıtılması için oluşturulmuş bölüm. void setup() komutu ile bu bölüm son bulur.

**Ayar bloğu:** void setup() komutu ile başlayan {} blok parantezleri arasında yer alan haberleşme protokollerinin başlatıldığı, algılayıcı (sensör) birimlerinin giriş, eyleyici birimlerinin(motor,led,buzzer v.b.) çıkış olarak tanımlandığı, birbiri ile haberleşen birden fazla cihazın adreslerinin tanımlandığı bloktur. Bu blok mikro denetleyicinin aktif olması durumunda sadece bir kez çalışmaktadır.

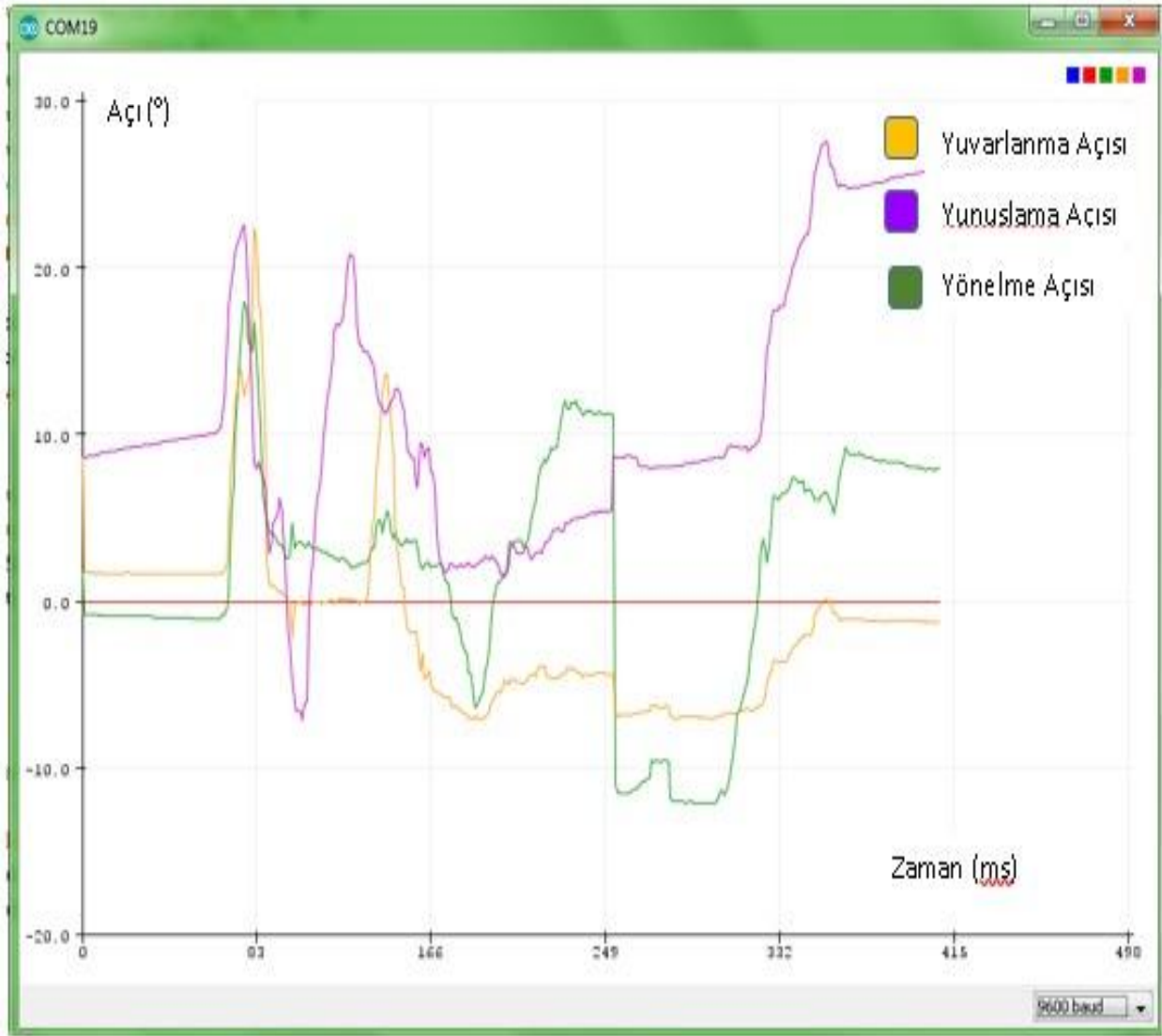
**Ana program bloğu:** void loop() komutu ile başlayıp {} blok parantezleri arasında yer alan ana program, gerçekleşmesi istenilen olayların programı bu blok içerisinde tanımlanır.

**Alt program(fonksiyon) bloğu:** Alt bölümlere ayrılan ana programın fonksiyon olarak tanımlanan program parçacıklarının bulunduğu alandır. Gerekli görüldüğü durumlarda ana programdan fonksiyonlar bu bölümden çağırılmaktadır. Void loop ana programının} blok parantezinin bitmesi ile bu bölüm başlamaktadır. Tercihe göre bu bölüm ayar bloğu ile ana program bloğu arasında da yer alabilmektedir.



Resim 12. Arduino IDE görünüm

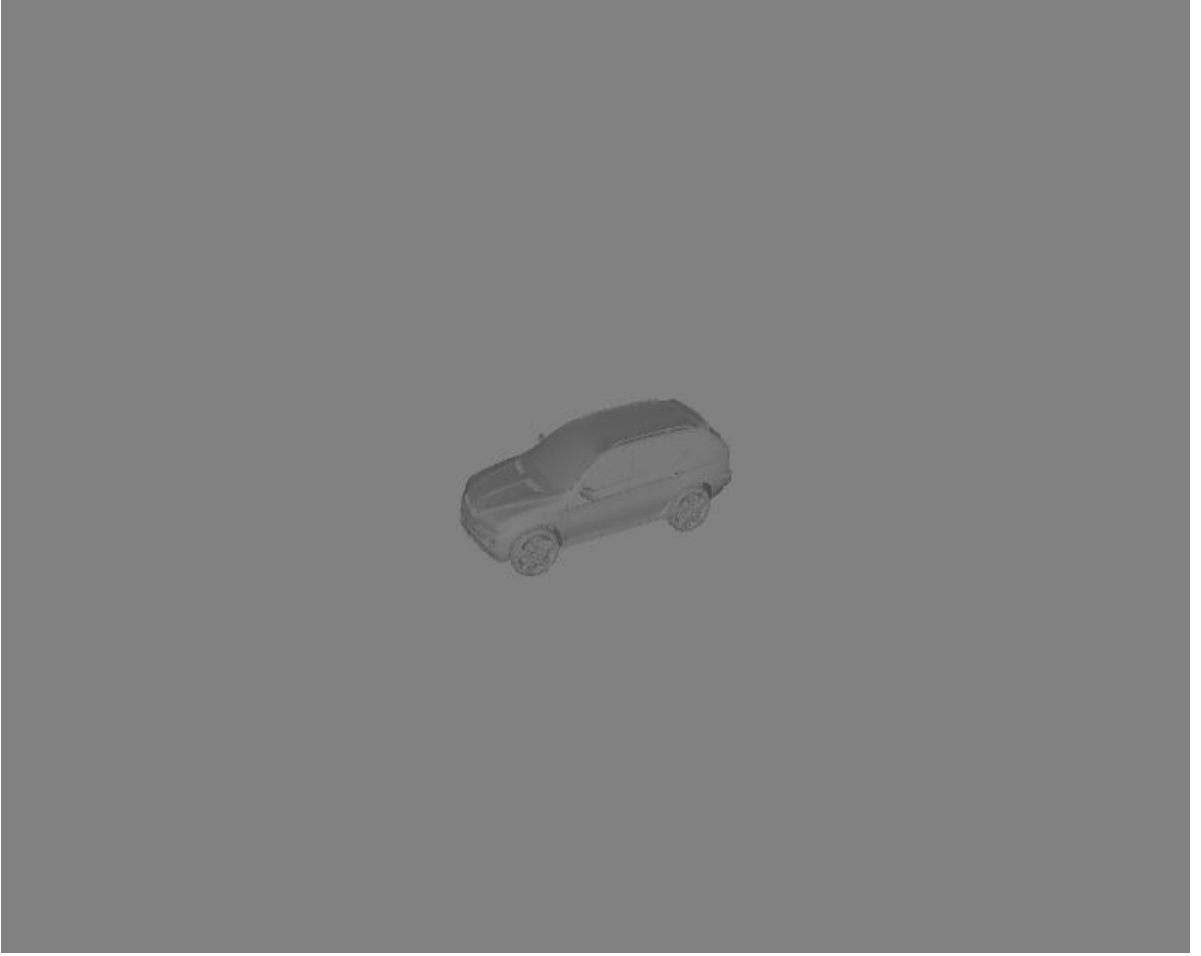




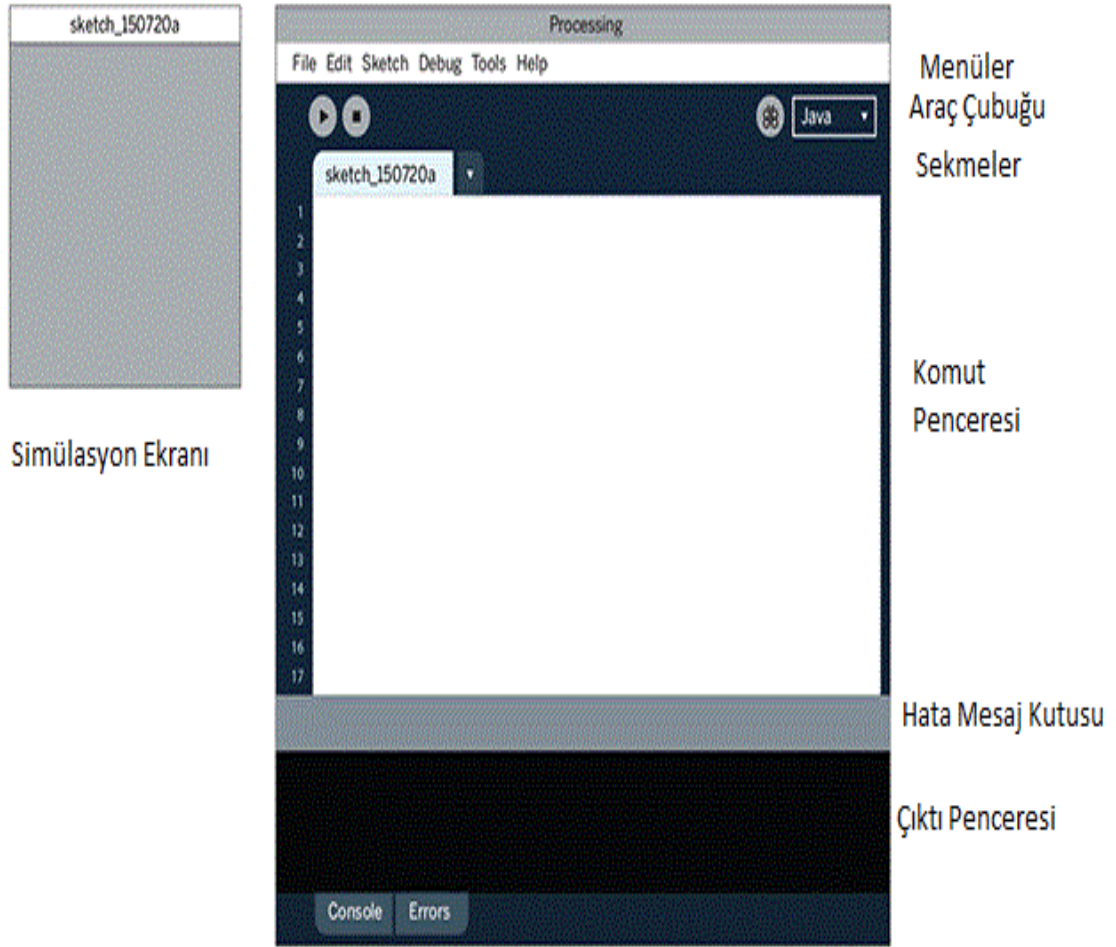
Resim 13. Arduino IDE'den üç eksene ait ölçüm alınması

### 3.3.2. Processing 2.2.1. Simülasyon Platformu

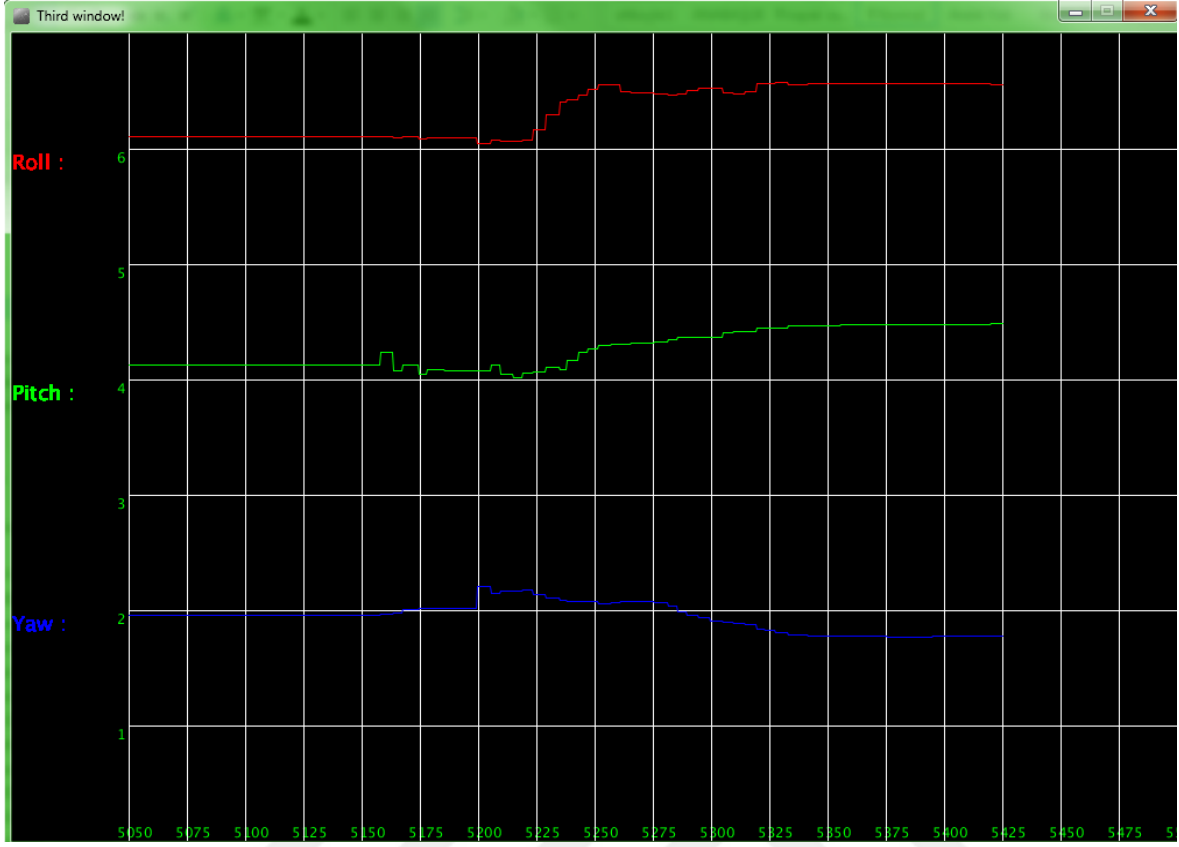
Java programlama dili ile görsel projeler oluşturmada kullanılan bir platformdur. Açık kaynak kodlu ve ücretsiz olması sebebiyle geniş bir kullanım alanına sahiptir. 3.x ve 2.x olmak üzere iki versiyon mevcuttur. Projede COM portlarından gelen verileri görselleştirmede kullanılmıştır. Aracın üç boyutlu görünümü modellenerek uzay boşluğunda hareketleri görülmektedir. X, Y ve Z eksenlerinde ki açısal değişimleri ise grafiksel olarak gösterilmektedir. Bunların yanında GPS ten alınan veriler doğrultusunda konumu harita üzerinde gösterilmektedir.



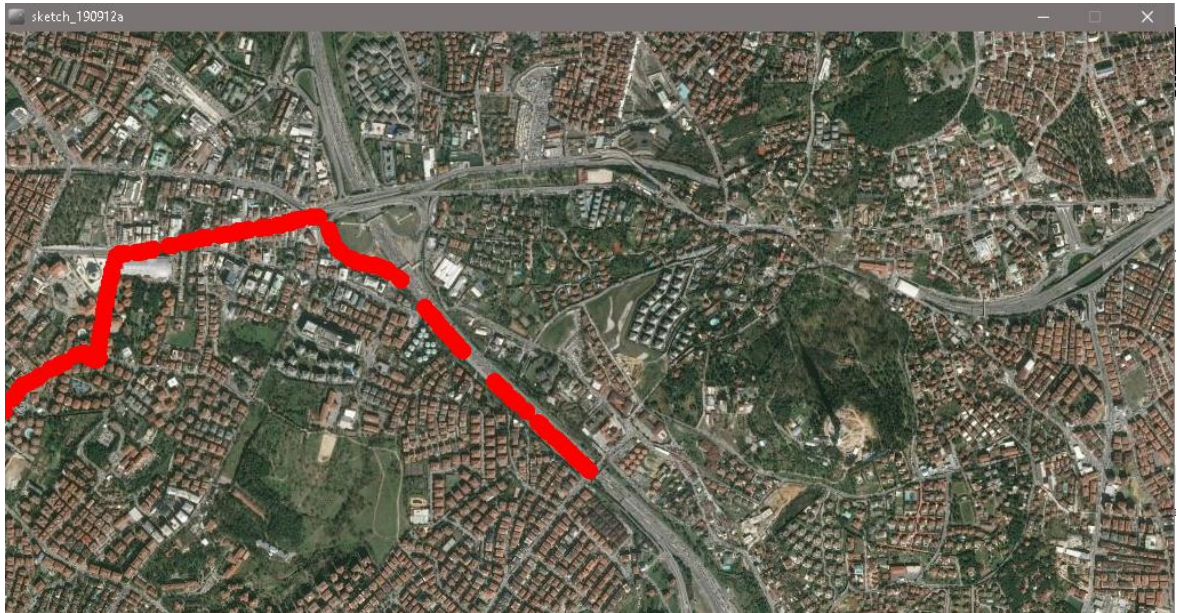
Resim 14. Aracın üç boyutlu uzaydan görünümü



Resim 15. Processing 2.2.1 IDE görünümü



Resim 16. Aracın üç eksene göre hareketlerinin grafiksel görünümü



Resim 17. Kaza olayının gerçekleştiği konumun uydu görüntüsü

### **3.3.3. Appinventor Android Yazılım Geliştirme Platformu**

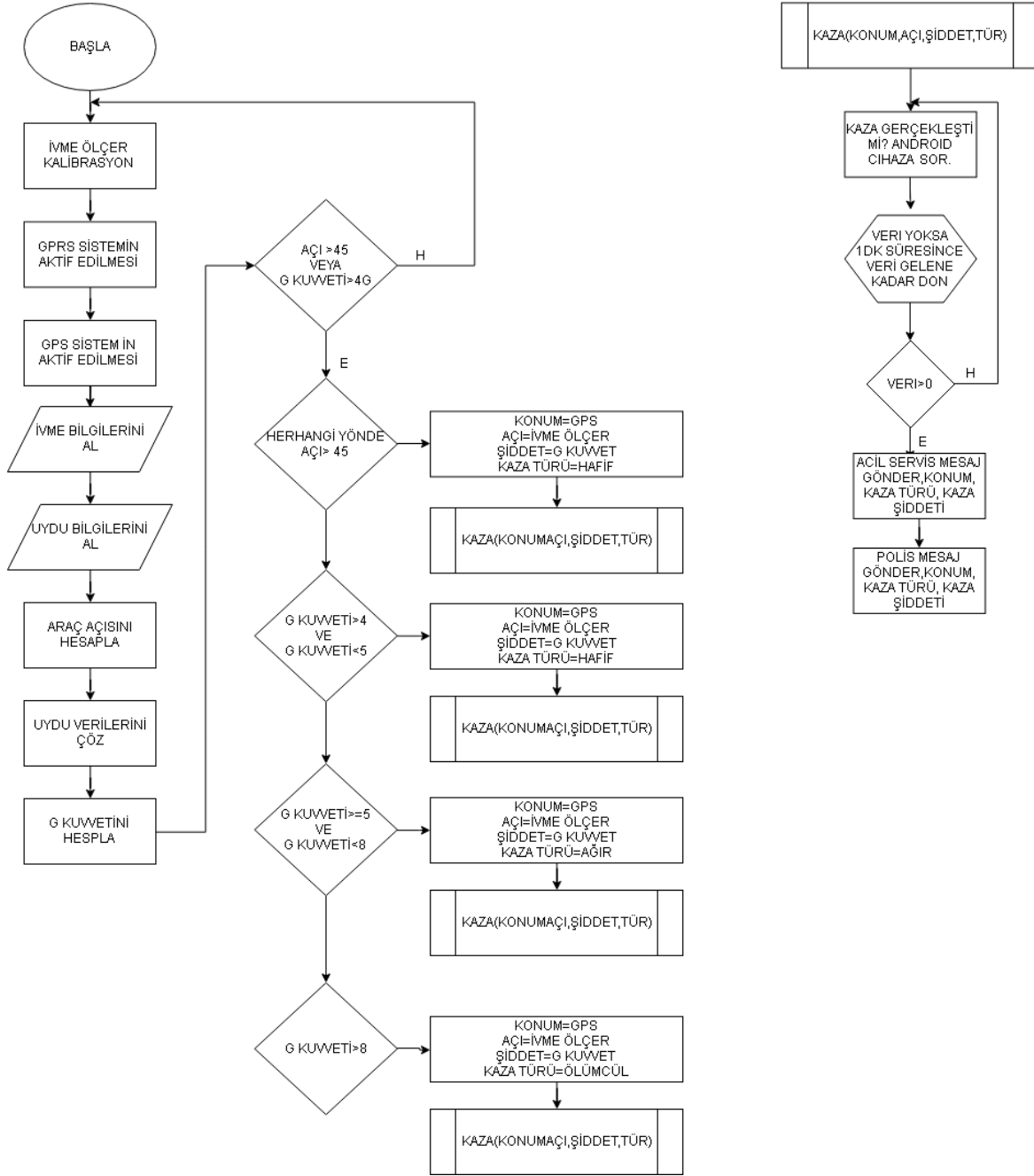
MIT (Machetutes University) tarafından geliştirilen bu platform ile yazılım bilgisi bilinmeden, kodlama mantık çerçevesinde android platformunda çalışacak uygulamalar geliştirmek mümkün olabilmektedir. Projede kullanıcın telefonu ile haberleşebilmesi için bu platformda dilediği gibi uygulama gerçekleştirebilmektedir.

### **3.4. Sistem Algoritmaları**

Bu bölümde, android ve mikrodeneyselci tabanlı bildirim ve uyarı sistemi android, processing ve arduino olmak üzere üç ayrı platformun her birinin kendisine ait akış diyagramları aşağıda gösterilmektedir.

### 3.4.1. Mikrodenetleyici Sistem Algoritması

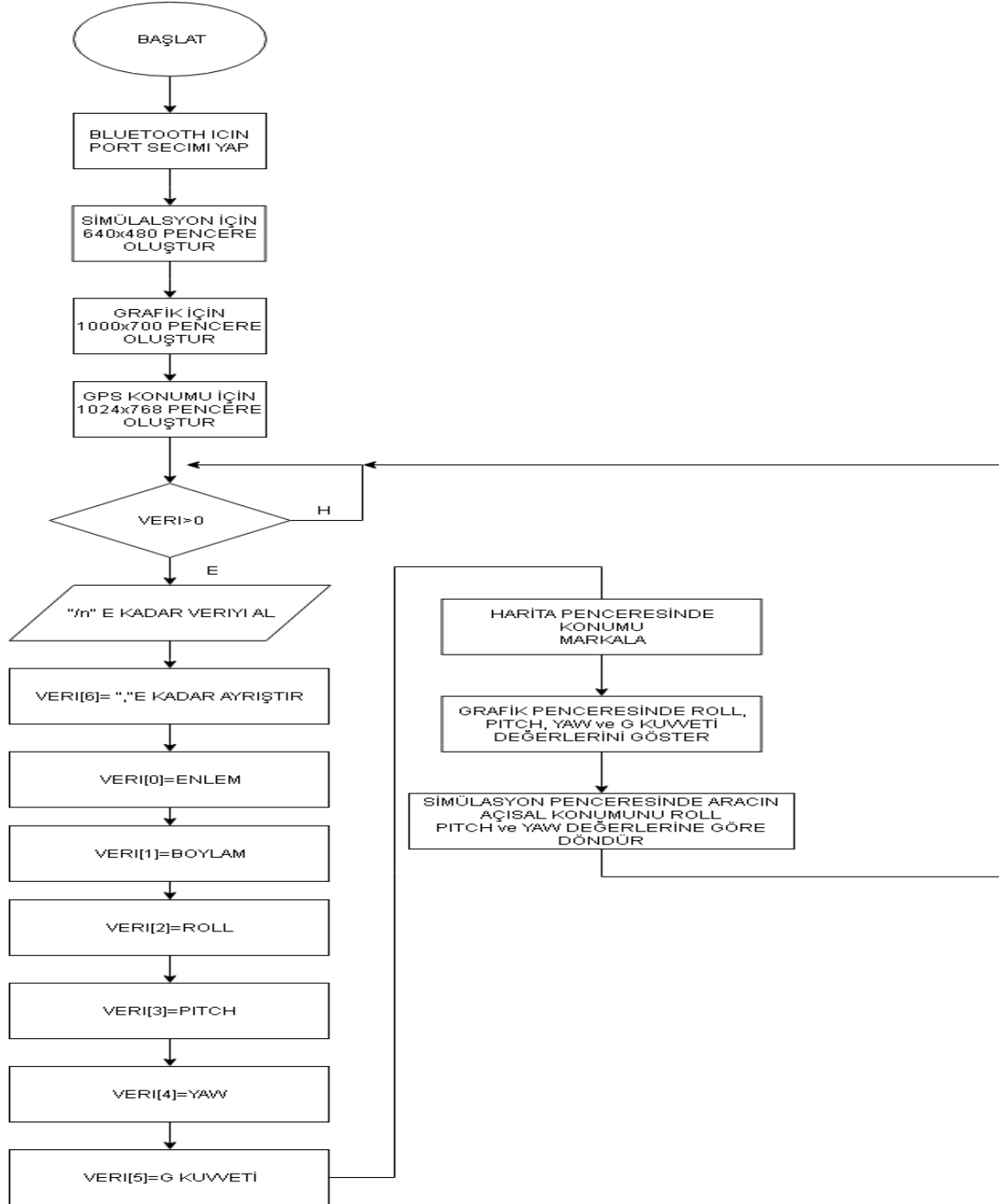
Kaza gerçekleşme anından mesaj bilgisinin gönderilme zamanına kadar olan Arduino ve Android Tabanlı Kaza Bildirim ve Uyarı Sisteminin çalışmasına dair akış diyagramı Şekil 14'te gösterildiği gibidir. Akış diyagramına ait C++ programı EK-1 başlığı altında açıklaması ile birlikte verilmiştir.



Şekil 14. Mikro denetleyicinin kaza bildirim ve uyarı sistemine ait akış diyagramı

### 3.4.2. Processing 2.2.1 Simülasyon Sistem Algoritması

Kazanın gerçekleştiği anda hafıza kartına kaydedilen verilerin zamana göre aracın konumu ve hareketlerinin benzetimini yapan processing programına ait sistemin akış diyagramı Şekil 15'te gösterilmiştir. Processing programında yazılan benzetim programına ait java kodu EK-2 de yer almaktadır.



Şekil 15. Processing simülasyon ortamına ait java programının akış diyagramı

#### 4. SONUÇLAR

Bu çalışmada; her yıl dünyada 1,2 milyondan fazla insanın trafik kazalarında hayatını kaybettiği ve kaza sonrası gerçekleşen ölümün, kaza mahallinde gerçekleşen ölümden fazla olduğu verisi göz önünde bulundurularak, bu sayıyı azaltmak adına tasarlanan arduino ve android tabanlı kaza bildirim ve uyarı sistemi geliştirilmiştir.

Acil müdahale ekiplerinin en kısa sürede trafik kazalarının gerçekleştiği olay mahalline ulaşmasını sağlamak ve kaza sonrası ölüm oranlarını azalmasına katkı sağlamak amaçlanmıştır. Aracın merkezi bir konumuna yerleştirilen platform üzerindeki ivme ölçer ile ölçülen G kuvveti ve aracın açısız konumu sayesinde kaza şiddeti ve türü belirlenmektedir. Platform üzerindeki GPS modülü ile konumu belirlenen araç, Yandex ve Google navigasyon sistemleri tarafından trafik yoğunluğuna göre en kısa yol rotasını çizmektedir. GPRS ile de sim kartının bulunduğu operatör ile SMS (Short Message Service-Kısa İleti Servisi) gönderilerek ilgili birime anlık olarak kazanın konumu, şiddeti ve kaza türünün iletilmesi ile acil müdahale ekiplerine anında müdahale imkânı sağlayabilecek bir sistem tasarlanmıştır.

Bu konuda yapılan çalışmalar analiz edildiğinde halihazırdaki ivme ölçerlerin birçoğunun akıllı telefon üzerinden yazılım ile yapıldığı ve sağlıklı verilerin alınmadığı görülmüştür. Bazı sistemlerde ise çoklu değişkenler arası bağlantı kurulabilmesi için maliyeti hayli yüksek ve hayli zahmetli sunucu ağları kurmak gereklidir. Bu alanda yapılan çalışmalara göre sistemin diğer sistemlere göre avantajları aşağıdaki tabloda belirtilmiştir.



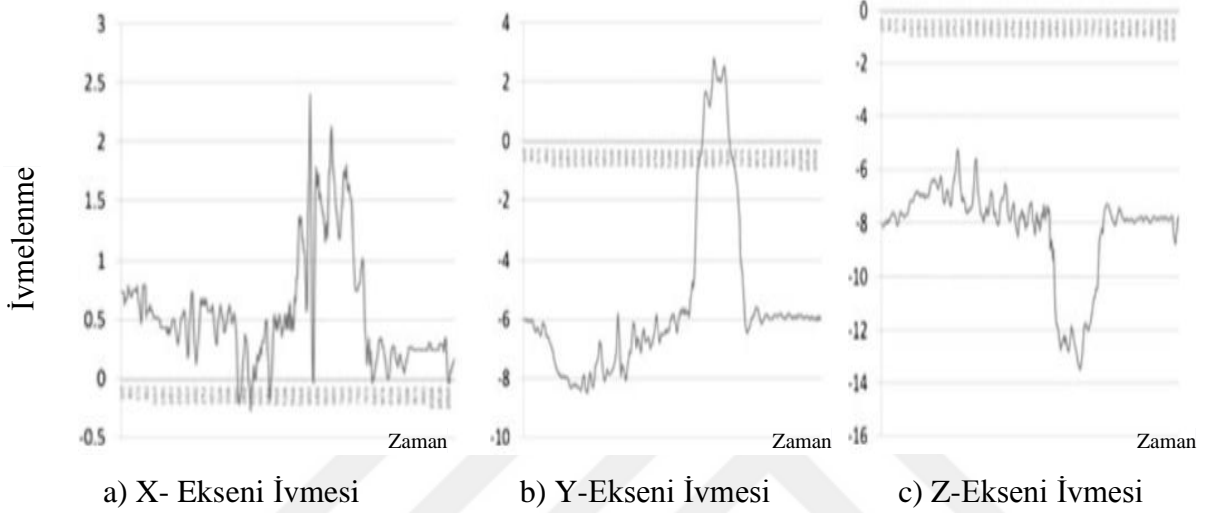
Tablo 7.Çalışmanın diğer çalışmalar ile karşılaştırılması

	Özgün Çalışma	Trafik Kazası otomatik algılama ve uzaktan alarm cihazı [7]	Çeşitli Bölgelerde Araç Hız Sınırı Uyarı ve Çarpışma Tespit Sistemi [8]	Akıllı Trafik Işık Sistemiyle Otomatik Kaza Tespit ve Ambulans Kurtarma [9]	Mobil Telefon ile Kaza Tespiti [11]	WreckWatch: Akıllı Telefonlarla Otomatik Trafik Kazası Tespiti ve Bildirimi [12]	Araçların Kazayla İzlenmesi için MEMS İvmeölçer ve GPS İzleme Kullanan Kablosuz Kara Kutu [13]
Kaza Bildirim ve Uyarı Sistemi	X	X	X	X	X	X	X
GPS	X	X	X	X	X	X	X
Bildirim Servisi	X	X		X	X	X	X
Kaza Kategorilendirme	X	X			X		X
İvme Verilerini Filtreleme	X	X		X	X		
Düşük Maliyet	X		X		X	X	X
Veri Depolama	X	X					
Simulasyon	X						
Mobil Uyarıcı Uygulama	X				X	X	X
Veri Güvenilirliği	X	X					

Çalışmada sözü geçen arduino ve android tabanlı kaza bildirim ve uyarı sisteminin, diğer sistemlerden farklı olarak GPS, ivme ölçer ve GPRS modüllerini tek bir platformda toplanmış olması sayesinde maliyet azalmakta, ivme ölçere uygulanan filtreleme ile sağlıklı

sonular alınabilmekte ve yukarıda bahsedilen olumsuzluklar tamamen bertaraf edilebilmektedir. Bu sistem sayesinde alınan veriler benzetim programı ile simüle edilerek kaza hakkında ivedi ve detaylı bilgi elde edilmiştir.

Alan taraması sonuçlarına göre Şekil 14.'de gösterildiği gibi Arduino ve Android tabanlı



Şekil 16. Aracın ani durmada üç eksene göre ivmelenmesi

kaza bildirim sistemi Tablo 7'de görüldüğü üzere diğer sistemler üzerinde bulunan diğer sistemler üzerindeki eksiklikler giderilmiştir. Yapılan test sonuçlarına göre SMS bilgisi 5-9 saniye arasında gecikme göstermiştir. Bu gecikme süresi kullanılan GSM operatörü kaza alanı yakınında bulunan baz istasyonunun varlığına ve doğal şekil ve engellerine göre değişmektedir.

Kaza haberinin gerekli birimlere zamanında ulaşmaması sonucu kaza yerine ulaşım sürelerinin uzun olması, acil müdahale merkezlerine yapılan çağrılarının çoğunun sahte olması nedeniyle ekiplerin gereksiz meşgul edilmesi de düşünüldüğünde, Arduino ve Android Tabanlı Kaza Bildirim ve Uyarı Sistemleri büyük önem taşımaktadır. Sistemin geniş bir ağ şeklinde tüm araçlarda bulunması halinde kaza tespit tutanaklarının otomatik düzenlenmesi ve hatalı sürücünün tespitinin kaza anında hızlı bir şekilde belirlenmesi de Emniyet Birimlerinin yükünü hafifletecek olabilir. Bu Sistem daha da geliştirilerek kaza sigorta şirketleri tarafından sahte kazaları belirlemelerinde de kullanılabilir.

## KAYNAKLAR

1. Çetin, B., Barış S. Ve Saroğlu, S. (2011). “*Türkiye’de Karayollarının Gelişimine Tarihsel Bir Bakış*” Çankırı Karatekin Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi, 123-150.
2. [https://www.who.int/violence\\_injury\\_prevention/road\\_safety\\_status/2015/TableA2.pdf?ua=1](https://www.who.int/violence_injury_prevention/road_safety_status/2015/TableA2.pdf?ua=1) 21. 07.2019
3. Keser, H., Ay, S., Çetin I., (2018). “*Ulaştırmada Karayolları: Türkiye’deki Gelecek Beklentileri*” TESAM Akademi Dergisi 63-93.
4. Karayolları Genel Müdürlüğü (2010), “*2010 Trafik ve Ulaşım Bilgileri*”, Strateji Geliştirme Daire Başkanlığı Ulaşım Maliyetleri ve Verimlilik Şubesi Müdürlüğü Mayıs 2011 (kgm.gov.tr)
5. Türkiye İstatistik Kurumu, “*Karayolu Trafik Kaza İstatistikleri*”, 2015 n.d. 13.01.2019
6. Ayten, S., (2015), “*Denizli ilinde 112 komuta merkezine yapılan aramalar ve acil ambulans hizmetlerinin kullanımının değerlendirilmesi*”. Pamukkale Üniversitesi Tıp Fakültesi.
7. Chaklader, S., Alam, J., Islam, M., Sabbir, A.S., (2014), “*Traffic accident automatic detection and remote alarm device*”, 2014 International Conference on Informatics, Electronics & Vision (ICIEV), IEEE, Dhaka, Bangladesh ,1–6.
8. Singh, D.N., (2013), “*Vehicle Speed Limit Alerting and Crash Detection System at Various Zones*”, International Journal of Latest Trends in Engineering and Technology, 26.
9. Nova college of Engg For Women, Jupudi, IBM, Vijayawada, Jyothi, R.J., Prasad, V.R., N Anuradha, N.A., (2012), “*Automatic Accident Detection and Ambulance Rescue With Intelligent Traffic Light System.*” IJSR 3177–3179.

10. Kansiz, A.O., Guvensan, M.A., (2013),” *Mobil Telefon ile Kaza Tespiti*”, 2013 21st Signal Processing and Communications Applications Conference, IEEE, Haspolat,1–4.
11. White, J., Thompson, C., Turner, H., Dougherty, B., Schmidt, D.C., (2011), “*WreckWatch: Automatic Traffic Accident Detection and Notification with Smartphones.*” *Journal Mobile Networks and Applications* Volume16 Issue3, 285–303.
12. Watthanawisuth, N., Lomas, T., Tuantranont, A., (2012),” *Wireless Black Box Using MEMS Accelerometer and GPS Tracking for Accidental Monitoring of Vehicles*”, Proceedings of 2012 IEEE-EMBS International Conference on Biomedical and Health Informatics,IEEE, Hong Kong ,847–850.
13. Namahoot, C.S., Brückner, M., (2015),”*Mobile and Wireless Technology 2015*”, Springer Berlin Heidelberg, Berlin, Heidelberg ,173–182.
14. Organization, W.H.O., (2015), “*Global Status Report on Road Safety 2015*,” World Health Organization.  
[https://www.who.int/violence\\_injury\\_prevention/road\\_safety\\_status/2015/en/](https://www.who.int/violence_injury_prevention/road_safety_status/2015/en/)
15. Karayolları Genel Müdürlüğü 2017, “*Karayolları Genel Müdürlüğü Sorumluluğundaki Yol Ağında Meydana Gelen Trafik Kazalarına Ait Özet Bilgiler*”.  
<http://www.kgm.gov.tr/SiteCollectionDocuments/KGMdocuments/Trafik/TrafikKazalariOzeti2017.pdf>
16. Arıkan, E., Çubuk, M., Hatipoğlu, S., Sinoplu, M. (2008), “*Türkiye’deki Karayolu Tünellerinde Trafik Güvenliği*”, *Teknik Dergi*, 19 (93), 4471-4486.
17. Şimşek, M., Erdemli, T., Aydemir, H., Taşdelen, K. (2013). “*Konum Destekli Acil Durum Koordinasyonu (Koday)*”. *Uluslararası Teknolojik Bilimler Dergisi*, 5 (3), 89-97.
18. Aydoğan, T., Çakır, A., Polat, Y. E. (2011). “*Cep Telefonu ile Acil Çağrı Otomasyonu*”, *Elektrik Elektronik ve Bilgisayar Sempozyumu*, 98 -101, Fırat Üniversitesi, Elâzığ.

19. Selig, J.M., (1992),” *Introductory robotics*” Prentice-Hall, Englewood Cliffs New Jersey.
20. Craig, J.J., Addison-Wesley, (1986) “*Introduction to robotics: mechanics and control*”, Addison Wesley Publishing Company, Reading, Massachusetts.
21. Noureldin, A., Karamat, T.B., Georgy, J., (2013), "Fundamentals of Inertial Navigation, Satellite-based Positioning and their Integration”, Springer Berlin Heidelberg, Berlin, Heidelberg ,21–63.
22. Sciavicco L., Siciliano B. (1996), “*Modeling and control of robot manipulators*”, McGraw-Hill, New York.
23. Asada H., Slotine J.-J. E., (1986), “*Robot analysis and control*”, John Wiley & Sons, New York.
24. Manisalı, M., (2017), “*İnsansız hava araçlarında denge sistem kontrolünün bulanık mantıklı fpga ve gömülü sistemler ile tasarımı ve uygulaması*”. Pamukkale Üniversitesi Fen Bilimleri Enstitüsü, 91.
25. Mert, GPS Nedir? Nasıl Çalışır? | Modülasyon, Çanak Anten, LNB n.d.27.02.2019
26. datasheet.pdf n.d. 27.02.2019
27. mega2.pdf n.d. 27.02.2019
28. arduino\_gsm\_gps\_jumper\_ayarları.pdf n.d. 02.03.2019
29. Gnd teknik eğitim teknolojileri elektronik otomasyon sistemleri | gsm / gps shield n.d. 19.04.2019

30. SIM900\_SIM300\_ATC\_Comparison\_V1 00.pdf n.d. 29.05.2019

31. UM10204 I2C-bus specification and user manual 201464.

32. PID control arduino drones mpu6050 mpu9250 gyro accelerometer euler msp432 rtos  
n.d. 01.06.2019



**EKLER**



EK-1 Arduino IDE Yazılımı

```
//*****KÜTÜPHANE TANIMLAMALARI*****
```

```
#include <SoftwareSerial.h>
```

```
#include <LiquidCrystal_I2C.h>
```

```
#include <TinyGPS++.h>
```

```
#include <String.h>
```

```
#include<stdlib.h>
```

```
#include <Wire.h>
```

```
#include <MPU6050_tockn.h>
```

```
//*****DEĞİŞKEN TANIMLAMALARI*****
```

```
String enlem,boylam;
```

```
LiquidCrystal_I2C lcd(0x27,20,4);
```

```
//GPS HABERLEŞME SANAL PORT AYARI
```

```
TinyGPSPlus gps;
```

```
int txPin = 4;
```

```
int rxPin = 3;
```

```
SoftwareSerial bluetooth(rxPin, txPin);
```

```
MPU6050 mpu6050(Wire);
```

```
long timer = 0;
```

```
float x,y,z;
```



```
float roll,pitch,yaw;

//*****İLK AYARLAMALARIN YAPILANDIRILMASI*****

void setup() {

//I2C HABERLEŞME BAŞLATMA

Wire.begin();

//İVME ÖLÇER BAŞLATMA

mpu6050.begin();

//İVME ÖLÇER KALİBRASYON

mpu6050.calcGyroOffsets(true);

//LCD BAŞLATMA

lcd.init();

lcd.backlight();

//BLUETOOTH HABERLEŞME AYARI

bluetooth.begin(9600);

//SERİ MONİTÖR HABERLEŞME AYARI

Serial.begin(9600);

// GPRS HABERLEŞME AYARI

Serial2.begin(9600);

lcd.print("Koordinat.");

delay(1000);
```

```
lcd.clear();

lcd.print("Koordinat..");

delay(1000);

lcd.clear();

lcd.print("Koordinat...");

delay(1000);

lcd.clear();

delay(500);

}

//*****ANA PROGRAM*****

void loop() {

//İVME VERİLERİNİ GÜNCELLEME

mpu6050.update();

//İVME AÇI HESAPLAMALARI

roll = 0.96 * mpu6050.getGyroAngleX() + 0.04 * mpu6050.getAccAngleX();

pitch = 0.96 * mpu6050.getGyroAngleY() + 0.04 * mpu6050.getAccAngleY();

yaw = mpu6050.getGyroAngleZ();

//GPS VERİ ALMA

while(Serial2.available(>0) gps.encode(Serial2.read());

enlem=String(gps.location.lat(), 6);
```

```
boylam=String(gps.location.lng(), 6);

lcd.setCursor(0,0);

lcd.print("Enlem :");

lcd.print(gps.location.lat(), 6);

lcd.setCursor(0,1);

lcd.print("Boylam:");

lcd.print(gps.location.lng(), 6);

//BLUETOOTH KONUM BİLGİSİ GÖNDERME

bluetooth.print(enlem);

bluetooth.print(",");

bluetooth.print(boylam);

bluetooth.print(",");

//BLUETOOTH AÇI BİLGİSİ GÖNDERME

bluetooth.print(roll);

bluetooth.print(",");

bluetooth.print(pitch);

bluetooth.print(",");

bluetooth.println(yaw);

Serial.print(enlem);

Serial.print(",");
```

```
Serial.print(boylam);

Serial.print(",");

Serial.print(roll);

Serial.print(",");

Serial.print(pitch);

Serial.print(",");

Serial.println(yaw);

Serial.print(",");

Serial.print(mpu6050.getAccX());

Serial.print(",");

Serial.print(mpu6050.getAccY());

Serial.print(",");

Serial.println(mpu6050.getAccZ());

if(mpu6050.getAccX()>2 || mpu6050.getAccY()>2 || mpu6050.getAccZ()>2) {

  SendTextMessageCarpismaliKaza();

  delay(3000);

  while(mpu6050.getAccX()<2 || mpu6050.getAccY()<2 || mpu6050.getAccZ()<2) break;

}

if(roll>45 || pitch>45 || yaw>45) {

  SendTextMessageTaklaliKaza();
```

```

delay(3000);

while(roll<45 || pitch<45 || yaw<45) break;

} if (Serial1.available()){

Serial.write(Serial1.read());

}

}

//*****ARACIN ÇARPIŞMASI SONRASI GÖNDERECEĞİ MESAJ*****

void SendTextMessageCarpismaliKaza ()

{

//*****-----GOOGLE HARİTALAR-----*****

Serial.println("Sending Text...");

//KARTI SMS MODUNA ALMAK İÇİN GEREKLİ AT KOMUTU

Serial1.print("AT+CMGF=1\r"); // Set the shield to SMS mode

delay(100);

//GÖNDERİLECEK NUMARANIN AT KOMUTU

Serial1.println("AT+CMGS = \"+905542958749\"");

delay(100);

//MESAJ İÇERİĞİNİN HAZIRLANMASI

String msg="Kaza Türü:Carpisma/n";

msg+="Konum:";

```

```
msg+="https://www.google.com.tr/maps/@"+String(gps.location.lat(),6)+" "+String(gps.lo
cation.lng(),6)+" ,17z?hl=tr"; Serial1.println(msg); delay(100);
```

```
//MESAJIN İLETİLMESİ İÇİN GEREKEN ASCII KODUN ONAYI
```

```
Serial1.print((char)26);
```

```
Serial1.println();
```

```
Serial.println("Text Sent.");
```

```
delay(10000);
```

```
//*****-----YANDEX HARİTALAR-----*****
```

```
Serial.println("Sending Text...");
```

```
Serial1.print("AT+CMGF=1\r");
```

```
delay(100);
```

```
Serial1.println("AT+CMGS = \"+905542958749\");
```

```
delay(100);
```

```
String String msg2="Kaza Türü:Carpisma/n";
```

```
msg2+="Konum:";
```

```
msg2+="https://yandex.com.tr/harita/?ll="+String(gps.location.lng(),6)+"%2C"+String(gps
.location.lat(),6)+"&z=20"; Serial1.println(msg2); //the content of the message
```

```
delay(100);
```

```
Serial1.print((char)26);
```

```
delay(100);
```

```

Serial1.println();

Serial.println("Text Sent.");

delay(10000);

}

//*****ARACIN TAKLA ATMASI DURUMUNDA GIDECEK MESAJ*****

void SendTextMessageTaklaliKaza(){

//*****-----GOOGLE HARİTALAR-----*****

Serial.println("Sending Text...");

//KARTI SMS MODUNA ALMAK İÇİN GEREKLİ AT KOMUTU

Serial1.print("AT+CMGF=1\r"); // Set the shield to SMS mode

delay(100);

//GÖNDERİLECEK NUMARANIN AT KOMUTU

Serial1.println("AT+CMGS = \"+905542958749\"");

delay(100);

//MESAJ İÇERİĞİNİN HAZIRLANMASI

String msg="Kaza Türü:Takla/n";

msg+="Konum:";

msg+="https://www.google.com.tr/maps/@"+String(gps.location.lat(),6)+" "+String(gps.lo
cation.lng(),6)+" ,17z?hl=tr"; Serial1.println(msg); delay(100);

//MESAJIN İLETİLMESİ İÇİN GEREKEN ASCII KODUN ONAYI

```

```
Serial1.print((char)26);

Serial1.println();

Serial.println("Text Sent.");

delay(10000);

//*****-----YANDEX HARİTALAR-----*****

Serial.println("Sending Text...");

Serial1.print("AT+CMGF=1\r");

delay(100);

Serial1.println("AT+CMGS = \"+905542958749\"");

delay(100);

String String msg2="Kaza Türü:Takla/n";

msg2+="Konum:";

msg2+="https://yandex.com.tr/harita/?ll="+String(gps.location.lng(),6)+"%2C"+String(gps
.location.lat(),6)+"&z=20"; Serial1.println(msg2); //the content of the message

delay(100);

Serial1.print((char)26);

delay(100);

Serial1.println();

Serial.println("Text Sent.");

delay(10000);
```



```
}
```

```
//ÇAĞRI YAPMA FONKSİYONU
```

```
void DialVoiceCall()
```

```
{
```

```
Serial1.println("ATD+905542958749;");//dial the number, must include country code
```

```
delay(100);
```

```
Serial1.println();
```

```
}
```

EK-2 Processing Yazılımı

/\*HARİTALAMA ve 3 BOYUTLU ÇİZİM İÇİN GEREKLİ KÜTÜPHANLERİN  
YÜKLENMESİ\*/

import de.fhpotsdam.unfolding.mapdisplay.\*;

import de.fhpotsdam.unfolding.utils.\*;

import de.fhpotsdam.unfolding.marker.\*;

import de.fhpotsdam.unfolding.tiles.\*;

import de.fhpotsdam.unfolding.interactions.\*;

import de.fhpotsdam.unfolding.ui.\*;

import de.fhpotsdam.unfolding.\*;

import de.fhpotsdam.unfolding.core.\*;

import de.fhpotsdam.unfolding.mapdisplay.shaders.\*;

import de.fhpotsdam.unfolding.data.\*;

import de.fhpotsdam.unfolding.geo.\*;

import de.fhpotsdam.unfolding.texture.\*;

import de.fhpotsdam.unfolding.events.\*;

import de.fhpotsdam.utils.\*;

import de.fhpotsdam.unfolding.providers.\*;

import processing.serial.\*;

import processing.serial.\*;

import java.awt.event.KeyEvent;

```
import java.io.IOException;

Serial myPort;

/*GRAFİKSEL ÇİZİM ve 3 BOYTULU SİMULASYON İÇİN GEREKLİ
DEĞİŞKENLERİN TANIMLANMASI*/

String data="";

float x=100.0,y=0.0,preY=0.0,preX=100.0;

float x2=100.0,y2=0.0,preY2=0.0,preX2=100.0;

float x3=100.0,y3=0.0,preY3=0.0,preX3=100.0;

int ekranBas=0,genislik=1000,yukseklk=700;

float enlem,boylam,roll, pitch,yaw;

UnfoldingMap harita;

int genislikX=1;

void setup(){

    background(255);

    size(1,1);

    noLoop();

}

void draw(){

/*BLUETOOTH veya ARDUİNO COM PORTUNDAN ALINACAK VERİLER İÇİN
SERİ HABERLEŞMENİN BAŞLATILMASI*/

    printArray(Serial.list());
```

```

myPort = new Serial(this, Serial.list()[0], 9600);

//SİMÜLASYON PENCERESİNİN OLUŞTURULMASI

String[] args2={"Secondary window!"};

SecondWindow window2=new SecondWindow();

PApplet.runSketch(args2, window2);

//AÇISAL VERİLERİN ÇİZİLECEĞİ GRAFİK PENCERESİNİN OLUŞTURULMASI

String[] args3={"Third window!"};

ThirdWindow window3=new ThirdWindow();

PApplet.runSketch(args3, window3);

/*GPS VERİLERİNİN GÖSTERİLECEĞİ HARİTA PENCERESİNİN
OLUŞTURULMASI*/

String[] args4={"Secondary window!"};

FourthWindow window4=new FourthWindow();

PApplet.runSketch(args4, window4);

}

//*****SİMULASYON*****

class SecondWindow extends PApplet{

void setup(){

size (640, 480,P3D);

}

void draw() {

```

```
//SATIR SATIR GELEN VERİNİN ALINMASI
```

```
data = myPort.readStringUntil('\n');
```

```
/*VERİNİN GELMESİ ANINDA “,” ile AYRIŞTIRILAN VERİNİN PARÇALANARAK  
GEREKLİ DEĞİŞKENLERE AKTARILMASI*/
```

```
if (data != null) {
```

```
data = trim(data);
```

```
String veri[]=split(data,',');
```

```
if(veri.length>1){
```

```
roll=float(veri[2]);
```

```
pitch=float(veri[3]);
```

```
yaw=float(veri[4]);
```

```
}
```

```
translate(width/2, height/2, 0);
```

```
//3 BOYUTLU PRİZMANIN DÖNÜŞÜ İÇİN GEREKLİ AÇI DEĞERLERİ DÖNÜŞÜMÜ
```

```
rotateX(radians(pitch));
```

```
rotateY(radians(yaw));
```

```
rotateZ(radians(roll));
```

```
fill(0, 76, 153);
```

```
background(233);
```

```
box (400, 40, 200);
```

```
}
```

```
}  
  
}  
  
//*****GRAFIK*****  
  
class ThirdWindow extends PApplet{  
  
    void setup(){  
  
size(genislik, yukseklik);  
  
    frameRate(100);  
  
    drawStuff(ekranBas);  
  
    }  
  
    void draw() {  
  
//*****SERIAL GELEN VERI*****  
  
        if (data != null) {  
  
            data = trim(data);  
  
            String veri[]=split(data,',');  
  
            //veri[]=);  
  
            if(veri.length>1){  
  
                println(data);  
  
                roll=float(veri[2]);  
  
                println(veri[2]);  
  
                pitch=float(veri[3]);
```

```
println(veri[3]);

yaw=float(veri[4]);

println(veri[4]);

println(veri.length);

}

}

//*****ROLL*****

float y = roll;

fill(255,0,0);

textSize(18);

textAlign(LEFT, TOP);

text("Roll :", 0, height-600);

x=preX+genislikX;

stroke(255,0,0);

line(preX, height-600+preY, x, height-600+y);

preY=y;

preX=x;

if(preX>genislik) {

clear();

x=100.0;
```

```
preX=100.0;

ekranBas++;

drawStuff(ekranBas);

println(ekranBas);

}

//*****PITCH*****

float y2 = pitch;
fill(0,255,0);

textSize(18);

textAlign(LEFT, TOP);

text("Pitch :", 0, height-400);

x2=preX2+genislikX;

stroke(0, 255,0);

line(preX2, height-400+preY2, x2, height-400+y2);

preY2=y2;

preX2=x2;

if(preX2>genislik) {

x2=100.0;

preX2=100.0;

}
```



```
/*******YAW*****
```

```
float y3 = yaw;
```

```
fill(0,0,255);
```

```
textSize(18);
```

```
textAlign(LEFT, TOP);
```

```
text("Yaw :", 0, height-200);
```

```
x3=preX3+genislikX;
```

```
stroke(0, 0, 255);
```

```
line(preX3, height-200+preY3, x3, height-200+y3);
```

```
preY3=y3;
```

```
preX3=x3;
```

```
if(preX3>genislik) {
```

```
    x3=100.0;
```

```
    preX3=100.0;
```

```
}
```

```
}
```

```
void drawStuff(int ekranDeger) {
```

```
    background(0);
```

```
    int ekranSon=ekranDeger*1000;
```

```
    for (int i = 100; i <= width; i += 50) {
```

```

fill(0, 255, 0);

textSize(12);

text((ekranSon+i)/2, i-10, height-15);

stroke(255);

line(i, height, i, 0);

}

for (int j = 100; j < height; j += 100) {

fill(0, 255, 0);

text(6-j/(height/6), 90, j);

stroke(255);

line(100, j, width, j);

}

}

}

//*****GPS*****

class FourthWindow extends PApplet{

void setup(){

size (1024, 768,P3D);

harita = new UnfoldingMap(this, new Microsoft.AerialProvider());

MapUtils.createDefaultEventDispatcher(this, harita);

```

```
printArray(Serial.list());

delay(5000);

}

void draw() {

    if (myPort.available() > 0) {

        data =myPort.readStringUntil('\n');

        if (data != null) {

            data = trim(data);

            String veri[]=split(data,',');

            if(veri.length>1){

                enlem=float(veri[0]);

                boylam=float(veri[1]);

            }

        }

        harita.draw();

        harita.zoomAndPanTo(new Location(enlem,boylam), 30);

        Location istanbulKonum = new Location(enlem,boylam);

        SimplePointMarker istanbulMarker = new SimplePointMarker(istanbulKonum);

        istanbulMarker.setColor(color(255, 0, 0, 100));

        istanbulMarker.setStrokeColor(color(255, 0, 0));
```

```
istanbulMarker.setStrokeWeight(4);  
  
harita.addMarkers(istanbulMarker);  
  
Location konum = harita.getLocation(mouseX, mouseY);  
  
fill(255);  
  
text(konum.getLat() + ", " + konum.getLon(), mouseX, mouseY);  
  
}  
  
}  
  
}
```



## ÖZGEÇMİŞ

### Kişisel Bilgiler

Soyadı, adı : Abdullah ÇETİN  
 Uyuğu : T.C.  
 Doğum tarihi ve yeri : 23/12/1983-Bayburt  
 Medeni hali : Evli  
 Telefon : 05349671741  
 Faks :  
 E-mail : abduallah.cetin17@gmail.com  
 Adres Bilgileri : İstiklal Mah. Krizantem Sk. No:31 Anadolu Apt D.8  
 Ümraniye-İSTANBUL



### Eğitim

Derece	Eğitim Birimi	Mezuniyet tarihi
Yüksek lisans	İstanbul Gelişim Üniversitesi Mekatronik Mühendisliği	2014-201..
Lisans	Gazi Üniversitesi Teknik Eğitim Fakültesi Bilgisayar Sistemleri Öğretmenliği-İÖ	2007 – 2011
Önlisans	Kocaeli Üniversitesi / Karamürsel M.Y. Gemi İnşaatı	2005-2007
Lise	Balıkesir 100. Yıl Endüstri Meslek Lisesi Bilgisayar Donanım	2001-2004

### İş Deneyimi

Yıl	Yer	Görev
2018-201..	Türkiye Teknoloji Takımı	Robotik Kodlama Eğitmenliği
		Elektronik Programlama Eğitmenliği
		Siber Güvenlik Eğitmenliği
2017-201..	Asfa Eğitim Kurumları	Kodlama Robotik Öğretmeni

2013-2017	Özel Ümraniye Gökkuşuğu Eğitim Kurumları	Bilişim Teknolojileri Öğretmeni-Formatör
2011-2012	Çanakkale Özel İsmail Kaymak Kolej	Bilişim Teknolojileri Öğretmeni-Formatör
2003-2004	Çanakkale Yonca Bilgisayar	Teknik Eleman
2010-2011	Ankara Atatürk Kız Teknik ve Meslek Lisesi	Öğretmenlik Uygulama Stajı
2010-2011	Ankara İskitler Teknik ve Endüstrimeslek Lisesi	Öğretmenlik Uygulama Stajı
2007-2008	Ankara Gazi Teknik ve Endüstri Meslek Lisesi	Öğretmenlik İzleme Stajı
2010-2011	ODTÜ Teknokent	Endüstri Stajı
2005-2007	KOSBAŞ– Soli Shipyard	Endüstri Stajı

#### Yabancı Dil

İngilizce

Pre-Intermediate

#### Yayın

2016 Mayıs	EEB(Elektrik Elektronik Bilgisayar Sempozyumu)Tokat Gazi Osman Paşa 2016 Kendini Dengeleyen Robot Makale	Üniversitesi
2011 Haziran	İşitme engelliler için metni işaret diline çeviren program- Lisans Tezi	Gazi Üniversitesi

#### Kullanılan Programlar

Program Adı	Seviye
Autocad	Orta
Solidworks	İyi
C#,C++,Python,Java	İyi
Proteus-Ares	İyi

CCS C, PIC ASSEMBLY	İyi
Php Programlama	Orta
Arduino Programlama	İyi
Processing	İyi
Robot Tx Programlama-Fischer Teknik	İyi
Robotis Dream-Darwin Mini-Bioloid-Darwin Op-Bioloid Gp	İyi

**Hobiler:** Robotik, Balık tutma, Yamaç Paraşütü, Amatör Telsizcilik

### Sertifika- Başarılar

Yıl	Sertifika - Başarılar	Yer
2018 Eylül	TEKNOFEST-İHA Destekli İKA Finalistliği	Teknofest
2015 Mart	Education and Culture Lifelong learning programme CEMONIUS	Bahçelievler Gökkuşığı Eğitim Kurumları
2015 Mayıs	Education and Culture Lifelong learning programme CEMONIUS	Özel İsmail Kaymak Koleji
2015 Eylül	UAV Turkey İnsansız Hava Araçları	TÜBİTAK

## Verilen-Alınan Eğitimler

Durum	Eğitim Adı	Yer-Tarih
Verildi	MEB Onaylı STEM Eğitimci Eğitimi	Birlik Vakfı-2018 Mayıs
Verildi	Türkiye Teknoloji Takımı Robotik Kodlama Eğitimi	Deneyap Atölyeleri 2018-2019 Eylül-Şubat
Verildi	Türkiye Teknoloji Takımı Elektronik Programlama Eğitimi	Deneyap Atölyeleri 2019 Şubat-Haziran
Verildi	Türkiye Teknoloji Takımı Siber Güvenlik Eğitimi	Deneyap Atölyeleri 2019 Haziran-....
Alındı	İş Güvenliği	İstanbul Gelişim Üniversitesi Mart 2017
Alındı	Siber Güvenlik Eğitimci Eğitimi	TÜBİTAK TÜSSİDE 2019 Haziran
Alındı	Akıl Oyunları Eğitimci Eğitimi	İstanbul Gelişim Üniversitesi 2015 Mayıs

## Referanslar

Türkiye Teknoloji Takımı Mütevelli Heyeti Başkanı Haluk BAYRAKTAR



Türkiye Teknoloji Takımı Deneyap Genel Müdürü Ömer KÖKÇAM

Özel İsmail Kaymak Koleji Kurucu Müdür

Belgin KAYMAK  
ÇIRAKOĞLU

Gökkuşığı Eğitim Kurumları Kurucu Müdür

Şule GAYRETLİ

Ümraniye Gökkuşığı Eğitim Kurumları Genel  
Müdürü

Ziya Köksal ÇAKMAK

Asfa Eğitim Kurumları Genel Müdürü

Bayram KUM

Mental HR & Consultancy Şirketinde EducaMental  
General Manager

Belma UYSAL



**GELİŞİM GELİŞMEKTİR...**

