



# **ELEKTRİKLİ BİR TAŞIT İÇİN GÖSTERGE PANELİ TASARIMI**

**İzzet Timur YARDIMCI**

**YÜKSEK LİSANS TEZİ  
MAKİNA EĞİTİMİ ANA BİLİM DALI**

**GAZİ ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ**

**ARALIK 2019**

İzzet Timur YARDIMCI tarafından hazırlanan “ELEKTRİKLİ BİR TAŞIT İÇİN GÖSTERGE PANELİ TASARIMI” adlı tez çalışması aşağıdaki jüri tarafından OY BİRLİĞİ ile Gazi Üniversitesi Makina Eğitimi Ana Bilim Dalında YÜKSEK LİSANS Tezi olarak kabul edilmiştir.

**Danışman:** Prof. Dr. Kemal ERŞAN

Otomotiv Mühendisliği Ana Bilim Dalı, Gazi Üniversitesi

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

.....

**Başkan:** Prof. Dr. Veli ÇELİK

Uzay ve Havacılık Mühendisliği Ana Bilim Dalı, Ankara Yıldırım Beyazıt Üniversitesi

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

.....

**Üye:** Doç. Dr. Fatih ŞAHİN

Otomotiv Mühendisliği Ana Bilim Dalı, Gazi Üniversitesi

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

.....

Tez Savunma Tarihi: 26/12/2019

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

.....

Prof. Dr. Sena YAŞYERLİ  
Fen Bilimleri Enstitüsü Müdürü

## ETİK BEYAN

Gazi Üniversitesi Fen Bilimleri Enstitüsü Tez Yazım Kurallarına uygun olarak hazırladığım bu tez 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.

İzzet Timur YARDIMCI

26/12/2019

ELEKTRİKLİ BİR TAŞIT İÇİN GÖSTERGE PANELİ TASARIMI  
(Yüksek Lisans Tezi)

İzzet Timur YARDIMCI

GAZİ ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ  
Aralık 2019

ÖZET

Bu çalışmada, elektrikli taşıtlar başta olmak üzere taşıt içi donanımların haberleşmesinin sağlandığı uygulama yapılmıştır. Taşıt içi donanımlardan, taşıt hızı, batarya göstergesi ve gösterge paneline bu bilgilerin yansıtılması ile beraber Matlab programında bu sonuçlar karşılaştırılarak geliştirilmesi için hazırlık yapılmıştır. Bu çalışmada elektrikli bir taşıtın simülasyon verilerinin Matlab programı kullanılarak kurulan prototip elektrikli taşıt donanımına uygulanması ve kontrolü gerçekleştirilmiştir. Uygulamanın doğrulanması için araç hızı göstergesi, enerji sarfiyatını gösteren batarya durum göstergesi ve gösterge paneline bu bilgilerin yansıtılması için gerekli elektronik donanımın kurulması ve kurulum aşamaları anlatılmıştır. Kurulan prototipe taşıt şehir çevrimi uygulanmış, sistem çalıştırılmış, elde edilen ölçüm verileri doğrulama amaçlı uygulanan girdi verileri ile aynı ekranda karşılaştırılarak doğrulanmıştır. Yapılan prototipin doğru olarak çalıştığı ve gelecek daha gelişmiş çalışmaların aynı yöntemle yapılabileceği sonucuna ulaşılmıştır.

Bilim Kodu : 93009  
Anahtar Kelimeler : Elektrikli taşıt, gösterge paneli, PWM, I2C, tasarım  
Sayfa Adedi : 68  
Danışman : Prof. Dr. Kemal ERŞAN

# DASHBOARD DESIGN OF AN ELECTRICAL VEHICLE

(M. Sc. Thesis)

İzzet Timur YARDIMCI

GAZİ UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

December 2019

## ABSTRACT

In this study, it is made that communication of in-vehicle equipment, especially electric vehicles, is provided. In the Matlab program, these results are compared to the vehicle speed, battery indicator and instrument panel. In this study, the simulation data of an electric vehicle is applied to the prototype electric vehicle equipment installed using Matlab program and its control is realized. In order to verify the application, the vehicle speed indicator, the battery status indicator showing the energy consumption and the installation of the necessary electronic equipment to reflect this information to the instrument panel are explained. Vehicle city cycle was applied to the prototype, the system was started and the measurement data obtained were compared with the input data applied for verification purposes and verified on the same screen. It is concluded that the prototype works correctly and that future more advanced studies can be done with the same method.

Science Code : 93008

Key Words : Electrical Vehicle, Dabsboard, PWM, I2C, design.

Page Number : 68

Supervisor : Prof. Dr. Kemal ERŞAN

## TEŞEKKÜR

Sistemin geliştirilebilir hale getirilmesinde katkılarından dolayı danışmanım Prof. Dr. Kemal ERŞAN'a, bugüne kadarki sabır ve desteklerinden dolayı aileme ve tezin geliştirilebilir bir hale getirmeme yardımlarından dolayı kardeşim Öğr. Gör. Emel Elif YARDIMCI'ya ve Yasemin ÖNDER VAN DER LEE'ye teşekkürü bir borç bilirim.

## İÇİNDEKİLER

	Sayfa
ÖZET .....	iv
ABSTRACT.....	v
TEŞEKKÜR.....	vi
İÇİNDEKİLER .....	vii
ŞEKİLERİN LİSTESİ .....	x
RESİMLERİN LİSTESİ .....	xii
SİMGELER VE KISALTMALAR.....	xiii
1. GİRİŞ.....	1
2. LİTERATÜR ÖZETİ .....	3
2.1. Elektrikli Taşıtlar .....	3
2.2. Gösterge Panelleri .....	4
2.2.1. Analog göstergeler .....	5
2.2.2. Dijital göstergeler.....	6
2.3. Kullanılan Sensörler ve Elektronik Elemanlar.....	9
2.3.1. Kızılötesi devir ve hız sensörü .....	10
2.3.2. Gerilim bölücüler .....	11
2.4. Grafik Bar 3.7V Pil Şarj Göstergesi.....	12
2.5. Mikrodenetleyiciler .....	12
2.5.1. Arduino MEGA.....	13
2.5.2. Arduino UNO-NANO.....	13
2.6. Bluetooth Modülü .....	15
2.7. L293 Motor Sürücü Kartı.....	16
2.8. Kullanılan Yazılımlar.....	16
2.8.1. Matlab yazılımı .....	16



	<b>Sayfa</b>
2.8.2. Arduino kod editörü .....	17
2.8.3. 3ds Max .....	18
2.8.4. AutoCAD .....	18
2.8.5. Photoshop .....	18
2.8.6. Unity .....	19
2.9. Mikroişlemcilerde Haberleşme sistemleri .....	19
2.9.1. I2C senkron haberleşme protokolü .....	19
2.9.2. CAN-BUS haberleşme protokolü .....	20
2.10. Android İşletim Sistemi .....	20
2.11. Programın Yazılması .....	20
2.12. Programın Kontrol Altına Alınması .....	21
<b>3. ELEKTRİKLİ ARAÇ GÖSTERGE PANELİ PROTOTİP TASARIMI .....</b>	<b>23</b>
3.1. Prototip Uygulamasında Yazılımların Programlanması .....	25
3.1.1. Matlab yazılımı üzerinde yapılan işlemler .....	25
3.1.2. UDC (urban drive cycle) dosyalarının kullanılması .....	25
3.1.3. Matlab programına UDC verilerinin aktarılması .....	27
3.1.4. Matlab ve arduino arasında bilgi akışının sağlanması .....	28
3.1.5. Arduino yazılımı ile Matlab arasında UDC bilgilerinin aktarılması .....	28
3.2. Prototip Uygulaması İçin Kullanılan Donanımlar .....	29
3.2.1. Mikroişlemciler .....	29
3.2.2. ATmega328P .....	30
3.2.3. HC-05 bluetooth modülü .....	31
3.2.4. L293 motor sürücü kartı .....	31
3.2.5. F249 kızılötesi hız sensörü .....	32
3.2.6. Batarya şarj kontrol devresi .....	32

	<b>Sayfa</b>
3.3. Araç İçi Gösterge Panelinin ve Görselinin Tasarımı .....	32
<b>4. PROTOTİPİN ÇALIŞTIRILMASI VE DEĞERLENDİRİLMESİ.....</b>	<b>37</b>
4.1. Prototipte Kullanılan Donanım Elemanları.....	37
4.1.1. UDC verilerinin derlenmesi .....	37
4.1.2. Matlab kontrol verilerinin hazırlanması.....	38
4.1.3. Matlab ve arduino haberleşmesinin sağlanması.....	38
4.1.4. Bilgisayar haberleşmesini sağlayan ünitenin yazılımı .....	38
4.1.5. Dc motor kontrol ünitesinin hazırlanması.....	39
4.1.6. Batarya şarj seviyesi ve sıcaklık kontrol modülünün hazırlanması .....	40
4.1.7. Bluetooth bağlantısının sağlanması .....	42
4.1.8. Android yazılımının hazırlanması.....	43
4.1.9. Ana kontrol ünitesinin hazırlanması .....	43
4.1.10. Uygulamada kullanılacak görsellerin hazırlanması .....	44
4.2. Elektronik Donanımın Hazırlanması.....	49
4.2.1. Bilgisayar haberleşme ünitesi .....	49
4.2.2. Dc motor kontrolü.....	49
4.2.3. Batarya kontrol devresi .....	50
4.2.4. Bluetooth haberleşmesi .....	50
4.2.5. Ana kontrol ünitesi.....	51
<b>5. SONUÇ VE ÖNERİLER .....</b>	<b>53</b>
<b>KAYNAKLAR .....</b>	<b>57</b>
<b>EKLER.....</b>	<b>65</b>
<b>EK-1.....</b>	<b>66</b>
<b>ÖZGEÇMİŞ .....</b>	<b>67</b>

## ŞEKİLERİN LİSTESİ

Şekil	Sayfa
Şekil 2.1. Mekanik göstergelerin çalışma mekanizmaları .....	6
Şekil 2.2. Sinyal bobinli hız ölçüm sensörü.....	9
Şekil 2.3. Hal etkili hız ölçüm sensörü .....	10
Şekil 2.4. Kızılötesi hız sensörü.....	10
Şekil 2.5. ATmega328P mikroişlemcisi çıkış pini özellikleri .....	15
Şekil 2.6. Arduino kod editörü ile mikroişlemciye yazılım yüklenmesi .....	17
Şekil 2.7. Bilgisayar ile mikroişlemcinin bilgisayara bağlanması. ....	18
Şekil 3.1. Prototip çalışma uygulama şeması.....	23
Şekil 3.2. Uygulamanın elektronik şeması .....	24
Şekil 3.3. ECE-15 şehir içi çevrimi .....	26
Şekil 3.4. ECE-15 şehir içi çevrimi 1. Kısım.....	26
Şekil 3.5. Görev yöneticisinde arduinonun 28 numaralı COM-port bağlantısı. ....	27
Şekil 3.6. Modelin gerçek elemanlarla kurulumu. ....	29
Şekil 3.7. ATmega238P mikroişlemcisinin kullanılan bacak yerleşimleri.....	30
Şekil 3.8. Arduino UNO geliştirme kartı kullanılan pin isimleri.....	30
Şekil 3.9. HC-05 bluetooth modülü ile arduino UNO bağlantısı kullanılmıştır. ....	31
Şekil 4.1. ECE-15 şehir içi çevrimi 0-195 saniye aralığı.....	37
Şekil 4.2. Görev yöneticisinde orijinal arduino COM port bağlantısı .....	39
Şekil 4.3. Gerilim bölücü hesaplama. ....	41
Şekil 4.4. Batarya ve sıcaklık ölçümü.....	42
Şekil 4.5. Analog veri okuma .....	42
Şekil 4.6. Mikroişlemci ve bluetooth modüllerinin birbirine bağlanması .....	43
Şekil 4.7. Arduinoların TWI protokolü ile haberleşmesi.....	44
Şekil 4.8. Panel görsel tasarımları.....	45
Şekil 4.9. Uygulama üzerinde gösterilecek görsel.....	45

<b>Şekil</b>	<b>Sayfa</b>
Şekil 4.10. Matlab arayüzünde oluşturulan grafik görsel .....	46
Şekil 4.11. Panel görseli.....	49
Şekil 4.12. Motor hız kontrolü.....	50
Şekil 4.13. Uygulamanın bağlantı şekli .....	51

## RESİMLERİN LİSTESİ

Resim	Sayfa
Resim 2.1. Analog gösterge (Mini Cooper (1964)) .....	6
Resim 2.2. Dijital gösterge Dodge 600 (1984) .....	7
Resim 2.3. Mercedes S65 Coupe (2015) .....	7
Resim 2.4. BMW 6 Series Coupe .....	8
Resim 2.5. Grafik bar şarj seviyesi göstergesi .....	12
Resim 2.6. Arduino MEGA .....	13
Resim 2.7. ATmega328P mikroişlemcisine sahip arduino NANO geliştirme kartı .....	14
Resim 2.8. ATmega328P mikroişlemcisine sahip arduino UNO geliştirme kartı .....	14
Resim 2.9. ATtiny85 .....	15

## SİMGELER VE KISALTMALAR

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

<b>Simgeler</b>	<b>Açıklamalar</b>
<b>CAD</b>	Computer Aided Design (bilgisayar destekli tasarım)
<b>CAN-BUS</b>	Controller Area Network Bus (Kontrol Alan Ağı Veri Yolu)
<b>ECU</b>	Engine Control Unit (Motor Kontrol Ünitesi)
<b>ESD</b>	Electrostatic Discharge
<b>HUD</b>	Head-Up Display
<b>I2C</b>	Inter-Integrated Circuit (Bütünleşik Devreler)
<b>IDE</b>	Integrated Development Environment
<b>ISO</b>	International Organization for Standardization (Uluslararası Standartlar Teşkilatı)
<b>PWM</b>	Pules Width Modulation (Dalga Genişlik Modülasyonu)
<b>SPI</b>	Serial Peripheral Interface (Seri Çevresel Arayüz)
<b>TÜBİTAK</b>	Türkiye Bilimsel ve Teknolojik Araştırmalar Kurumu
<b>USB</b>	Universal Serial Bus (Evrensel Seri Veri Yolu)

## 1. GİRİŞ

Taşıt gösterge panellerinin geliştirilmesinde sürücüye gerekli bilgilerin zamanında ve uygun şekilde gösterilmesi ön plana çıkmaktadır. Bu sebeple gösterge panelleri taşıt ön panel tasarımının ana omurgasını teşkil etmektedir. Ergonomik ve kullanılabilir bilgileri içeren gösterge paneli tasarımı, sürücünün sürüş performansını artırmakta ve sağlığını olumlu şekilde etkilemektedir.

Elektrikli ve hibrit taşıtlarda kullanılan güç kaynağı sayısı donanımdan kaynaklı bilgilerin yoğunluğu klasik taşıtlara göre daha fazladır. Bu taşıtların enerji yönetim sistemleri, bilgi akışı ve kontrol sistemleri taşıta gücü sağlayan kaynakların organizesini sağlamakta ve taşıtta optimum enerji harcanmasını gerçekleştirmektedir. Sürüş esnasında sürücünün gerekli hız, şarj durumu ve diğer donanımların durumunu görmesi ve sürüşünü en ekonomik şartlarda gerçekleştirmesi büyük oranda gösterge panelinin görselliği ve sürücü tarafında bunun algılanmasına bağlıdır.

Bunun sağlanması, gösterge panelinin tasarımına, bilgilerin yerleşimine ve renk seçimine bağlıdır. Gösterge panelinde gerekli bilgilerin oluşturulması birçok ünitenin bilgilerinin değerlendirilerek panele yansıtılmasına bağlıdır.

Elektrikli bir otomobilin çalışabilmesi için, birbirinden bağımsız çalışan ünitelerin ana kontrol ünitesi tarafından kontrol edilip birbirleriyle uyumlu çalışacak şekilde denetlenmeleri gerekir. Bu çalışmanın amacı, böyle bir kontrol ünitesi ve buna bağlı belirli görevleri yerine getiren kontrol üniteleri ile haberleşmelerini tasarlamaktır. Bu amaçla bu çalışma, sistemin geliştirilebilirliği de ön planda tutularak UDC (şehir içi çevrim) bilgilerinin yüklü olduğu Matlab kurulu bir bilgisayar ve bu bilgisayarın haberleştiği mikroişlemci grubu üzerine kurulması düşünülmüştür.

Gösterge panelinin oluşturulmasında tüm bilgilerin panele aktarılması, oldukça zorlu bir süreç olduğundan, belirli sayıda sistemin bilgi aktarımının ve iletişiminin gerçekleştirilmesi planlanmıştır. Bu sistemlerin iletişimi sağlanırken modüllerin her birini ayrı ayrı oluşturmak yerine, belirli görevler yerine getirilirken belirlenen ayrı görevleri birbirleriyle uyumlu çalışacak şekilde farklı görevler seçilmiştir. Bu görevler sırasıyla, Matlab tarafından gönderilen UDC bilgileri doğrultusunda bir dc motorun uygun devirlerde döndürmek ve bu

devri kontrol etmek, anlık batarya seviyelerini kontrol ederken, batarya sisteminin sıcaklığını kontrol etmek, bu bilgileri bilgisayarda ve gösterge panelini temsil eden tablet üzerinde göstermektir.

Yapılan çalışmalarda belirlenen bu görevleri yerine getirebilmek için tasarım, yazılım ve donanım bileşenlere ihtiyaç olduğu görülmüştür. Bu çalışmada gösterge panelinin yukarıda açıklandığı şekilde farklı görevleri gerçekleştirebilmesi için UDC bilgilerinin gönderilebilmesi için Matlab programı, gösterge panelinin görsel tasarımı için AutoCAD, 3d max ve Photoshop programı, donanımın oluşturulması ve geliştirilmesi için arduino yazılım geliştirme ortamı, bilgilerin kablosuz olarak iletilebilmesi için HC-05 bluetooth modülü, dc motoru kontrol etmek için L293 motor sürücü, gösterge panelini temsilen android işletim sistemi kurulu bir tablet ve telefonda gerekli grafiklerin oluşturulabilmesi için Unity 3d programı kullanılmıştır.

Bu çalışma ile yapılacak prototip çalışmanın, gelecekte elektrikli taşıt kontrol sistemlerinin geliştirilebilmesi ön çalışmasını teşkil edeceği düşünülmektedir.

Bu sebeple elektrikli bir taşıtın, donanımlarının istenen şartlarda çalışmasını sağlamak amacıyla, donanımların ayrı modüller olarak kurulması gerekmektedir. Sistem, ana mikroişlemciyle farklı mikroişlemcilerin haberleşmesi ve gelecek çalışmalar için, sonradan eklemeler yapılacak şekilde düzenlenmelidir. Donanımlar, prototipin gereksinimleri göz önüne alınarak, ön çalışma amacıyla yapılmalıdır. Kullanılan yazılımlar, gerçek veriler kullanılarak, donanımın kontrol edilmesine olanak sağlamaktadır (Allio, 2012).



## 2. LİTERATÜR ÖZETİ

Otomobil endüstrisinin gelişmesiyle birlikte gösterge panelleri araç seyir halinde iken sürücüyü araç hakkında bilgilendirmek ve ikaz etmek için kullanılmıştır. Kullanılan göstergeler, aktif ve pasif güvenlik sistemleri, motor bilgileri, taşıt hızı, ikaz sistemleri ve aydınlatma gibi bilgiler ile sürücüyü uyarmakta veya yapılması gereken için bilgilendirmektedir (Sureshababu, Mohan, Santhiyakumari ve Thangavel, 2015; Riechers, 2019). Bununla beraber dijital gösterge panellerinin, analog gösterge panellere göre belirli üstünlükleri vardır (Shashidhara, Deepak, Manjunath, Murthy ve Ruckmongathan, 2004).

### 2.1. Elektrikli Taşıtlar

Günümüzde genel olarak içten yanmalı motorlar kullanılmasına rağmen çevresel etkenlerden dolayı elektrikli taşıtlar trafikte daha fazla yer almaya başlamıştır (Cadex Electronics Inc., 2019). İçten yanmalı motorlu taşıtlara göre en önemli farkları, taşıta hareketin dc motorları tarafından sağlanıyor olmasıdır (Martino, 2012). Elektrikli taşıtlarla ilgili araştırmalar yoğun bir şekilde devam ederken, en fazla uygulama alanı, insansız ve otonom araçlar üzerine devam etmektedir (Ayas, 2015). Elektrikli taşıtlarda taşıt seyir halinde iken bataryaların enerji seviyelerinin sürekli kontrol edilmeleri gerekmektedir (Uğur, 2011; Muratoğlu, 2017). Elektrikli taşıt bataryalarının taşıta sağlayabildikleri güç, genellikle batarya kapasitelerine bağlıdır (Sarıkurt ve Balıkçı, 2017; Yol 2019). Bu sebeple, elektrikli taşıtlar içten yanmalı motorlu taşıtlarla kıyaslandığında, ilave batarya kontrol sistemlerine ihtiyaç duyarlar. Bu bataryalar özelliklerinden dolayı özel şarj cihazlarına ihtiyaç duyarken şarj sürelerine bu şarj cihazları büyük oranda etkili olmaktadır (Tezde ve Okumuş, 2018).

Türkiye’de sürdürülen tahrik sistemi olarak dc motoru kullanan elektrikli taşıtların ülkemizde üretilmesi için Haziran 2015’da başlayan “new Swedish Saab” ve TÜBİTAK ortak girişimi sayesinde Saab 9-3 modelinin, 2020 yılında piyasaya sürülmesi planlanmaktadır (Feasibility Study Automotive Production in Konya, 2016). Bu sayede ülkemizde de elektrikli taşıt sistemleri ile ilgili “know how” çalışmalarının gelişen teknoloji ile birlikte ilerlemesi gerçekleştirilebilecektir. Bu sebeple elektrikli taşıtlarda insan-makine etkileşimi için ara yüzü geliştirme çalışmalarının (Gerlach, 2011), gösterge panelinde dikkat dağıtıcı ikaz durumunda, sürücü davranışının incelenmesi (Morrow, Cole ve Davis, 2011)

ve yeni nesil gösterge panelleri tasarımı (Jeon, Schuett, Yim, Raman ve Walker, 2011) gösterge panellerinde ISO standartlarının (Johann ve Mahr, 2011) bilinmesi önem arz etmeye başlamıştır. “Open Car User Experience Lab” ile kullanıcı ara yüzü oluşturulması ve gösterge panellerinin araç yönetim sistemlerinde, sürücüye gerekli bilgileri göstermesinin sağlanabilmesi amaçlanmıştır (Wilfinger, Murer ve Tscheligi, 2011). Taşıtta arıza durumlarında hata tanılama sistemlerinin geliştirilip olası arızaların sürücü tarafından algılanıp gerekli tedbirlerin alınması gerekmektedir (Falke ve Krömker, 2011). Bunların en başında şarj gereksinimi gelmektedir (Fotouhi, Propp ve Auger, 2015). Sürücünün bu gibi durumlarda önceden bilgilendirilip, o anda gidilecek mesafeye göre şarj ihtiyacını gidermesi gerekmektedir. Bunun haricinde geliştirme çalışmalarında sistem bilgilerinin gerek lokal, gerekse internet aracılığıyla depo edilmesi gereklidir (Marques ve Pitarma, 2019). Sistem geliştiricileri gerçek zamanlı bilgilere ulaşılırken önlemlerini önceden alıp gerekli programlamayı yapabilirler.

Bu alanda yapılan çalışmalar arasında kullanıcı denetiminde olan sesli komut ile çalıştırma (Hofmann, Ehrlich ve Andreas, 2011), el hareketleri ile donanım kontrolü (Riener ve Rossbory, 2011), mesafe sensörü ve artırılmış gerçeklik ile HUD yardımı ile konsol kontrolü (Riener ve Wintersberger, 2011) sayılabilir. Seyir halinde trafik işaretlerinin tanınarak sürücüye bildirilmesi (Hyun, Hwang, Ryu ve Cho, 2011) ve güvenli sürüş ikaz sistemleri (Zhou, Itoh ve Inagakı, 2011) ortam bilgilerini sürücüye bildiren sistemlerdir.

## **2.2. Gösterge Panelleri**

Gösterge panelleri analog ve dijital göstergeler olmak üzere iki grupta incelenebilmektedir. Gösterge paneli, göstereceği bilgileri kendisinin de dâhil olduğu gömülü sistemlerden almaktadır (Noergaard, 2005). Araç sürücülerinin çalıştıkları bölgenin zayıf tasarımı sebebiyle uzun süre direksiyon başında çalışan sürücülerde ergonomik tasarım eksiklikleri sebebiyle ki bunların başında gösterge panelleri gelmektedir, kalıcı duruş bozukluklarına sebep olmaktadır (Samuel, Israel ve Moses, 2016). Gösterge panel tasarımı günümüzde tekrar elden geçmeli, tasarım değiştirilmelidir (Everdell, 2019). Gösterge panellerinin dijital gelişimi sayesinde araç sürücülerine daha ergonomik ve kullanışlı paneller oluşturulabilir hale gelmiştir (Gibson, Butterfield ve Marzano, 2016).

Sürüş esnasında, yola verilen dikkat ile gösterge panelinin de olduğu alanı aynı anda görmek için bakış alanını taşıt içinden bakıldığında gösterge paneli ve yolun aynı anda görülmesine dikkat edilmesi gerekmektedir. Görülmesi gereken alan yol görüntüsünden geniştir. Ergonomik olarak hem yol görülmeye hem de gösterge paneli görülmeye çalışıldığında alan genişlediği için sürücüler için bir karmaşa oluşturmaktadır. Gösterge paneli tasarımı yapılırken bu hususların ön planda tutulması gerekmektedir (Pitts, Wellings, Attridge ve Williams, 2011). Tasarım esnasında, sanal görselleştirme ve gerçek sürüş deneyimi esnasında görsel ve tasarımsal etkenlerden dolayı farklılıkların oluşmaktadır. Bu farklılıklar, son tasarımın şekillenebilmesi için yapılacak geliştirmelerin sürekli olması gerektiğini göstermektedir.

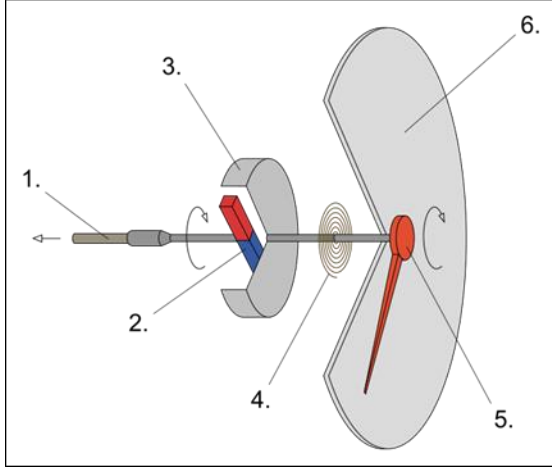
Sanal ortam ve gerçek sürüş esnasındaki farklılıklar, tasarımı yapacak kişinin, genel değişkenleri bilerek ve kendi test ederek yapacağı geliştirmeler sonucunda uygulamaya konulabilmiştir. Sürücünün fiziksel özellikleri de tasarım esnasında göz önünde bulundurulmalıdır (Baldauf, Schabus, Fuchs, Fröhlich ve Suetter, 2011).

### **2.2.1. Analog göstergeler**

Analog gösterge sistemi, hız göstergesi km göstergesi, şarj göstergesi, yakıt göstergesi, motor devir göstergesi ve hararet göstergesini içermektedir. Bu sistemlerde sistemin çalışmasında kablo ile sistemin bilgisi gösterge paneline aktarılmaktadır. Örneğin hız, vites kutusu çıkış milinden, yakıt, yakıt deposundan, hararet ise motordan sıcaklık müşirinden alınan gerilimler veya mekanik hareketin etkisi ile gösterge paneline aktarılmaktadır.

Analog gösterge panellerinde, sistemin çalışması mekanik olduğundan tasarım esnasında boyutlandırma sebebiyle kısıtlamalar oluşmaktadır. Bu gösterge panellerinde mekanik parçalar bulunduğundan belirli bir alana ihtiyaç duymakta, gerekli bilgilerin gösterilebilmesi için gerekli bilginin de mekanik olması sebebiyle her bir bilgi için ayrı kablo veya mekanik elemana ihtiyaç duyulmaktadır. Bu da tasarım esnasında gerek bağlantı sayısı, gerekse panelin boyutlandırılması açısından kısıtlamalar getirmektedir. Şekil2.1’de hareketli bir milden alınan dönme hareketi, gösterge paneline kadar getirilerek hareket, gösterge panelindeki ibreyi hareket ettirmekte ve o anki durum bilgisini ön panel üzerinde sürücünün bilgisine sunmaktadır (Franco ve diğerleri, 2017). Ayrıca mekanik gösterge panellerinde her bir ikaz lambası için, bilginin alınmasında ayrı bir kablo grubuna ihtiyaç duyulmaktadır. Bu

da dijital gösterge panellerini tasarım esnekliği sebebiyle ön plana çıkarmaktadır. Resim 2.1’de 1964 model bir Mini Cooper’a ait analog gösterge paneli görülmektedir.



Şekil 2.1. Mekanik göstergelerin çalışma mekanizmaları



Resim 2.1. Analog gösterge (Mini Cooper (1964))

### 2.2.2. Dijital göstergeler

Dijital gösterge panellerinde, bilgiler, dijital olarak panele gelmekte, panel üzerinde bulunan mikro işlemciler sayesinde panel üzerindeki gerekli göstergelere bilgi ulaştırılarak sürücünün anlayacağı formatta görselleştirilmesini sağlamaktadır.

Bu gösterge sisteminde yedi bölümlü sayısal ekranlardan oluşmakta olup, göstergelerin bir kısmı sayısal ekranda, diğer bir kısmı ise analog görüntülü olarak gösterilmektedir. Analog görüntülerin gösterilmesinde ibrelerin hareketi için step motorlar kullanılmıştır. Bu motorların kullanılması gerekli enerji ihtiyacını artırmaktadır.

Bu sistem daha da ilerletilerek göstergelerin bir kısmı LCD ekranlara aktarılmaya başlanmıştır. Bu ekranların kullanılmasıyla bilgiler sensörler vasıtası ile ECU'dan alınmış ve ekranlara yansıtılmıştır. İlk uygulamaları yol bilgisayarlarına olan LCD ekran uygulaması, daha sonra aydınlatma sisteminde farların durumunu ve cam sileceğinin sıvı seviyesinin gösterimine kadar geniş bir alanda kullanılmaya başlanmıştır. Resim 2.2'de 1984 model Dodge 600'e ait dijital gösterge görülmektedir (Lavery, 2019).



Resim 2.2. Dijital gösterge Dodge 600 (1984)

Resim 2.3'te ise LCD ekranlı Mercedes S64 Coupe'ye ait dijital bir gösterge görülmektedir (Jasper, 2014). Bu tür göstergelerde görüntü LCD panelde gösterilmekte olup, görüntü modları istenildiği gibi değiştirilebilmektedir. Bu sistemlerin enerji gereksinimleri sabit olup, enerji sarfiyatının düşük olması sebebiyle elektrikli taşıtlarda tercih edilmektedirler (Zhu, Wang, Henaut ve Beeby, 2014; Scott, Vunakece, Vosawale, Assaf ve Mehta, 2019).



Resim 2.3. Mercedes S65 Coupe (2015)

Günümüzde HUD (Head-Up Display) göstergeler de dijital göstergeler arasında yer almaktadır. Resim 2.4'te BMW 6 serisi Coupe araca ait ön cama yansıtılmış HUD görsel görülmektedir (Meister, 2015).



Resim 2.4. BMW 6 Series Coupe

Gösterge panellerinin tasarımı konusunda yapılan çalışmalar (Zhou, 2011) ve (Jınadasa, 2016) tarafından detaylı olarak incelenmiş ve tasarımlar yapılmıştır.

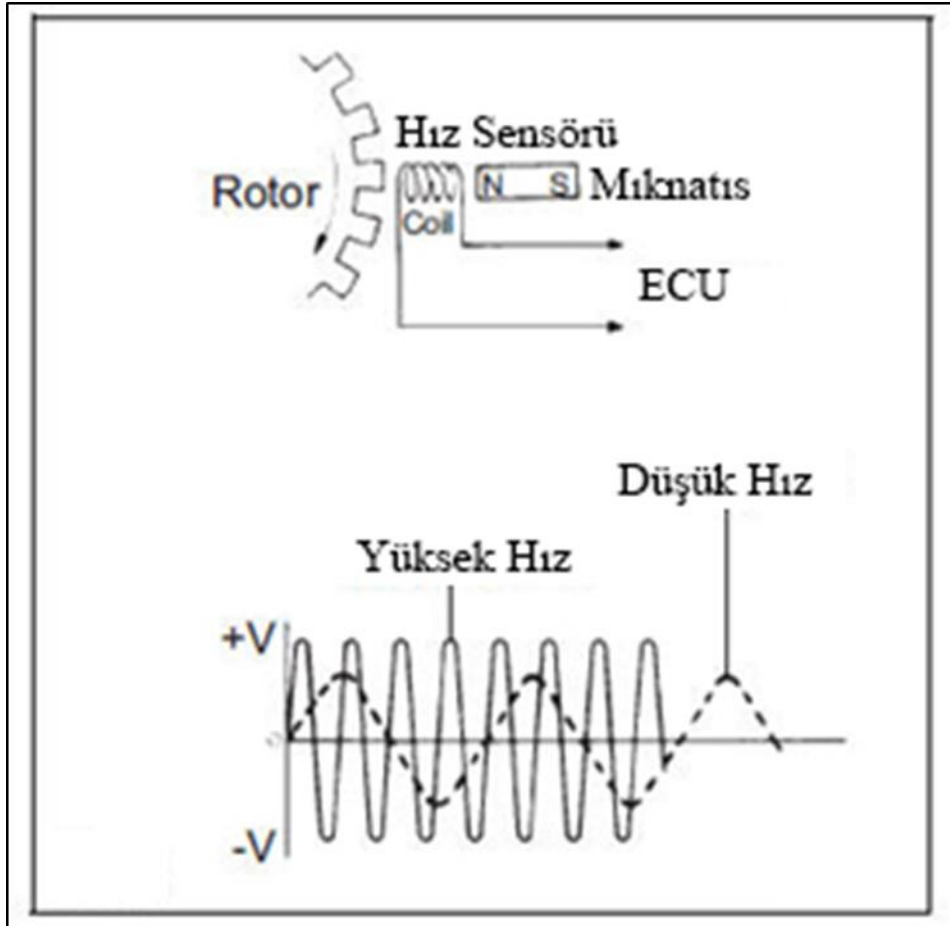
Dijital gösterge panellerinde, tasarım süreçleri belirli aşamalardan oluşmakta ve bu aşamalar esnasında tasarım, gerek konum gerekse görsel görünüm için kullanılacak alan sebebiyle sürekli değişmektedir (Vilarinho, Lopes ve Sousa, 2017; Mantovani, Lo Presti, Cavazzoni ve Baldini, 2017; Porter, Case, Freer ve Bonney, 1993).

Günümüzde dijital gösterge panelleri tasarlanırken araç gereksinimleri, sürücü davranışları ve araç üzerinde bulunan donanımlara bağlı olarak bilgisayar destekli tasarım ortamlarının yanında, test aşamalarında artırılmış gerçeklik uygulamaları kullanılmakta, araç içi kontrol ünitelerinin haberleşmelerinde kablolu karmaşası ve maliyetlerini düşürmek için BUS tipi araç içi haberleşme sistemleri kullanılmaktadır (Gröger, Hillmann, Hahn, Mitschang ve Westkämper, 2013; Gibson, Butterfield ve Marzano, 2016).

Taşıt gösterge paneli tasarımında gerek mevcut sistemler üzerinde, gerekse yeni geliştirilen sistemler üzerinde kullanıcı testleri yapılarak panel üzerinde gösterilecek bilgiler belirlenmektedir (Albert, Toledo ve Grimberg, 2011). Yapılan bu testler esnasında, sürücü davranışları ve dikkat dağıtıcı unsurlar da incelenerek gerekli görüldüğü durumlarda, park sensörü gibi, ikazlara ses ikazı da eklenmekte, sürücünün durum hakkında daha fazla bilgiye ihtiyaç duyduğu vurgulanmaktadır (Neureiter, Meschtscherjakov, Wilfinger ve Tscheligi, 2011; Islinger, Köhler ve Ludwig, 2011; Osswald ve Tscheligi, 2011).

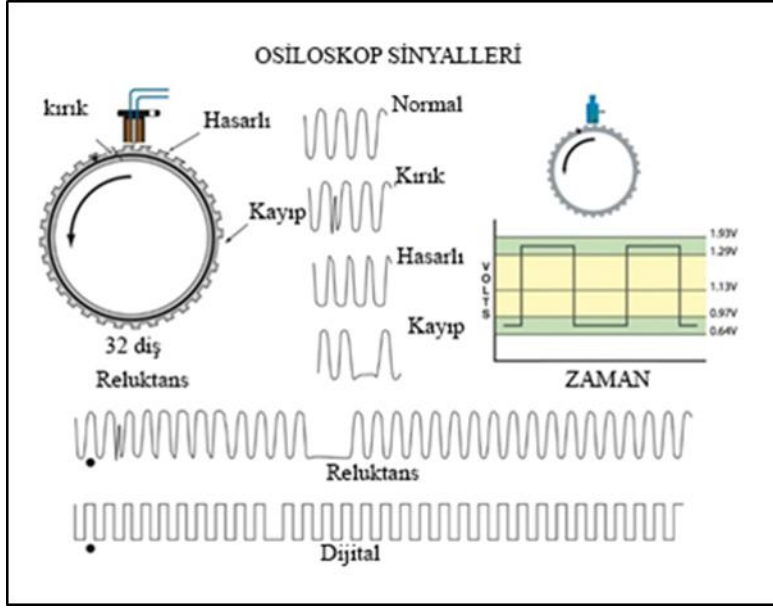
### 2.3. Kullanılan Sensörler ve Elektronik Elemanlar

Gösterge panellerine gerekli bilginin aktarılması ve görüntülenmesinde sensörler büyük öneme sahiptir. Gerek kontrol ünitesinin diğer ünitelerle iletişime geçmesi ve gerekse sistemlerin kontrolünde, ünitelerin görevleri yerine getirmesinde sensörlere ihtiyaç vardır. Hızın belirlenmesi için tekerlek veya motor hız sensörüne, batarya kapasitesinin tanımlanması için batarya kapasitesi ile gerilim arasında ilişkiyi tanımlayan devre ve gerilim sensörü gerekmektedir. Şekil 2.2’de Toyota Siena’ya ait taşıt hız sensörünün çalışması temsili olarak görülmektedir (Toyota Sienna, 2019). Şekil 2.3’de ise dişleri hasarlı bir devir/hız sensöründe oluşan sinyal hataları görülmektedir (Markel, 2013).



Şekil 2.2. Sinyal bobinli hız ölçüm sensörü

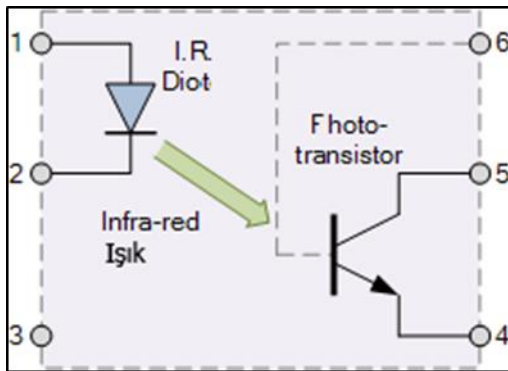
Sensörler genel olarak gerilim üreten elemanlar olup bu elemanlar analog gerilimleri dijital sinyallere dönüştüren mikroişlemcilerle göndermektedir. Mikroişlemciler için yazılan programlar bu bilgi ile değerlendirme yaparak bilginin iletişim veya kontrol birimine aktarılmasını sağlamaktadır.



Şekil 2.3. Hal etkili hız ölçüm sensörü

### 2.3.1. Kızılötesi devir ve hız sensörü

Bir adet kızılötesi LED ve karşısında fototranzistör bulunan hız sensörüdür (Uzun ve Ayan, 2015). Genellikle karşılıklı kurulu halde bulunmaktadır ve gerekli devre elemanları ile beraber, mikroişlemciye, ışık ve sensör arasından her bir nesne geçişinde mikroişlemciye gönderilecek sinyal üretmektedir. Şekil 2.4'te kızılötesi hız sensörüne ait devre şeması görülmektedir.



Şekil 2.4. Kızılötesi hız sensörü

Taşıtın hızının belirlenmesinde tekerlek yarıçapından faydalanılmaktadır. Bir taşıtın hızı, tekerinin çevresel hızına eşittir. Eş.1'de, tekerin çevresel hızı dolayısıyla taşıtın hızını veren bağıntı verilmiştir.



$$\omega v = V/R \quad (2.1)$$

Formülde;

$\omega v$  = tekerin çevresel hızını,

$v$  = aracın doğrusal hızını,

$R_r$  = teker yarı çapını,

$\omega v$  = tekerin açısal hızını göstermektedir (The MathWorks, Inc., 2019b).

Şekil 2.2 ve Şekil 2.3 göz önüne alındığında, her bir turda rotor disk üzerinde bulunan diş sayısı kadar darbe sinyali ECU'ya gönderilmektedir. Rotor diski üzerinde bulunan diş sayısı formül üzerine P olarak yazıldığında taşıt hızını hesaplamak için kullanılacak eşitlik Eşitlik 2'de verilmiştir.

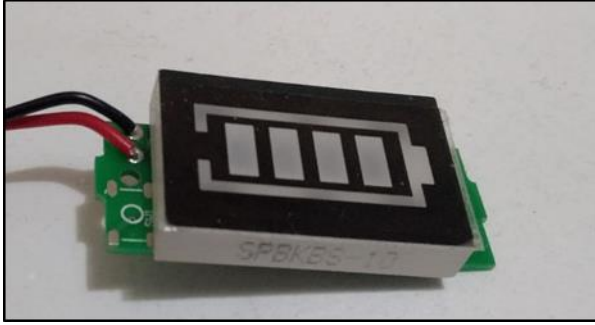
$$\omega v = v/(R \times P) \quad (2.2)$$

Bu eşitlikte hızın ( $v$ ) birimi m/s olması durumunda eşitliğin 60 ile bölünmesi sonucunda devir/dakika cinsinden tekerin açısal hızı bulunmaktadır.

### 2.3.2. Gerilim bölücüler

Elektronik devrelerde elektrik seviyelerinin ölçümü esnasında, ADC devreleri kullanılmaktadır (Scherz ve Monk, 2013). Elektrikli taşıtlarda kullanılan batarya sistemleri, aracın dc motorlarını hareket ettirecek seviyeye yükseltilerek aracı hareket ettirir. Buna karşı elektronik devrelerin çalışma gerilimleri 5V seviyesindedir. Batarya seviyelerini doğru ölçmek için kullanılacak gerilim bölücü, ikili direnç takımından oluşmaktadır ve batarya gerilimini oranlayarak şarj seviyesinin ölçülmesine olanak sağlamaktadır. Gerilim bölücü hesaplaması, batarya şarj devresinin kurulması bölümünde detaylı bir şekilde anlatılacaktır.

#### 2.4. Grafik Bar 3.7V Pil Şarj Göstergesi



Resim 2.5. Grafik bar şarj seviyesi göstergesi

Batarya durumunu göstermek için çalışmalarda mikroişlemcilerden hariç, grafik bar 3,7V şarj seviyesi göstergesi kullanılmaktadır. Ölçüm uçları arasında gerilim olduğunda dış kısımdaki gösterge yanmaktadır. Şarj seviyesine göre 3,2V-3,7V aralığına voltajı %25 seviyelerinde ölçmektedir. Daha yüksek seviyelerde voltajları ölçebilmek için ölçülecek voltaj ile grafik bar ölçüm uçları arasına gerilim bölücü devre elemanlarının eklenmesi gerekmektedir (Cadex Electronics Inc., 2019). Resim 2.5'te batarya şarj seviyesini göstermek üzere özel olarak üretilen bir şarj durum göstergesi görülmektedir.

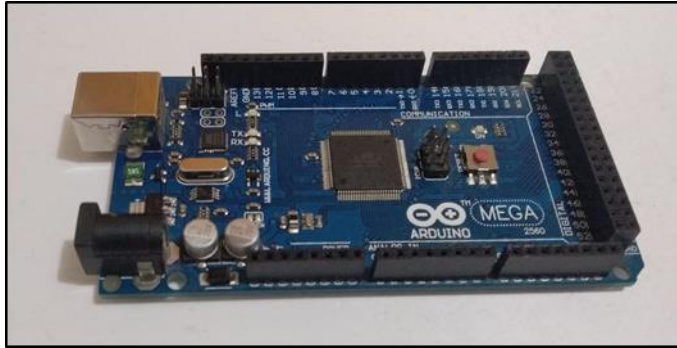
#### 2.5. Mikrodenetleyiciler

Veri haberleşme yapıları ve üzerine kolay kod yazılabilmesinden dolayı arduino yazılım geliştirme platformu ile uyumlu mikroişlemciler kullanılmaktadır. Arduino uygulama geliştirme kartları farklı özelliklerde bulunmalarına rağmen, arduino UNO, arduino MEGA ve ATtiny85 geliştirme kartları çeşitli devrelerde kullanılmıştır (Topsakal, Yılmaz ve Tayfur, 2016; Tarlak ve İşen, 2018).

Arduino uygulama geliştirme kartları, bilgisayarın COM portu üzerinden RS232 protokolü ile haberleşmektedir (Uzun ve Ayan, 2015). Arduino geliştirme kartları üzerinde bulunan mikroişlemcilerde, UART, SPI, I2C ve seri haberleşme pinleri bulunmakla beraber, bu pinler, gerekli durumlarda dijital giriş-çıkış pini olarak da kullanılmaktadır (Uzun ve Ayan, 2015; Köylü, 2011; Microchip, 2019).

### 2.5.1. Arduino MEGA

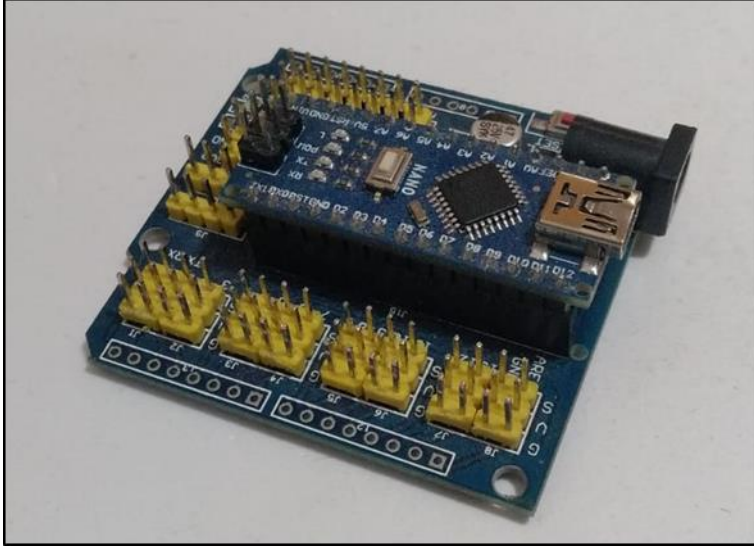
Çalışma voltajı 5V'tur ve üzerindeki voltaj regülatörü sayesinde 7-12V arasındaki voltajlarla da beslenebilmektedir. Üzerinde ATmega2560 mikroişlemcisi bulunmaktadır. 15 adet çıkışı PWM olmak üzere 54 adet dijital giriş-çıkış pini, 16 adet analog giriş pini sayesinde çok sayıda giriş-çıkış sinyaline ihtiyaç duyulduğunda kullanılmaktadır (Uzun ve Ayan, 2015). 256KB program hafızası ile yüklü program gerektiren uygulamalarda kullanılmaktadır. Çalışmalarda ana işlemci olarak kullanılmıştır. Resim 2.6'da bir arduino MEGA görülmektedir.



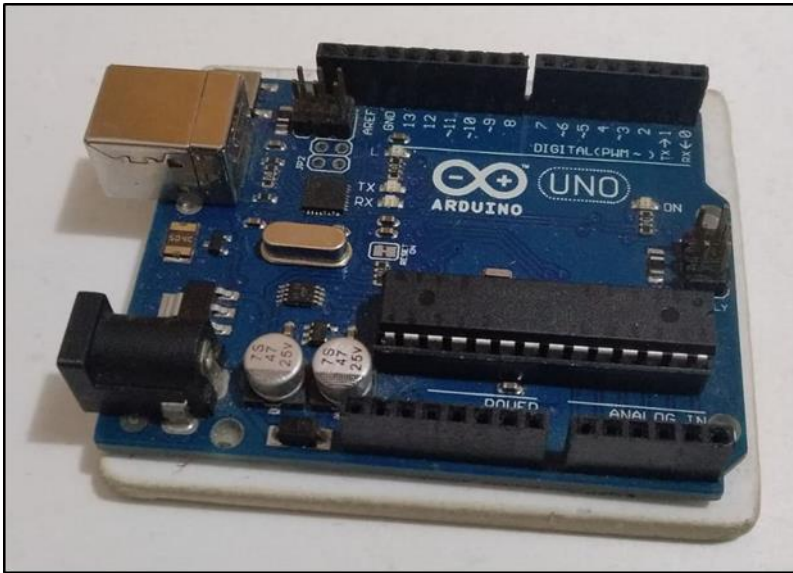
Resim 2.6. Arduino MEGA

### 2.5.2. Arduino UNO-NANO

Üzerinde 6 adet çıkışı PWM özellikli olmak üzere 14 dijital, 6 analog giriş pini bulunan geliştirme kartıdır (Topsakal, Yılmaz ve Tayfur, 2016). Üzerinde bulunan işlemci ATmega328P mikroişlemcisidir. 5V ile çalışmaktadır ve üzerinde bulunan voltaj regülatörü sayesinde 7-12V aralığında voltajlarla dahi çalışmaktadır. 32KB program hafızası ve 16Mhz ile çalışmaktadır (Uzun ve Ayan, 2015). Bilgisayar haberleşmesi, motor sürücüyü kontrol etme ve motor devrini ölçme amacıyla kullanılmışlardır. Arduino UNO ve arduino NANO üzerinde aynı mikroişlemci bulunmaktadır (Kandemir, 2011). Aralarında yapısal olarak elektronik kart ölçü farklılığı ve bilgisayar bağlantı için kullanılan USB soket giriş farkı bulunmaktadır (Uzun ve Ayan, 2015). Resim 2.7'de sensör kartı üzerine monte edilmiş bir arduino NANO görülmektedir. Resim 2.8'de ise bir arduino UNO uygulama geliştirme kartı görülmektedir.



Resim 2.7. ATmega328P mikroişlemcisine sahip arduino NANO geliştirme kartı



Resim 2.8. ATmega328P mikroişlemcisine sahip arduino UNO geliştirme kartı

Üzerinde 6 adet giriş-çıkış pini bulunmaktadır. Bunlardan 4 tanesi analog pin olarak kullanılabilir. batarya şarj seviyesini ölçmek için. 8KB program hafızası bulunmaktadır. 5V ile çalışmaktadır ve üzerinde bulunan voltaj regülatörü sayesinde 7-16V ile çalıştırılabilir. Şekil 2.5'te ATmega328P mikroişlemcisinin pinleri ve özellikleri görülmektedir. Resim 2.9'da ise bir ATtiny85 deneme kartı görülmektedir.

Arduino function					Arduino function
reset	(PCINT14/RESET) PC6	1	28	PC5 (ADC5/SCL/PCINT13)	analog input 5
digital pin 0 (RX)	(PCINT16/RXD) PD0	2	27	PC4 (ADC4/SDA/PCINT12)	analog input 4
digital pin 1 (TX)	(PCINT17/TXD) PD1	3	26	PC3 (ADC3/PCINT11)	analog input 3
digital pin 2	(PCINT18/INT0) PD2	4	25	PC2 (ADC2/PCINT10)	analog input 2
digital pin 3 (PWM)	(PCINT19/OC2B/INT1) PD3	5	24	PC1 (ADC1/PCINT9)	analog input 1
digital pin 4	(PCINT20/XCK/T0) PD4	6	23	PC0 (ADC0/PCINT8)	analog input 0
VCC	VCC	7	22	GND	GND
GND	GND	8	21	AREF	analog reference
crystal	(PCINT6/XTAL1/TOSC1) PB6	9	20	AVCC	VCC
crystal	(PCINT7/XTAL2/TOSC2) PB7	10	19	PB5 (SCK/PCINT5)	digital pin 13
digital pin 5 (PWM)	(PCINT21/OC0B/T1) PD5	11	18	PB4 (MISO/PCINT4)	digital pin 12
digital pin 6 (PWM)	(PCINT22/OC0A/AIN0) PD6	12	17	PB3 (MOSI/OC2A/PCINT3)	digital pin 11(PWM)
digital pin 7	(PCINT23/AIN1) PD7	13	16	PB2 (SS/OC1B/PCINT2)	digital pin 10 (PWM)
digital pin 8	(PCINT0/CLKO/ICP1) PB0	14	15	PB1 (OC1A/PCINT1)	digital pin 9 (PWM)

Digital Pins 11, 12 & 13 are used by the ICSP header for MOSI, MISO, SCK connections (Atmega168 pins 17, 18 & 19). Avoid low-impedance loads on these pins when using the ICSP header.

Şekil 2.5. ATmega328P mikroişlemcisi çıkış pini özellikleri



Resim 2.9. ATtiny85

## 2.6. Bluetooth Modülü

Gösterge paneline hız, devir ve batarya şarj seviyesi bilgileri kablosuz olarak gönderilebilmektedir (Houda ve Lakel, 2015). Bu amaçla mikroişlemci ile seri haberleşme özelliğine sahip HC-05 bluetooth modülü kullanılmıştır. HC-05 modülü 2.4GHz frekans bandında çalışmaktadır. 3,6V- 5V gerilim aralığında çalışabilmektedir (Wikizero, 2019).

## **2.7. L293 Motor Sürücü Kartı**

Motor sürücü kartı azami 2A akım ve 5V - 36V gerilim aralığında dc motorların hız ve devirlerini dijital sinyallerle kontrol edilmelerine olanak sağlamaktadır.

## **2.8. Kullanılan Yazılımlar**

### **2.8.1. Matlab yazılımı**

Matlab, mühendislik hesaplamaları için kullanılan akademik bir yazılımdır (NAM, 2019). Farklı dillerle olan oryantasyonunun yanında 2D ve 3D grafik oluşturulabilmektedir. Dış donanımlarla haberleşme olanakları ve sürekli kullanılacak döngüler yazılması gerektiğinde, alt programlara bölerek sistemin oluşturulmasında esneklik sağladığı için tercih edilmiştir (Erşan ve Suvak, 2016).

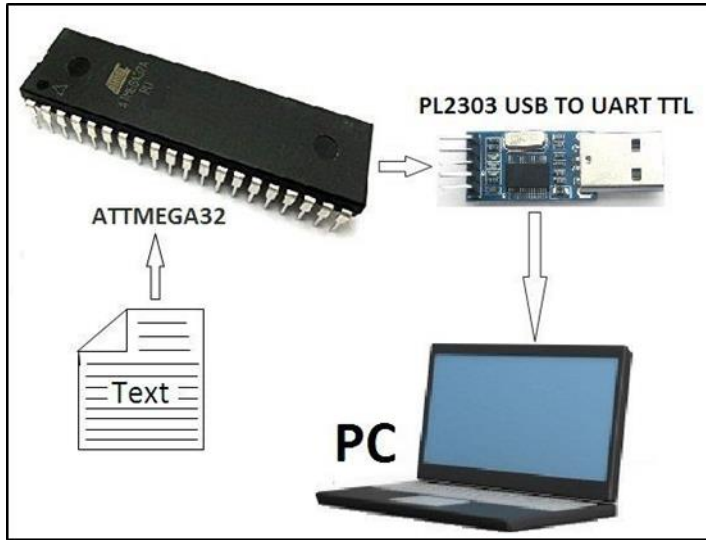
Burada geliştirilen yazılım, şehir içi çevrim bilgilerini dış donanımlara göndererek, dış donanımlardan gelen veriler değerlendirilecek şekilde yazılmıştır. İleriki aşamalarda Matlab ve bilgisayar devreden çıkartılarak araç verileri otomatik çalışma modunda, aracın gerçek seyri esnasında test edilmesine olanak verecek şekilde yapılabilecektir (Richard, Carlson ve Duoba, 2007).

Matlab yazılımı, matematiksel hesaplamaları yapabilmesinin yanında, bilgisayara kurulabilen yazılımlarla da entegre şekilde çalışabilmektedir. Excel, notepad ve Word belgeleri ile veri giriş-çıkışı yapabilmesinin yanında, ilave donanım eklendiğinde, kamera görüntüsü ile işlem yapabilme, sensör ve anahtarların durumlarını okuyabilmek için seri haberleşme protokolü RS232 haberleşmesi de sağlamaktadır (The MathWorks, Inc., 2019a ; The MathWorks, Inc, 2019c).

Matlab ile kod yazımı esnasında sürekli tekrarlayan işlemler için alt fonksiyonlar oluşturulurken (Rouleau, 2019), prototip geliştirme sürecinde harici donanımların test edilmesi işlemlerinin yanında gerekli durumlarda (Soyaslan, 2015), verilerin hesaplanabilmesi için simülasyon desteği de sunmaktadır (Lousberg, 2016; McDonald, 2012; Baysallı, 2010).

### 2.8.2. Arduino kod editörü

Arduino kod editörü, dünya genelinde kullanılan, akademik ve prototip çalışmalar, hobi devreleri ve yarı-profesyonel kullanıcılar için, farklı mikroişlemcilere program yazabilmek için kullanılan bir programlama ara yüzüdür.



Şekil 2.6. Arduino kod editörü ile mikroişlemciye yazılım yüklenmesi

Yaygın olarak ATmega328P mikroşlemcisi kullanılsa da daha gelişmiş ATmega1280, ATmega2560 mikroşlemcilerinin programlanması, Wi-fi destekli ESP8266, ESP32 mikroşlemcileri ve daha düşük pin sayısı ve kapasiteye sahip ATtiny85 mikroşlemcilerini de programlayabilmektedir. Şekil 2.6’da bir mikroşlemcinin elde ettiği verileri bilgisayara aktarma şekli görülmektedir.

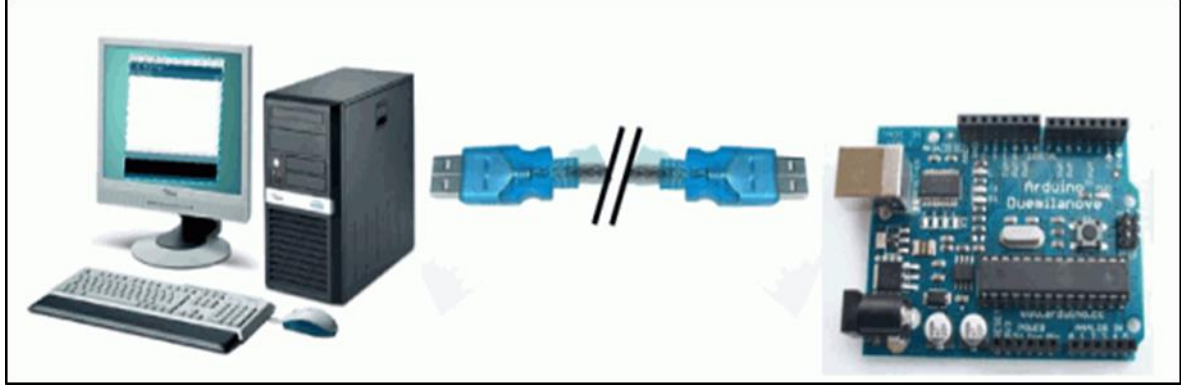
Programın bünyesinde bulunan seri monitör sayesinde, farklı mikroşlemcilerle bilgi alışverişinin yanında, gerekli kütüphanelerin eklenmesi, geliştirilmesi veya yazılması sayesinde daha karmaşık yazılımları sayısız mikroşlemciye yükleyebilen bir yazılımdır.

Kullanım alanının geniş olması, yukarda sayılan özellikleri ve açık kaynaklı bir yazılım olması, arduino kod editörünü elektronik ve yazılım konusuna yeni başlayanların yanında, profesyonellerin de tercihi olmasını sağlamıştır.

Yazılımın kurulmasından sonra sistemde yapılması gereken, uygun USB ara kablosu ile bilgisayara bağlanması, uygun yazılımın yazılıp yüklenmesidir. Şekil 2.7’de bilgisayar ve



mikroişlemcinin haberleşmesi için gerekli bilgisayar bağlantısının yapılış şekli görülmektedir.



Şekil 2.7. Bilgisayar ile mikroişlemcinin bilgisayara bağlanması.

### 2.8.3. 3ds Max

Autodesk firması tarafından sunulan 3B ürün tasarımı, modelleme, görselleştirme ve animasyon programıdır. Ölçülü modelleme ve görselleştirme imkânı sunan program ile araç sürücü kabini modeli üzerinden, gösterge paneli görselinin yerleşimi ve gösterilecek bilgilerin konumları belirlenebilmektedir. Aynı zamanda ergonomi ve görsellerin konumlandırılmalarının da tasarımı yapılabilmektedir.

### 2.8.4. AutoCAD

Autodesk firması tarafından sunulan ve imalat için ölçülendirme amaçları için kullanılan bilgisayar destekli tasarım programıdır. 3ds Max programında hazırlanan 3B modeller bu AutoCAD programı ile ölçülü hale getirilip tasarımın ölçülendirilmesi yapılabilmektedir.

### 2.8.5. Photoshop

Adobe firmasının sunduğu, resim oluşturma, işleme ve düzenleme programıdır. AutoCAD programında, ölçülü ve dijital taslak görsel üzerinden tablet üzerinde gösterilmek istenen resmin son hali oluşturulabilmektedir. Katman özellikleri sayesinde kadranslar üzerinde gösterilecek olan ibreler de bu program ile yapılabilmektedir.



### 2.8.6. Unity

Yapay zekâ desteği de sunan, çoklu platform desteği bulunan, 2B, 3B uygulama, simülasyon, oyun ve mühendislik programıdır. Unity içerisinde, PC, android ve IOS işletim sistemleri için uygulamalar geliştirebilen modüller bulunmaktadır. Bu program ile android işletim sistemi bulunan tablet için photoshop ile hazırlanılan görseller ve Java, C#, ve phyton programları ile entegre çalıştırılarak tabletin kablosuz bluetooth modülü aktif edilip, elektronik cihazların birbiri ile haberleşmesi sağlanabilmektedir. Android işletim sistemli cihazlarda bluetooth modülünün aktif edilmesi kod içerisinde gerekli izinlerin verilmesi sayesinde yapılabilmektedir.

## 2.9. Mikroişlemcilerde Haberleşme sistemleri

Her mikroişlemci, Uygulama alanlarına göre dış donanımlarla farklı şekillerde haberleşebilmektedir. Bu haberleşme yöntemleri, her mikroişlemcide farklılık gösterirken, haberleşme yöntemlerinin aktif edilmeleri, mikroişlemciye yazılan yazılımlar tarafından belirlenmektedir (San, 2015). Arduino uygulama geliştirme yazılımı ile kullanılacak haberleşme protokolünün aktif edilebilmesi için program içerisine, önceden yazılmış olan kütüphaneler dâhil edilmelidir. Standart olarak uygulama ile gelen kütüphaneler yükleneceği gibi, uygulama gereksinimlerine göre, özel olarak yazılan kütüphanelerde eklenebilir. Arduino’da seri haberleşme için harici bir kütüphaneye gerek duyulmazken I2C, TWI, CAN-BUS protokolleri, özel haberleşme yöntemleri olduğu için uygun olan kütüphanenin kod satırı başında işleme dâhil edilmesi gerekmektedir. Haberleşme protokollerinin aktif edilme çalışmanın ileri aşamalarında detaylı olarak anlatılmıştır ve gerekli kod dizilimleri arduino kodları kısmında ek olarak verilmiştir.

### 2.9.1. I2C senkron haberleşme protokolü

I2C haberleşme protokolü, senkron haberleşme protokolüdür (Topsakal, Yılmaz ve Tayfur, 2016). Sistemde bir işlemci Master görevini üstlenirken diğer işlemcilerin, Master işlemciden gelen komutlar doğrultusunda kendilerine yazılan yazılımlar doğrultusunda işlemlerini yapmaktadır (Topsakal, Yılmaz ve Tayfur, 2016). Master haricindeki mikroişlemciler, sistem içerisinde slave olarak çalışmaktadır ve üzerlerine yazılan yazılım doğrultusunda master mikroişlemcisinden herhangi bir komut gelmesi dahi işlemlerini

yaparken master mikroişlemcisinden gelen komutlar doğrultusunda ölçtükleri değer sonuçlarını master mikroişlemcisine göndermektedir veya master mikroişlemcisinden gelen değerler doğrultusunda, işlemlerine devam temektedir (Demirten, 2019).

### **2.9.2. CAN-BUS haberleşme protokolü**

CAN-BUS haberleşme protokolü, asenkron haberleşme protokolüdür (Ünalnı, 2015). Sisteme bağı olan mikroişlemcilerin, yazılan yazılım doğrultusunda öncelikleri vardır. BUS üzerinden gelen verileri okuyabiliyorken BUS üzerinden veri gönderirken bu öncelik sırasına göre sisteme veri yüklemektedir. I2C haberleşme protokolüne göre, haberleşme daha yavaş olmasına rağmen, 11 bitlik adresleme satırı sayesinde aynı BUS üzerine daha fazla mikroişlemci bağlanabilmektedir (Özdemir, 2014; Wikipedia. 2019).

### **2.10. Android İşletim Sistemi**

İşlemci teknolojisinin gelişmesi sayesinde günümüzde bilgisayarlarla aynı özelliklere sahip cep telefonlar ve tabletler üretilebilmektedir. Ebat olarak küçük olmalarına rağmen yüksek işlem kapasiteleri sayesinde bu cihazlar birden fazla işlemi aynı anda yapabilmektedir. Günümüzde bu cihazların çoğunda android işletim sistemi mevcuttur. Kullanım amaçları haberleşme üzerine kurulu olduğu için, tablet ve cep telefonları farklı donanımlarla haberleşerek hem bilgi alış-verişi hem de işlem yaptırabilmeleri sağlanabilmektedir (Topsakal, Yılmaz ve Tayfur, 2016). Araştırmamızda android sistemin bluetooth haberleşme özelliklerinden yararlanılarak prototip ile haberleşme sonucunda hız ve batarya durum bilgilerinin anlık olarak android cihaz ekranından gösterilmesi sağlanabilmektedir (Çavdaroğlu ve Ergen, 2015).

### **2.11. Programın Yazılması**

Entegre devrelerle diğer bileşenlerin çalışması veya güç kontrolünün yapılması için mikrodeneleyicilere program yazılması gerekmektedir.

## 2.12. Programın Kontrol Altına Alınması

Elektronik elemanlar genellikle fiziksel işlemlerden daha hızlıdır. İşlemlerin gerçekleştirilebilmesi ve gerçek zamanlı kontrolü için elektronik elemanların hızının kontrol edilmesi gerekmektedir (Playground.arduino.cc, 2016).

Yukarıda literatür özetinde açıklandığı gibi bir sistemin tasarım ve donanımlarının kurulumunda gerekli sensör, yazılım, donanım ve iletişimin istenen şekilde gerçekleştirilebilmesi tüm bu bileşenlerin uygun elemanların uyumlu seçilmesi ile gerçekleştirilebilmektedir.

Planlanan gösterge panelinin oluşturulabilmesi ve kontrolünün sağlanabilmesi için literatürden yapılan incelemeler sonucunda aşağıdaki eleman ve yazılımlara ihtiyaç olduğu belirlenmiştir.

- Matlab yazılım ve şehir içi çevrimi
- Arduino geliştirme yazılımı
- Atmel 328PU
- Unity 3D yazılımı
- HC-05 bluetooth modülü
- Arduino UNO
- L293 motor sürücü
- Kızılötesi hız sensörü

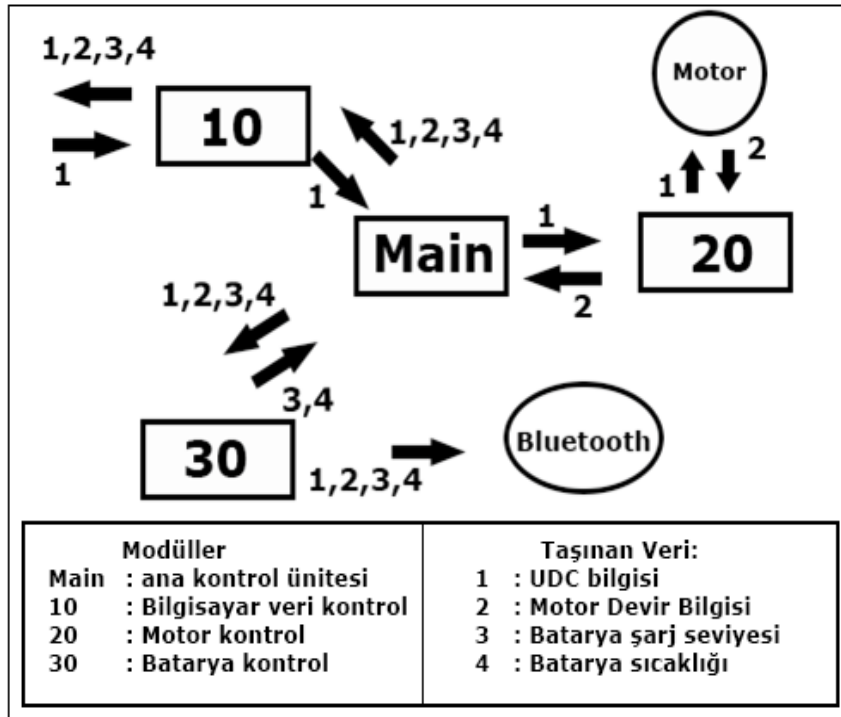


### 3. ELEKTRİKLİ ARAÇ GÖSTERGE PANELİ PROTOTİP TASARIMI

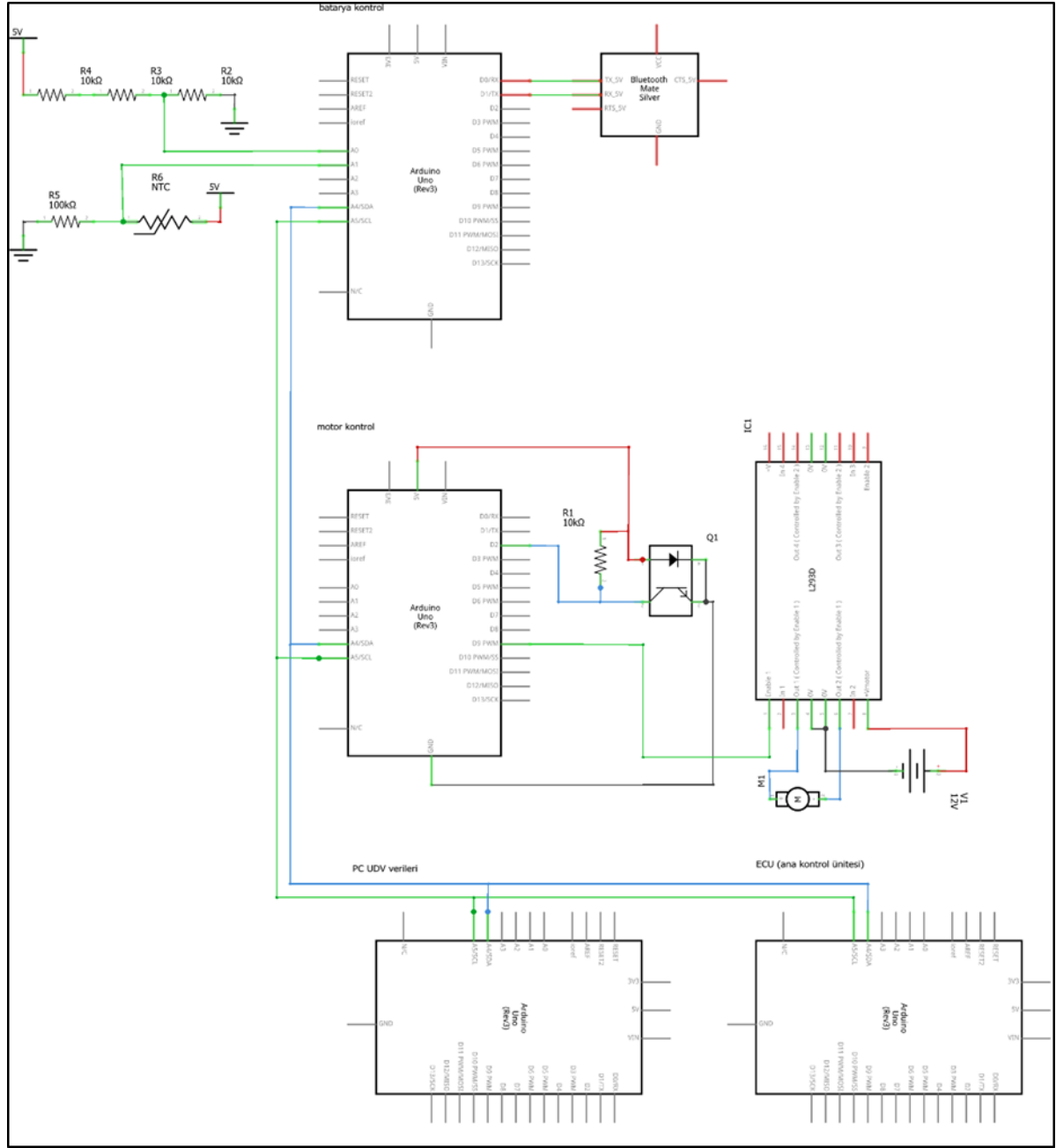
Prototip çalışması için literatürde incelenerek belirlenen donanım elemanları ve yazılımlar bu bölümde detaylı olarak ele alınmıştır.

Prototip test çalışmasında gerek yazılımsal gerekse donanımsal olarak temini kolay, kullanım alanları geniş olan donanımlar seçilmiştir. Test edilecek araçlar üzerinde, bağlantıların kurulması, geliştirilmesi ve gerektiği durumlarda sisteme direkt adaptasyonu sağlanabilen bu donanımlar, herhangi bir değişiklik yapmadan gerçek araçların üzerinde de kullanılabilirliği düşünülmüştür.

Bu kapsamda düşünülen prototip için gerekli donanımların kurulmasında kullanılan model Şekil 3.1’de ve uygulamanın elektronik devre şeması Şekil 3.2’de verilmiştir.



Şekil 3.1. Prototip çalışma uygulama şeması



Şekil 3.2. Uygulamanın elektronik şeması

Tasarımı düşünülen prototip çalışmasında gerek sistemin test edilebilirliği gerekse araç üzerine direkt uygulanabilmesi açısından aşağıdaki sistemler kurularak test edilmiştir. Hem bilgisayarda hem de gösterge panelinde hazırlanan prototipin sonuçları gösterilmiştir.

Yapılan çalışmada şehir içi çevrim bilgileri doğrultusunda dc motoru hareket ettirilmiş, dc motorunun devri okunmuş, sisteme elektrikli araçlarda kullanılan Li-Ion pil eklenerek gerçek zamanlı olarak şarj seviyesi ölçülmüş, ölçülen değerler, Bluetooth haberleşmesi ile gösterge paneline aktarılmıştır.

### **3.1. Prototip Uygulamasında Yazılımların Programlanması**

#### **3.1.1. Matlab yazılımı üzerinde yapılan işlemler**

Prototiple çalışma amacıyla, mikroişlemcilere gereken bilgilerin hesaplanması ve gönderilmesi için Matlab yazılımı kullanılmıştır. Kullanılan veriler, şehir içi çevrim dosyasından alınan, gerçek araca ait zamana bağlı hız verilerini kapsamaktadır. Prototip çalışmanın kodları EK 1’ de verilmiştir.

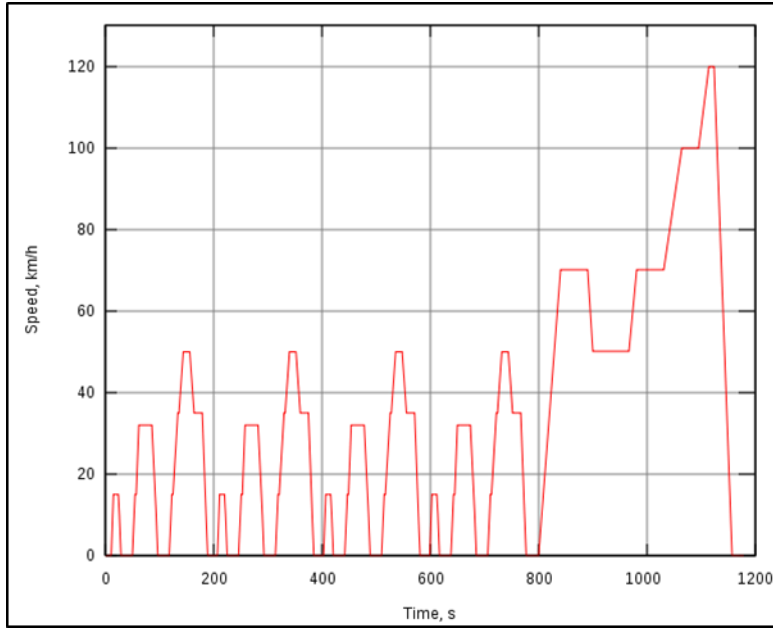
Aracın kontrol ünitelerini temsil eden mikroişlemcilere yazılımların yazılıp yüklenebilmesi için hem örnek dosyaların kolay bulunması hem de mikroişlemcilere kolay yazılım yükleme yapılabilmesi için arduino kod editörü kullanılmıştır.

Mikroişlemcilerle android işletim sistemi yüklü tabletin haberleşebilmesi için, tablet üzerine gelen verilerin kontrol edilebilmesi, bluetooth bağlantısı ile haberleşebilmesi ve görsel içerik konusunda hazırlanan içeriği düzenleme ve uygulama olanakları sunan unity yazılımı kullanılmıştır.

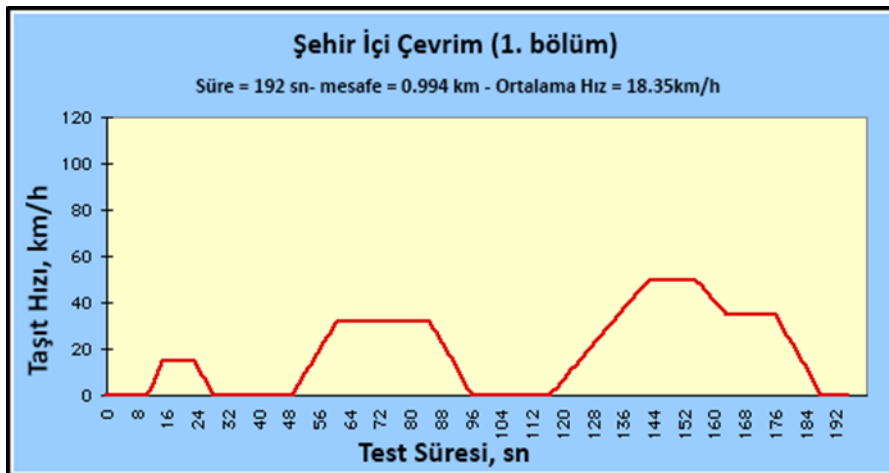
#### **3.1.2. UDC (urban drive cycle) dosyalarının kullanılması**

Şehir içi çevrim dosyaları, araç üzerinde emisyon sistemi değişikliği yapıldığında, bu değişikliğin gerekli sonuçları verip vermediğini denetlemek amacıyla, aracın güç aktarma sistemlerini kontrol etmek amacıyla kullanılan verilerdir (DieselNet.com, 2019).

Aracın zamana bağlı olarak hız değerlerini gösteren bir bilgi dizisidir. Çoğu gelişmiş ülke, ülkelerine ihraç edilecek araçlar için kendi belirledikleri çevrimlere uygunluğu şart koşturmaktadır. Bu sayede araç emisyonlarından dolayı çevreye verilecek zararları kontrol altına almaktadırlar. Sunulan bu çalışmada ülkemizin 1995 yılında araç düzenlemelerinin uyumlaştırılması kapsamında kabul edilen ECE-15 çevriminin tamamı Şekil 3.3’te, uygulamamızda kullanılan 195sn’lik birinci kısım ise Şekil 3.4’te verilmiştir.



Şekil 3.3. ECE-15 şehir içi çevrimi



Şekil 3.4. ECE-15 şehir içi çevrimi 1. Kısım

Kullanılacak şehir içi çevrim bilgileri genel olarak aşağıdaki gibidir;

Süre : 195sn

Mesafe: 0,944km

Ortalama hız : 18,35km/sa

Azami hız : 50km/sa olarak görülmektedir



### 3.1.3. Matlab programına UDC verilerinin aktarılması

ECE 15 şehir içi çevrim bilgileri, excel dosyası içerisine yazılarak ‘ececol.xlsx’ ismi ile kaydedilmiştir. Matlab dosyası içerisine aşağıdaki kodlar yazılmıştır.

```
clear all
```

```
clc
```

```
T = xlsread('ececol.xlsx');
```

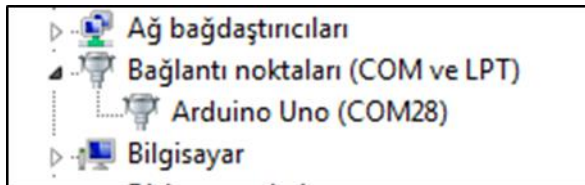
```
time=T(:,1);
```

```
speed=T(:,2);
```

```
plot(time,speed)
```

Bu sayede excel dosyasındaki birinci sütun içerisindeki bilgiler time değişkenine, ikinci sütundaki bilgiler de speed değişkeni içerisine aktarılarak ‘ececol.xlsx’ dosyasına ait grafik çizdirilmiştir. Grafik verilerinin aktarılması ve grafiğin çizilmesi için kullanılan kodlar EK 1’de verilmiştir.

Arduino UNO uygulama geliştirme kartı bilgisayara bağlandıktan sonra görev yöneticisinden geliştirme kartı için atanan port bulunarak Matlab yazılımının bu port ile haberleşmesi sağlanmıştır. Bağlantı portunun bulunması Şekil 3.5’te verilmiştir.



Şekil 3.5. Görev yöneticisinde arduinonun 28 numaralı COM-port bağlantısı.

Matlab komut satırına aşağıdaki kodlar yazılarak Matlab ve arduino arasında veri iletişimi için bağlantı sağlanmıştır.

```
a=serial('COM28','baudrate',9600)
```

#### 3.1.4. Matlab ve arduino arasında bilgi akışının sağlanması

Matlab kodu çalıştırılıp haberleştirildikten sonra, Matlab ile arduino bağlantısını sonlandırmak için Matlab kod satırının sonuna

```
delete(a)
```

satırı eklenmiştir. Bu sayede Matlab kodu tekrar çalıştırılıp arduino ile haberleşmesi istendiğinde Matlab programının hata vermesi engellenmiştir.

#### 3.1.5. Arduino yazılımı ile Matlab arasında UDC bilgilerinin aktarılması

Arduino ve Matlab haberleşmesi sağlandıktan sonra, UDC bilgilerinin Matlab'dan mikroişlemciye aktarım süreleri kontrol edilmiş, bilgisayar hızına bağlı olarak bu bilgilerin aktarım zamanlarında farklılıklar gözlemlenmiştir. UDC bilgileri, gerçek zamanlı olmayıp, var olan verilerden sağlandığı için, Matlab yazılımı içerisinde bir alt program oluşturularak gerekli zamanlarda mikroişlemciye gönderilmesi sağlanmıştır.

Arduinoya gelen bilgiler(HEX, DEC, OCTA) farklı formatlarda olduğu için bunlar sayısal değerlere (int) dönüştürülmüştür. Bu sayede dc motoru sürmek için kullanılan motor sürücüsüne gerekli PWM sinyallerine dönüştürülmeleri kolaylaşmıştır. Burada bilinmesi gereken, bu PWM bilgilerinin, farklı motorlarda farklı sonuçlar vermesidir. Bu sebeple kullanılacak motor katalog verileri incelenmiş, PWM sinyalleri motor sürücüye gönderilirken, uygun katsayılar kullanılarak gönderilmiştir. Matlab ve arduinonun haberleşebilmesi için kullanılan kodlardan Matlab kodları EK1'de, arduino kodları ise EK2'de verilmiştir.

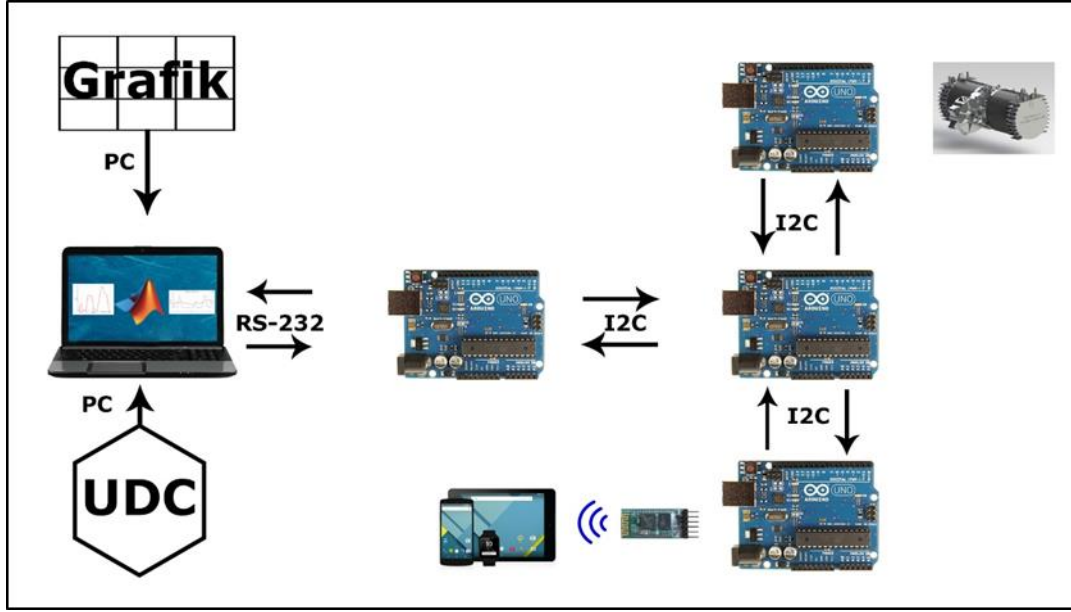
Bu aşamada bilgisayarla bağlantının kurulması ve gelen verilerin okunması için kullanılan kod dizilimi aşağıdaki gibidir:

```
if (Serial.available())
```

```
gelenBilgi = (Serial.read());
```

### 3.2. Prototip Uygulaması İçin Kullanılan Donanımlar

Modeli Şekil 3.1’de verilen prototip modelin şematik şeklinin uygulanması için gerekli uygun donanım elemanları literatürde incelenerek seçilmiş, modelin gerçek elemanlarla kurulumu gerçekleştirilmiştir (Şekil 3.6).



Şekil 3.6. Modelin gerçek elemanlarla kurulumu.

Test uygulamasında, çalışma sonucundaki beklentiler doğrultusunda uygun işlemci ve donanımlar seçilmiştir.

Sistemin farklı işlemcilerle uyumlu çalışabilirliği amacıyla, istenen minimum gereksinimleri karşılayacak asgari işlemciler seçilmeye çalışılmasının yanında, kolay kurulum ve test sonuçlarının hızlı bir şekilde alınabilmesi için, çoğunlukla arduino UNO ile çalışma tercih edilmiştir.

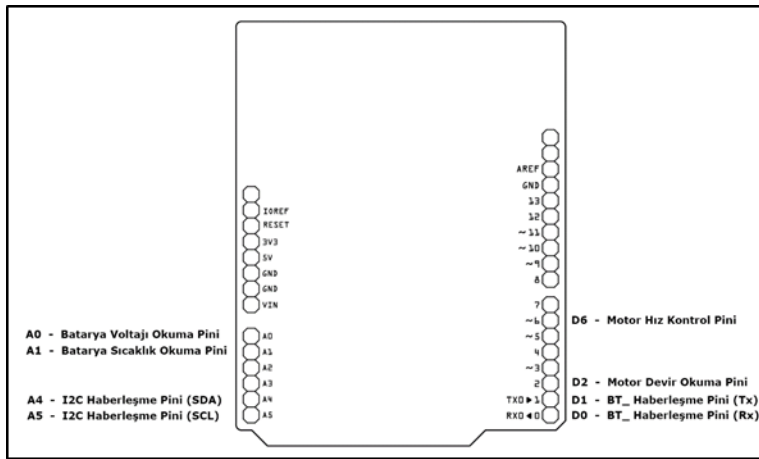
#### 3.2.1. Mikroişlemciler

Günümüzde farklı görevleri yerine getirebilecek şekilde imal edilmiş binlerce mikroişlemci mevcuttur. Bunlardan bazıları belirli görevleri yerine getirecek şekilde imal edilmiş olup, bazıları ise geliştiricilerin geliştirecekleri sistemlere göre içerisine yazılım yükleyebilecekleri şekilde imal edilmektedir.

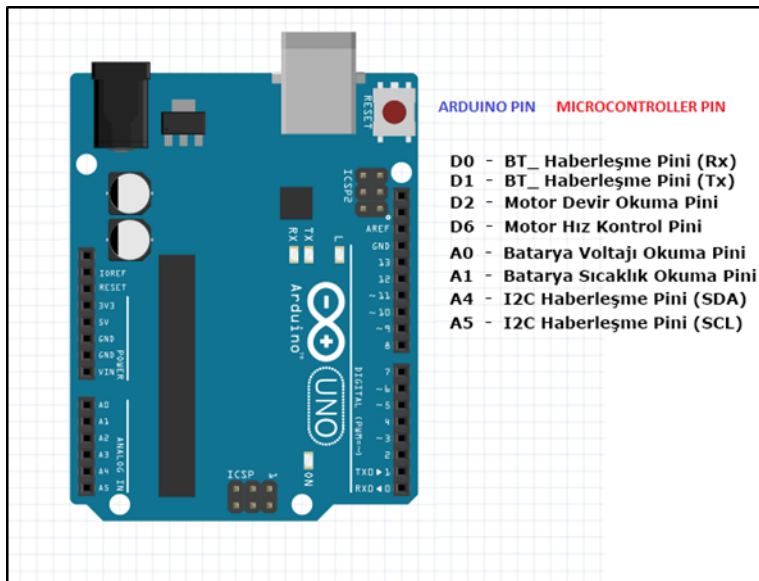
Sunulan bu çalışmada geliştirilen uygulamada, farklı özelliklere sahip, bir kısım değiştirilmeyen, diğer bir kısmı ise, kullanılacak yere göre içerisine yazılım yüklemesi yapılabilecek özelliklere sahip mikroişlemciler kullanılmıştır.

Kullanılacak işlemlere göre, pin sayısı düşük veya fazla, program hafızası düşük veya yüksek, kullanım amaçlarına göre programlanabilir veya üzerinde sabit yazılım bulunan farklı mikroişlemciler kullanılmıştır. Bunlar genel özellikleri itibarı ile aşağıdaki gibidir.

### 3.2.2. ATmega328P



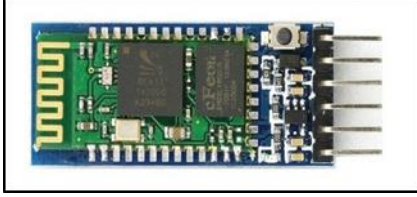
Şekil 3.7. ATmega238P mikroişlemcisinin kullanılan bacak yerleşimleri



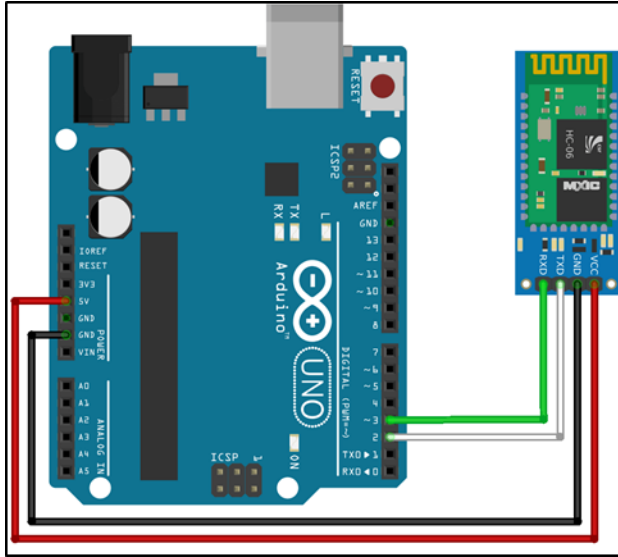
Şekil 3.8. Arduino UNO geliştirme kartı kullanılan pin isimleri

### 3.2.3. HC-05 bluetooth modülü

HC-05 Bluetooth modülü, kurulu sistemlere kablosuz haberleşme özelliği kazandırmaktadır. Mikroişlemci ile Rx-Tx girişlerine bağlanarak android cep telefonunun ekranında gerekli bilgilerin okunmasını sağlamaktadır. Resim 3.1’de bluetooth modülü Şekil 3.9 da ise bluetooth modülü ile arduiononun bağlantı şekli görülmektedir.



Resim 3.1. HC-05 bluetooth modülü



Şekil 3.9. HC-05 bluetooth modülü ile arduino UNO bağlantısı kullanılmıştır.

### 3.2.4. L293 motor sürücü kartı

L293 Motor sürücü kartı, orta güçte dc motorların hızının kontrol edilmesi için kullanılan L293 entegresi adapte edilmiş elektronik motor kontrol kartıdır. Motor kontrol gerilimi olarak 4.5V – 36 V gerilimlerle çalışabilmektedir. Kontrol sinyali olarak lojik 5V sinyaller ile motorun sürülmesi sağlanabilmektedir. Entegre içerisinde dahili ESD koruması ve H köprüsüne sahip yapısı ile 4 adet çıkışı mevcuttur. Bu sayede 4 adet DC motorun ayrı ayrı hızlarının kontrol edilmesine veya 2 adet DC motorun hız ve yönlerinin kontrol edilmesine veya 1 adet step motorun kontrol edilmesine olanak sağlamaktadır.

### 3.2.5. F249 kızılötesi hız sensörü

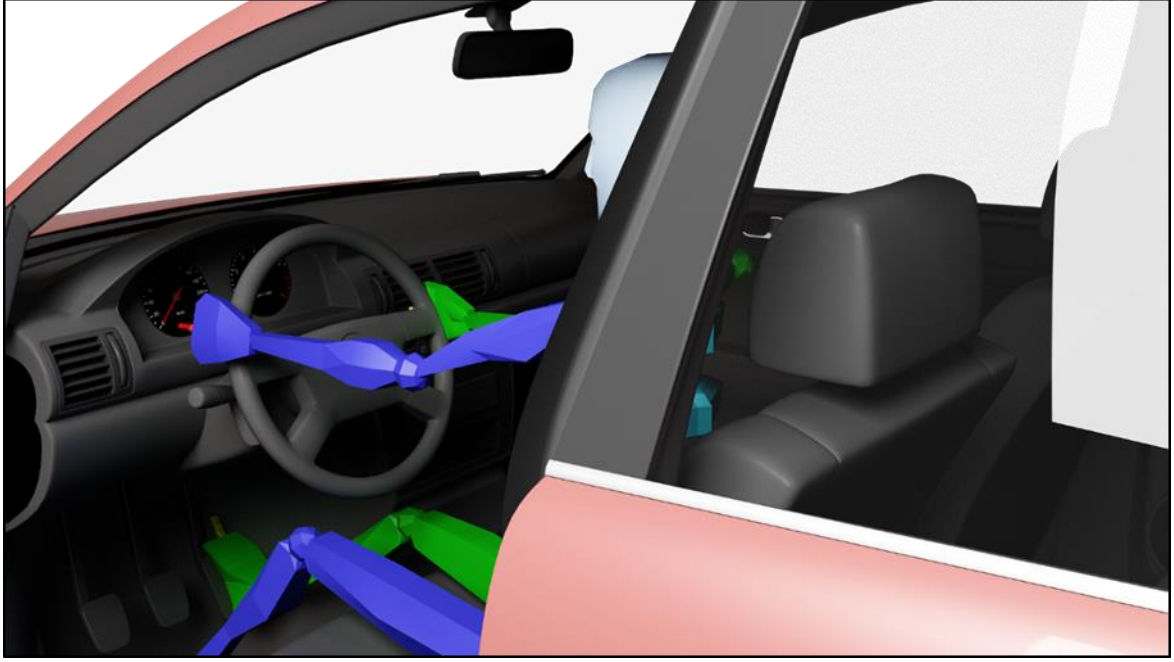
4 pinli 3.3V- 5.5V aralığında çalışan dijital ve analog çıkış verebilen devir kontrol kartıdır. Yapısında bulunan kızılötesi alıcı ve verici LED'ler sayesinde, motor üzerinde bulunan diskler üzerinden motorun devrini ölçebilmektedir.

### 3.2.6. Batarya şarj kontrol devresi

Batarya şarj kontrol devreleri, elektrik ve elektronik sistemlere güç sağlayan bataryaların şarj durum bilgisini görsel olarak veren devrelerdir. Yapı olarak şarj seviyelerini göstermenin yanında, ilave elektronik donanımlar ile bataryayı aşırı şarjdan koruma ve şarj seviyesi belli bir seviyenin altına düştüğünde devreye olan beslemeyi kesme görevleri de üstlenebilmektedir. Yapılan çalışmada, kullanılan batarya 12V gerilim sağlamakta iken elektronik sistemin 5V ile çalışmasından dolayı bir gerilim bölücü ile kullanılarak gerilim gerekli oranında düşürülmekte, okunan değer sisteme dijital bilgi olarak verilmektedir.

## 3.3. Araç İçi Gösterge Panelinin ve Görselinin Tasarımı

Autodesk firması tarafından sunulan 3B ürün tasarımı, modelleme, görselleştirme ve animasyon programı 3ds max ölçülü modelleme ve görselleştirme için kullanılmıştır. Program ile araç sürücü kabini modeli üzerinden, gösterge paneli görselinin yerleşimi ve gösterilecek bilgilerin konumları belirlenip ergonomi ve görsellerin konumlandırılmaları yapılarak ön tasarım yapılmıştır. Resim 3.2 ve resim 3.3'de tasarım için kullanılan araç içerisinde sürücünün gösterge panelini görüş şekli görülmektedir.



Resim 3.2. Araç içyapısı

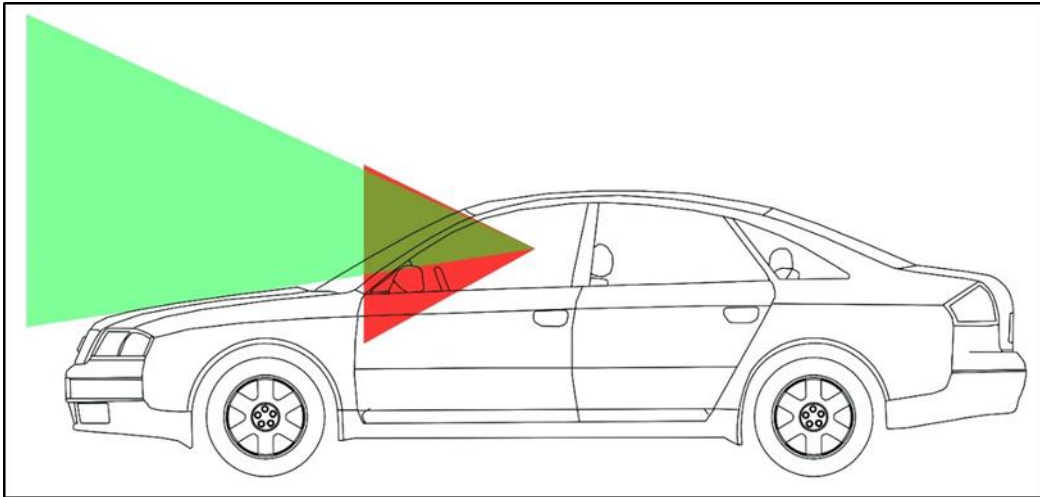


Resim 3.3. Sürüş esnasında sanal ortamda araç içi görünümü



Resim 3.4. Sürüş esnasında araç içi yol ve kadrın görünümü

Autodesk firması tarafından sunulan ve imalat için ölçülendirme amaçları için kullanılan bilgisayar destekli tasarım programı AutoCAD, 3ds Max programında hazırlanan 3B modeller ölçülü hale getirilip tasarımın ölçülendirilmesi yapılmıştır. Tarafımdan çekilen gerçek bir aracın sürüş esnasında tasarıma benzeyen yol ve kadrın görünüşü resim 3.4'te görülmektedir.

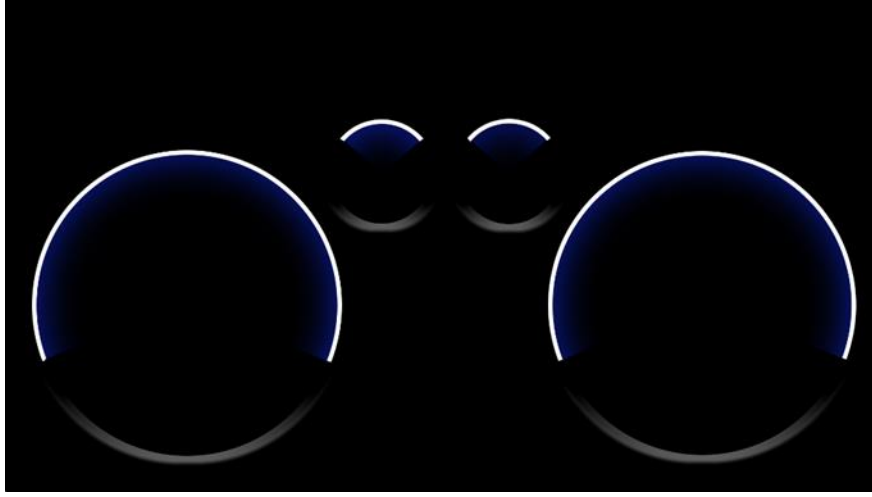


Resim 3.5. Sürüş esnasında yol ve kadrın görüş açıları

Adobe firmasının sunduğu, resim oluşturma, işleme ve düzenleme programı photoshop programı kullanılarak, AutoCAD programından, ölçülü ve dijital taslak görsel üzerinden, tablet üzerinde gösterilmek istenen resmin son hali oluşturulmuştur. Katman özellikleri



sayesinde kadr nlar  zerinde g sterilecek olan ibreler olu turulup ayrı ayrı kaydedilmi tir. Resim 3.6’da tasarım g rselinin nihai hali g r lmektedir.



Resim 3.6. G sterge paneli g rseli

Yapay zek  desteęi ve  oklu platform desteęi sunan, 2B, 3B uygulama, sim lasyon, oyun ve m hendislik programı Unity kullanılarak android i letim sistemi bulunan tablet i in photoshop ile hazırlanılan g rseller y klenmi tir ve Java, C#, ve phyton programları ile entegre  alı ması sayesinde tabletin kablosuz bluetooth mod l  aktif edilerek kurulan elektrikli ta ıt prototipi ile haberle mesi saęlanmı tır.



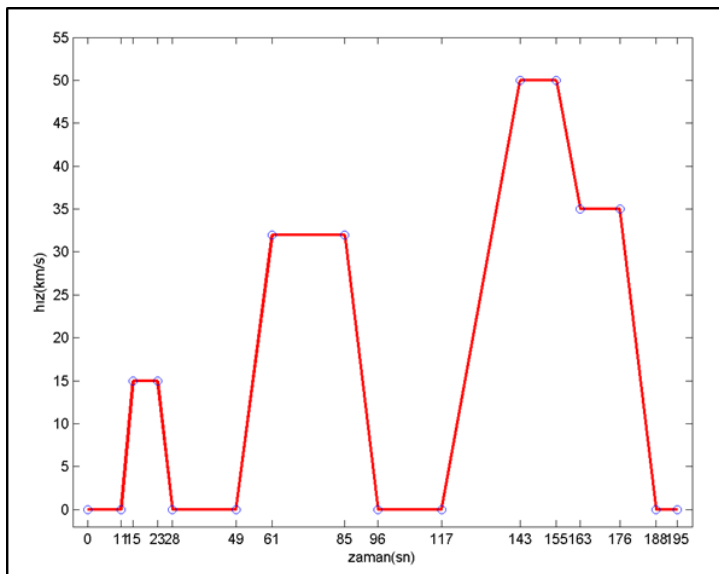
## 4. PROTOTİPİN ÇALIŞTIRILMASI VE DEĞERLENDİRİLMESİ

Modele göre eleman seçimleri yapıp kurulan prototipin gerçek elemanları ve bu elemanların çalışması ile sistemin komple çalıştırılması aşağıda detaylı olarak anlatılmıştır.

### 4.1. Prototipte Kullanılan Donanım Elemanları

#### 4.1.1. UDC verilerinin derlenmesi

Şehir içi çevrim verileri araştırıldıktan sonra, ülkemizde de standart olarak kullanılan ECE-15 verilerinin kullanılmasına karar verilmiştir. ECE-15 verilerinin matematiksel modeli oluşturularak Matlab programında kullanılmak üzere matematiksel denkleme dönüştürülmüştür. ECE-15 verilerinin ilk 195 saniyesi düşük motor yükü ve düşük motor sıcaklıkları için kullanılmaktadır. Bu kısım TİP-1 olarak adlandırılmakta olup bu çalışmada Matlab kodları TİP-1 için hazırlanmıştır. 0-780 saniye arası, motor uygun çalışma sıcaklığına ulaşmış motorlar için kullanılmaktadır. 780-1180 saniye aralığı, ağır yük koşulları için kullanılmakta olup yüksek hacimli dizel motorlar için kullanılmaktadır. Ek 1’de ECE-15 TİP-1 testi için kullanılan Matlab kodları verilmiştir. Bu kodun çalıştırılması sonucunda elde edilen grafik görüntüsü Şekil 4.1’de verilmiştir. ECE-15 verilerinin matematiksel hale getirilmiş şekli ve grafiğin oluşturularak mikroişlemciye gönderilmesi için kullanılan kod ek 1’de verilmiştir.



Şekil 4.1. ECE-15 şehir içi çevrimi 0-195 saniye aralığı

#### 4.1.2. Matlab kontrol verilerinin hazırlanması

Matlab ile mikroişlemcilerin haberleşmesi, bilgisayarlarda var olan seri port haberleşmesi üzerinden yapılmıştır. Bunun için mikroişlemcinin bağlı olduğu port bulunarak kod üzerinde gerekli düzenlemeler yapıldı. Anlık olarak zamana bağlı konumlar, şehir içi çevrim değişkenlerinden alınacağı için bu bilgiler, 'y' değişkenine zamana bağlı olarak atanarak 'write(s,y)' komutu ile mikroişlemciye gönderildi. Okunan verilerin belirlenmesi ve bunların grafik olarak gösterilmesi için ise 'plot(x,y)' komutu kullanıldı. Gösterilmesi istenen 4 adet grafik olduğu için 'subplot(2,2,2)' ile grafik 4'e bölünerek grafik görseli oluşturuldu. Ayrı bir grafik ile data değişkenine 'data = fgets(s)' komutu atanan veriler, 'plot(x,y,x,y2,x,y3,x,y4)' komutu ile grafik penceresinde çizdirilerek grafik.1'deki görsel elde edildi. Kullanılan kod yapısının tamamı Ek 1'de verilmiştir.

#### 4.1.3. Matlab ve arduino haberleşmesinin sağlanması

Matlab seri haberleşmesinin sağlanabilmesi için mikroişlemcinin bilgisayara bağlı olduğu seri port adresinin bilinmesi gereklidir. Bilgisayarda aygıt yöneticisi üzerinden bağlantı noktaları sekmesinden kontrol edilerek bağlı bulunana port numarası bulundu. Bu port Matlab kod editöründe 's=serial('com3')' komutu ile s değişkenine atanarak tanıtılması yapıldı. Mikroişlemciden gelen veriler 'data=fgets(s)' komutu ile data değişkenine aktarılarak bu değişkenler sayesinde dc motor hızı, batarya şarj seviyesi ve batarya sıcaklığı bilgileri okunarak bu veriler doğrultusunda grafiğin çizdirilmesi sağlandı. Matlab'dan mikroişlemciye veri gönderme işlemi, belirlenen değişkenin 'fprintf(s, '%\n', y) komutu ile gönderilmesi sağlandı. Değişken gönderme komutundan önce, bağlantının sağlanabilmesi için 'fopenf(s)' komutu ile port üzerinden bağlantının başlatılması ve 196 saniye sonunda veri alış-verişinin sonunda bağlantı portunun kapatılması için 'close all' ve portun serbest bırakılması için 'delete(instrfind)' komutları gönderilmiştir.

#### 4.1.4. Bilgisayar haberleşmesini sağlayan ünitenin yazılımı

Arduino yazılım geliştirme ortamı içerisinde, dahili seri haberleşmenin sağlanabilmesi için

```
void setup(){
```

```
Serial.begin(9600);
```

```
}
```

komutu ile ayar tanımlamaları içerisinde 9600 baud hızında tanımlaması yapıldı. Arduino kodu olarak haberleşmede gelen veri ve giden veri olmak üzere iki farklı haberleşme yönlendirmesi vardır. Gelen verinin okunabilmesi ve işlemcinin gereksiz çalışmasını engellemek ve gelen verinin okunması için

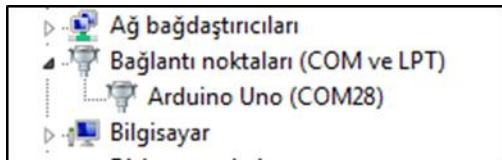
```
if (softSerial.available()>0)
```

döngüsü tanımlanır. Bu şekilde seri porttan veri gelmesi durumunda tanımlanan sonraki komutlar çalıştırılır. Seri haberleşmede veri göndermenin sağlanabilmesi için ise

```
Serial.print(x) ve
```

```
Serial. write (x)
```

komutları ile daha önceden tanımlanan 'x' değişkeni seri haberleşme ile gönderilir.



Şekil 4.2. Görev yöneticisinde orijinal arduino COM port bağlantısı

Donanımda bilgisayar ile haberleşmeyi sağlayan mikroişlemcinin kodları EK 2’de verilmiştir. Şekil 4.2’de arduinonun bilgisayar bağlantısı yapıldığında bilgisayar tarafından otomatik olarak atanan COM bağlantısının belirlenmesi görülmektedir.

#### 4.1.5. Dc motor kontrol ünitesinin hazırlanması

Arduino mikroişlemcilerinin kontrol pinlerinden en fazla 200 mA akım çekilebilmektedir. Bu akım seviyesi prototip uygulamada kullanılacak enkoderli dc motorun çalıştırılması için yetersiz kalmaktadır. Bu sebeple lojik voltaj seviyeleri ile yüksek güçteki motorların sürülebilmesine olanak sağlayan motor sürücülerini kullanılmaktadır. Prototip çalışmada kullanılan enkoderli motorun çalışma şartı 12V ve 800 mA olduğu için L293 motor sürücü devresi kullanılmıştır. Motoru değişik hızlarda kullanabilmek için PWM sinyali

kullanılmaktadır. Dc motor için tam güçte çalıştırmak için gerekli PWM seviyesi 255'dir. Servo motorun 180 derecelik tam dönüşü için gerekli sinyal bilgisi ise 180 birimdir. Arduino IDE içerisinde MAP fonksiyonu kullanılarak bilgisayardan gelen UDC bilgisi dc motor için 255'e, servo motor için ise 180'e ölçeklendirilmiştir. Bunun için kullanılan kod dizilimi;

```
srv= map(UDC,0,55,0,180) //servo motor ölçeklendirmesi
```

```
mtr= map(UDC,0,55,0,255) //DC motor ölçeklendirmesi
```

şeklindedir. Servo motor 0-180 derece arasında çalışmaktadır. DC motor sürücü için 8 bitlik PWM sinyalleri çalıştırılacaktır. 0-55km hız aralığı servo motor için 0-180, dc motor için ise 0-255 değerlerin ölçeklendirilmiştir. Motorun şehir içi çevrim bilgilerine göre çalıştırılması ve devrinin ölçülmesi için kullanılan kod EK 4'te verilmiştir.

#### 4.1.6. Batarya şarj seviyesi ve sıcaklık kontrol modülünün hazırlanması

Sıcaklık kontrolü ve batarya şarj seviyelerinin kontrolü için mikroişlemcinin ADC özelliği kullanılmıştır. Mikroişlemcinin analog girişlerinde 10 bitlik ADC çözünürlüğü bulunmaktadır.  $2^{10}=1024$  bitlik analog dönüşüm yapıldığı için voltaj okuma ve sıcaklık okuma değerleri için

```
V = map(BAT,0,1023,0,100) //batarya voltaj seviyesi
```

```
DGR = map(DGR,0,1023,0,100) //batarya sıcaklık seviyesi
```

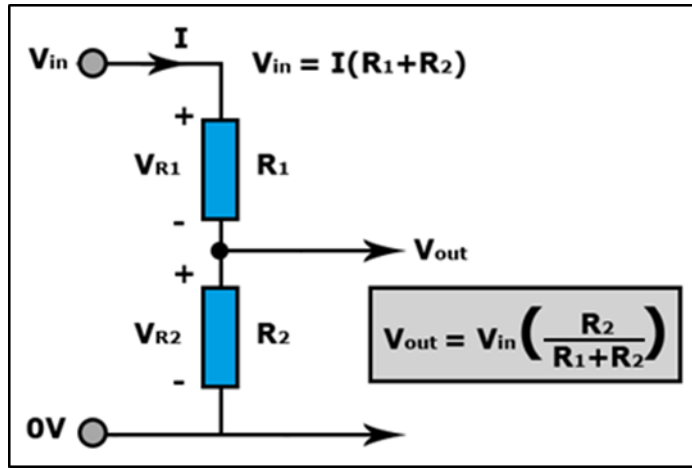
dönüşümleri kullanılmıştır. Uygulamanın batarya verilerini okuduğu yazılım, EK 3'te verilmiştir. Sıcaklık ölçümünün yapılabilmesi için  $10k\Omega$  değerinde NTC kullanıldığından, gerilim bölücü olarak seri bağlantı ile  $10k\Omega$  direnç bağlanarak sıcaklık ölçümü yapılmıştır. Kullanılan dc motor 12V ile çalışmaktadır. 3 adet Li-Po pil seri bağlanarak;

Tam şarjda  $4.7 V \cdot 3 = 14.1V$

En düşük şarjda  $3.5V \cdot 3 = 9.6V$

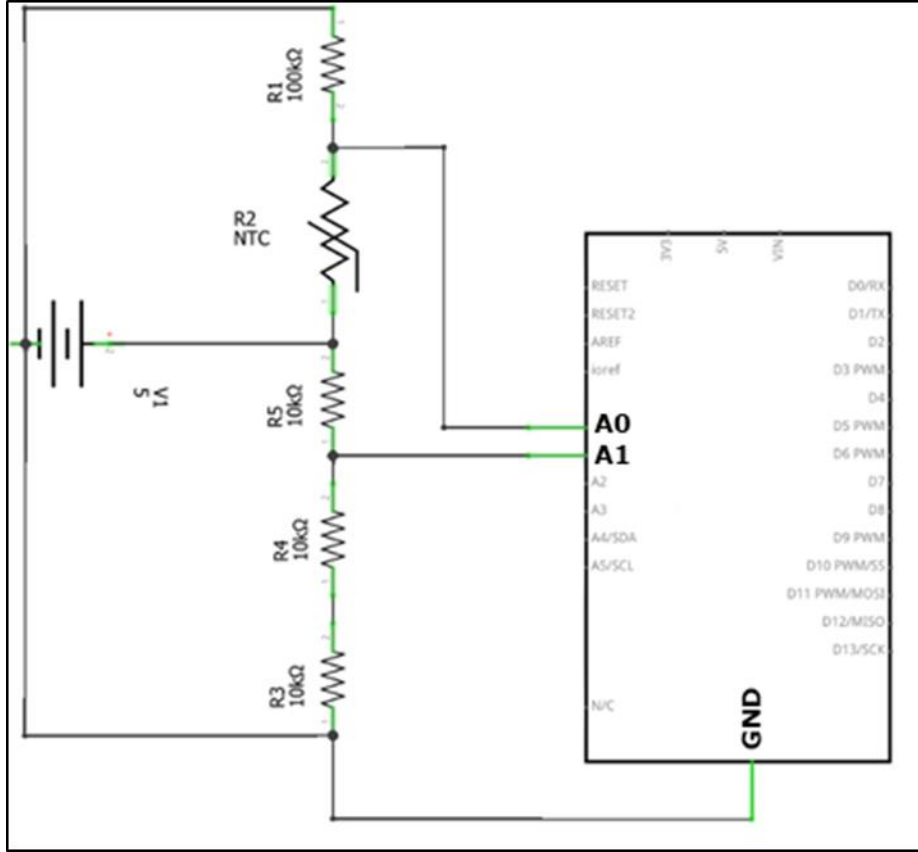
gerilim sağlandığından öncelikle mikroişlemcinin hasar görmemesi için uygun orana sahip gerilim bölücü kullanılmıştır. Bu sayede 9.6V - 14.7 V düşük ve yüksek voltaj değerleri boş ve dolu bataryayı temsil etmek üzere ölçeklendirme yapılmıştır ve batarya voltaj değişkenine atanmıştır. Değerin bulunması için batarya gerilim voltaj değerinin 3'te biri alındıktan sonra lityum polimer batarya için geliştirilen matematik model kullanılarak batarya kapasite değeri yüzde (%) olarak elde edilmekte ve grafik olarak gösterilmektedir. Polimer batarya için gerekli matematik model (Erşan, Kasım ve Burhan, 2018) literatür verileri kullanılarak elde edilmiştir. Elde edilen bağıntı (Eşitlik 4.1)'te verilmiştir.

$$K = -0.04798V^2 + 4.3254V - 8.7316 \quad (4.1)$$

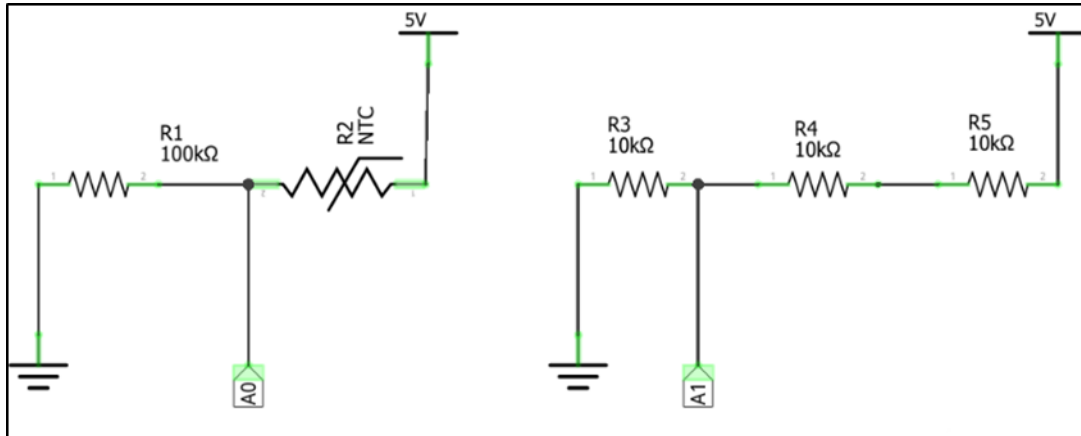


Şekil 4.3. Gerilim bölücü hesaplama.

Batarya voltaj seviyesinin ölçülmesi ve sıcaklık değerinin bulunması için kullanılan kod EK 5'te verilmiştir. Şekil 4.3'de bir gerilim bölücünün oluşturulması ve hesaplaması görülmektedir. Şekil 4.4'te batarya voltajını okumak ve batarya sıcaklığını okumak için yapılan bağlantı şekli, şekil 4.5'te ise gerilim bölücü için kullanılan devre şeması ayrı olarak görülmektedir.



Şekil 4.4. Batarya ve sıcaklık ölçümü



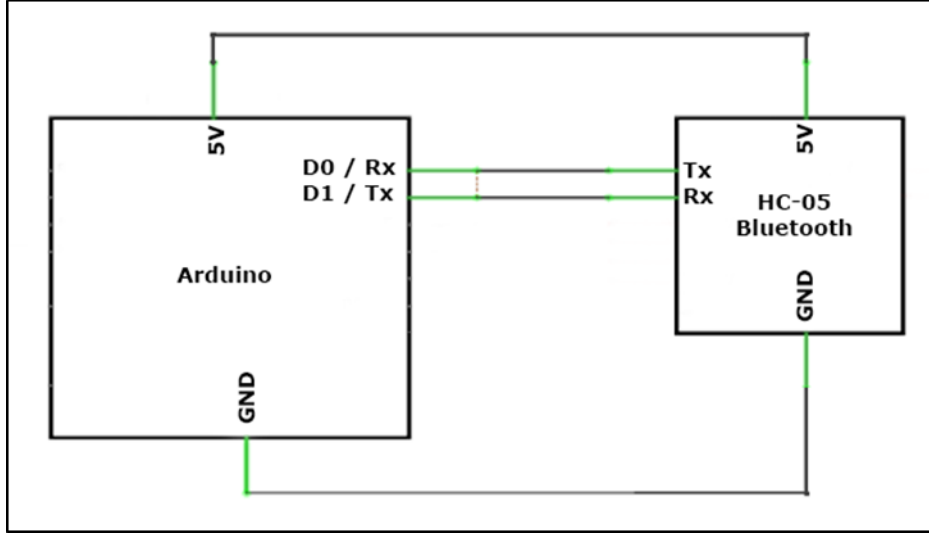
Şekil 4.5. Analog veri okuma

#### 4.1.7. Bluetooth bağlantısının sağlanması

Bluetooth modülü seri haberleşme protokolü üzerinden kablosuz haberleşme sağlamaktadır. Bu sebeple mikroişlemcinin seri haberleşme pinleri bu işlem için kullanılmıştır. RS-232 seri haberleşmede karşılıklı modüllerin Rx-Tx uçları Şekil 4.6’de görüldüğü gibi bağlanmıştır. Bu sayede cihaz özelliğinden bağımsız olarak bluetooth desteği olan bilgisayar, tablet gibi



diğer cihazlarla gerekli haberleşme sağlanmıştır. Mikroişlemci ile android işletim sistemi yüklü tabletin haberleşmesi için mikroişlemcide kullanılan kod EK 6’da verilmiştir.



Şekil 4.6. Mikroişlemci ve bluetooth modüllerinin birbirine bağlanması

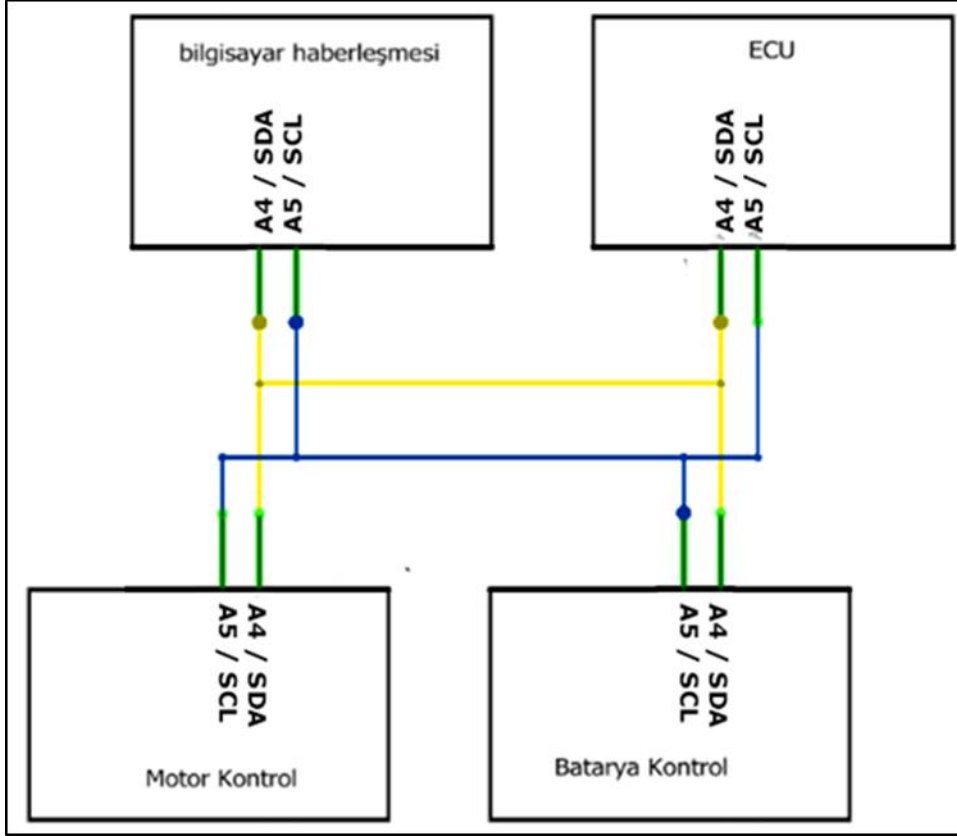
#### 4.1.8. Android yazılımının hazırlanması

Bluetooth haberleşme protokolünde aktarılan veri ASCII formatında olduğundan tablet içerisinde kullanılan yazılımda gerekli dönüşümlerin yapılması gereklidir. Bu sebeple byte olarak gelen veri önce string formatına dönüştürülmüştür. Sonrasında kodlamalarda yapılan düzenleme sayesinde gelen veriler UDC, V, DC ve DGR değişkenlerine atanarak tasarımı yapılan görsel üzerinde taşıt hızı, voltaj seviyesi ve sıcaklık değerlerini gösteren ibrelere atanmıştır.

#### 4.1.9. Ana kontrol ünitesinin hazırlanması

Sistemde 3 adet mikroişlemci birbirinden bağımsız çalışmaktadır. Sistemin çalışmasını sürdürmesi ve göstergenin gerekli bilgileri gösterebilmesi için elde edilen verilerin sistemin diğer elemanlarına uygun şekilde aktarılması gerekmektedir. Bu sebeple sistemde gerekli bilgi akışının sağlanabilmesi için ana bir mikroişlemci kullanılmıştır. Bu mikroişlemci bilgisayardan gelen UDC bilgilerini, motor kontrol ünitesinden gelen hız bilgilerini ve batarya kontrol ünitesinden gelen şarj seviyesi ve batarya sıcaklıklarını gerekli değişkenlere atayarak yine bu mikroişlemcilere göndermektedir. Şekil 4.7’de ECU kontrol ünitesi olarak

kullanılan mikroişlemcinin gelen verileri sakladığı değişkenler ve bu değişkenleri gönderdiği mikroişlemciler ile olan bağlantı şekli gösterilmiştir.

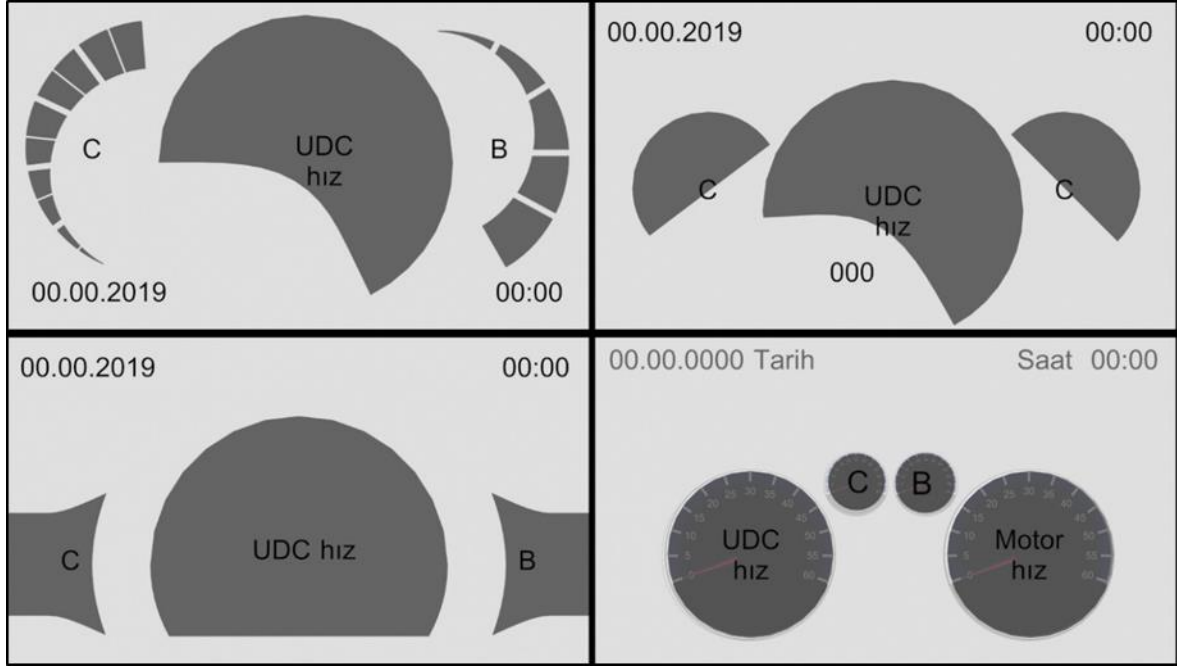


Şekil 4.7. Arduinoların TWI protokolü ile haberleşmesi

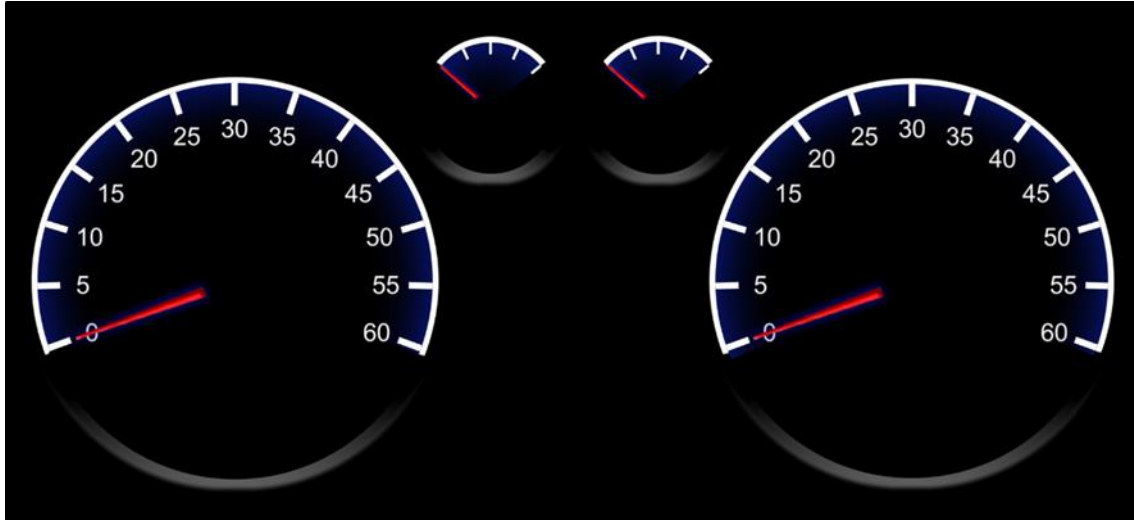
#### 4.1.10. Uygulamada kullanılacak görsellerin hazırlanması

- Sistemde kullanılan veriler ve bunların görselleştirilmesi
- Android işletim sistemi yüklü cep telefonu
- Bilgisayar monitöründen ve
- Analog ve dijital olarak panel üzerinden

olmak üzere 3 farklı şekilde uygulanmıştır. Bunlardan birincisi kurulumun amacı olan panel görselinin hazırlanmasıdır. Burada gösterilecek 4 bilgi şehir içi çevrim bilgisi, dc motorun dönme hızı, batarya voltaj seviyesi ve batarya sıcaklık bilgileridir. Bu bilgilerin gösterilmesi için Şekil 4.8'daki görseller hazırlanmış olup amaca göre bunlar arasında kullanım amacına göre seçim yapılması sağlanmıştır. Şekil 4.9'de ise kullanılacak görsel görülmektedir.

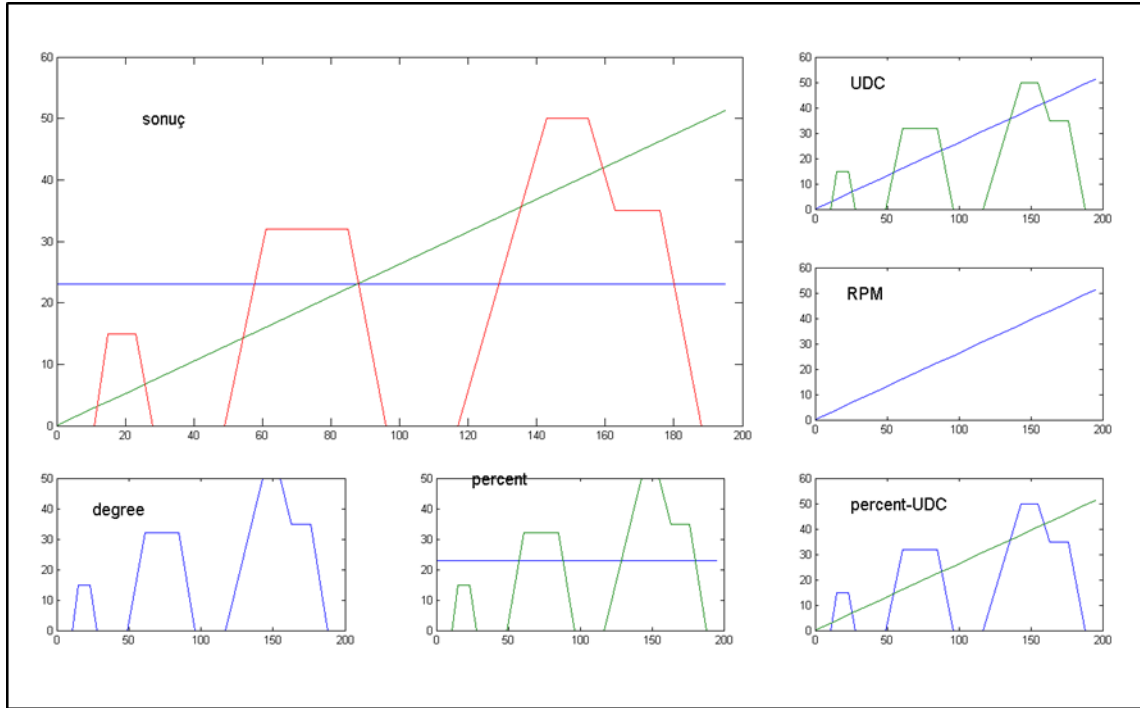


Şekil 4.8. Panel görsel tasarımları



Şekil 4.9. Uygulama üzerinde gösterilecek görsel

İkinci görsel bilgisayar ara yüzünde verilerin kontrol edilmesi ve değerlendirilmenin yapılabilmesi için hazırlanan Şekil 4.10'daki görseldir. Bu görselde, Matlab haberleşmesi sonucunda gelen veriler şehir içi çevrim bilgileri, bu bilgilerle eş zamanlı olarak çalıştırılan dc motorun okunan devir bilgisi ve bu esnada bataryada oluşan gerilim değişimleri ile sıcaklık değişimleri grafikte eşzamanlı olarak görülebilmektedir. Görselin tablet üzerinde sistemin çalışması ile eşzamanlı olarak kontrol edilmesi için kullanılan kod takımı EK 7, EK 8 ve EK 9'da verilmiştir.



Şekil 4.10. Matlab arayüzünde oluşturulan grafik görsel

Bilgisayarda grafik Şekil 4.10'da görülen görseli oluşturması için gerekli Matlab kod satırı aşağıda verilmiştir.

```
clear all;
```

```
clc;
```

```
close all;
```

```
delete(instrfind);
```

```
p=getserialport;
```

```
[m,n] = listdlg('PromptString','Select Your Bluetooth
```

```
device:','SelectionMode','single','ListString',p);
```

```
com=cell2mat(p(m));
```

```
s=serial(com);
```

```
fopen(s)
```

```
figure(1)
```

```
a = [0,11,15,23,28,49,61,85,96,117,143,155,163,176,188,195];
```

```
b = [0,0 ,15,15,0 ,0 ,32,32,0 , 0 ,50 ,50 ,35 ,35 ,0 ,0];
```

```
x = a;
```

```
y1 = (x-x+23)
```

```
y2 = x/3.8;
```

```
y3 = b
```

```
plot(a, b, x, y2, x, y3), text(25,50,'UDC', 'FontSize',15)
```

```
hold on
```

```
scatter(a,b)
```

```
set(gca,'XTick',[0,11,15,23,28,49,61,85,96,117,143,155,163,176,188,195] );
```

```
set(gca,'XTickLabel',[0,11,15,23,28,49,61,85,96,117,143,155,163,176,188,195] );
```

```
figure(2)
```

```
subplot(3,3,6)
```

```
plot(x, y2), text(22,50,'RPM', 'FontSize',15)
```

```
subplot(3,3,3)
```

```
plot(x, y2 ,x , y3), text(25,50,'UDC', 'FontSize',15)
```

```
subplot(3,3,7)
```

```
plot(x, y3), text(25,40,'degree', 'FontSize',15)
```

29.12.2019 23:06 I:\\_dizi\...\subPlot\_UDC.m 2 of 2

```
subplot(3,3,8)
```

```
plot(x, y1
```

```
, x, y3), text(25,50,'percent', 'FontSize',15)
```

```
subplot(3,3,9)
```

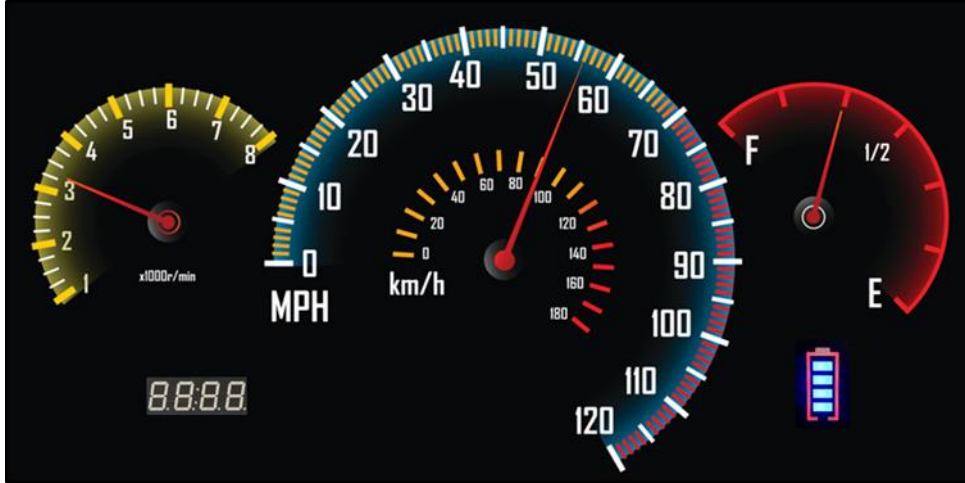
```
plot(x, y3, x, y2), text(25,50,'percent-UDC', 'FontSize',15)
```

```
subplot(1.4,1.4,1)
```

```
plot(x, y1, x, y2, x, y3), text(25,50,'sonuç', 'FontSize',15)
```

Arduinodan gelen grafik verilerinin eşzamanlı olarak bilgisayarda çizdirilmesi için kullanılan kod takımı EK 1’de verilmiştir.

Üçüncü görsel ise bir araç üstü gösterge için yapılacak panel üzerinde hız, voltaj ve batarya sıcaklığı ve şarj seviyesinin gösterildiği Şekil 4.11’deki görseldir.



Şekil 4.11. Panel görseli

Bu görsellerden bilgisayar arayüzünde kullanılacak görseller Matlab programı içerisinde 'plot()' komutu ile oluşturulmuştur. Tablet ve panel üzerindeki görseller photoshop programı yardımıyla oluşturulmuştur. Bunun için arka plan ve ibreler ayrı ayrı hazırlanmıştır.

## 4.2. Elektronik Donanımın Hazırlanması

Sistem kurulumu için birden fazla kontrol ünitesinin kullanılmasının sebebi tek bir mikroişlemcinin yeterli sayıda giriş ve çıkış pinine sahip olmamasının yanında, bir mikroişlemcinin çok sayıda işlem yapması sırasında yaşanacak gecikmeler sebebiyle sonuçların gecikmeli çıkmasıdır. Birden fazla işlemcinin aralarında haberleşmeleri, kablolu sorunlarını da beraberinde getirebileceği için sistemde I2C olarak bilinen ve Arduino IDE içerisinde TWI olarak adlandırılan haberleşme protokolü kullanılacaktır.

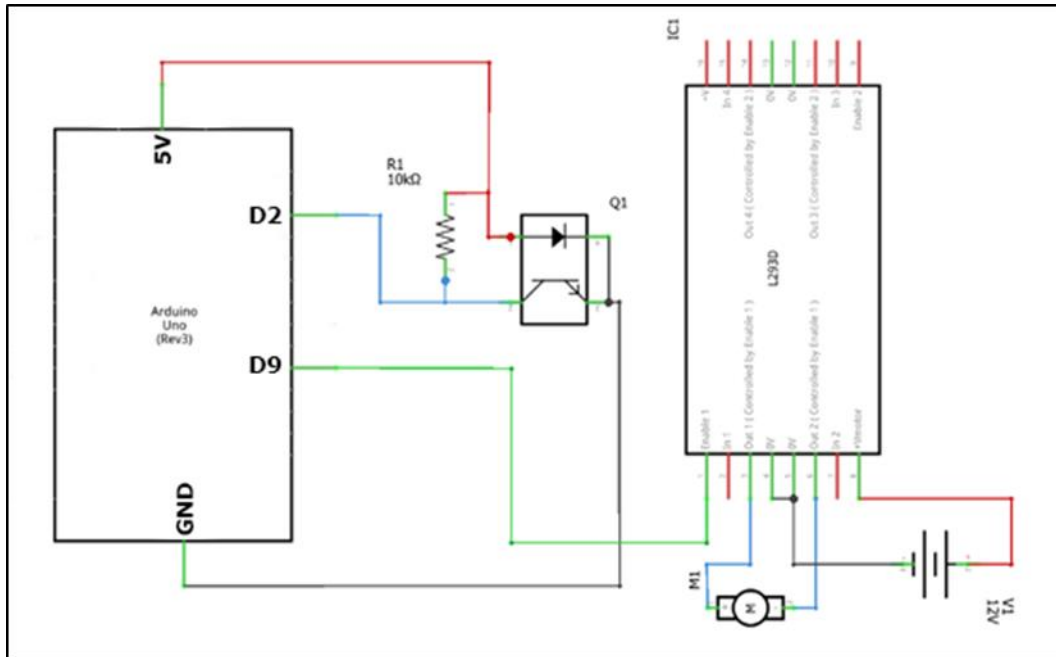
### 4.2.1. Bilgisayar haberleşme ünitesi

Mikroişlemcilerin bilgisayarla haberleşmesini sağlamak için, USB-TTL dönüştürücü olarak kullanılan CH340 FTDI çipi kullanılmaktadır. Bu sayede bilgisayar bağlantısı USB üzerinden direkt yapılabilir.

### 4.2.2. Dc motor kontrolü

Motor kontrolünün sağlanacağı mikroişlemcide kesme pini olarak atanması gerektiğinden, hız ölçümü için D2 pini devir ölçümü için kullanılmıştır. Motoru istenen devirde

döndürebilmek için PWM çıkış desteği olan 9 nolu pin kullanılmıştır. Kullanılan bütün mikroişlemcilerde haberleşmenin sağlanabilmesi için A4 ve A5 pinleri TWI haberleşme protokolü için kullanılmıştır. Şekil 4.12’de motor kontrolü için kullanılacak mikroişlemcinin bağlantı şeması görülmektedir.



Şekil 4.12. Motor hız kontrolü

### 4.2.3. Batarya kontrol devresi

Batarya şarj seviyesi ve sıcaklık kontrolü için A0 ve A1 analog girişleri kullanılmıştır. Şarj seviyesi için 15V max şarj seviyesi olan 3 adet Li-Po pil kullanılacaktır. Bu amaçla 3 adet 10k direnç kullanılarak bir gerilim bölücü oluşturulmuştur. Pillerin sıcaklıklarını kontrol etmek için 100k NTC ve buna seri bağlı 100k direnç ile bir gerilim bölücü oluşturulmuştur. Oluşturulan gerilim bölücü ve mikroişlemciye bağlantı şekilleri Şekil 4.4 ve Şekil 4.5'te gösterilmiştir. Bu sayede sıcaklık değişimleri 10 byte'lık hassasiyetle ölçülebilmektedir.

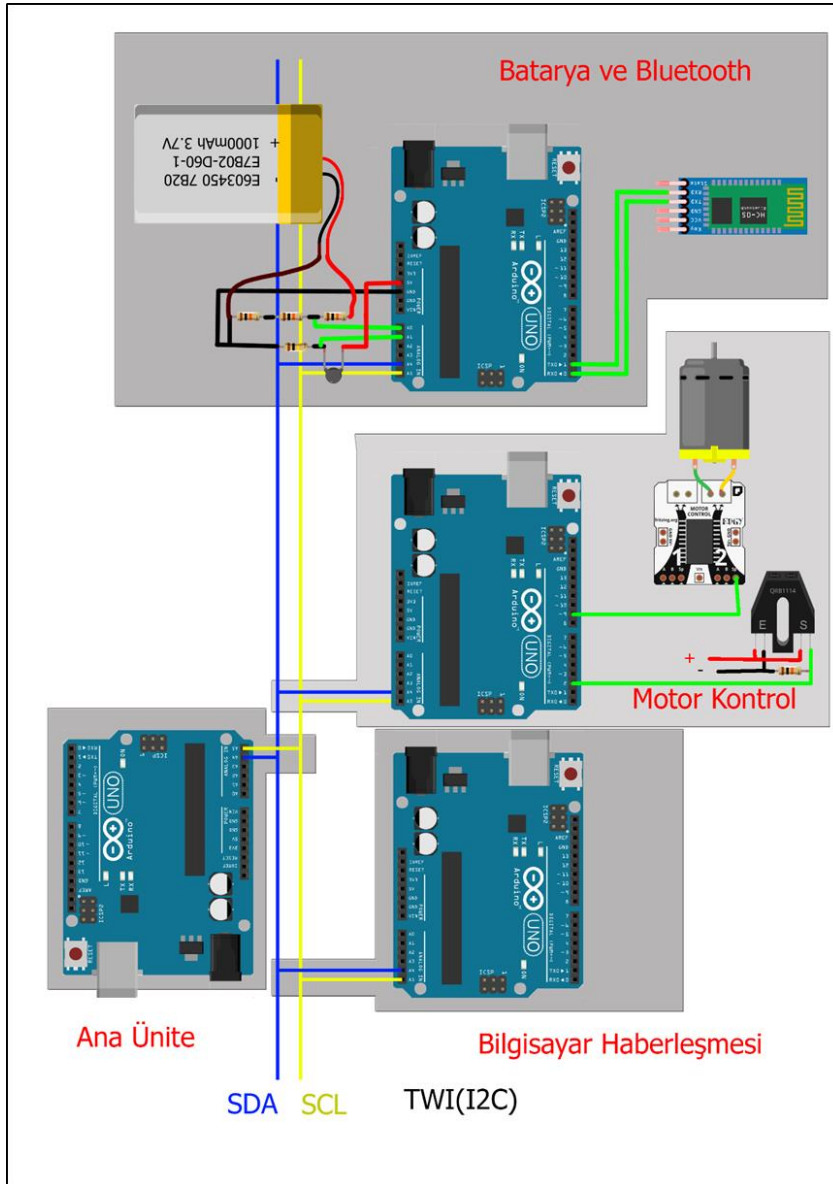
#### 4.2.4. Bluetooth haberleşmesi

Bluetooth modülü sisteme ECU kontrol ünitesi üzerinden bağlanmıştır. Bu amaçla mikroişlemcinin 0 ve 1 nolu pinlerine HC-05 bluetooth modülü bağlanmıştır. Bağlantı şekli Şekil 4.6’te görülmektedir.



#### 4.2.5. Ana kontrol ünitesi

Ana kontrol ünitesi sistemdeki haberleşmenin temelini oluşturmaktadır. Bu haberleşme protokolü sayesinde 7 bitlik adresleme imkânı sayesinde  $2^7 = 128$  adet farklı görevler yapabilen mikroişlemcinin birbiri ile uyum içerisinde çalışması sağlanmaktadır. Haberleşme esnasında 32byte veya 256 bitlik veri haberleşme için kullanılabilir. TWI haberleşme protokolü ile bütün data, 2 adet hat üzerinden aktarılabilir. Besleme gerilimi de ayrı 2 adet kablo üzerinden yapıldığı için ana kontrol ünitesi ve diğer ve diğer ünitelerin birbirleri ile olan bağlantısı Şekil 4.13’de görülmektedir.



Şekil 4.13. Uygulamanın bağlantı şekli



## 5. SONUÇ VE ÖNERİLER

Bu çalışmada, simülasyon verileri ile gerçek bir dc motorun kontrol edilmesi ve bu esnada elde edilen verilerin bilgisayar ortamında, bluetooth bağlantısı ile tablet üzerinde ve oluşturulan ayrı bir panel üzerinde gösterilmesi sağlanmıştır. Konuların ayrı ayrı ele alınması sonucunda, farklı cihazların birbirleri bilgi alış-verişi esnasında çoklu bilgi aktarımında yaşanan sorunlar ve bunların çözümleri sağlanmış, farklı mekanik sistemlerden ve dijital verilerden yararlanılarak gösterge sistemi oluşturulmuştur.

Çalışılan sistem ile bilgisayar, cep telefonu ve tabletlerin kullanılmasıyla gösterge sisteminde cihazlar arasında seçim serbestliği sağlanmıştır.

Taşıt içi gösterge panellerinde, taşıtın çalışabilmesi için gerekli analog ve dijital sinyallerin işlenmesi, bunların görsel hale getirilmesi ve bu görsellerin değiştirilebilir olması, kullanıcı açısından ön planda tutulan bilgilerin gösterilmesi, ihtiyaç duyulmayan görsellerin ise önem derecelerine göre gizlenmesi veya daha küçük gösterilmesi, dijital göstergeler sayesinde mümkün olmaktadır. Değiştirilebilir bu ara yüzlerde, çalışmada kullanılan 4 unsur olan şehir içi çevrim verileri, taşıt motorunun devri, kullanılan bataryanın şarj seviyesi ile sıcaklığı farklı şekillerde gösterilmiş olup verilerin daha fazla olması durumunda bu verilerin kaydedilmesi ve simülasyon olarak tekrar oynatılması sayesinde gerekli araştırma, inceleme ve geliştirmelerin yapılması sağlanabilir.

Tablet ekranlarının küçük olması sebebi ile bu çalışmada da gösterildiği gibi, bilgisayar ekranlarında bu bilgilerin gösterilmesi ve incelenmesi sağlanabilir. Günümüz bilgisayarlarının hızları ve kapasiteleri göz önüne alındığında bir taşıtın kontrol edilmesinin dışında, test standartlarının oluşturulmasında kullanılabilir. Bu testlerin tekrarlana bilirliği sayesinde gerekli değişiklikler yapılan taşıtlar için performans test ortamları geliştirilebilir. Ayrıca deney sonuçları kaydedilerek uygun görülen geliştirmelerin daha ileri seviyelere taşınması sağlanabilir.

Bu çalışmada oluşturulan standart bir gösterge panelinin oluşturulması sağlanmıştır. Bu birden fazla kişinin, çalışma değişkenleri üzerinde ayar yapmasını ve gerçek zamanlı olarak elde edilen veriler sayesinde anlık değişikliklerin yapılmasını gerekli geliştirmelerin daha hızlı yapılmasını sağlayabilir.

Bu çalışmada kullanılan elektronik donanım, özellikle kolay bulunan parçalardan seçilmiştir. Bu sayede gerek bu çalışmanın geneli gerek belirli kısımları, gerekse yapılabilecek eklemelerle daha karmaşık veya daha gelişmiş sistemlerin kurulması sağlanmıştır.

Günümüz enerji depolama sistemlerinden enerjinin taşınabilir halde olması, Li-Po bataryalar sayesinde mümkün olmaktadır. Bu çalışmanın farklı aşamalarında kullanılan 3 adet Li-Po bataryada, farklı şarj seviyeleri kullanıldığından, sistemin genel performansı en zayıf bataryaya göre belirlenebilmiştir. Bu sebeple gerçek bir elektrikli taşıt yapılması durumunda, birbirine yakın özelliklere sahip, özdeş bataryaların kullanılması ile taşıtın batarya şarj seviyesi daha kolay yönetilebilecektir.

Bu çalışma ile Li-Po bataryaların yüksek performansının yanında, şarj süreleri sebebi ile uzun süreli kullanımlarda, bu bataryaların doldurulmalarının, bu bataryaların özelliklerinden olan “C” oranlarının hesaplanması ile olabileceği görülmüştür. Bu değer, bataryanın sabit olarak verebileceği maksimum akım miktarını ve şarj esnasında şarj seviyesini de belirlemektedir.

Kablosuz ağ sistemi araç içi ve araç dışı olarak iki kısımdır. Uygulaması yapılan gösterge sisteminde cep telefonunda bilgilerin gösterimi bluetooth ile sağlanmıştır. Bu uygulama ile yeni nesil araçlarda uygulaması başlatılan ve ilk olarak araç içinde lastik basınç sensörlerinde uygulanan araç içi kablosuz iletişim yöntemi, bu çalışmada taşıt hızı ve batarya enerji seviyesinin gösterimi ile gösterge paneline taşınmıştır. Bu sistemlerin bundan sonraki uygulamalar için kablosuz iletişim ağına katkı sağlayacağı düşünülmüştür.

Bu tezde yapılan çalışmalar otomotiv alanında ileriki çalışmalarda elektrikli taşıt için daha gelişmiş sistemlerin ön çalışmasını teşkil edeceği düşünülmüştür.

Elektronik sistemler ve mekanik sistemlerin birbirlerine göre belirli üstünlükleri bulunmaktadır. Hareket iletimi esnasında mekanik sistemlerde sistemin büyüklüğüne göre atalet sorunları olmasına karşı elektronik sistemlerde gerekli olan enerji miktarı artmaktadır. Sistemlerin birlikte çalıştırılması sayesinde daha ergonomik, verimli ve düşük maliyetli sistemler ortaya çıkarılabilmektedir. Mekanik sistemlerin kurulumu esnasında yapılan

mekanik hesaplamalar, elektronik sistemlerin kullanılması sayesinde birkaç test sayesinde çözülebilmektedir.

Gösterge panelinde elektronik devrelerin çalışması fiziksel olarak çalışan sistemlerden daha hızlıdır. Gerçek zamanlı ölçümde ve gösterimde ileriki çalışmalarda elektronik sistem donanımlarının gelişmesine bağlı olarak daha iyi sistemler geliştirilebilir.



## KAYNAKLAR

- Albert, G., Toledo, T. and Grimberg, E. (2011). *The use of in vehicle data recorders and self-reported data for evaluating driving behavior of young drivers*. 3rd International Conference on Automotive User Interfaces and Interactive Vehicular Applications. University of Salzburg, Salzburg, 5-8.
- Allio, M. K. (2012). Strategic dashboards: Designing and deploying them to improve implementation. *Strategy & Leadership*, 40(5), 4-13.
- Ayas, F. Y. (2015). İnsansız. *Mekatronik Dergisi*, 29-32.
- Baldauf, M., Schabus, D., Fuchs, M., Fröhlich, P. and Suetter, S. (2011). *The roadsafe toolkit: rapid prototyping for road-testing novel automotive user interfaces*. 3rd International Conference on Automotive User Interfaces and Interactive Vehicular Applications. University of Salzburg, Salzburg, 55-56.
- Baysallı, O. U. (2010). *Otonom mayın tarama robotu*. Yüksek Lisans Tezi, İstanbul Teknik Üniversitesi Fen Bilimleri Enstitüsü, İstanbul.
- Carlson, R. W., Duoba, M. J., Bohn, T. P. and Vyas, A. D. (2007). *Testing and analysis of three plug-in hybrid electric vehicles*. SAE Technical Paper, 2007 World Congress, Detroit, Michigan, 1-4.
- Çavdaroglu, G. Ç. and Ergen, E. (2015, June). *Derlenen yol kenarı ünitesi, Araç CANBus ve konum verilerinin android*. 3rd International Symposium on Innovative Technologies in Engineering and Science, Universidad Politecnica de Valencia, Valencia, Spain, 742-750.
- Erşan, K., Kasım, L. and Burhan, C. (2018). *Elektrikli taşıtlarda kullanılan lityum polimer bataryalar için matematik model*. International Symposium on Multidisciplinary Studies and Innovative Technologies. IEEE Turkey Section, Ankara, 233-239,
- Falke, S. and Krömker, H. (2011). *Personas for on-board diagnostics – user types of drivers dealing with malfunction in e-cars*. 3rd International Conference on Automotive User Interfaces and Interactive Vehicular Applications. University of Salzburg, Salzburg, 29-30.
- Fotouhi, A., Propp, K. and Auger, D. (2015, September). *Electric vehicle battery model identification and state of charge estimation in real world driving cycles*. Proceedings of the 7th Science And Electronic Engineering Conference (CEEC 2015), Colchester.
- Franco, F., Stella, G. D., Neme, J. H., Santos, M. M., Stevan Jr, S. L. and da Rosa, J. N. (2017). *Teaching model-in-the loop: A case study for controller of distributed dashboard in a road vehicle*. 25th International Symposium on Industrial Electronics, Santa Clara, CA, USA, 863–868.
- Gerlach, S. (2011). *The HMISL language for modeling HMI product lines*. 3rd International Conference on Automotive User Interfaces and Interactive Vehicular Applications. University of Salzburg, Salzburg, 15-16.

- Gibson, Z., Butterfield, J. and Marzano, A. (2016). *User-centered design criteria in next generation vehicle consoles*. 5th CIRP Global Web Conference Research and Innovation for Future Production, Patras, Greece, 260 – 265.
- Gröger, C., Hillmann, M., Hahn, F., Mitschang, B. and Westkämper, E. (2013). *The operational process dashboard for manufacturing*. Forty Sixth CIRP Conference on Manufacturing Systems 2013, Setubal, Portugal, 205 – 210.
- Hofmann, H., Ehrlich, U. and Andreas, E. (2011). *Information extraction from the world wide web using a speech interface*. 3rd International Conference on Automotive User Interfaces and Interactive Vehicular Applications. University of Salzburg, Salzburg, 9-10.
- Houda, K. and Lakel, R. (2015). *Synchronized communication in a set of autonomous mobile*. The International Conference on Advanced Wireless, Information, and Communication. Algeria, Tunisia, 154-161.
- Hyun, J., Hwang, J., Ryu, J. and Cho, K. (2011). *Designing a spatial haptic cue-based navigation system*. 3rd International Conference on Automotive User Interfaces and Interactive Vehicular Applications. University of Salzburg, Salzburg, 17-18.
- Islinger, T., Köhler, T. and Ludwig, B. (2011). *Driver distraction analysis based on FFT of steering wheel angle*. 3rd International Conference on Automotive User Interfaces and Interactive Vehicular Applications. University of Salzburg, Salzburg, 21-22.
- İnternet: Bosch Sanayi Ve Ticaret A.Ş. (2019, 06 17). 1960-1989 yeni iş kolları ve elektronik. URL: <https://www.bosch.com.tr/kesfet/1960-1989-yeni-is-kollari-ve-elektronik/>, Son Erişim Tarihi: 17.06.2019
- İnternet: Cadex Electronics Inc. (2019). BU-602: How does a Battery Fuel Gauge Work? URL: [https://batteryuniversity.com/learn/article/the\\_battery\\_fuel\\_gauge](https://batteryuniversity.com/learn/article/the_battery_fuel_gauge), Son Erişim Tarihi: 18.06.2019
- İnternet: Demirten, M. (2019) I2C Protokolünün Tanıtılması. URL: [https://demirten.gitbooks.io/gomulu-linux/i2c\\_protokolunun\\_tanitilmasi.html](https://demirten.gitbooks.io/gomulu-linux/i2c_protokolunun_tanitilmasi.html), Son Erişim Tarihi: 17.06.2019.
- İnternet: DieselNet.com. (2019). ECE 15 + EUDC / NEDC. URL: [https://www.dieselnat.com/standards/cycles/ece\\_eudc.php](https://www.dieselnat.com/standards/cycles/ece_eudc.php), Son Erişim Tarihi: 18.06.2019
- İnternet: Everdell, C. (2019). Why The Car Industry Needs To Rethink The Dashboard User Interface Design. URL: <https://techcrunch.com/2015/10/08/why-the-car-industry-needs-to-rethink-the-dashboard-user-interface-design/>, Son Erişim Tarihi: 18.06.2019
- İnternet: Feasibility Study Automotive Production in Konya. (2016). Feasibility Study Automotive Production in Konya Version 13.1. Konya. URL: <http://www.investinkonya.gov.tr/images/dosya/KONYA%20FEASIBILITY%20STUDY%20FOR%20AUTOMOTIVE%20DFD.pdf>, Son Erişim Tarihi: 18.06.2019



- İnternet: Köylü, S. (2011). Gömülü Sistemler ve. Özgür Yazılım ve Linux Günleri 2011. İstanbul: İstanbul Bilgi Üniversitesi Bilgisayar Bilimleri Bölümü. URL: [https://seminer.linux.org.tr/wp-content/uploads/Navi-Linux-2011\\_serdar\\_koylu.pdf](https://seminer.linux.org.tr/wp-content/uploads/Navi-Linux-2011_serdar_koylu.pdf), Son Erişim Tarihi: 18.06.2019
- İnternet: Lavery, J. (2019). Don't Call it an Aries: Dodge 600ES Convertible. URL: <https://barnfinds.com/dont-call-it-an-aires-dodge-600es-convertible/>, Son Erişim Tarihi: 18.06.2019.
- İnternet: Markel, G. (2013). Servicing Wheel Bearings And Wheel Speed Sensors. URL: <https://www.underhoodservice.com/servicing-wheel-bearings-and-wheel-speed-sensors/>, [https://www.underhoodservice.com/wp-content/uploads/Articles/03\\_01\\_2009/1145100chartjpg\\_00000064878.jpg](https://www.underhoodservice.com/wp-content/uploads/Articles/03_01_2009/1145100chartjpg_00000064878.jpg), Son Erişim Tarihi: 18.06.2019
- İnternet: Meister, U. (2015). Auto esiklaasist saab arvutiekraan. URL: <https://tehnika.postimees.ee/3085819/auto-esiklaasist-saab-arvutiekraan>, Son Erişim Tarihi: 09.02.2019.
- İnternet: Microchip. (2014). Device Overview. URL: <https://www.microchip.com/wwwproducts/en/ATmega328p>, Son Erişim Tarihi: 17.06.2019.
- İnternet: Nam, E. (2019). Drive Cycle Development and Real-world data in the United States. URL: <https://www.slideserve.com/osgood/drive-cycle-development-and-real-world-data-in-the-united-states>, Son Erişim Tarihi: 18.06.2019.
- İnternet: Playground.arduino.cc (2019, 06 17). URL: <https://www.arduino.cc/>, Son Erişim Tarihi: 17.06.2019
- İnternet: Riechers, A. (2019). The Typography & Design of Vehicle Dashboards. URL: <https://www.printmag.com/typography/driven-distraction-dashboard-design-typography/>, Son Erişim Tarihi: 18.06.2019.
- İnternet: Shashidhara, A., Deepak, G., Manjunath, B., Murthy, A. and Ruckmongathan, T. (2004). Liquid crystal display for an automobile dashboard. Raman Research Institute, C.V. Raman Avenue, Sadashivanagar, Bangalore, India 560080, URL: <http://dspac.rri.res.in/bitstream/2289/1171/1/2004%20Photonics%20Liquid%20crystal....pdf>, Son Erişim Tarihi: 18.06.2019.
- İnternet: The MathWorks, Inc. (2019a). Getting started with serial port communication. URL: [https://nl.mathworks.com/help/matlab/matlab\\_external/getting-started-with-serial-i-o.html](https://nl.mathworks.com/help/matlab/matlab_external/getting-started-with-serial-i-o.html), Son Erişim Tarihi: 18.06.2019.
- İnternet: The MathWorks, Inc. (2019b). Modeling an anti-lock braking system. URL: <https://nl.mathworks.com/help/simulink/slref/modeling-an-anti-lock-braking-system.html>, Son Erişim Tarihi: 18.06.2019.
- İnternet: The MathWorks, Inc. (2019c). Videos and Webinars <https://nl.mathworks.com/videos/reading-excel-data-into-matlab-with-a-gui-part-1-98224>, Son Erişim Tarihi: 18.06.2019.

- İnternet: Toyota Sienna. (2019). Toyota Sienna Service Manual: Right Front Wheel Speed Sensor Signal. URL: [http://www.tsienna.net/right\\_front\\_wheel\\_speed\\_sensor\\_signal-1173.html](http://www.tsienna.net/right_front_wheel_speed_sensor_signal-1173.html), Son Erişim Tarihi: 17.06.2019.
- İnternet: Wikipedia. (2019). CAN bus. URL: <http://www.wikizero.biz/index.php?q=aHR0cHM6Ly9lbi53aWtpcGVkaWEub3JnL3dpa2kvQ0FOX2Jlcw>, Son Erişim Tarihi: 17.06.2019.
- İnternet: Wikizero. (2019). Bluetooth. <http://www.wikizero.biz/index.php?q=aHR0cHM6Ly90ci53aWtpcGVkaWEub3JnL3dpa2kvQmx1ZXRVb3Ro>, Son Erişim Tarihi: 17.06.2019.
- Jasper. (2014). Duur, Duurder, Mercedes S65 AMG Coupé. URL: <http://www.drivessential.com/duur-duurder-mercedes-s65-amg-coupe/>, Son Erişim Tarihi: 14.07.2019.
- Jeon, M., Schuett, J., Yim, J.-B., Raman, P. and Walker, B. N. (2011). *ENGIN (Exploring Next Generation IN-vehicle INterfaces): Drawing a new conceptual framework through iterative participatory processes*. 3rd International Conference on Automotive User Interfaces and Interactive Vehicular Applications. University of Salzburg, Salzburg, 13-14
- Jinadasa, G. (2016). *Study and designing of instrument cluster*. University of Wolverhampton for the degree of Bachelor of Engineering, Wolverhampton.
- Johann, V. and Mahr, A. (2011). *A new icon selection test for the automotive domain based on relevant ISO usability criteria*. 3rd International Conference on Automotive User Interfaces and Interactive Vehicular Applications. University of Salzburg, Salzburg, 25-26.
- Kandemir, V. (2011). *Genel amaçlı programlanabilir kontrolör*. Yüksek Lisans Tezi, İstanbul Teknik Üniversitesi Fen Bilimleri Enstitüsü, İstanbul.
- Lousberg, T. (2016). *MATLAB/simulink ile kamyon simülâtörü oluşturmak*. Eindhoven: Eindhoven Teknoloji Üniversitesi.
- Mantovani, S., Lo Presti, I., Cavazzoni, L. and Baldini, A. (2017). *Influence of manufacturing constraints on the topology optimization of an automotive dashboard*. 27th International Conference on Flexible Automation and Intelligent Manufacturing, Modena, Italy, 1700 – 1708.
- Marques, G. and Pitarma, R. (2019). An internet of things-based environmental quality management system to supervise the indoor laboratory conditions. *Applied Sciences*, 9(3), 438.
- Martino, R. D. (2012). *Otomobilin dinamik davranışlarının simülasyonu ve modellenmesi (Modelling and simulation of the dynamic behaviour of the automobile)*. Mulhouse: Haute Alsace Üniversitesi.
- McDonald, D. (2012). *Electric vehicle drive simulation with matlab/simulink*. Michigan: American Society for Engineering Education.

- Morrow, J. D., Cole, K. and Davis, J. (2011). *On detecting distracted driving using readily-available vehicle sensors: preliminary results*. 3rd International Conference on Automotive User Interfaces and Interactive Vehicular Applications. University of Salzburg, Salzburg, 31-32.
- Muratoğlu, Y. (2017). *Elektrikli araçlarda kullanılan lityum iyon pillerin şarj durumlarının kokusuz Kalman filtresi ile kestirilmesi*. Yüksek Lisans Tezi, Mersin Üniversitesi Fen Bilimleri Enstitüsü, Mersin.
- Neureiter, K., Meschtscherjakov, A., Wilfinger, D. and Tscheligi, M. (2011). *Investigating the usage of multifunctional rotary knobs in the center stack with a contextual inquiry*. 3rd International Conference on Automotive User Interfaces and Interactive Vehicular Applications. University of Salzburg, Salzburg, 19-20.
- Noergaard, T. (2005). *Embedded System Architecture a comprehensive guide for engineers and programmers*. Washington: Newnes, 672.
- Osswald, S. and Tscheligi, M. (2011). *Gas station creative probing: The front seat passenger*. 3rd International Conference on Automotive User Interfaces and Interactive Vehicular Applications. University of Salzburg, Salzburg, 23-24.
- Özdemir, A. (2014). *Asansör haberleşme sistemlerinde canbus hatatolerans modu kullanımı*. İzmir: Tmmob Makina Mühendisleri Odası İzmir Şube ve Elektrik Mühendisleri Odası Birlikteliği, 289-294.
- Pitts, M. J., Wellings, T., Attridge, A. and Williams, M. A. (2011). *An approach to user-focused development of high-value human-machine interface concepts*. 3rd International Conference on Automotive User Interfaces and Interactive Vehicular Applications. University of Salzburg, Salzburg, 51-52.
- Porter, J., Case, K., Freer, M. and Bonney, M. (1993). *Computer aided ergonomics design of automobiles*. London: Loughborough University.
- Riener, A. and Rossbory, M. (2011). *Natural and intuitive hand gestures: A substitute for traditional vehicle control?* 3rd International Conference on Automotive User Interfaces and Interactive Vehicular Applications. University of Salzburg, Salzburg, 33-34.
- Riener, A. and Wintersberger, P. (2011). *3D theremin: A novel approach for convenient "point and click"-interactions in vehicles*. 3rd International Conference on Automotive User Interfaces and Interactive Vehicular Applications. University of Salzburg, Salzburg, 35-36.
- Rouleau, G. (2019, 06 18). Creating Your Own Block With Continuous States (MATLAB S-Function), URL: <https://blogs.mathworks.com/simulink/2012/06/25/creating-your-own-block-with-continuous-states-matlab-s-function/>, Son Erişim Tarihi: 18.06.2019.
- Samuel, O. A., Israel, D. and Moses, F. (2016). Anthropometry survey of nigerian. *World Journal of Engineering and Technology*, 4(02), 176-182.
- San, S. (2015). PIC ile USB. *Mekatronik Dergisi*, 33-39.

- Sarı Kurt, T. and Balıkçı, A. (2017). Tam elektrikli araçlar için özgün bir enerji yönetim sistemi uygulaması. *Gazi Üniversitesi Mühendislik-Mimarlık Fakültesi Dergisi*, 32(2), 323-333.
- Scherz, P. and Monk, S. (2013). *Practical electronics for inventors*. New York: McGraw-Hill, 29.
- Scott, G., Vunakece, S., Vosawale, R. D., Assaf, M. and Mehta, U. V. (2019). A flexible dashboard panel system for electric vehicle. *World Scientific and Engineering Academy and Society Transactions on Electronics*, 10, 33-41.
- Soyaslan, M. (2015). DC servo motor simulink modeli. *Mekatronik Dergisi*, 4-11.
- Sureshbabu, P., Mohan, P., Santhiyakumari, N. and Thangavel, M. (2015). Analysis and design of digital meter in vehicles to control fuel consumption using UTLP. *International Journal of Innovative Research in Science, Engineering and Technology*, 4(6), 635-642.
- Suvak, H. (2016). The simulation of a full electric vehicle using the city cycle. *International Journal of Automotive Engineering and Technologies*, 5(2), 38-46.
- Tarlak, H. and İşen, E. (2018). Elektrikli araçlar ve akü şarj sistemleri. *Kırklareli Üniversitesi Mühendislik ve Bilim Dergisi*, 124-141.
- Tezde, E. İ. and Okumuş, H. İ. (2018). Batarya modelleri ve şarj durumu (SoC) belirleme. *Elektrik Mühendisleri Odası Bilimsel Dergi*, 8(1), 21-25.
- Topsakal, M., Yılmaz, E. and Tayfur, Y. (2016). *Quadcopter uçuş kontrolcüsünün tasarlanması*. Trabzon: Karadeniz Teknik Üniversitesi.
- Uğur, E. (2011). *Prototip bir elektrikli araç üzerine enerji yönetim sisteminin uygulanması*. İstanbul: Yıldız Teknik Üniversitesi.
- Uzun, H. İ. and Ayan, A. Ö. (2015). *Renk okuyucu robot*. Trabzon: Karadeniz Teknik Üniversitesi.
- Ünalmiş, B. (2015). Endüstriyel. *Mekatronik Dergisi*, 19.
- Vilarinho, S., Lopes, I. and Sousa, S. (27-30 June 2017). *Design procedure to develop dashboards aimed at improving the performance of productive equipment and processes*. 27th International Conference on Flexible Automation and Intelligent Manufacturing, FAIM2017, Modena, Italy, 1634 – 1641
- Wilfinger, D., Murer, M. and Tscheligi, M. (2011). *Open car user experience lab: A practical approach to evaluating car HMIs holistically*. 3rd International Conference on Automotive User Interfaces and Interactive Vehicular Applications. University of Salzburg, Salzburg, 37-38.
- Yol, Ö. (2019). DMA Forklift Batarya Sistemi. URL: <https://docplayer.biz.tr/108843297-Onder-yol-dma-yonetim-kurulu-baskani-dma-forklift-batarya-sistemi.html>, Son Erişim Tarihi: 18.06.2019.

- Zhou, H., Itoh, M. and Inagaki, T. (2011). *Situation-adaptive driver assistance systems for safe lane change via inferring driver intent*. 3rd International Conference on Automotive User Interfaces and Interactive Vehicular Applications. University of Salzburg, Salzburg, 39-40.
- Zhou, S. (2011). *Automotive cockpit design 2020*. Göteborg: Chalmers University of Technology.
- Zhu, D., Wang, L., Henaut, J. and Beeby, S. (2014). *Comparisons of energy sources for autonomous in-car wireless tags for asset tracking and parking applications*. *Procedia Engineering*, 87, 783-786.



## **EKLER**

EK-1.

GAZİ ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ MÜDÜRLÜĞÜNE

Makine Eğitimi Ana Bilim Dalı Yüksek Lisans tez öğrencisi olan 010150301 numaralı İzzet Timur YARDIMCI'nın "ELEKTRİKLİ BİR TAŞIT İÇİN GÖSTERGE PANELİ TASARIMI" başlıklı tezinin, Yükseköğretim Kurulu Lisansüstü Tezlerin Elektronik Ortamda Toplanması, Düzenlenmesi ve Erişime Açılmasına ilişkin Yönergesinin 6.2. Maddesi gereğince yeni teknik materyal ve metotların kullanıldığı, henüz makaleye dönüşmemiş veya patent gibi yöntemlerle korunmamış ve internetten paylaşılması durumunda 3. şahıslara veya kurumlara haksız kazanç imkanı oluşturabilecek bilgi ve bulguları içermesi sebebiyle uygulamanın geliştirilmesi için kullanılan Ekler kısmında verilen kodlar kaldırılmıştır.

02.01.2020

İzzet Timur YARDIMCI



## ÖZGEÇMİŞ

### Kişisel Bilgiler

Soyadı, adı : YARDIMCI, İzzet Timur  
 Uyuğu : T.C.  
 Doğum tarihi ve yeri : 23.04.1975, HOLLANDA  
 Medeni hali : Bekar  
 Telefon : 0 (555) 432 98 76  
 e-mail : izztti@gmail.com



### Eğitim

Derece	Eğitim Birimi	Mezuniyet Tarihi
Yüksek lisans	Gazi Üniversitesi / Makine Eğitimi	Devam ediyor
Lisans	Osmangazi Üniversitesi / Makine Mühendisliği	Devam Ediyor
Lisans	Gazi Üniversitesi / Otomotiv Öğretmenliği	2005
Ön lisans	Akdeniz Üniversitesi / Motor	2000
Lise	Atatürk Endüstri Meslek Lisesi / Metal İşleri	1993

### İş Deneyimi

Yıl	Yer	Görev
2018-2019	TG Otomasyon	Ar-Ge ve Tasarım
2015-2016	Kuasar Bilişim	Ürün Tasarım Uzmanı
2013-2014	Şahlar Kabin	Kaynak Robotları Sorumlusu
2001-2001	Raf Sistemler	Mekik Sistemi Tasarımı
2012-2013	Albayrak Makine	Estram Hattı Durak Tasarımı
2008-2009	ATY Tanıtım	Aydınlatma Tasarımı
2006-2007	Kentimaj Kentsel Tasarım	3B görselleştirme uzmanı

### Yabancı Dil

İngilizce  
 Flemenkçe

**Yayınlar**

Yardımcı, İ. T. ve Erşan, K. (2019, Haziran). *Elektrikli araçlar için araç iç ünitelerinin haberleşme prototipinin geliştirilmesi*. VI. Uluslararası Fen, Mühendislik ve Mimarlık Bilimlerinde Akademik Çalışmalar Sempozyumu, Ankara, Türkiye.

**Hobiler**

3D oyun tasarımı, animasyon ve simülasyon uygulamaları, 3D model tasarımı.



*GAZİ GELECEKTİR..*