

T.C.
FIRAT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ



ELEKTRİKLİ ARAÇLARDA GÜVENLİK ÖNLEMLERİ

Merve YILDIRIM

Doktora Tezi

ELEKTRİK ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI

Elektrik Makinaları Bilim Dalı

OCAK 2022

T.C.
FIRAT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

Elektrik Elektronik Mühendisliği Anabilim Dalı

Doktora Tezi

ELEKTRİKLİ ARAÇLARDA GÜVENLİK ÖNLEMLERİ

Tez Yazarı
Merve YILDIRIM

Danışman
Prof. Dr. Hasan KÜRÜM

OCAK 2022
ELAZIĞ

T.C.
FIRAT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

Elektrik Elektronik Mühendisliği Anabilim Dalı

Doktora Tezi

Başlığı: Elektrikli Araçlarda Güvenlik Önlemleri

Yazarı: Merve YILDIRIM

İlk Teslim Tarihi: 10.01.2022

Savunma Tarihi: 25.01.2022

TEZ ONAYI

Fırat Üniversitesi Fen Bilimleri Enstitüsü tez yazım kurallarına göre hazırlanan bu tez aşağıda imzaları bulunan jüri üyeleri tarafından değerlendirilmiş ve akademik dinleyicilere açık yapılan savunma sonucunda OYBİRLİĞİ ile kabul edilmiştir.

Danışman:	Prof. Dr. Hasan KÜRÜM Fırat Üniversitesi, Mühendislik Fakültesi	İmza Onayladım
Başkan:	Dr. Öğr. Üyesi Abuzer ÇALIŞKAN Fırat Üniversitesi, Mühendislik Fakültesi	Onayladım
Üye:	Doç. Dr. Eyyüp ÖKSÜZTEPE Fırat Üniversitesi, Sivil Havacılık Yüksekokulu	Onayladım
Üye:	Dr. Öğr. Üyesi Taner GÖKTAŞ Dokuz Eylül Üniversitesi, Mühendislik Fakültesi	Onayladım
Üye:	Dr. Öğr. Üyesi Zeki OMAÇ Munzur Üniversitesi, Mühendislik Fakültesi	Onayladım

Bu tez, Enstitü Yönetim Kurulunun/...../20..... tarihli toplantısında tescillenmiştir.

İmza
Prof. Dr. Kürşat Esat ALYAMAÇ
Enstitü Müdürü

BEYAN

Fırat Üniversitesi Fen Bilimleri Enstitüsü tez yazım kurallarına uygun olarak hazırladığım “Elektrikli Araçlarda Güvenlik Önlemleri” Başlıklı Doktora Tezimin içindeki bütün bilgilerin doğru olduğunu, bilgilerin üretilmesi ve sunulmasında bilimsel etik kurallarına uygun davrandığımı, kullandığım bütün kaynakları atıf yaparak belirttiğimi, maddi ve manevi desteği olan tüm kurum/kuruluş ve kişileri belirttiğimi, burada sunduğum veri ve bilgileri unvan almak amacıyla daha önce hiçbir şekilde kullanmadığımı beyan ederim.

25.01.2022

Merve YILDIRIM



ÖNSÖZ

Son yıllarda Elektrikli Araçlara (EA) olan ilgi büyük ölçüde artmaktadır. Bunun en önemli nedenleri, çevre ve hava kirliliğini önlemesi, sessiz çalışmaları ve enerji tasarrufu sağlamaları gibi avantajlara sahip olmasıdır. Bu yüzden bu alanda yapılan çalışmalar büyük önem kazanmıştır. EA'larda kullanılacak motorların seçimi de oldukça önemlidir. Motorlardan başarılı kontrol performansı, hızlı moment cevabı, güvenilirlik ve yüksek verim beklenmektedir. Bu yüzden bu özellikleri karşılayan Sürekli Mıknatıslı Senkron Motorlar (SMSM), EA'lar için uygundur ve bu çalışmada da tercih edilmiştir.

Günümüzde mekanik diferansiyeller ve aktarma organları EA'nın ağırlığını arttırmaktadır. Gelişen motor tasarımları sayesinde motorlar tekerlek içlerine yerleştirilerek her bir tekerlek bağımsız olarak kontrol edilebilir ve geleneksel mekanik diferansiyeller yerine Elektronik Diferansiyel Sistemler (EDS) kullanılarak aracın ağırlığı azaltılabilmektedir. Bunun yanı sıra, arıza bakım masrafları ve mekanik kayıplar da azaltılarak EA'nın performansı artırılmaktadır. Böylece bu çalışmada tasarlanan EDS ile dört tekerlekten tahrikli bir EA için direksiyon açısına ve aracın hızına göre tekerlek hızları değiştirilerek farklı yol durumlarında sürüş stabilitesi sağlanmaktadır.

EA'lara olan ilginin ve bilimsel çalışmaların artması beraberinde bu araçlarda uygulanacak olan güvenli sürüş uygulamalarının da önem kazanmasını sağlamıştır. Bu yüzden bu çalışmada geliştirilen Sürücü Destek Sistemleri (SDS) ile donanımsal bir aktif güvenlik sistemi tasarlanarak akıllı sistemlerin geliştirilmesi hedeflenmektedir. Kullanılan Kontrol Alan Ağı (CAN-Bus) haberleşme protokolünün getirdiği avantajlar ile dört adet SMSM'ler, sürücüler, işlemciler ve sensörlerden oluşan tüm sistemin yüksek hızda, güvenilir şekilde haberleşmesi sağlanmaktadır. Gerçekleştirilen kamera tabanlı görüntü işleme uygulamalarıyla SMSM'lere CAN-Bus hattı üzerinden bilgi gönderilerek motorlar kontrol edilebilmektedir. Ayrıca, EA'nın farklı yol durumlarında güvenli ilerleyebilmesi için sürücüyü uyaran bir SDS geliştirilerek risk analizi yapılmaktadır. Böylece bu çalışma ile otonom araçların önemli bir basamağı oluşturularak teknolojiye ve bilime katkı sağlanmaktadır.

Çalışmalarım boyunca değerli bilgi birikimleriyle yardımlarını eksik etmeyen, tecrübeleriyle yolumu aydınlatan ve çalışmalarımın gerçekleşmesi için gerekli zaman ve imkânları sağlayan değerli danışman hocam Sayın Prof. Dr. Hasan KÜRÜM'e en içten teşekkürlerimi sunarım.

Zaman ve mekân gözetmeksizin yardımlarını hiçbir zaman esirgemeyen, değerli bilgileri ve deneyimleriyle tez çalışmama önemli katkılar sağlayan çok kıymetli hocalarım Doç. Dr. Eyyüp ÖKSÜZTEPE'ye, Dr. Öğr. Üyesi Özgür KARADUMAN'a, Dr. Öğr. Üyesi Haluk EREN'e ve Dr. Öğr. Üyesi Abuzer ÇALIŞKAN hocalarıma da teşekkürü bir borç bilirim. Fırat Üniversitesi Fırat Racing Formula Student Elektrikli Araç yarış projesi kapsamında gerçekleştirdiğimiz tez çalışmama vermiş oldukları önemli finansal desteklerden dolayı Else Enerji'ye ve üniversitemize de çok teşekkür ederim. Deneysel çalışmalarım boyunca uzaktan da olsa her zaman desteğini esirgemeyen EHR Elektronik Hidrolik ve Otomasyon A.Ş. sahibi Sayın Oktay KARAKAYA'ya da çok teşekkür eder, şükranlarımı sunarım.

Hayatım boyunca desteklerini her zaman hissettiğim, moral ve motivasyonlarıyla hep yanımda olan canım aileme ve tüm yakınlarıma çok teşekkür ederim. Ayrıca akademik hayatım boyunca her zaman bana yoldaş olan ve çalışmalarımına destek veren canım dostum Arş. Gör. Dr. Güllü BOZTAŞ'a ve yanımda olan tüm arkadaşlarıma çok teşekkür ederim.

Merve YILDIRIM

ELAZIĞ, 2022

İÇİNDEKİLER

	Sayfa
ÖNSÖZ.....	iv
İÇİNDEKİLER	v
ÖZET	viii
ABSTRACT	ix
ŞEKİLLER LİSTESİ	x
TABLolar LİSTESİ	xv
SİMGELER	xvi
KISALTMALAR	xix
1. GİRİŞ	1
1.1. Genel Bilgi	1
1.2. Literatür Araştırması.....	2
1.3. Tezin Organizasyonu	11
1.4. Tezin Katkısı ve Amacı	12
2. ELEKTRİKLİ ARAÇLAR	15
2.1. Elektrikli Araçların Tarihi	16
2.2. Elektrikli Araç Teknolojisi	18
2.2.1. Bataryalı Elektrikli Araç	18
2.2.2. Hibrit Elektrikli Araç	19
2.2.3. Yakıt Hücresi Elektrikli Araç	21
2.2.4. Elektrikli Araçların Diğer Çeşitleri	21
2.3. Tekerlek İçi Motorlu Elektrikli Araç Mimarisi	22
2.4. Elektrikli Araçların Çalışma Prensipleri	22
2.4.1. Elektrikli Araçların Avantaj ve Dezavantajları	25
3. SÜREKLİ MİKNATISLI SENKRON MOTORLAR.....	27
3.1. Sürekli Mıknatıslı Senkron Motorların Yapısı	27
3.1.1. Yüzey Mıknatıslı Sürekli Mıknatıslı Senkron Motorlar	29
3.1.2. Dâhili Sürekli Mıknatıslı Senkron Motorlar.....	30
3.1.3. Radyal Akıllı Motorlar	31
3.1.4. Eksenel Akıllı Motorlar	32
3.2. Sürekli Mıknatıslı Senkron Motorların Çalışma Prensipleri.....	33
3.3. Sürekli Mıknatıslı Senkron Motorların Dinamik Modeli	33
3.4. Sürekli Mıknatıslı Senkron Motorların Kontrolü	39
4. ARAÇLARDA VE ENDÜSTRİDE AĞ SİSTEMLERİ	41
4.1. Motor Yönetim Sistemleri	41
4.1.1. EKÜ'nün Yapısı ve Çalışması	41
4.2. Veri İletim Sistemleri	42
4.2.1. Seri Haberleşme	43
4.2.2. Paralel Haberleşme.....	43
4.3. Haberleşme Protokolleri	43
4.4. Elektronik Kontrol Üniteleri Arasında Haberleşme	45
4.4.1. CAN-Bus Haberleşme Protokolü	45
4.4.2. Yerel Bağlantı Ağı (LIN)	55

4.4.3. Medya Odaklı Sistemleri Taşıma Ağı (MOST)	56
5. SAYISAL GÖRÜNTÜ İŞLEME.....	58
5.1. Görüntü İşlemede Temel Kavramlar	58
5.1.1. Sayısal Görüntü	58
5.1.2. Renk Uzayları.....	58
5.2. Görüntü İşleme	60
5.3. Görüntü İşleme Kullanım Alanları	61
5.4. Sayısal Görüntü İşlemede Temel Adımlar.....	61
5.5. Görüntü İşleme Algoritmaları.....	62
5.5.1. Renkli Görüntülerin Griye Dönüştürülmesi	62
5.5.2. Impuls Cevabı ve Konvolüsyon	63
5.5.3. İki Boyutlu Lineer Konvolüsyon.....	63
5.5.4. Medyan Filtre	66
5.5.5. Sobel Kenar Çıkarma Filtresi	66
5.5.6. Prewitt Kenar Çıkarma Filtresi.....	69
5.5.7. Robert Kenar Çıkarma Filtresi	70
5.5.8. Canny Kenar Çıkarma Filtresi.....	70
6. TEKERLEK İÇİ MOTORLU ELEKTRİKLİ ARAÇLAR İÇİN ELEKTRONİK DİFERANSİYEL SİSTEM TASARIMI.....	72
6.1. Giriş	72
6.2. Elektronik Diferansiyel Sistemler.....	72
6.3. Ackermann Jeantand Modeli	73
6.4. Elektronik Diferansiyel Sistemin Modellenmesi	75
6.5. HY-TTC60 İşlemci.....	76
6.6. PCAN-USB CAN Hattı	77
6.7. CoDeSys Yazılımı	78
6.7.1. CoDeSys Fonksiyonları.....	79
6.7.2. CoDeSys Proje Bileşenleri	79
6.8. Deney Düzeneği	85
6.9. Dört Tekerlek İçi Motorlu Bir Elektrikli Aracın Dinamik Modeli	91
6.9.1. Hava Sürtünme Direnci	94
6.9.2. Lastik Teker Dinamiği	95
6.10. Dört Tekerlek İçi Motorlu Bir Elektrikli Aracın Kontrolü	97
7. FPGA VE RASPBERRY PI İLE GÖRÜNTÜ İŞLEME VE GÜVENLİ SÜRÜŞ UYGULAMALARI	103
7.1. Kullanılan Donanımlar	103
7.1.1. Raspberry Pi	103
7.1.2. Alanda Programlanabilir Kapı Dizileri (FPGA).....	103
7.2. Gerçekleştirilen Görüntü İşleme Uygulamaları	105
7.2.1. Xilinx System Generator Kullanılarak Görüntü ve Video İşleme Uygulamaları	105
7.2.2. RGB Görüntülerin Gri Seviyeye Dönüştürülmesi	105
7.2.3. Görüntüyü Tersine Çevirme	106
7.2.4. Görüntü Geliştirme.....	107
7.2.5. Eşikleme (Thresholding)	108
7.2.6. Erozyon ve Genişletme	109
7.2.7. Kenar Çıkarma	110

7.3. Raspberry Pi’de Gerçekleştirilen Görüntü İşleme Uygulamaları	111
7.3.1. Canny Kenar Çıkarma Algoritması	111
7.3.2. Yol Çizgilerinin Takibi	111
7.3.3. Nesne Tanıma.....	112
7.3.4. Yüz ve Göz Tanıma.....	113
7.3.5. Hareket Algılama	113
7.4. Dört Tekerlek İçi Motorlu Elektrikli Araçta Güvenli Sürüş Uygulamaları	114
7.5. Viraj Yönü ve Açısının Tespit Edilmesi ve Risk Hesabı.....	117
7.6. Araca Etki Eden Kuvvetlerin İncelenmesi ve Risk Hesabı.....	120
7.7. Metodun Gerçekleştirilmesi	121
7.7.1. Şeritler ve Hedef Noktaların Belirlenmesi	121
7.7.2. Şerit Piksellerini Filtreleme: Gradyant ve Renk Eşikleme	121
7.7.3. Polinom Uydurma ve Eğrilik Ölçümü.....	123
7.7.4. Segmentasyon (Bölütleme)	124
8. SONUÇLAR.....	125
8.1. Elektrikli Aracın Dinamik Modelinden Alınan Sonuçlar	125
8.2. Codesys, Matlab/Simulink Simulasyonları ve Deneysel Sonuçlar	128
8.3. Elektrikli Aracın Güvenli Sürüşü Üzerine Yapılan Çalışmaların Sonuçları	139
8.3.1. Kameradan Alınan Görüntülere Göre Engel Tespiti	139
8.3.2. Viraj Yönü ve Açısının Tespiti ile Risk Hesabı	141
8.4. Sonuçlar ve Değerlendirme	145
ÖNERİLER	147
KAYNAKLAR.....	148
ÖZGEÇMİŞ	

ÖZET

Elektrikli Araçlarda Güvenlik Önlemleri

Merve YILDIRIM

Doktora Tezi

FIRAT ÜNİVERSİTESİ
Fen Bilimleri Enstitüsü

Elektrik Elektronik Mühendisliği Anabilim Dalı

Ocak 2022, Sayfa: xix + 152

Son yıllarda teknolojinin gelişimiyle artan araç kullanımı, bazı sorunları da beraberinde getirmiştir. Fosil yakıtlı araçların çoğalmasıyla, yaydığı zararlı gazlar hava ve çevre kirliliğine yol açmaktadır. Bunu ortadan kaldırmak için Elektrikli Araçların (EA) kullanımı önem kazanmıştır.

EA’larda kullanılacak motorların seçimi oldukça önemlidir. Bu motorlardan başarılı kontrol performansı, hızlı moment cevabı ve yüksek verim istenir. Sürekli Mıknatıslı Senkron Motorlar (SMSM) bu özellikleri karşıladıklarından EA’lar için uygundur ve bu çalışmada da tercih edilmiştir. Gelişen motor tasarımları sayesinde EA’nın ağırlığını azaltmak için motorlar tekerleklerin içine yerleştirilir ve Elektronik Diferansiyel Sistemler (EDS) kullanılmaya başlanmıştır.

Bu çalışmada, dört tekerlekten tahrikli bir EA için EDS tasarlanarak dinamik modeli oluşturulmuş ve EA’larda güvenlik önlemleri üzerine çalışmalar yapılmıştır. EA’larda yer alan motor, sürücü, batarya gibi tüm donanımın güvenli şekilde kontrol edilmesi oldukça önemlidir. Bunun için Kontrol Alan Ağı (CAN-Bus) protokolüyle sisteme bağlanan tüm işlemci ve sensörler arasında yüksek veri hızında güvenilir bir haberleşme gerçekleştirilmiştir. Tasarlanan sistemde, her bir SMSM tekerlek içine yerleştirilir ve motorlar bağımsız olarak kontrol edilir. Direksiyon açısına ve aracın hızına bağlı olarak dört SMSM hızları simülasyonlarla ve deneysel çalışmalarla hesaplanmıştır. Araç dinamiklerini de katarak oluşturulan modelde, SMSM’lerin gerçek hızları, EDS’den alınan referans hızlarla karşılaştırılarak bir kontrolör tasarlanmış ve sistemin stabilitesi sağlanmıştır. Önerilen bu kontrol yöntemi, literatürdeki karmaşık yöntemlerden daha basit bir yaklaşım sunmakta ve işlem kolaylığı sağlamaktadır. Daha sonra, gömülü sistem platformları ile kameradan alınan gerçek zamanlı görüntüler işlenmiş ve araca tehlike oluşturabilecek durumlar önlenerek motorlar kontrol edilmiştir. Böylece bu çalışmayla, tüm üniteler arasında sağlam ve hızlı iletişim sağlanarak kararlı ve güvenilir bir sistem tasarlanmıştır.

Anahtar Kelimeler: Elektrikli araç, Elektronik diferansiyel sistem, Kontrol alan ağı, Sürekli mıknatıslı senkron motor, Görüntü işleme, Güvenli sürüş.

ABSTRACT

Safety Precautions in Electric Vehicles

Merve YILDIRIM

Ph.D. Thesis

FIRAT UNIVERSITY

Graduate School of Natural and Applied Sciences

Department of Electrical and Electronics Engineering

January 2022, Pages: xix + 152

In recent years, increasing use of the vehicles with development of technology has brought some problems with it. Due to widespread use of internal combustion engine vehicles, harmful gases cause significant environmental pollutions. To eliminate them, use of Electric Vehicles (EVs) has gained importance.

The selection of the motor to be used in the EVs is very important. Successful control performance, fast torque response, and high efficiency are desired from these motors. Therefore, Permanent Magnet Synchronous Motors (PMSMs) are preferred in this study. Thanks to developing motor designs, the PMSMs are placed into the wheels to reduce weight of the EV and Electronic Differential Systems (EDS) are used.

In this study, a dynamic model of the EV is created by designing an EDS with four In-wheel PMSMs and safety precautions are studied. It is very important to safely control all equipment such as motor, driver, battery in EVs. Hence, Controller Area Network-Bus communication with high baud-rate is used. According to steering angle and speed of the EV, the PMSM speeds are calculated by simulations and experimental works. Then, a controller is designed and actual speeds of the PMSMs are compared with speeds taken from the EDS. This proposed method offers a simpler approach than complex methods in the literature. After, real-time images taken from a camera are processed by embedded system platforms and motors are controlled by preventing dangerous situations for the EV. Thus, a stable and reliable system is designed by providing robust and fast communication between all units.

Keywords: Electric vehicle, Electronic differential system, Control area network-bus, Permanent magnet synchronous motor, Image processing, Safe driving.

ŞEKİLLER LİSTESİ

	Sayfa
Şekil 2.1. İcat edilen ilk elektrikli araç [52].	16
Şekil 2.2. La Jamais Contente elektrikli aracı [52].	17
Şekil 2.3. Tesla Roadster ve Mitsubishi MiEV [54].	18
Şekil 2.4. Seri hibrit elektrikli aracın tasarımı [52].	19
Şekil 2.5. Paralel hibrit elektrikli aracın tasarımı [52].	20
Şekil 2.6. Yakıt hücresel elektrikli aracın yapısı [52].	21
Şekil 2.7. Tekerlek içi motor konfigürasyonu [52].	22
Şekil 2.8. Elektrikli araç yapısı [55].	23
Şekil 3.1. Bir Yüzeye Monte Sürekli Mıknatıslı Senkron Motorun yapısı	28
Şekil 3.2. Yüzey mıknatıslı rotor yapısı	29
Şekil 3.3. Dâhili mıknatıslı rotor yapısı	30
Şekil 3.4. Motor yapısına göre farklı SMSM çeşitleri	31
Şekil 3.5. a) YMSMSM b) DSMSM c) Spoke SMSM d) Bariyer SMSM rotor çeşitleri [64].	31
Şekil 3.6. Bir dış rotor SMSM'nin yapısı [64].	32
Şekil 3.7. Eksenel SMSM'nin yapısı [64].	33
Şekil 3.8. Bir SMSM'nin üç faz eşdeğer devresi [63].	34
Şekil 3.9. Bir DSMSM'nin motor hız kontrol şeması	39
Şekil 4.1. Haberleşme sistemlerinin kullanım yerleri [72].	45
Şekil 4.2. CAN-Bus haberleşme hattı [73].	47
Şekil 4.3. CAN-Bus veri yolu dönüştürücüsü [73].	48
Şekil 4.4. Standart veri çerçevesi	49
Şekil 4.5. Genişletilmiş veri çerçevesi	49
Şekil 4.6. SOF biti [73].	49
Şekil 4.7. Standart çerçeve denetim alanı	50
Şekil 4.8. Genişletilmiş çerçeve denetim alanı	50
Şekil 4.9. Kontrol alanı	51
Şekil 4.10. Veri alanı	51
Şekil 4.11. CRC alanı	51
Şekil 4.12. ACK alanı	52
Şekil 4.13. Çerçeve sonu alanı	52
Şekil 4.14. Standart istek çerçevesi	52

Şekil 4.15.	Genişletilmiş istek çerçevesi.....	53
Şekil 4.16.	Hata çerçevesinin yapısı.	53
Şekil 4.17.	Taşma çerçevesinin yapısı.	54
Şekil 4.18.	Bit gönderme işlemi.....	54
Şekil 4.19.	Bir araçta CAN ve LIN hatlarının gösterimi [75].	56
Şekil 4.20.	MOST hattında kullanılan sistemler [76].....	57
Şekil 5.1.	YUV renk yoğunlukları ve değerleri [77].	59
Şekil 5.2.	HSV renk uzayı [79].	60
Şekil 5.3.	Sayısal görüntü işleme adımları [77].	61
Şekil 5.4.	Konvolüsyon işlemi	63
Şekil 5.5.	2 boyutlu konvolüsyon.....	65
Şekil 5.6.	Görüntüye uygulanan konvolüsyon işlemi ve sonucu.....	65
Şekil 5.7.	Konvolüsyon işleminde çıkış pikselinin hesaplanması.....	66
Şekil 5.8.	Medyan filtrenin çıkışı.....	66
Şekil 5.9.	Bir görüntünün Kartezyen koordinat sisteminde gösterimi [84].....	67
Şekil 5.10.	Sobel filtre matrisleri G_x ve G_y	69
Şekil 5.11.	Sobel filtresi ve çıkışı [86].....	69
Şekil 5.12.	Prewitt filtre matrisleri G_y ve G_x	70
Şekil 5.13.	Robert filtre matrisleri G_y ve G_x	70
Şekil 6.1.	Ackermann-Jeantand modeli [1].	73
Şekil 6.2.	EDS'nin Matlab/Simulink modeli	75
Şekil 6.3.	HY-TTC 60 işlemci [89].....	76
Şekil 6.4.	CAN veri hattı [90].	76
Şekil 6.5.	PCAN-USB kablosu	77
Şekil 6.6.	PCAN-USB bağlantı uçları [92].	78
Şekil 6.7.	PCAN-USB bağlantı şeması [92].	78
Şekil 6.8.	IL dilinde bir fonksiyon örneği [93].	80
Şekil 6.9.	IL dilinde bir örnek fonksiyon bloğu.	81
Şekil 6.10.	IL dilinde örnek bir program.....	81
Şekil 6.11.	Değişkenlerin izlenmesi.....	84
Şekil 6.12.	Deney düzeneği (1. Dört adet DSMSM'ler, 2. Sürücüleri, 3. PC, 4. 24 V DC kaynak, 5. HY-TTC60 işlemci, 6. Enkoder, 7. Gaz pedalı).	85
Şekil 6.13.	Dört adet DSMSM ve sürücüleri	85
Şekil 6.14.	Tüm parametreleri aktif etme [94].	86

Şekil 6.15.	Veri alma ve gönderme adresleri [94].....	88
Şekil 6.16.	PCAN-View'den alınan ve gönderilen mesajlar.....	90
Şekil 6.17.	EDS'nin akış diyagramı.....	91
Şekil 6.18.	Elektrikli aracın fiziksel modeli.....	92
Şekil 6.19.	Lastik temas yüzeyindeki alanın basınca göre değişimi	95
Şekil 6.20.	Dinamik tekerleğin serbest cisim diyagramı.....	96
Şekil 6.21.	Dört tekerlek içi motorlu SMSM'li bir EA'nın dinamik modeli.....	97
Şekil 6.22.	EA'nın ön ve arka tekerlekleri için gerçek sistem ve EDS'nin çıkışından alınan hız farklarının PID kontrolörden geçirilmesi.....	98
Şekil 6.23.	Dört tekerden tahrikli bir EA'nın dinamik modeli.....	99
Şekil 6.24.	Kuvvet ve momentlerin hesabı	100
Şekil 6.25.	Tahrik kuvvetlerinin alt sistemi	100
Şekil 6.26.	Yanal kuvvetlerin hesabı	101
Şekil 6.27.	Momentlerin hesabı	101
Şekil 6.28.	Gerçek sistem hızı ile referans hız farkının PID kontrolörü	101
Şekil 7.1.	Raspberry Pi GPIO pinleri.....	103
Şekil 7.2.	Basys 3 FPGA kartı	104
Şekil 7.3.	Basys 3 güç devresi.....	104
Şekil 7.4.	Simulink'te gerçekleştirilen ön işleme alt bloğu.....	105
Şekil 7.5.	Simulink'te gerçekleştirilen son işleme alt bloğu	105
Şekil 7.6.	RGB'den griye dönüşüm Simulink modeli.....	106
Şekil 7.7.	RGB'den griye dönüşüm işlemi Simulink alt bloğu	106
Şekil 7.8.	RGB'den griye dönüşümden elde edilen sonuç	106
Şekil 7.9.	Görüntünün tersini alma Simulink modeli.....	107
Şekil 7.10.	Tersi alınan görüntünün çıkışı	107
Şekil 7.11.	Tersi alınan görüntü için Simulink'te oluşturulan farklı bir algoritma	107
Şekil 7.12.	Görüntü geliştirmenin Simulink modeli	107
Şekil 7.13.	Gri seviye görüntü ve geliştirilmiş görüntü	108
Şekil 7.14.	Gri seviye görüntünün eşikleme işleminin Simulink modeli	108
Şekil 7.15.	Gri seviye görüntünün eşikleme işlemi sonucu elde edilen yeni görüntü	108
Şekil 7.16.	İkili tabanda görüntünün erozyon algoritmasının Simulink modeli.....	109
Şekil 7.17.	İkili tabanda görüntünün erozyon algoritması sonucu elde edilen görüntü	109
Şekil 7.18.	İkili tabanda görüntünün genişletme algoritmasının Simulink modeli	109
Şekil 7.19.	İkili tabanda görüntünün genişletme algoritması sonucu elde edilen görüntü	110

Şekil 7.20.	Video kenar çıkarma algoritması Simulink modeli	110
Şekil 7.21.	RGB video görüntüsünü gri seviyeye dönüştürme	110
Şekil 7.22.	Gri seviye video görüntüsüne kenar çıkarma metodu uygulama	111
Şekil 7.23.	Canny kenar çıkarma algoritmasının sonucu	111
Şekil 7.24.	Yol çizgilerinin takibi	112
Şekil 7.25.	Çeşitli nesneleri tanıma.....	112
Şekil 7.26.	Yüz ve göz tanıma	113
Şekil 7.27.	Hareket algılama	114
Şekil 7.28.	CAN-Bus hattına bağlı üniteler	114
Şekil 7.29.	Tüm deney düzeneği.....	115
Şekil 7.30.	Raspberry Pi ve kamera modülü	115
Şekil 7.31.	Raspberry Pi ve FPGA'nın deney düzeneğine bağlantı şeması	116
Şekil 7.32.	Tüm sistemin akış diyagramı	117
Şekil 7.33.	Şehirlerarası bir yolda viraj geometrisi	118
Şekil 7.34.	(a) Viraj sağa doğru (b) Düz yol, (c) Viraj sola doğru [97].	118
Şekil 7.35.	Viraj geometrisi	119
Şekil 7.36.	Virajlı yolda araca etki eden kuvvetler	120
Şekil 7.37.	Kaynak noktaları ve şerit çizgilerinin belirlenmesi	121
Şekil 7.38.	Şerit piksellerini filtreleme	122
Şekil 7.39.	Şerit piksellerini maskeleme	122
Şekil 7.40.	Polinom uydurma ve eğrilik ölçümü.....	123
Şekil 8.1.	EA'nın gerçek hız ve referans hız eğrileri	125
Şekil 8.2.	Dinamik modelde kullanılan direksiyon açısının eğrisi	125
Şekil 8.3.	Farklı direksiyon açıları için EA'nın yol eğrisi.....	126
Şekil 8.4.	EDS'den hesaplanan ön iç tekerlek hızı	126
Şekil 8.5.	EDS'den hesaplanan ön dış tekerlek hızı.....	127
Şekil 8.6.	EDS'den hesaplanan arka iç tekerlek hızı.....	127
Şekil 8.7.	EDS'den hesaplanan arka dış tekerlek hızı	127
Şekil 8.8.	a) Kontrolörsüz sistem b) Kontrolörlü sistem için EDS ile gerçek sistemin sağ ve sol tekerlek hız farkları.....	128
Şekil 8.9.	50 km/s araç hızı için ön iç tekerlek hızının Codesys, Matlab ve deneysel sonuçları.....	132
Şekil 8.10.	100 km/s araç hızı için ön iç tekerlek hızının Codesys, Matlab ve deneysel sonuçları.....	132
Şekil 8.11.	150 km/s araç hızı için ön iç tekerlek hızının Codesys, Matlab ve deneysel sonuçları.....	133
Şekil 8.12.	50 km/s araç hızı için ön dış tekerlek hızının Codesys, Matlab ve deneysel sonuçları.....	133

Şekil 8.13.	100 km/s araç hızı için ön dış tekerlek hızının Codesys, Matlab ve deneysel sonuçları.....	133
Şekil 8.14.	150 km/s araç hızı için ön dış tekerlek hızının Codesys, Matlab ve deneysel sonuçları.....	134
Şekil 8.15.	50 km/s araç hızı için arka iç tekerlek hızının Codesys, Matlab ve deneysel sonuçları.....	134
Şekil 8.16.	100 km/s araç hızı için arka iç tekerlek hızının Codesys, Matlab ve deneysel sonuçları.....	134
Şekil 8.17.	150 km/s araç hızı için arka iç tekerlek hızının Codesys, Matlab ve deneysel sonuçları.....	135
Şekil 8.18.	50 km/s araç hızı için arka dış tekerlek hızının Codesys, Matlab ve deneysel sonuçları.....	135
Şekil 8.19.	100 km/s araç hızı için arka dış tekerlek hızının Codesys, Matlab ve deneysel sonuçları.....	135
Şekil 8.20.	150 km/s araç hızı için arka dış tekerlek hızının Codesys, Matlab ve deneysel sonuçları.....	136
Şekil 8.21.	50 km/s araç hızında direksiyonun tersi yönü için ön iç tekerlek hızının Codesys, Matlab ve deneysel sonuçları.....	138
Şekil 8.22.	50 km/s araç hızında direksiyonun tersi yönü için ön dış tekerlek hızının Codesys, Matlab ve deneysel sonuçları.....	138
Şekil 8.23.	50 km/s araç hızında direksiyonun tersi yönü için arka iç tekerlek hızının Codesys, Matlab ve deneysel sonuçları.....	138
Şekil 8.24.	50 km/s araç hızında direksiyonun tersi yönü için arka dış tekerlek hızının Codesys, Matlab ve deneysel sonuçları.....	139
Şekil 8.25.	EA 50 km/s hızda giderken frenleme yapılması durumunda hızın değişimi.....	140
Şekil 8.26.	EA 50 km/s hızda giderken frenleme yapılması durumunda aldığı yol eğrisi	140
Şekil 8.27.	EA 100 km/s hızda giderken frenleme yapılması durumunda hızın değişimi.....	141
Şekil 8.28.	EA 100 km/s hızda giderken frenleme yapılması durumunda aldığı yol eğrisi	141
Şekil 8.29.	100 km/s araç hızında sağ ve düz yol için viraj bilgilerinin gösterimi.....	142
Şekil 8.30.	100 km/s araç hızında sağa ve sola kıvrılan yol için viraj bilgilerinin gösterimi	142
Şekil 8.31.	200 km/s araç hızında sol ve düz yol için viraj bilgilerinin gösterimi	143
Şekil 8.32.	200 km/s araç hızında sağa kıvrılan yol için viraj bilgilerinin gösterimi	143
Şekil 8.33.	Morfolojik işlemler sonucu.....	144
Şekil 8.34.	Yol kenar çizgileri ve ufuk noktasının hesabı.....	144
Şekil 8.35.	Segmentasyon işlemi ve yol viraj yönünün belirlenmesi.....	144

TABLÖLAR LİSTESİ

	Sayfa
Tablo 4.1. Açık ve kapalı haberleşme sistemleri [69].	44
Tablo 4.2. Seri haberleşme protokollerinin özellikleri [71].	44
Tablo 4.3. Veri ağı uzunluğu ile haberleşme hızı arasındaki ilişki	49
Tablo 6.1. EDS tasarımında kullanılan parametreler	75
Tablo 6.2. HY-TTC60 bağlantı pinleri [91].	77
Tablo 6.3. Integer veri tipleri	83
Tablo 6.4. SMSM sürücü parametre ayarları [94].	87
Tablo 6.5. Veri alma ve gönderme adresleri [94].	88
Tablo 6.6. Sürücülere gönderilecek mesajlar [94].	89
Tablo 6.7. EA'nın dinamik modelinde kullanılan sabit parametreler ve değerleri.	95
Tablo 6.8. Yuvarlanma direnci katsayıları	97
Tablo 8.1. 50 km/s araç hızı ve farklı direksiyon açıları için Codesys, Simulink ve deneysel sonuçlar	129
Tablo 8.2. 100 km/s araç hızı ve farklı direksiyon açıları için Codesys, Simulink ve deneysel sonuçlar	130
Tablo 8.3. 150 km/s araç hızı ve farklı direksiyon açıları için Codesys, Simulink ve deneysel sonuçlar	131
Tablo 8.4. 50 km/s araç hızı ve direksiyonun tersi yönü için Codesys, Simulink ve deneysel sonuçlar	137

SİMGELER

A	: Aracın dikey kesit alanı
B	: Sürtünme momenti
B_r	: Artık mıknatıslanma özelliği
C_w	: Direnç katsayısı
d	: Havanın yoğunluğu
d_r	: Arka tekerlekler arasındaki mesafe
dx	: Hareketli koordinat sisteminde x eksenindeki bir döngülük yol farkı
dy	: Hareketli koordinat sisteminde y eksenindeki bir döngülük yol farkı
D	: Tekerleklerle etki eden yanal kayma kuvveti
$D_{öi}$	: Ön iç tekerin yanal kuvveti
$D_{öd}$	: Ön dış tekerin yanal kuvveti
D_{ai}	: Arka iç tekerin yanal kuvveti
D_{ad}	: Arka dış tekerin yanal kuvveti
e_a, e_b ve e_c	: a, b, c fazlarına ilişkin statorda indüklenen zıt emk
f	: Frekans
F_{ad}	: Arka dış tekerin boyuna kuvveti
F_{ai}	: Arka iç tekerin boyuna kuvveti
F_m	: Merkezkaç kuvveti
F_{net}	: Tekerin hareket yönünde yere uygulayabileceği maksimum çekiş kuvveti
$F_{öd}$	: Ön dış tekerin boyuna kuvveti
$F_{öi}$	: Ön iç tekerin boyuna kuvveti
f_{ro}	: Yuvarlanma direnci katsayısı
f_s	: Evirici frekansı
F_s	: Sürtünme direnci
$F_{sürtünme}$	: Sürtünme kuvveti
F_w	: Uygulanan tahrik kuvveti
g	: Yerçekimi ivmesi
$ G $	: Gradyan büyüklüğü
G_x	: Yatay eksen matrisi
G_y	: Düşey eksen matrisi
H	: Çekirdek matrisi
H_c	: Koersif kuvvet
i_d ve i_q	: d-q eksen stator akımları
i_a, i_b ve i_c	: a, b, c fazlarına ilişkin stator akımları
I_z	: z eksenine göre EA'nın kütleli atalet momenti
J	: Atalet momenti
k	: Sürtünme katsayısı
K	: Sağ ve sol king pim arasındaki mesafe

l_f	: Ön tekerleklerin ağırlık merkezine olan mesafeleri
l_r	: Arka tekerleklerin ağırlık merkezine olan mesafeleri
L_{aa}, L_{bb} ve L_{cc}	: a, b, c fazlarına ilişkin faz sargılarının toplam indüktansları
L_{ab}, L_{bc} ve L_{ca}	: a, b, c fazlarına ilişkin fazlar arasındaki karşılıklı indüktansları
L_{ms}	: Faz sargısı öz indüktansı
L_{ss}	: Faz sargısı toplam indüktansı
L_l	: Kaçak indüktans
L_d ve L_q	: d-q eksen indüktansları
L_{yay}	: Yay uzunluğu
m	: Aracın kütlesi
M	: Fazlar arasındaki karşılıklı indüktanslar
M_a	: Tahrik momenti
n	: Devir sayısı
$n_{öd}$	: Ön dış tekerlek hızı
$n_{öi}$	: Ön iç tekerlek hızı
n_{ad}	: Arka dış tekerlek hızı
n_{ai}	: Arka iç tekerlek hızı
p	: Kutup sayısı
Q	: Gradyan yönü
q_1	: Aracın x-eksenindeki ötelenmesi
q_2	: Aracın x-eksenindeki hızı
q_3	: Aracın y-eksenindeki ötelenmesi
q_4	: Aracın y-eksenindeki hızı
q_5	: Aracın z-ekseni etrafındaki dönme açısı
q_6	: Aracın z-eksenindeki açısal hızı
r	: Viraj yarıçapı
r_w	: Tekerlek yarıçapı
R	: Stator direnci
R_a, R_b ve R_c	: a, b, c fazlarına ilişkin faz sargı dirençleri
R_{ad}	: Arka dış tekerleğin viraj yarıçapı
R_{ai}	: Arka iç tekerleğin viraj yarıçapı
R_{cg}	: Elektrikli aracın ağırlık merkezinin yarıçapı
R_{merkez}	: Virajın merkeze olan yarıçapı
$R_{öd}$	: Ön dış tekerleğin viraj yarıçapı
$R_{öi}$	: Ön iç tekerleğin viraj yarıçapı
R_{ro}	: Yuvarlanma direnci
r_w	: Tekerlek dinamik yarıçapı
T_e	: Elektriksel moment
T_L	: Yük momenti
V	: Aracın hızı

V_a, V_b, V_c	: a, b, c fazlarına ilişkin stator gerilimleri
V_d, V_q	: d-q eksen gerilimleri
y	: Viraj ufuk noktasına olan dik uzaklığı
W	: Tekerleğe gelen tepki kuvveti
$w_{öd}$	: Ön dış tekerleğin açısal hızı
$w_{öi}$	: Ön iç tekerleğin açısal hızı
w_{ad}	: Arka dış tekerleğin açısal hızı
w_{ai}	: Arka iç tekerleğin açısal hızı
ω_r	: Rotor hızı
X	: Giriş görüntüsü
Y	: Çıkış görüntüsü
α	: Tekerin kayma açısı
$\alpha_{öd}$	: Ön dış tekerin yana kayma açısı
$\alpha_{öi}$	: Ön iç tekerin yana kayma açısı
α_{ad}	: Arka dış tekerin yana kayma açısı
α_{ai}	: Arka iç tekerin yana kayma açısı
λ_f	: Zıt emk sabiti
λ_d ve λ_q	: d-q eksen stator akı bağlantıları
λ_{af}	: Statoru bağlayan rotor mıknatısları nedeniyle akı bağlantısı
θ	: EA'nın yana kayma açısı
θ_r	: Rotor pozisyonu
δ	: Direksiyon açısı
$\delta_{öd}$	: Ön dış tekerlek açısı
$\delta_{öi}$	: Ön iç tekerlek açısı
σ	: Gauss filtre parametresi
γ	: Viraj açısı
ψ_a, ψ_b, ψ_c	: a, b, c fazlarına ilişkin manyetik akılar

KISALTMALAR

AA	: Alternatif Akım
ACC	: Uyarlanabilir Hız Sabitleyici
ACK	: Alındı Bilgisi Alanı
ADASIS	: Gelişmiş Sürücü Asistanı Sistemleri Ara yüzü Spesifikasyonu
AlNiCo	: Alüminyum-Nikel-Kobalt
BLDC	: Fırçasız Doğru Akım Motoru
BMS	: Batarya Yönetim Sistemi
CAN-Bus	: Kontrol Alan Ağı
CCS	: Hız Sabitleme Sistemi
CFC	: Sürekli Fonksiyon Şeması
CNN	: Evrişimli Sinir Ağı
CO	: Karbon Monoksit
CRC	: Dönüşel Fazlalık Kontrol Alanı
CSMA/CD	: Çarpışma Algılayıcıyla Taşıyıcı Dinleyen Çoklu Erişim
DA	: Doğru Akım
DLC	: Veri Uzunluk Kodu
DSMSM	: Dâhili Sürekli Mıknatıslı Senkron Motor
EA	: Elektrikli araç
EDS	: Elektronik Diferansiyel Sistem
EEPROM	: Elektronik Olarak Silinebilir ve Programlanabilir Salt Okunur Bellek
EH	: Elektronik Ufuk
EKÜ	: Elektronik Kontrol Ünitesi
EMK	: Elektromotor Kuvvet
EOF	: Çerçeve Sonu
EPCU	: Elektrik Gücü Kontrol Birimi
EPROM	: Silinebilir Programlanabilir Salt Okunur Bellek
FBD	: Fonksiyon Blok Diyagram
FPGA	: Alanda Programlanabilir Kapı Dizileri
GPIO	: Genel Amaçlı Giriş/Çıkış
HC	: Hidrokarbon
HD	: Hibridizasyon Derecesi
HEA	: Hibrit Elektrikli Araç
HSV	: Ton, Doygunluk ve Parlaklık Değeri
IDE	: Tanımlayıcı Bit
IL	: Açıklama Listesi
INT	: Çerçeveler Arası Boşluk
LD	: Merdiven Diyagramı
LDC	: Alçak Gerilim DA-DA Dönüştürücü
LIN	: Yerel Bağlantı Ağı
MOST	: Medya Odaklı Sistemleri Taşıma Ağı
MPC	: Model Öngörülü Kontrol
NdBF _e	: Neodimyum-Bor-Demir
NMOG	: Metan Olmayan Organik Gaz
NMS	: Maksimum Olmayan Bastırma İşlemi
NO	: Nitrojen Oksit

OBC	: Yerleşik Şarj Cihazı
PI	: Oransal İntegral
PID	: Oransal İntegral Türev
PLC	: Programlanabilir Mantıksal Denetleyici
POU	: Program Organizasyon Birimi
RAM	: Rastgele Erişimli Bellek
RGB	: Kırmızı, Yeşil, Mavi
ROM	: Sadece Okunabilir Bellek
RTDAS	: Gerçek Zamanlı Sürücü Destek Sistemi
RTR	: Uzak Gönderim İstek Biti
RX	: Alıcı
SDS	: Sürücü Destek Sistemleri
SFC	: Sıralı Fonksiyon Şeması
SmCo	: Samaryum-Kobalt
SMSM	: Sürekli Mıknatıslı Senkron Motor
SOF	: Çerçeve Başlangıcı
SRR	: Yedek Uzak İstek Biti
ST	: Yapılandırılmış Metin
TX	: Verici
UART	: Evrensel Asenkron Alıcı Verici
VCU	: Araç Kontrol Ünitesi
YMSMSM	: Yüzeye Monte Sürekli Mıknatıslı Senkron Motor
V2I	: Araçtan Altyapıya
V2V	: Araçtan Araca

1. GİRİŞ

1.1. Genel Bilgi

Son yıllarda elektrikli otomobile olan ilgi hızla artmaktadır. Bunun en önemli nedenleri, dünya genelinde enerji kaynaklarının azalmaya başlamasıyla verimliliğin ön plana çıkması, fosil yakıtların tükenmeye yüz tutması, batarya teknolojisindeki gelişmeler, elektrik motor sürücülerinde kullanılan güç elektroniği elemanlarındaki ilerlemeler ve gelişen işlemci teknolojisidir. Elektrikli Araçlar (EA), zararlı gaz emisyonlarını ortadan kaldırarak çevre kirliliğini önlemektedir. Ayrıca sessiz çalışmaları, yüksek verime sahip olmaları, bakım masraflarının az ve elektrik motorlarının daha ucuz olması gibi önemli avantajları vardır. Bu nedenle EA'lar öncelikli konular arasına girmiştir.

Son zamanlarda elektrik motorları hemen hemen her endüstride çeşitli yükleri sürmede kullanılmaktadır. Dünya genelinde farklı oranlarda güç gerektiren çeşitli uygulamalarda kullanılan çok sayıda elektrik motoru bulunur. Büyük güçlü motorlar daha ağır uygulamalarda kullanılırken, küçük güçlü motorlar ev aletlerinde kullanılırlar. Elektrik motoru olarak Sürekli Mıknatıslı Senkron Motor (SMSM) yaygın şekilde tercih edilirken, uzay, otomasyon, pompa, fan, medikal, havacılık, asansör gibi pek çok endüstride de bu motorlar kullanılmaktadır. SMSM'lerin veriminin yüksek ve ömürlerinin uzun olması, düşük atalet, yüksek çıkış gücü, yüksek dinamik cevap ve daha iyi moment-hız karakteristiği sağlaması gibi pek çok önemli avantajları vardır. Bu yüzden bu çalışmada da motor olarak SMSM'ler tercih edilmiştir.

Araçlarda tekerleklerin hızını ayarlayarak aracın dengeli ve güvenli sürüşünü sağlamak için diferansiyel sistemler kullanılmaktadır. Bu sistemler araç viraja girdiğinde dönüş yönüne göre, dıştaki tekerlek hızının içteki tekerlek hızından daha yüksek olmasını sağlayarak aracın savrulmasını önler. Genellikle araçlarda mekanik diferansiyeller kullanılmaktadır. Bunların getirdiği arıza, bakım masrafları gibi önemli dezavantajlar vardır. Bunları ortadan kaldırmak için Elektronik Diferansiyel Sistemler (EDS) tercih edilir. EDS'de motorlar tekerlek içlerine yerleştirilerek aktarım organlarının kullanımı ortadan kaldırılır. Böylece aracın ağırlığı azaltılarak bakım maliyeti de ortadan kalkar ve motorların kontrolü elektronik bir sistem üzerinden gerçekleştirilir.

Ulaşımın en önemli unsurların başında güvenlik gelir, daha sonra da konfordur. Sürücülerin ve yolcuların rahat ulaşımını sağlamak için gerçekleştirilen tüm donanımlar konforun içerisinde yer almaktadır. Bunun yanı sıra araç için önemli olan yakıt tüketimi, haberleşme, navigasyon ve sensörler gibi tüm unsurları da kapsar. Güvenlik ise, ulaşımın en önemli unsurudur. Bu nedenle, aracın kaza risklerini en aza indirgeyerek can kaybına neden olacak durumları ortadan kaldırmak için alınacak güvenlik tedbirleri oldukça önemlidir. Aktif ve pasif güvenlik

olmak üzere iki çeşit güvenlik vardır. İkisinde de amaç, kazaları önleyerek can ve mal güvenliğini sağlamaktır. Hava yastığı gibi kaza anında devreye giren ve akıllı olmayan güvenlik sistemleri pasif güvenliğı oluşturur. Aktif güvenlik ise, araç yolda giderken çeşitli analizler yaparak sürücüye destek olan, güvenli sürüşün sağlanması için gerekli olan akıllı sistemlerin hepsini içermektedir. Aktif güvenlik kapsamında Sürücü Destek Sistemleri (SDS) geliştirilmiştir. Buna göre sürücünden, araçtan ve çevreden alınan bilgiler değerlendirilir ve sürücünün güvenli şekilde gitmesini sağlayarak kazalara karşı önlemler alınır. Bu nedenle son yıllarda bu alanda yapılan çalışmalar oldukça önem kazanmıştır.

1.2. Literatür Araştırması

Bu bölümde, EA'larda kullanılan EDS'ler, sürüşe yardımcı sistemler ve güvenli sürüş uygulamaları ile ilgili yapılan çalışmalar anlatılmıştır.

Literatürde EDS üzerine yapılan çalışmalar detaylı olarak araştırılmıştır. Bu çalışmalardan ilki 2009 yılında, Y.E. Zhao, J.W. Zhang ve X.Q. Guan tarafından gerçekleştirilmiştir. Bu çalışmada iki tekerlek içi motorlu bir EA için bulanık mantık kontrol yöntemine dayanan bir EDS yer almaktadır [1]. Bu yöntem, her bir tekerleğin kayma oranını hesaplamak için kullanılmıştır. Önerilen sistemi doğrulamak için bir Matlab/Simulink simülasyonu gerçekleştirilmiştir. Bu yöntemle EA'nın kavisli yollarda stabilitesi sağlanarak kayma hızının optimize edildiğı sonuçlardan görülmüştür. [2]'deki yazarlar, dört anahtarlı üç fazlı evirici kullanarak doğrudan tekerlekten tahrikli bir EA için Kayma Mod Kontrolörlü bir EDS sunmuşlardır. Tasarlanan sistem, iki arka tekerlek içi motorlu bir araç üzerinde doğrudan kontrol edilerek başarılı bir şekilde test edilmiştir. 2010 yılında Y. Zhou ve arkadaşlarının gerçekleştirdiğı çalışmada [3], sürüş koşullarına dayalı olarak dört tekerlek içi motorlu bir EA için EDS'nin kontrol stratejisi incelenmiştir ve Matlab/Simulink simülasyonu yapılmıştır. EA'lar için uygun bir EDS tasarlanmıştır. 2011 yılında yapılan bir başka çalışmada L. Zhai ve S. Dong, bağımsız olarak dört tekerlek içi motor tarafından sürülen bir araç için Yapay Sinir Ağları Oransal-İntegral-Türevsel Kontrol Yöntemi (Proportional-Integral-Derivative, PID) kullanarak EDS tasarımı önermişlerdir [4]. Direksiyon açısına göre dört tekerlek hızını hesaplamış ve dört tekerlek içi motorda moment dağılımını sağlamak için dört PID kontrolör kullanmışlardır. Buna göre EDS'nin farklı direksiyon açıları ve hızları için başarılı bir şekilde tasarlandığı sonuçlardan görülmüştür. 2013 yılında Draou tarafından iki tekerlek içi SMSM tarafından sürülen bir araç için EDS tasarlanmıştır [5]. Her motor için doğrudan moment kontrolü kullanılmıştır. Yol koşullarına göre farklı simülasyonlar yapılmış ve simülasyonlar ile eğimli yolda araç stabilitesi sağlanmıştır. İki tekerlek içi Fırçasız DA (BLDC) motor tarafından tahrik edilen bir EDS [6]'da, bir Bulanık Mantık Hız Denetleyicisi kullanılarak tasarlanmıştır. Bu sistem EA'nın arka tekerlekleri için yapılmıştır. Aynı zamanda farklı yol koşulları için EDS'nin bir Matlab/Simulink analizi gerçekleştirilmiştir. Böylece bu

yöntemle optimal simülasyon sonuçları elde edilmiştir. [7]'de, iki arka tekerlek içi SMSM tarafından tahrik edilen bir EA için hataya dayanıklı bir EDS çalışılmıştır. Ackermann-Jeantand geometrisi kullanılarak iki arka tekerlek hızı hesaplanmıştır. Tasarlanan sistemin performansı simülasyonlar ile doğrulanmıştır. 2014 yılında A. Bahri ve arkadaşları tarafından [8], iki BLDC motorla tahrik edilen arka tekerlekler için yeni bir Dalgacık Kontrolörü (Wavelet Controller) kullanılarak bir EDS modellenmiş ve simüle edilmiştir. Simülasyonlarda farklı yol koşulları göz önüne alınmıştır. Sonuçlar, PID gibi diğer kontrolörlerle karşılaştırılmıştır. 2015 yılında S. Sharma ve arkadaşları, Doğru Akım (DA) motorlu bir EA'nın iki arka tekerleği için bir EDS önermişlerdir [9]. Bu çalışmada, EA hızına ve dönüş açısına göre arka tekerlek hızları mikrodenetleyici tabanlı bir kontrol yaklaşımı kullanılarak tahmin edilmiştir. Sonuçlar deneysel çalışmalarla doğrulanmıştır. [10]'da, iki BLDC motorlu bir EA için gömülü bir EDS araştırılmıştır. Verimlilik testi, farklı yük koşulları için BLDC motor sürüş sisteminde de gerçekleştirilmiştir. Böylece bu çalışmayla geri beslemeli gömülü EDS'nin gerçek hızı referans hıza eşit olacak şekilde ayarlanarak verimli bir sistem tasarlanmıştır. [11]'de iki SMSM tarafından tahrik edilen EA'nın arka tekerlekleri için bir elektronik diferansiyel modeli sunulmaktadır. Simülasyonlarda farklı yol koşulları incelenmekte ve bu yöntemle eğimli yolda sürüş stabilitesi sağlanmaktadır. [12]'de, Matlab/Simulink kullanılarak tekerlekten tahrikli bir araç için Bulanık Mantık Kayma Mod Kontrolörlü bir EDS açıklanmıştır. Bu denetleyici yöntemi, PID ve Kayma Mod Kontrolör yöntemleriyle karşılaştırılmıştır. Bulanık Mantık Kontrolörlü EDS'nin diğer kontrol yöntemlerine göre daha iyi performans gösterdiği sonuçlardan görülmüştür. 2017 yılında D. Yin ve arkadaşları [13], dört tekerlekle sürülen bir EA için EDS tasarımı üzerine çalışmalar yürütmüşlerdir. EDS'nin analizinde direksiyon açısına bağlı olarak simülasyonlar gerçekleştirilmiştir. Böylece EDS'nin kontrol performansı üzerinde geri besleme kazancının çok önemli olduğu sonuçlardan gözlenmiştir. İki tekerlek içi motorlu bir EA için EDS'nin modellenmesi ve simülasyonu, [14]'teki Bulanık Mantık Kontrol yöntemine dayalı olarak gerçekleştirilmiştir. Bu yöntem, her bir tekerleğin kayma oranını tahmin etmek için kullanılmıştır. EDS modeli, Matlab/Simulink sonuçları ile doğrulanmış ve farklı yol koşulları için EA'nın stabilitesi sağlanarak kayma hızı optimize edilmiştir. [15]'te yazarlar, EA'nın arka tekerlekleri için bir EDS modeli önermiştir. Arka tekerleklerde iki SMSM kullanılmıştır. Düz ve kavisli yollar gibi çeşitli yol koşullarına göre farklı simülasyonlar gerçekleştirilmiştir. Buna göre virajlı yollarda EA'nın stabilitesinin sağlandığı görülmüştür. [16]'da, Ackermann geometrisi kullanılarak bir tekerlek içi motor tarafından tahrik edilen EA için EDS çalışılmıştır. Farklı direksiyon açılarında ve gerçek hızlarda Matlab/Simulink simülasyonu yapılmıştır. Sonuçlar, önerilen EDS modelinin tekerlek içi motorlu bir EA için uygun olduğunu göstermiştir. [17]'de EA'nın arka tekerlekleri için bir EDS tasarlanmıştır. İki katlı kontrol döngüleri dış ve iç tekerlekler için çalışmaya dâhil edilmiştir. EA'nın yalpalama hareketi ve sağlamlığı bu döngüler tarafından

incelenmiştir. İç kontrol döngüsü Matlab/Simulink kullanılarak simüle edilmiştir ve dış kontrol döngüsü ise bir EPF10KE Alanda Programlanabilir Kapı Dizisi (FPGA) tarafından gerçekleştirilmiştir. Tasarlanan EDS, ticari bir EA üzerinde başarıyla test edilmiştir. [18]'de yüksek performanslı EA'lar için EDS tasarımı önerilmiştir. Momenti ayarlamak için bir kapalı döngü kontrol sistemi kullanılmış ve simülasyonlar EA'nın arka tekerlekleri için yapılmıştır. Sonuçlar, tasarlanan EDS'nin zorlu sürüş koşullarında yüksek performanslı EA'lar için uygun olduğunu göstermiştir. [19]'da, doğrudan moment kontrolü ve adaptif akı ve hız gözlemci tabanlı bir algoritma kullanılarak iki asenkron motorlu EA için bir EDS önerilmiştir. Algoritma sonuçları, bir dijital sinyal işlemcisi TMS320LF2407 tarafından alınan deneysel sonuçlarla doğrulanmış ve sürüş stabilitesi sağlanmıştır. [20]'de, arka tekerlek içi motorlar ile tahrik edilen bir EA için EDS deneysel olarak tasarlanmıştır. Bu çalışmada, gömülü bir sistem kullanılarak motorların bulanık mantık kontrolü gerçekleştirilmiştir ve sonuçlar deneysel çalışmalarla doğrulanmıştır. 2018 yılında Y.C. Chang ve arkadaşları [21], kentsel EA'lar için bir EDS kontrol stratejisi önermiştir. Bu çalışmayla motordan tekerleğe güç aktarılırken oluşan kayıp, doğru bir kontrol yöntemi ile azaltılmaktadır. Bu nedenle, bu çalışma yeni bir kontrol yöntemi sunmaktadır ve her bir tekerlek bağımsız olarak kontrol edilebilmektedir. [22]'de iki ön tekerlek içi SMSM'li bir EA için dalgacık dönüşümüne dayalı bir EDS önerilmiştir. EDS'nin testi ve simülasyonu, Carsim/Simulink programı kullanılarak çeşitli sürüş koşulları için gerçekleştirilmiştir. Sonuçlar, EA'nın kararlılığının ve sağlamlığının bu çalışma ile sağlandığını göstermektedir. Bir diğer çalışmada [23], bir EDS'nin modellenmesi ve simülasyonu, iki BLDC motor tarafından tahrik edilen EA'nın arka tekerlekleri için yeni bir dalgacık dönüşümü kontrolörü kullanılarak gerçekleştirilmiştir. Farklı yol durumları için simülasyonlar yapılmış ve bu simülasyon sonuçları ile EA performansının kararlılığı sağlanmıştır. [24]'te, Ackermann direksiyon diferansiyeline dayalı bir EDS, tekerlek içi motorlu bir EA için Matlab/Simulink ile modellenmiştir. Simülasyonlar, çeşitli hızlar ve direksiyon açıları için gerçekleştirilmiştir. Sonuçlara göre, önerilen yöntem, bir tekerlek içi motor tarafından sürülen EA'lar için kullanışlı ve verimlidir. [25]'te yazarlar, hafif EA'lar için bir EDS'nin tasarımını ve uygulamasını açıklamaktadır. EA'nın bağımsız kontrolü için tekerleklere iki BLDC motor yerleştirilmiştir. Dönüş açısını ve tekerlek hızlarını ölçen herhangi bir sensör bulunmamaktadır. Böylece tasarlanan EDS ile EA virajlı yolda giderken motorların farklı devirlerde dönüşü sağlanmıştır. [26]'da, tekerlek içi motorlu bir EA için bulanık kontrol yöntemi kullanılarak moment dengeleyen bir adaptif elektronik diferansiyel kontrol yöntemi sunulmaktadır. Adaptif moment dengeleme için araç direksiyonunun kinematiği ve dinamikleri, her bir tekerleğin hızı ve kuvveti incelenmiştir. Böylece bu çalışma ile optimal bir EDS tasarımı gerçekleştirilmiştir. [27]'de H.T. Al-Fiky ve arkadaşları tarafından, motorların enerji tüketimini azaltacak optimal bir EDS modeli sunulmuştur. Bunun için tekerlek kaymasını en aza indirecek ve motorun moment saturasyonunu önleyecek şekilde bir kontrolör

geliştirilmiştir. Bu çalışmada, üç fazlı BLDC motor modellenmiş ve tasarlanmıştır. Böylece bir elektronik diferansiyelin performansı iyileştirilerek enerji tüketimi azaltılmıştır.

Literatürdeki çalışmalara bakıldığında, genellikle EA'nın ön veya arka tekerlekleri için EDS modeli üzerine çalışıldığı görülmektedir. Aksine bu tez çalışmasında, EA'nın dört tekerlek içi motoru için EDS tasarımı gerçekleştirilmiştir. Bunun için dört adet SMSM, EA'nın tekerleklerine yerleştirilir. Böylece EA kavisli yolda giderken, farklı direksiyon açıları ve araç hızları için motorların farklı hızlarda dönmesi sağlanmıştır. Dolayısıyla moment, dört SMSM için eşit olarak dağıtılmış ve her bir motor ayrı ayrı kontrol edilmiştir. Ayrıca bu durum, mekanik aktarma organlarının kullanımını ortadan kaldırarak EA'nın kütlesini azaltır. Böylece, EDS'de gerçek zamanlı bir güç yönetimi sağlanmış ve dört tekerlek için farklı moment referansları üretilerek her tekerleğin hızı ayarlanmıştır.

Son yıllarda SDS ile ilgili pek çok çalışma gerçekleştirilmektedir. Bunlardan ilkinde [28], gerçek zamanlı uygulamalar için şerit tespiti ve takibi yapılmıştır. Etkili bir şerit algılama algoritması, sürüş güvenliğini destekleyen önemli bilgiler sağlar. Bunun için gelişmiş bir SDS'nin temel bir bileşenidir. Şerit tespit ve takip algoritmasının karşılaştığı zorluklar, şerit işaretlerinin net olmaması, kötü hava nedeniyle zayıf görüş, aydınlatma ve ışık yansıması, gölgeler ve yoğun yol tabanlı talimatları içerir. Bu çalışmada, yüksek gürültü seviyesini ve hesaplama süresini azaltmak için etkin bir ilgi alanına sahip, sağlam ve gerçek zamanlı görüş tabanlı şerit algılama algoritması önerilmiştir. Önerilen algoritma, şerit işaretlerinin özelliklerini doğrulamak için bir gradyan işaretini, bir renk işaretini ve tarama çizgisi testleri ile bir çizgi kümelemeyi birlikte işler. Alınan görüntüdeki herhangi bir yanlış şerit işaretini kaldırır ve toplanmış istatistiksel verileri kullanarak gerçek şerit işaretlerini takip eder. Deneysel sonuçlar, önerilen algoritmanın doğru sonuçlar verdiğini ve düşük hesaplama gücüne sahip gömülü sistemlerde gerçek zamanlı çalışma gereksinimini karşıladığını göstermiştir. [29]'da, EA'lar ve Hibrit Elektrikli Araçların (HEA) enerji ekonomisinin iyileştirilmesi için Gerçek Zamanlı Sürücü Danışma Sistemi (Real-Time Driver Advisory System, RTDAS) sunulmuştur. Sistem, Gelişmiş Sürücü Asistanı Sistemleri Ara yüzü Spesifikasyonu (ADASIS v2) protokolünün ikinci versiyonuna uygun olarak sağlanan bir yol haritası verilerine dayanır. Yol haritası verileri daha sonra Elektronik Ufuk'a (Electronic Horizon, EH) dönüştürülmüştür. EH, enerji ekonomisi için çeşitli algoritmalar geliştirerek aracın önündeki yol özelliklerinin izlenmesini sağlar. Sistem, yol eğimi, eğrilik, yükselme, hız sınırları ve trafik işaretleri gibi yolun önündeki statik özelliklere odaklanmıştır. Bu verilere ve araç parametrelerine dayalı olarak sistem, sürücüye gaz pedalını bırakması için bir mesaj gönderilip gönderilmeyeceğini ve ne zaman gönderileceğini belirlemek ve daha yüksek enerji ekonomisi, konfor ve sürüş güvenliği sağlamak için algoritmalar kullanır. Gaz pedalının serbest bırakılması, aracın mekanik frenler kullanmadan kinetik enerji geri kazanımının optimizasyonunu sağlayan kontrollü rejeneratif frenleme ile hızı ayarlamak için bir algoritma sunmuştur. [30]'da, hareket

halindeki bir aracın ön paneline monte edilmiş bir kamera kullanılarak şerit algılama, izleme, yol yüzeyi işaretleme ve tanıma için yeni bir gerçek zamanlı denetimsiz öğrenme çerçevesi önerilmiştir. Akıllı araçlarda otomatik şerit ve yol yüzeyi işaretleme, algılama ve kameralardan tanıma önemli bir rol oynamaktadır. Gelişmiş SDS'lerde sürüş güvenliğini artırmaya ve otonom sürüş araçlarında küresel yerelleştirmeyi ve yol planlamayı geliştirmeye yardımcı olur. Harita oluşturma ve güncelleme, şeritlerin ve yol yüzeyi işaretlerinin otomatik olarak algılanmasından da yararlanabilir. Öğrenme çerçevesinde, şeritleri ve yol yüzeyi işaretlerini eşzamanlı olarak tespit etmek için eğri uydurma ile birleştirilmiş bir uzay-zamansal artan kümeleme algoritması kullanılmıştır. Çerçeve, sürekli, sürekli olmayan, düz veya kavisli olması, şerit sayısı ve türünden bağımsızdır. Ayrıca, yol yüzeyi işaretlerinin tüm türleri, boyutları ve yönelimleri için çalışır. Ancak eğitim verilerinde bulunan yol yüzeyi işaretlerinin türü ile sınırlıdır.

Geleneksel derin öğrenme tabanlı araç algılama yöntemleri genellikle birden çok ölçek ve boyuta sahip bir filtre piramidi kullanılarak tasarlandığından kullanılan çok sayıda ölçek nedeniyle ve sınıflandırıcı tüm ölçeklerde çalıştığı için işlem süresi yavaştır. Son zamanlarda, giriş görüntüsünün boyutundan bağımsız olarak ağı yalnızca bir kez kullanan araçları tespit etmek için bir derin öğrenme tabanlı bölge öneri ağı sunulmuştur [31]. Nesne algılamada, derin öğrenme tabanlı bölge önerisi ağları, doğruluk açısından yüksek performansa ulaşmıştır. Bu sistemler, normal sürüş koşullarında çok yüksek bir doğruluk sağlar; ancak kar, yağmur veya sis gibi zorlu sürüş koşullarında performansları düşer. Ek olarak, mevcut son teknoloji sistem tabanlı bölge öneri ağları SDS'lerin gerçek zamanlı gereksinimlerini hala karşılayamamaktadır. Daha yakın zamanlarda, yerel kalıpların tanımlanmasının geleneksel derin öğrenme sistemlerinin performansını iyileştirdiği gösterilmiştir. Bu nedenle 2019 yılında V.D. Nguyen ve arkadaşları tarafından [31], doğruluklarını arttırmak için bölge öneri ağlarındaki yerel kalıplar araştırılmıştır. Mevcut bölge öneri ağlarının işlem süresini iyileştirmek için derinlik bilgileri de incelenmiştir. DeneySEL sonuçlar, önerilen sistemin hem doğruluk hem de işlem süresi açısından son teknoloji nesne bölgesi algılama sistemlerinden daha iyi performans elde ettiğini göstermiştir. [32]'de, trafik ışıklarında sürüş güvenliğini arttırmak için, akıllı bir güvenli sürüş asistanı sunulmuştur. Sürüş asistanı, trafik ışığı aşamalarını gözlemlemenin zorluklarını azaltabilen tek bir yerleşik kameralı bir görüntü tanıma sisteminden oluşur. Tanıma sistemi, bir Evrişimli Sinir Ağı kullanarak trafik ışığı süresinin geri sayım bilgilerini elde eder ve trafik ışığı bilgilerinin sonuçlarını kullanarak geri sayım süresini tahmin eder. Ayrıca kamera ve trafik ışığı bilgilerini kullanarak trafik ışığı ile araç arasındaki mesafeyi hesaplamak için bir model geliştirilmiştir. Trafik ışığı evresine ve elde edilen mesafeye bağlı olarak, sürüş asistanı, sürücünün güvenliğini arttırmak için bir hız kontrol stratejisi sağlamaktadır. Önerilen yöntemlerin etkinliğini doğrulamak için kapsamlı deneyler yapılmıştır. [33]'te yazarlar, kazaların büyük çoğunluğunun dikkatsizlik, sinyal vermeme, uyuşukluk, cep telefonu ile konuşma vb. hatalardan kaynaklandığını

gözlemlemiş ve bu tür hataların kontrol altına alınmasına ve kazaların önlenmesine yardımcı olacak bir sistem oluşturmuşlardır. Önerilen sistem, sürücünün aracı sürmesi gerekip gerekmediğine ilişkin kararlara dayalı olarak sürücünün durumunu tespit etmek için makine öğrenme algoritmalarını kullanmıştır. İlk olarak, bir kamera ve görüntü işleme kullanılarak sürücünün yüz özelliklerinin tespiti yapılmıştır. Öznitelikleri yakalandıktan sonra, sürücü durumunu elde etmek için eğitilmiş bir yapay sinir modeli oluşturulmuştur. Ayrıca, sürücünün cep telefonu kullanıp kullanmadığını tespit etmek için de Evrişimli Sinir Ağı (CNN) kullanılmıştır.

Son zamanlarda, gerçek zamanlı araç tespiti, SDS çarpışma azaltma, farlardan kaçınma ve farları kısma gibi önemli görevler büyük bir zorluk haline gelmiştir. Bununla birlikte, bu alandaki çoğu araştırmacı araştırmalarını gündüz iyi ışık koşullarında gerçekleştirmektedir. Bunun aksine [34]'te, görüntü işlemeye dayalı, gerçek zamanlı olarak gece vakti etkin bir araç tespit yöntemi çalışılmıştır. Önerilen yöntemde, eşikleme yöntemi ve çapraz korelasyon kullanılarak araç farları ve arka lambalar tespit edilmiştir. Deneyler, önerilen yöntemin doğruluk ve sağlamlık açısından iyi bir performansa sahip olduğunu ve gereksinimleri gerçek zamanlı olarak karşılayabildiğini göstermiştir. Deneyler, TERASIC tarafından sağlanan bir VEEK-MT2S içine yerleştirilmiş Donanım İşlemci Sistemi üzerinde C++ ve OpenCV kullanılarak yapılmıştır.

Son yıllarda çalışılan en önemli konulardan biri otonom araçlardır. Otonom araçların en büyük faydası güvenlidir. Araba kazalarının çoğuna insan hataları neden olur. Bu araçlar, toplu taşımanın daha fazla kullanılması ve böylece trafikte araç sayısının azalmasını sağlamaktadır. Aynı zamanda karbondioksit (CO₂) emisyonlarını azaltarak çevreye katkı sağlarlar. Bunun yanı sıra bazı olumsuz tarafları da vardır. Otonom araçlar, kötü hava koşullarında çok iyi çalışamazlar ve herhangi bir olay olması durumunda yasal sürecin tanımlanması gerekir. 2020 yılında S. Coicheck ve arkadaşlarının yaptığı bir çalışmada [35], bu teknolojinin mevcut durumu ve bir aracın insan müdahalesi olmadan kullanılmasına izin veren akıllı sistemlerin uygulamasının zorlukları anlatılmıştır.

[36] ise optimal kontrole dayalı SDS ile ilgili bir çalışma sunar. Bu tür sistemlerin amacı, sürücüyü olası tehlikeli durumlara karşı uyarmak için sürücü-aracı davranışını ve çevreyi izlemek ve gerçek araç manevrasını optimum kontrol ile önerilen bir referans manevrayla sürekli olarak karşılaştırmaktır. Optimum kontrol referans manevrası, uygun bir risk fonksiyonunun en aza indirildiği güvenlik kriterlerine dayanmaktadır. Bu çalışmada güvenli hız durumu ön planda ele alınmaktadır. Bunun için farklı risk fonksiyonu formülasyonlarının sistem performansı üzerindeki etkisi araştırılmıştır. S şeklinde bir virajda sabit hızda giden bir araç için simülasyonlar yapılmış ve güvenli, optimum referans manevraları baz alınarak sürüş şekilleri analiz edilmiştir.

Gelişmiş SDS'ler, son zamanlarda akıllı araçlarla ilgili en aktif konulardan biri haline gelmiştir. Bu tür bir yardım, sürücülere daha fazla kontrol ve gelişmiş bir sürüş deneyimi sağlayarak aracın çalışmasını kolaylaştırır ve hatta otonom araç çalışması için otomatik yardım

şeklinde gelir. [37]'de, genel sistem ticari bir araca kurulu olarak düşük maliyetli bir gömülü dijital sinyal işlemcisi kullanan bir sürücü yardım sistemi tasarlanmıştır. Gerçek zamanlı görüş sistemi, araçtan araca iletişim sistemi ve araç içi sensörler gibi çoklu sensörler tarafından sağlanan sürüş bilgilerine dayanır. Önerilen sistem, sollama manevralarını gerçekleştirirken karar vermeyi ve sürüş görevlerini yerine getirmeyi kolaylaştırabilir. Bu çalışmada, sollanan aracın ve komşu şeritteki diğer araçların çarpışmalarını önlemek için hesaba katıldığı bir çarpışma uyarı algoritmasına dayanan bir veri birleştirme programı geliştirilmiştir. Sistemde, insanlar tarafından gerçekleştirilen sürüş şekillerini ele almak için direksiyon ve hız otomasyonunda bulanık kontrol kullanılmıştır. Önerilen sistemin uygulanabilirliği gerçek bir yol ortamında incelenmiştir ve deneysel sonuçlar çeşitli sürüş manevraları yapılırken ilgili koordinasyon stratejilerinin kullanılmasının uygun olduğunu göstermiştir.

[38]'de, önden çekişli, tekerlek içi motorlu EA'lara yeni bir dinamik hareket stabilizasyonu yaklaşımı önerilmiştir. Yaklaşım, çekiş kontrol sistemi, EDS ve elektronik stabilite kontrolü gibi işlevler sunmuştur ve bir prototip EA "Corsa-EV" üzerinde test edilmiştir. Deneysel sonuçlar, sunulan yöntemin etkinliğini ve uygulanabilirliğini göstermiştir.

Bir başka çalışmada [39], bir EA'nın elektrikli tahrik arızası sırasında bir araç stabilite kontrolü önerilmiştir. Bu EA üç çekiş motoru tarafından tahrik edilir. Bir çekiş motoru iki ön tekerlektir, iki tekerlek motoru ise arka tekerleklerden sorumludur. Çekiş motorlarından herhangi biri arızalandığında önerilen kontrol, bir dijital sinyal işlemcisi kullanılarak bir doğrudan sapma momenti kontrolörüne ve bir kayma oranı kontrolörüne göre yeni moment dağılımını hesaplamaktadır. Daha sonra, arka tekerlek motorlarından biri olarak hareket eden bir motora sahip bir dinamometre üzerinde gerçek zamanlı bir çevrim içi donanım testi yapılmıştır. EA ve diğer iki çekiş motoru ticari olarak mevcut simülasyonlarla modellenmiştir. Önerilen kontrol stratejisinin elektrikli tahrik arızası tespit edildikten sonra EA'yı güvenli ve kararlı tutabildiği deneysel sonuçlarla gösterilmiştir.

Güvenli sürüş, trafik sıkışıklığı önleme ve sürüş yükünün azaltılmasına çözüm olarak kameralı ve sensörlü sürüş asistan sistemleri ve mevcut sistemlerin yardımcı sistemi olarak kablosuz teknoloji kullanan araçtan araca iletişim sistemleri önerilmiştir. Diğer yandan toplumun yaşlanma hızının artmasıyla birlikte yaşlıların sebep olduğu trafik kazaları çözülmesi gereken ciddi bir sorun haline gelmiştir. Bu yüzden [40]'ta, akıllı otomatik sürüş sistemi, kıdemli sürücüler için otomatik sürüş sistemine dayalı bir akıllı kontrol sistemi gerçekleştirmek için geliştirilmiştir. Bunun için küçük otomatik EA için gömülü bir iletişim sistemi geliştirilmiş ve geliştirilen sistemde iletişim çakışmasını önlemek için bir algoritma uygulanmıştır. Araçtan araca (V2V) iletişime dayalı olarak, araç kesintilerini ve ayrımlarını içeren bir kontrol sistemi başarıyla tasarlanmıştır. Küçük bir EA için birkaç kontrol modelinden alınan deneysel sonuçlar, geliştirilen sistemin etkinliğini göstermiştir.

[41]'de, Çoklu Trafik Işığı ile yol kesimine yaklaşan bir EA'nın kontrolü için yenilikçi bir yaklaşım sunulmuştur. V2V ve Araçtan Altyapıya (V2I) iletişim kullanılarak manevranın tamamlanması için enerji tüketimi azaltılabilir. Bunun için Model Öngörülü Kontrol (MPC) yöntemi kullanılmıştır. Sistemin performansı, trafik koşulları dahil edilerek, güç aktarma organları, sürücü ve yolu içeren karmaşık bir simülasyon ile değerlendirilmiştir. Sonuçlar, "normal" insan sürücü davranışına kıyasla idealleştirilmiş bir durum için % 29 ve gerçek bir yol simülasyonu senaryosu için toplam enerji tüketiminde % 17'lik bir azalma olduğunu göstermiştir. Bir başka çalışmada [42], otomotiv uygulamalarında ve özellikle EA'larda kullanılan SMSM motor sürücüleri için elektronik kontrol biriminin tasarımı geliştirilmiştir. Elektrik motor kontrol biriminin donanım tasarımı, otomotiv donanım tasarım standartlarına uygun olarak yapılmıştır. Yazılımı ise, AUTOSAR programı ile geliştirilerek donanımın gerçek zamanlı olarak güvenli şekilde çalışması sağlanmıştır. Otomotiv standartlarının güvenlik şartlarını yerine getirmek için 32 bitlik ana mikrodeneleyici RTOS ile birlikte ek bir 8 bitlik mikrodeneleyici kullanılmıştır. Geliştirilen yazılım ile algoritma ve ara yüz geliştirme aşamaları birbirinden ayrılarak donanım ve yazılımın daha az hatayla tasarlanması sağlanmıştır.

[43]'te araştırmacılar, SMSM ile çalışan bir EA'nın otomatik hız izleme kontrolüyle ilgili problemlerini incelemişlerdir. Sensör arızaları durumunda iyi bir kontrol performansını sürdürmek için Yüksek Dereceli Kayan Mod Kontrolörüne dayalı bir yeniden yapılandırma şeması önerilmiştir. Buna göre kontrollü motor çıkış momenti, aracın kesintisiz güvenli çalışmasını sağlamak için istenen araç referans hızını izlemek üzere EA'yı tahrik eder. Bu çalışmada, CarSim yazılım paketi kullanılarak EA'nın, aerodinamik yük kuvveti ve bozuk yol şartlarında oluşan sensör hatalarına karşı bir hız izleme kontrolü geliştirilmiştir. Önerilen arıza tespit şemasının geçerliliğini göstermek için 26 W üç fazlı bir SMSM üzerinde deneyler gerçekleştirilmiştir.

[44]'te, yalnızca tek tahrik tekerleğinin etkin olması koşuluyla, tüm tekerleklerden bağımsız tahrikli ve direksiyonlu EA için hata toleransı yüksek bir kontrol algoritması tasarlanmıştır. Bu hata toleranslı kontrol algoritmasının hedefi, araç performansını korumak ve hata durumunda sürücüye destek olmaktır. İstenen bileşke kuvveti ve momenti hesaplamak için Kayan Mod Kontrol yöntemi kullanılmıştır. Kontrol algoritması, her bir lastiğin boylamsal ve yanal kuvvetlerini tespit etmek için uygulanmıştır. Ayrıca, hatalı aracın davranışını düzeltmek ve güvenli sürüşe katkıda bulunmak amacıyla geliştirilmiştir. Gerçekleştirilen simülasyonlarla önerilen algoritma doğrulanmıştır.

Sürüş ortamlarında, aracın içinden en sık görülen nesneler trafikteki diğer araçlardır. Bu nedenle, güvenli ve verimli otomatik sürüş için yüksek doğrulukta araç tanıma algoritmalarının geliştirilmesi esastır. Ancak, sedan, otobüs, kamyon gibi farklı görünümlere sahip çeşitli araç türleri bulunduğundan, araçları sürekli olarak yüksek doğrulukla tespit etmek zordur. Bu sınıf içi

varyasyonun ele alınması gerekir veya araç tipine bağlı olarak düzensiz tanıma performansı oluşabilir. Geleneksel makine öğrenimi tabanlı algoritmalar, çoğunlukla genel örnekler üzerinde eğitildikleri için bu sorunu çözmede yetersizdir. Araç görünümündeki geniş çeşitlilik göz önüne alındığında, her araç tipinden numune toplamak ideal olmayabilir. Bu yüzden [45]'te, her araç tipinin ortak görünüm özelliklerini kullanan bir araç tanıma algoritması önerilmiştir. Dikdörtgen şekiller bir 3D LIDAR tarafından, lastikler ve tamponlar ise bir monoküler kamera tarafından yakalanmıştır. Bu ortak görünümelerden çıkarılan açısız özellikler daha sonra araç tanıma için Dempster-Shafer teorisi tarafından birleştirilmiştir. Deneysel çalışmalar gerçekleştirilerek, önerilen algoritma tarafından yakalanan ortak görünümelerin, kentsel sürüş ortamlarında çeşitli araç tiplerini tanımak için yeterince genelleştirilmiş özellikler sağladığını göstermişlerdir.

[46]'da, gençler ve daha yaşlı sürücüler için simülasyon sonuçları ile geliştirilmiş bir elektronik güvenlik sistem tasarımı sunulmuştur. Bu tasarımda, sürüş hatalarının sayısı ve hata süresi gibi iki parametre dikkate alınmıştır. Sürüş hatalarının sayısı belirli bir hata sınırını aşması durumunda, güvenlik tepkisi olarak sistemden uygun bir cevap alınmıştır. Simülasyon sonuçlarından, güvenli ön mesafe, kısa ön mesafe alarmı ve uzun ön mesafe alarmı olan üç sürüş koşulu durumu arasında tanıma yapılabildiği anlaşılmıştır. Elde edilen sonuçlara göre, sistemin tepki verme açısından etkinliğinin yüksek olduğu ve sürüş güvenliği alanlarında yüksek performanslı bir sistem elde edilebileceği görülmüştür.

[47]'de, planlanan bir rota üzerinde enerji tüketimini en aza indirmek amacıyla birbirleriyle bağlı araçlar için hız sistemleri önerilmiştir. Böylece araç güvenliği sağlanarak, bağlı ve otomatik araçlarda optimum şekilde hızlanma veya yavaşlama için güvenli bir eko-sürüş kontrol sistemi tasarlanmıştır. Asgari araçlar arası mesafe ve azami yol hız sınırı, durum kısıtlamaları olarak tanımlanmış ve enerji tüketimini en aza indiren optimal bir kontrol problemi sunulmuştur. Gerçek zamanlı kullanım için analitik durum kısıtlamalı bir çözüm türetilmiştir. Önerilen sistemde, önceki aracın çeşitli sürüş senaryoları için simülasyonlar gerçekleştirilmiştir. Sonuçlar, enerji tüketimini önemli ölçüde azaltabileceğini, yolculuk süresini artırmadan çarpışmayı önleyebileceğini ve enerji açısından verimli, gelişmiş bir hız sabitleyici olarak kullanılabileceğini de göstermiştir.

Diğer bir çalışma [48]'de, otomatik araç takibini gerçekleştirmek için yerleşik radar sensörünü ve V2V iletişimini benimseyen bir kontrol çerçevesi geliştirilmiştir. İlk olarak, öndeki aracın hızını takip etmek ve araçlar arası güvenli bir mesafe sağlamak için dizi kararlılığı tarafından sınırlandırılan bir doğrusal ileri besleme ve geri besleme denetleyicisi tasarlanmıştır. Daha sonra, V2V tarafından iletilen geçmiş hareket verilerine dayanarak önceki aracın yörüngesini hesaplamak için bir yol tahmin yöntemi önerilmiştir. Ön tekerleğin direksiyon açısını düzenlemek için model öngörülü kontrol yöntemi uygulanmıştır ve önerilen yöntemin sağlamlığı ve uyarlanabilirliği, çeşitli sürüş koşulları altında simülasyonlar ve testler ile doğrulanmıştır.

[49]'da, EA'nın menzilini iyileştirmek için bir hız sabitleme sistemi (CCS) çalışılmıştır. Geleneksel uyarlanabilir hız sabitleyici (ACC) ile EA, önde giden bir aracı takip eder veya hız sınırına yakın bir hızda gidecek şekilde kontrol edilir. Bu da EA'nın hızının, ortalama trafik hızından önemli ölçüde yüksek olmasına neden olabilir. Bu çalışmada, bu sorunu ele almak için bir CCS önerilmiştir. Buna göre bir EA'nın hızı, ortalama trafik hızının üzerine çıkmasını engelleyecek şekilde kontrol edilerek araçlar arası güvenli mesafe sağlanmıştır. Bu CCS'nin tasarımı ve simülasyonu bir MATLAB simülasyonu ile yapılmıştır. Simülasyon analizinde, önerilen sistemin EA'nın enerji tüketimini önemli ölçüde azalttığı görülmüştür.

1.3. Tezin Organizasyonu

Bu kısımda tezin bölümleri ile ilgili kısa açıklamalar yer almaktadır.

İkinci bölümde, EA'ların temel yapıları ve tarihsel gelişimleri açıklanmıştır. Daha sonra, çeşitleri, çalışma prensipleri ve tekerlek içi motor mimarisi detaylı anlatılmıştır.

Üçüncü bölümde, bu tez çalışmasında kullanılan SMSM'lerin yapısı, çeşitleri ve çalışma prensipleri ele alınmıştır. Matematiksel denklemlerle dinamik modeli açıklanmıştır.

Dördüncü bölümde, araçlarda ve endüstride kullanılan ağ sistemleri anlatılmıştır. Öncelikle motor yönetim sistemleri açıklanmış, bir Elektronik Kontrol Ünitesi (EKÜ)'nün yapısı ve çalışması gösterilmiştir. Daha sonra, haberleşme protokolleri hakkında bilgiler verilmiş, bu çalışmada da kullanılan CAN-Bus haberleşme protokolü ve çalışması detaylı açıklanmıştır. Ayrıca, bu tezde tercih edilmesinin nedenleri ve önemli avantajları üzerinde durulmuştur.

Beşinci bölümde, güvenli sürüş uygulamalarını gerçekleştirmek için çalışılan sayısal görüntü işleme ve algoritmaları açıklanmıştır. Bir görüntü işlemede temel adımlar, morfolojik işlemler ve kenar çıkarma algoritmaları detaylı olarak verilmiştir.

Altıncı bölümde, dört tekerden tahrikli bir EA için bu tezde gerçekleştirilen EDS tasarımı yer almaktadır. Bu çalışmada kullanılan Ackermann modelinden çıkartılan matematiksel denklemler verilerek, EDS'nin modellenmesi öncelikle Matlab/Simulink'te gerçekleştirilmiştir. Daha sonra bu matematiksel denklemler CoDeSys V2.3 yazılımına gömülerek endüstriyel bir işlemci ile programlanmıştır. Kullanılan bu yazılım paketi ve fonksiyonları da bu bölümde açıklanmıştır. Deneysel düzenek hazırlanarak, direksiyon açısına ve aracın hız bilgisine göre, dört tekerlek hızları hesaplanmıştır.

Bu bölümün ikinci kısmında, dört tekerden tahrikli bir EA'nın dinamik modeli Matlab/Simulink'te gerçekleştirilmiştir. Model, EDS, dört SMSM ve sürücüler, tekerlek ve araç dinamiklerinden oluşmaktadır. Gerçek sistemden alınan tekerlek hızları EDS çıkışından alınan hızlarla karşılaştırılarak bir kontrolör tasarlanmıştır. Tasarlanan sistem modeli ve teorisi kapsamlı olarak verilmiştir.

Yedinci bölümde, bu tez kapsamında gerçekleştirilen görüntü işleme uygulamaları yer almaktadır. FPGA ve Raspberry Pi 4 gibi gömülü sistem platformları kullanılarak yapılan nesne tanıma, hareket algılama, Canny kenar çıkarma algoritması, yüz tanıma, viraj yönü tespiti ve risk hesabı gibi uygulamalar anlatılmıştır. Ayrıca Xilinx System Generator kullanılarak gerçekleştirilen görüntü işleme uygulamaları ve sonuçları da yer almaktadır. Raspberry Pi'ye bağlı bir kameradan alınan görüntüler gerçek zamanlı olarak işlenmiş ve buna göre motorlar kontrol edilmiştir. Daha sonra viraj yönü, açısı ve risk hesabına yönelik çalışmalar gerçekleştirilmiş ve bu çalışmaların teorisi açıklanmıştır.

Sekizinci bölümde, dört tekerlek içi motorlu bir EA için gerçekleştirilen dinamik model ve EDS tasarımının sonuçları detaylı olarak verilmiştir. CoDeSys ve Simulink'ten elde edilen simülasyon sonuçları, deneysel sonuçlarla karşılaştırılmıştır. Tasarlanan hız kontrolörünün sonuçları da kapsamlı şekilde gösterilmiştir. Ayrıca EA'nın güvenli sürüşü üzerine yapılan viraj yönü ve açısının tespiti ile risk hesabının sonuçları da bu bölümde yer almaktadır. Buna göre sürücüye uyarı verebilen güvenilir ve stabil bir sistem tasarlanmıştır.

Son bölümde ise, tezden çıkartılan sonuçlara göre gelecekte yapılması düşünülen çalışmalar anlatılmaktadır.

1.4. Tezin Katkısı ve Amacı

Bu tez çalışmasında, EA'lardaki güvenlik önlemleri üzerine çalışmalar gerçekleştirilmiştir.

- İçten yanmalı motor kullanan araçlar çok sayıda karmaşık sisteme ihtiyaç duymaktadır. Bu araçlarda mekanik şanzıman, aks, dişli sistemi, mekanik diferansiyel gibi pek çok aktarım organı yer alır. EA'larda ise hareketli parçaların sayısı daha az olduğundan aracın ağırlığı daha düşüktür.

- İçten yanmalı motor kullanan araçlarda aktarım organlarının getirdiği arıza ve bakım masrafları da EA'lara göre oldukça yüksektir. Bu yüzden bu çalışmada, motorlar tekerleklerin içerisine yerleştirilerek aktarım organlarının kullanımı ortadan kaldırılır ve tüm sistemin kontrolü elektronik olarak gerçekleştirilir. Bunun için dört adet SMSM tekerleklerin içerisine yerleştirilir ve dört motordan tahrikli bir EA için EDS tasarlanır. Böylece mekanik parçaların yerine elektronik sistemlerin kullanılması aracın kütlesini azaltırken, bunların getirdiği bakım maliyetleri de önemli ölçüde azalmış olur.

- Bunun yanı sıra çalışmada kullanılan SMSM, yüksek verim ve moment/güç yoğunluğuna sahip olması gibi önemli avantajlarından dolayı EA'larda yaygın kullanılmaktadır. Bu yüzden bu çalışmada da tercih edilmiştir.

- Daha sonra dört tekerden tahrikli EA'nın dinamik modeli oluşturularak araç dinamikleri de sisteme dahil edilmiştir. Tasarlanan model, EDS, dört SMSM ve sürücüler, lastik ve araç

dinamiklerinden oluşmaktadır. SMSM'lerin yük koşulları da yolun eğimine ve viraja göre değerlendirilmektedir. Buna göre EDS çıkışından alınan referans hız farkları, gerçek sistemden alınan hız farklarıyla karşılaştırılarak bir kontrolör tasarlanmıştır. Tasarlanan hız kontrolörü literatürdeki karmaşık yöntemlerin aksine daha basit bir yöntem sunarak işlem kolaylığı sağlamaktadır. Böylece bu kontrol yöntemiyle sistemin stabilitesi sağlanarak moment ve güç tekerleklere dengeli dağıtılmıştır.

- Bu çalışmada öne çıkan bir diğer yapı, tasarlanan tüm sistemin güvenli şekilde yüksek haberleşme hızında çalışmasıdır. Bunun için kullanılan CAN-Bus haberleşme protokolüyle sisteme bağlanan tüm işlemci ve sensörler arasında 1 Mbit/s olan hattaki en yüksek veri hızında haberleşme gerçekleştirilir. Aynı zamanda bu protokolün önemli avantajlarından biri olan mesaj öncelikli yapıyla istenilen bilgilerin önce gönderilmesi sağlanır. Böylece hatta veriler karışmadan tüm sistem birbiriyle sorunsuzca haberleşebilmektedir.

- Tasarlanan EDS'de, direksiyon açısına ve aracın hızına bağlı olarak dört motorun dönmesi gereken hız değerleri simülasyonlarla ve deneysel çalışmalarla ayrı ayrı elde edilerek her bir motorun farklı devirde dönmesi sağlanmıştır. Burada kullanılan ana işlemci HY-TTC60 adında araç endüstrisinde yaygın şekilde kullanılan bir ARM işlemcidir. Bu işlemci literatür çalışmalarında yer almayan endüstriyel bir üründür. İçerisinde iki CAN ve LIN haberleşme hatları ve çok sayıda analog/dijital girişler bulunur. Bu yüzden diğer işlemcilerde kullanılamayacak sayıda sensör ve cihazı bu işlemcinin pinlerine bağlayarak haberleştirmek mümkündür. Aynı zamanda iki CAN hattı bulunduğundan ayrı bir kontrol sistemi gerektiğinde ikinci CAN hattı üzerinden tek yazılımla programlamak mümkündür. Örneğin, birinci CAN hattından motorların kontrolü yapılırken, ikinci CAN hattından batarya yönetim sistemi haberleştirilebilir. Bu işlemcinin kullanımı da bu nedenle önemli avantajlar sağlar.

Elektrikli Araçlar İçin Güvenlik Kavramı

Güvenlik Nedir?

Araçlarda güvenlik sistemleri aktif ve pasif güvenlik olmak üzere ikiye ayrılır. Pasif güvenlik can kaybını önleyebilmek amacıyla araçta yer alan hava yastığı, emniyet kemeri gibi kaza olduğunda devreye giren ve akıllı olmayan güvenlik unsurlarıdır. Aktif güvenlik ise kendi içerisinde ikiye ayrılır. Bunlardan ilkinin, SDS'ler oluşturur. Bunlar, araç yolda giderken sürücüye destek olan, çevredeki olası durumları analiz ederek kazaları önlemeye yönelik hesaplar ve çıkarımlar yapabilen akıllı sistemlerdir. Aktif güvenliğin ikincisini ise, aracın yolda dengede gidebilmesini ve tüm birimlerin güvenli şekilde çalışmasını sağlayan donanımsal ve tasarıma dayalı sistemler oluşturmaktadır.

Elektrikli Araçlarda Güvenlik

EA'larda yer alan motor, sürücü, batarya gibi tüm donanımın güvenli şekilde kontrol edilmesi oldukça önemlidir. Bu yüzden bu tez çalışmasında, aktif güvenliğin donanımsal tarafını oluşturan bir güvenlik sistemi tasarlamak amaçlanmıştır. EDS optimal olarak tasarlandıktan sonra HY-TTC60 ana işlemcisine Raspberry Pi ve FPGA bağlanarak çeşitli görüntü işleme uygulamaları gerçekleştirilmiş ve tüm sistemin güvenilir şekilde çalışması sağlanmıştır. Raspberry Pi'ye bağlı bir kameradan alınan gerçek zamanlı görüntüler işlenerek aracın önüne engel çıkması durumunda HY-TTC60 ana işlemci, motorlara CAN hattı üzerinden bilgi göndermektedir. Böylece motorlar durdurularak güvenli sürüş sağlanmıştır. Ayrıca sistemde kullanılan üç işlemci, enkoder, gaz pedalı gibi tüm sensörler ve motor-sürücü sistemi senkronize olarak, 1 Mbit/s haberleşme hızında sorunsuz ve güvenilir şekilde çalıştırılmıştır. Böylece bu çalışmayla görüntü işleme yöntemlerinin elektrik makinalarına uygulaması gerçekleştirilmiş ve CAN-Bus protokolüyle de haberleşme alanında çalışmalar yapılarak tüm alanları birleştiren disiplinler arası bir çalışma gerçekleştirilmiştir.

2. ELEKTRİKLİ ARAÇLAR

Ulaşım, dünyadaki en önemli araştırma alanlarından biridir. İnsanların taşınması için iki tür taşıma vardır: Kişisel ulaşım ve ortak taşımadır [50]. Otomobiller en önemli ve en çok kullanılan kişisel ulaşım araçlarıdır. Son birkaç yılda, daha fazla karbondioksit (CO_2) üretimi nedeniyle araştırmacılar bu endüstri için yeni teknolojiler aramaya başladılar. Bu yeni gelişen teknolojiler ile ulaşım artık CO_2 emisyonları için bir tehdit olmayacak ve otomobiller yakın gelecekte tükenen bir enerjiyi kullanmayacaktır. Son araştırmalar, içten yanmalı motorlu araçların her zaman petrole bağımlı olacağını gösteriyor. Dünya petrol rezervleri gün geçtikçe azalırken, içten yanmalı motorlu araçların yerini EA'ların alacağı ve HEA'ların da kullanılmaya başlanacağı öngörülmüştür.

1900'ların başında, elektrik, buhar ve benzinle çalışan üç otomobil çeşidi birbiriyle rekabet ediyordu [51]. On yıldan biraz daha uzun bir süre içinde, teknolojik yenilikler ve araçtan hız ve güç beklentisinin artması, benzinle çalışan otomobillerin ortaya çıkmasıyla ulaşımın geleceğini değiştirmiştir. 1990'larda, EA'ların ön plana çıkmasının toplumsal nedenleri, içten yanmalı motorların neden olduğu çevresel hasarı azaltmak, enerji maliyeti ve ulusal bağımlılıktaki maliyeti azaltmaktır. Benzinle çalışan araçlar, metan olmayan organik gaz (NMOG) emisyonlarının % 43'ünü, nitrojen oksit (NO) emisyonlarının % 57'sini ve karbon monoksit (CO) emisyonlarının % 82'sine neden olur [51]. EA'ların kullanımıyla bu emisyonların yayılımı ortadan kaldırılmıştır.

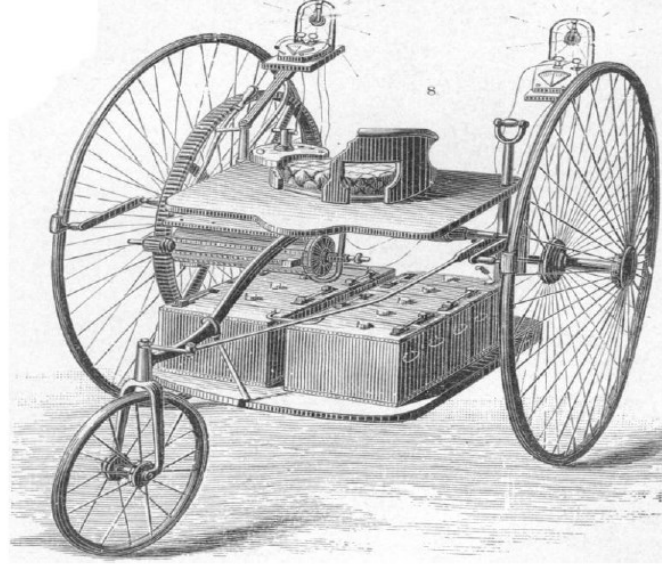
Günümüzde içten yanmalı araçlar, tahrikleri için gerekli enerjiyi hidrokarbon (HC) yakıtlarının yanmasından elde ederler. Yanma, yakıt ile hava arasında ısı ve yanma ürünlerini açığa çıkaran bir reaksiyondur. Isı, bir motor tarafından mekanik güce dönüştürülür ve yanma ürünleri atmosfere verilir. Bir HC, karbon ve hidrojen atomlarından oluşan molekülleri olan kimyasal bir bileşiktir. İdealde, bir HC'nin yanması sadece çevreye zarar vermeyen CO_2 ve su verir. Yeşil bitkiler fotosentez yoluyla bu CO_2 'yi kullanırlar. Yanma ürünleri CO_2 ve suyun yanı sıra, tamamı insan sağlığı için toksik olan belirli miktarda NO, CO ve yanmamış HC'ler içerir. Fosil yakıtların tükenmeye yüz tutması ve yaygın kullanımının getirdiği çevresel sorunlar, EA'ların ortaya çıkmasına zemin hazırlamıştır. Günümüzde üç çeşit EA yer almaktadır. Bunlar; tamamen EA'lar, yakıt hücreli EA'lar ve HEA'lardır. Pil ve yakıt hücreli EA'lar, akım varken yalnızca elektrik gücüyle çalışırlar. HEA'larda ise, hem elektrik motoru hem de içten yanmalı bir motor bulunur. Bu araçlar, yüksek enerji depolama kapasitesine ve büyük elektrik yükü şarj gereksinimlerine sahip pillerin kullanılmasını gerektirir. Bu yüzden elektrik güç sistemi tasarımında ve işletiminde önemli etkilere sebep olurlar.

2.1. Elektrikli Araçların Tarihi

İlk EA'lar, 18. yüzyıl başlarında, 1830'larda ortaya çıkmıştır. EA'ların evrimi yaklaşık 200 yıldır devam etmektedir. Günümüzde artan çevre kirliliği, egzoz emisyonu, yakıt talebi ve sınırlı yakıt kaynağının varlığı, insanları yeni temiz enerji kaynağı aramaya yönlendirmiştir. Böylece elektrik gücünü yeniden keşfetmek ve araçlara uyarlamak için harekete geçmişlerdir.

1830'larda EA'larda kullanılması nedeniyle şarj edilemeyen piller vardı [52]. Ticari EA'ların üretilmesi için yarım asır beklemek gerekiyordu. Bu yüzden EA'ların gündeme gelmesini batarya teknolojisindeki gelişmeler sağlamıştır.

İlk EA, 1881'de Fransız Elektrik Mühendisi Gustave Trouvé tarafından yapılmıştır [53]. Bu EA, Şekil 2.1'de gösterilen kurşun-asit pillerle beslenen, 0,1 Hp DA motorla çalışan bir üç tekerlekli bisikletti. Tüm araç ve sürücüsü yaklaşık 160 kg ağırlığındaydı.

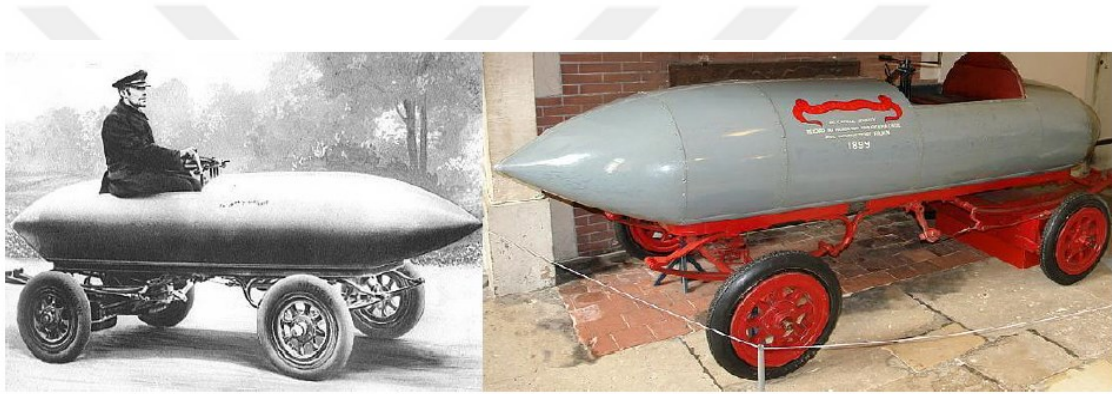


Şekil 2.1. İcat edilen ilk elektrikli araç [52].

Buna benzer bir araç 1883'te iki İngiliz Profesör tarafından yapılmıştır. Ancak zayıf teknolojileri nedeniyle halktan fazla ilgi görmemiştir. Benzinli araçlar, 15 km/s hız ve 16 km menzile sahipti. Bu nedenle EA kullanımı tercih edilmezdi. 1864'te Paris'ten Rouen'e yarışı her şeyi değiştirmiştir. 1135 km'lik yarış, ortalama 23.3 km/s hızla 48 saat 53 dakika sürmüştür. Bu hız, at arabalarının hızından çok daha üstündü. Dolayısıyla halk, otomobillere ilgi duymaya başlamıştır. Sonraki 20 yıl, EA'ların benzinli araçlarla rekabeti devam etmiştir. Bu, özellikle birkaç şehrin dışında çok fazla asfalt yolun bulunmadığı Amerika Birleşik Devletleri için geçerliydi. Bununla birlikte, Avrupa'da hızla artan asfalt yol sayısı sebebiyle benzinli araçlar tercih edilmeye başlanmıştır. Morris ve Salom, Electroboat adında ilk ticari EA'yı üretmişlerdir. Bu araç, mucitleri tarafından yaratılan bir şirket tarafından New York'ta bir taksi olarak

çalışıyordu. Electrobat, daha yüksek bir satın alma fiyatına rağmen emsallerinin aksine daha karlı olduğunu kanıtlamıştır. Araç, üç vardiya ve doksan dakikalık şarj süreleri ile dört saat kullanılabilirdi. Bu EA, iki adet 1.5 HP motorla sürülmüştür. Aracın maksimum hızı ve menzili, 32 km/s ve 40 km'ye ulaşmıştır.

Daha sonra 1897'de, Fransız M.A. Darracq tarafından ortaya çıkarılan rejeneratif frenleme icadı o dönemin en önemli gelişmesi olmuştur. Bu yöntem, aküleri yeniden şarj ederken aracın kinetik enerjisini geri kazanmayı sağlar ve bu da sürüş menzilini büyük ölçüde arttırır. Şehir içi sürüşte enerji verimliliğine her şeyden daha fazla katkıda bulunduğu için EA ve HEA teknolojisine yapılan en önemli katkılardan biridir. Ayrıca o dönemin en önemli EA'larından biri de Fransız Camille Jenatzy tarafından yaptırılan "La Jamais Contente"di. Bu araç, 100 km'ye ulaşan ilk araçtı. Hala müzede sergilenmektedir ve Şekil 2.2'de gösterilmiştir.



Şekil 2.2. La Jamais Contente elektrikli aracı [52].

EA'lardaki gelişmeler devam ederken benzinli motorlu araçlar daha güçlü ve esnek olduklarından daha yaygın kullanılmaya başlanmıştır. Sınırlı sürüş menzili ve düşük performans, EA'ları benzinli araçların gerisinde bırakıyordu. Son EA'lar 1905 tarihinde piyasaya sürülmüştür. Sonrasında yaklaşık 60 yıl, sadece golf arabaları ve dağıtım araçları EA'lar olarak üretilmiştir.

1945'te Bell Laboratuvarları'ndaki üç araştırmacı tarafından transistörün icadı, elektrik ve elektronik alanında önemli bir gelişme olmuştur. Daha sonra tristörler icat edilmiştir. Bunlar, yüksek gerilimlerde çalışabilen elemanlardı. Bu anahtarlama elemanları, elektrik motoruna gelen gücü daha verimli hale getirir ve çeşitli frekanslarda Alternatif Akım (AA) motorların çalışmasına izin vermektedir.

1980'lere kadar EA'larla ilgili çok az çalışma vardı. Çevre ile ilgili endişeler, bu konudaki araştırmaları arttırmıştır. Pil teknolojisi ve güç elektroniği alanındaki ilerlemelere rağmen, araç yelpazesi ve performansı hala yetersizdi. Modern EA çağı, 1980'lerde ve 1990'ların başında GM gibi firmaların EV1 ve Peugeot Société Anonyme (PSA) 106 Electric ile birkaç gerçekçi aracı piyasaya sürmesiyle doruğa ulaşmıştır. Bu araçlar, daha öncekilere kıyasla başarılı sonuçlar verse

de 1990'ların başında elektrikli otomobillerin benzinli otomobillerle menzil ve performans bakımından asla rekabet edemeyeceğini göstermiştir. Bunun nedeni, bataryalarda enerjinin, aynı enerji içeriği için benzinden çok daha ağır olan elektrotların metalinde depolanmasıdır.

Batarya teknolojisi hala büyük bir dezavantaj olarak EA'ların önünde durmaktadır. Bataryalar, EA'ların menzilinı kısıtlamakta ve aracın performansını düşürmektedir. Yüksek gerilim ve akım sağlayabilirler; ancak son yıllara kadar bir araç boyutuna yeterli bataryayı sığdırmak mümkün değildi. Birim ağırlık ve hacim başına enerji depolama kapasitesi çok zayıf olduğundan, bataryaların bu kapasitesi 21. yüzyılın başlarına kadar EA'ları sınırlamıştır.

2000'li yılların başlarında, yakıt tüketimini azaltmak için küçük araçların ve EA ve HEA'ların kullanımını arttırmaya yönelik çalışmalara başlanmıştır. 2004'te, Kaliforniyalı otomobil üreticisi Tesla Motors tarafından Tesla Roadster geliştirilmiş ve 2008 yılında piyasaya çıkmıştır. 2010 yılında, Mitsubishi MiEV ve Nissan Leaf EA'ları Japonya ve Amerika gibi çeşitli ülkelerde satışa çıkmıştır. Tesla Roadster ve Mitsubishi MiEV EA'ları Şekil 2.3'te gösterilmiştir. 2012 yılından beri ise, Citroen C1, Mercedes-Benz Vito E-cell, REVAi, Buddy, Wheego Whip LiFe, Transit Connect Electric, Tazzari Zero, Smart ED, Mia Electric, BYD e6, Ford Focus Electric, BMW ActiveE, Honda Fit, Coda, Renault Fluence Z.E.. Tesla Model S. gibi pek çok elektrikli otomobil piyasada yerini almıştır [54].



Şekil 2.3. Tesla Roadster ve Mitsubishi MiEV [54].

2.2. Elektrikli Araç Teknolojisi

EA'lar, 19. ve 20. yüzyıllardan günümüze kadar gelen araç teknolojisindeki gelişmelerle artan bir ivme kazanmıştır ve altı temel çeşitte sınıflandırılmaktadır.

2.2.1. Bataryalı Elektrikli Araç

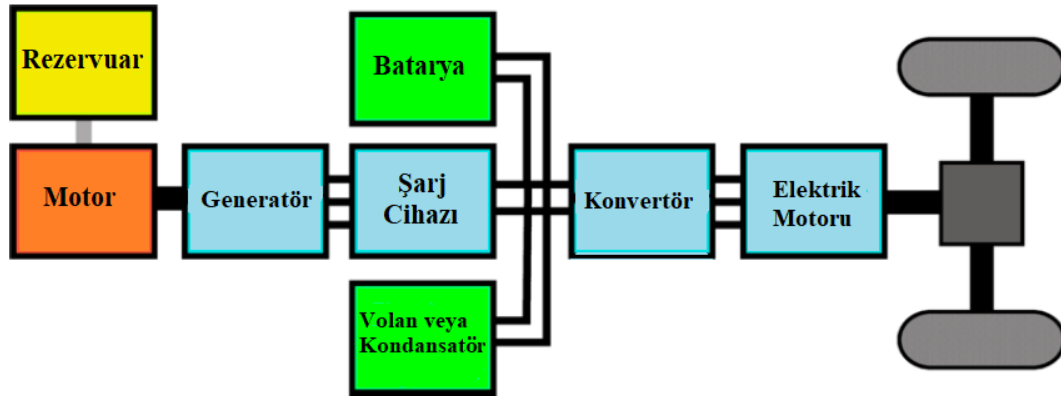
EA'nın keşfedildiği ilk zamandan beri var olan ilk bataryalı EA çeşididir. Bu araç, diğer çeşitlere kıyasla kolaylıkla tasarlanabilmektedir. Araç, enerji depolama ünitesi için bir batarya, bir elektrik motoru ve bir kontrolör içerir [52]. Bataryalar, şarj edilebilir ve bir şarj ünitesine takarak

yeniden enerjilenebilir. Şarj üniteleri aracın içine yerleştirilebilir veya aracın dışında yer alabilir. Kontrolör, elektrik motorunun hızını kontrol etmek için kullanılır. Aracın optimum güç koşullarında çalışmasını destekler. Elektrik motorları aynı zamanda generatör olarak da çalışabilir. Kontrolör, optimum güç dağılımına ulaşmak için elektrik motorunun çalışma moduna da karar verir. Bu tip EA'lar, EA'ların ilk modellerini oluşturmaktadır. Elektrikli bisikletten elektrikli arabalara, dağıtım kamyonlarından otobüslere kadar pek çok örneği vardır. Sınırlı menzile ve performansa sahip olması gibi bir dezavantajı vardır. Ancak bunlar belirli amaçlar için yeterlidir.

2.2.2. Hibrit Elektrikli Araç

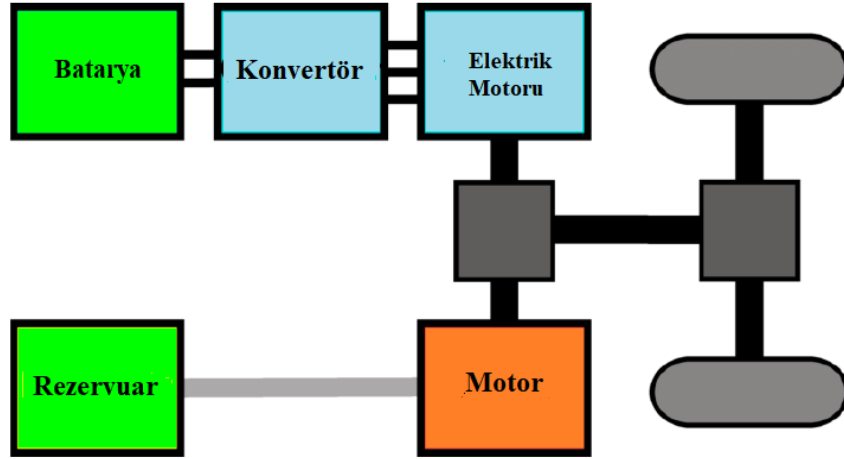
İkinci EA çeşidi, bir pil ve bir içten yanmalı motorun bir araya gelmesinden oluşan HEA'dır. Bunlar genelde son yıllarda incelenmiştir. Bir hibrit araç, iki veya daha fazla güç kaynağı kullanır. Güç çiftlerinin birçok kombinasyonu vardır. Hibrit araç, içten yanmalı motor, elektrik motoru ve generatörden oluşur. Seri ve paralel hibrit olmak üzere iki çeşitten oluşur.

Seri HEA'larda, elektrik motoru aracı hareket ettirir. Elektrik motoru, bir elektrik pilinden veya içten yanmalı bir motorla çalıştırılan bir generatörden beslenir ve her iki güç kaynağı da elektrik motoruna enerji vermek için kullanılabilir. Seri HEA'larda, aracın çekiş gücü sadece elektrik motorundan gelmektedir. Diğer enerji kaynakları elektrik motorunu besler. Seri hibrit aracın yapısı Şekil 2.4'te gösterilmiştir.



Şekil 2.4. Seri hibrit elektrikli aracın tasarımı [52].

Paralel hibrit araçlarda ise, hem içten yanmalı motor hem de elektrik motoru kullanılır. Bu motorlar aracın güç aktarım sistemine bağlıdır ve aynı anda aracı hareket ettirebilir. Paralel hibrit aracın yapısı Şekil 2.5'te gösterilmiştir.



Şekil 2.5. Paralel hibrit elektrikli aracın tasarımı [52].

İçten yanmalı motor ve generatör, hibrit araçların hareketi sırasında aküyü şarj eder. Bu, akünün EA'lara göre daha küçük olmasını sağlar. Ayrıca sistem, HEA'ların çalışma şekli olan rejeneratif frenlemeye de izin verir. Bu modda elektrik motoru generatör gibi çalışır ve elektrik enerjisi üretir. Her iki hibrit çeşidinde de birden fazla elektrik motoru olabilir.

Seri ve paralel hibrit araçlar farklı uygulamalarda kullanılmak üzere geliştirilmiştir. Seri hibritler özel uygulamalarda kullanılırken, paralel hibritler geniş bir uygulama alanına sahiptir. Hibrit dizel motorlu trenler ve gemiler, seri hibritler için iyi örnekler olabilir. Generatörlerin hibritlerde bulunması gerekir. Çünkü elektrik enerjisini aküde depolamayı sağlar. Generatörün bu sistemlere eklenmesi maliyeti arttırmaktadır. Paralel hibritler, içten yanmalı motor ile sürülebildiğinden küçük elektrik motorlu araca imkân sağlar. Ayrıca farklı güç kaynakları kombinasyonları vardır. Geniş alanda kullanılmasının temel nedeni budur. Enerjinin kaynaklara göre dağıtılması için optimum kombinasyona karar verilir. Egzoz emisyonları, yakıt tüketimi ve şehir içinde veya dışında sürüş enerji dağıtım kriterleridir.

Paralel hibritlerde asıl amaç, motor devrini ve momentini kontrol ederek egzoz emisyonlarını ve yakıt tüketimini en aza indirmektir. Bu durum, Denklem 2.1'de gösterilen "Hibridizasyon Derecesi (HD)" adı verilen bir değişkenle tanımlanır.

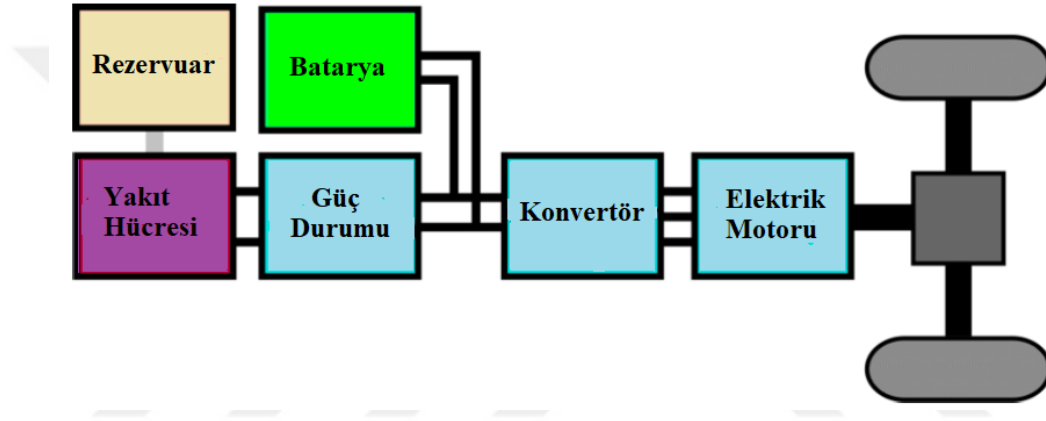
$$HD = \frac{\text{Elektrik motor gücü}}{\text{Elektrik motor gücü} + \text{İçten yanmalı motor gücü}} \quad (2.1)$$

Denklem (2.1), daha yüksek derecede hibritleşmenin minimum içten yanmalı motor gücü gerektirdiğini gösterir. Daha büyük bir HD, çoğu durumda daha iyi bir verimlilik sağlar. İçten yanmalı motor kapalıyken egzoz emisyonu olmaz; açık olması durumunda olur. HEA'lar bu

nedenle “kısmi sıfır emisyonlu araçlar” olarak adlandırılmaktadır. Hibrit araç teknolojisindeki gelişmeler fiyatlarını düşürerek onları daha ulaşılabilir hale getirmiştir.

2.2.3. Yakıt Hücresli Elektrikli Araç

Yakıt hücresli EA’lar, aracın enerji kaynağı olarak yakıt hücresini kullanır. Yakıt hücresi, enerji taşıyıcısını sıvı veya gaz formunda kullanan kimyasal bir yapıdır. Bir kimyasal reaksiyondan sonra enerji üretir. Yakıt hücrelerinin icadı yaklaşık 1840 yılında olmuştur. Yakıt hücresli EA’lar sadece kara araçları değil aynı zamanda gemilerdir. Pahalı bir teknoloji olması dezavantajlarıdır. Yakıt hücresli EA’nın yapısı Şekil 2.6’da gösterilmiştir.



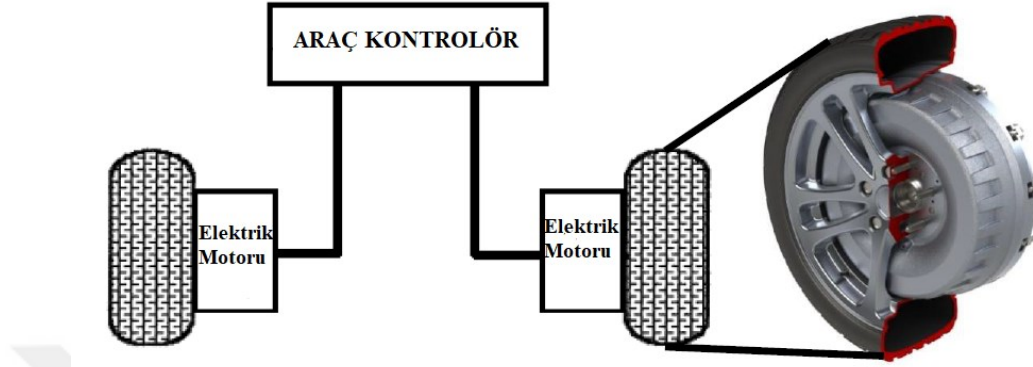
Şekil 2.6. Yakıt hücresli elektrikli aracın yapısı [52].

2.2.4. Elektrikli Araçların Diğer Çeşitleri

Güneş radyasyonu, EA’lar için başka bir enerji kaynağı türüdür. Bu türün yapısı sadece akülü EA’ya benzer. Bu tipler, elektrik motoru, batarya ve bataryayı şarj etmek için güneş ışığından faydalanır. Güneş, doğrudan doğal ve sonsuz bir enerji kaynağıdır. Bu tip araçların güneş ışığının yüksek olduğu bölgelerde kullanımı en idealdir. Bunun dışında, trolleybüs ve tramvay, besleme hatları ile çalışan iyi bilinen araçlardır. Araçların içinde küçük bir akü bulunur. Besleme hatları aküye gerekli elektrik enerjisini sağlar. Sistem bir önceki yüzyıldan beri kullanılmaktadır. Bataryalı EA’larda alternatif depolama elemanı olarak volanlar ve süper kapasitörler kullanılmaktadır. Bu elemanlar genellikle yardımcı enerji kaynağı olarak kullanılır ve elektrik gücünü depolar. Bu tür enerji depolama elemanları hızla şarj olur. Yüksek güçleri olabilir, ancak kapasiteleri küçüktür. İyi bir güç yoğunluğuna ve zayıf bir enerji yoğunluğuna sahiptirler.

2.3. Tekerlek İçi Motorlu Elektrikli Araç Mimarisi

Tekerlek içi motor, tekerleğin göbeğine bir elektrik motorunun monte edildiği bir konfigürasyondur. Tekerlek içi motorun konfigürasyonu Şekil 2.7'de gösterilmektedir.

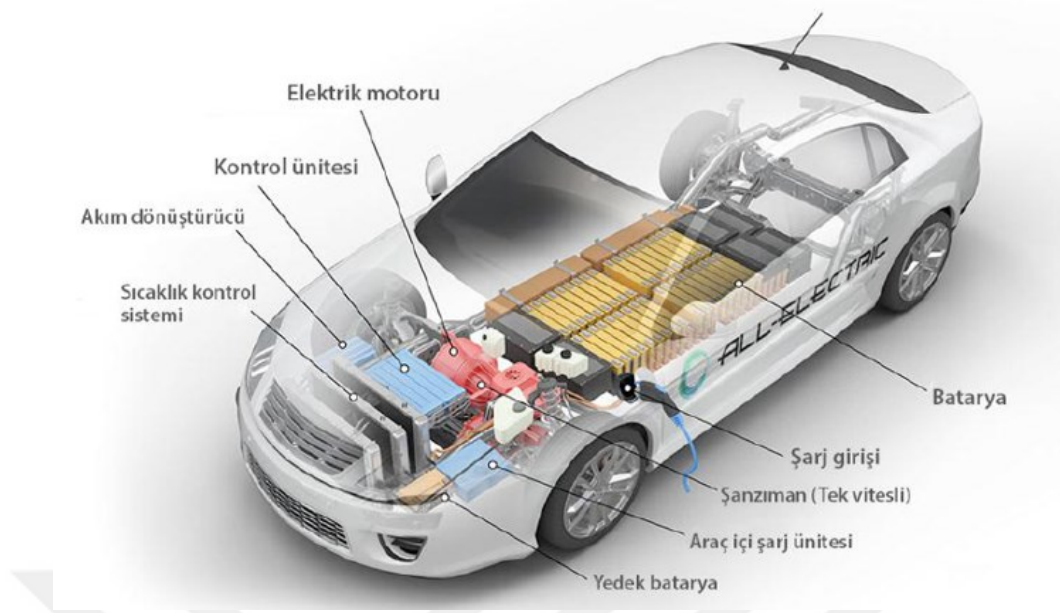


Şekil 2.7. Tekerlek içi motor konfigürasyonu [52].

Tekerlek içi motorların bazı önemli avantajları vardır. Aktarma organlarını ve diferansiyel mekanizmasını araç yapısından kaldırarak araç kütlesinin azalmasını sağlarlar. Motorlar bağımsız olarak kontrol edilebilir. Bu da aracın esnek bir yapıya sahip olmasını sağlar. Tekerlek içi motorlu araçlar, diğer mimarilere göre farklı yol koşullarında daha güçlü hale getirilebilir. Bu yapı, bir çekiş kontrol sistemiyle araca optimum çekiş tahriki sağlar.

2.4. Elektrikli Araçların Çalışma Prensibi

EA, tahrik için bir veya daha fazla elektrik motoru veya çekiş motoru kullanan bir araçtır. Araç dışındaki kaynaklardan elektrikle bir kolektör sisteminden çalıştırılabilir veya yakıtı elektriğe dönüştürmek için batarya, güneş panelleri, yakıt hücreleri veya elektrik jeneratörü kullanılabilir. Bir EA'nın yapısı detaylı olarak Şekil 2.8'de gösterilmiştir.



Şekil 2.8. Elektrikli araç yapısı [55].

Bir EA'da yer alan tüm elemanlar sırasıyla şöyledir:

Batarya: Bilindiği gibi EA'lar, motoru döndürmek ve sürüş için gerekli gücü üretmek için aküde depolanan elektrik enerjisini kullanır. EA'larda motorun bu gücü üretmek için batarya kullanması, fosil yakıt tüketen içten yanmalı araçlardan en büyük farkıdır. Bu nedenle EA'ların içten yanmalı araçlar için en önemli bileşenlerden ikisi olan motor ve şanzımana ihtiyacı yoktur. Bunun yerine, EA'lar elektrik gücü için birkaç bileşen taşır. Bunlar; motor, batarya, yerleşik şarj cihazı ve Elektrik Gücü Kontrol Birimi'dir (EPCU). Bunların tümü, batarya elektriğinin EA'yı ileriye/geriye doğru hareket ettiren kinetik kuvvete dönüştürülmesini sağlamak için gerekli bileşenleridir.

Batarya teknolojisindeki son gelişmeler sayesinde, daha yeni EA'lar, batarya yoğunluğu ve sürüş mesafesi açısından eski modellere göre önemli gelişmelere sahiptir. Örneğin Kia Soul Booster EA, maksimum 386 km mesafeye (Kore sertifika standartlarına göre) dayanan 64 kWh lityum iyon batarya ile donatılmıştır. Batarya ömründe de önemli gelişmeler görülmüştür. Normal bir kullanım düzenini varsayarsak, Soul Booster EA'nın bataryası oldukça uzun ömürlüdür.

Şarj girişi: Şarj girişi, çekiş aküsü paketini şarj etmek için aracın harici bir güç kaynağına bağlanmasına izin verir.

Batarya Yönetim Sistemi (Battery Management System, BMS): Bataryanın birçok hücrelerini yöneterek tek bir varlıkmiş gibi çalışabilmelerini sağlar. EA'nın bataryası, binlerce mini hücreden oluşur ve bataryanın dayanıklılığını ve performansını optimize etmek için her hücrenin diğerleriyle benzer durumda olması gerekir.

BMS esas olarak hücrenin şarj/deşarj durumunu denetler. Ancak arızalı bir hücre gördüğünde, hücrenin güç durumunu (açık/kapalı) bir röle mekanizması (diğer devreleri açmak/kapatmak için koşullu mekanizma) aracılığıyla otomatik olarak ayarlar. BMS genellikle pilin gövdesine yerleştirilir. Ancak bazen Elektrik Gücü Kontrol Ünitesinde (EPCU) de yer alabilir.

Batarya Isıtma Sistemi: Daha düşük sıcaklıklarda, pil hem şarj kapasitesinde hem de hızda bir düşüş görür. Akü ısıtıcısı aküyü ideal sıcaklık aralığında tutmak, mevsimsel performans düşüşlerini önlemek ve maksimum sürüş mesafesini korumak için mevcuttur. Sistem şarj olurken de çalışır ve şarjın daha verimli olmasını sağlar.

DA/DA dönüştürücü: Bu cihaz, çekiş akü paketinden gelen yüksek gerilimli DA gücünü, araç aksesuarlarını çalıştırmak ve yardımcı aküyü yeniden şarj etmek için gereken düşük gerilimli DA gücüne dönüştürür.

Elektrik motoru: Elektrik enerjisini tekerlekleri hareket ettiren kinetik enerjiye dönüştürür. Böylece, otomobillerdeki gürültü ve titreşim en aza indirilir. Sessiz ve konforlu bir sürüş sağlanmasına imkân verir. Ayrıca, EA güç aktarma sistemi motordan daha küçüktür. Bu nedenle verimli araç tasarımı için genişletilmiş kabin alanı veya depolama gibi birçok ek alan sağlar.

Motor aynı zamanda kısmen bir elektrik generatörüdür. Vites boştayken (örneğin, araba yokuş aşağı giderken) üretilen kinetik enerjiyi aküde depolanabilen elektrik enerjisine dönüştürür. Aynı enerji tasarrufu fikri, otomobilin hızını düşürdüğü ve "rejeneratif fren sistemi" olarak adlandırılan sistemle sonuçlandığında da geçerlidir. Bazı araçlar, hem sürüş hem de rejenerasyon işlevlerini yerine getiren motorlar kullanır. Hyundai Motor Group'un EA'larından bazıları, rejeneratif frenleme seviyelerini kontrol edebilir ve yakıt ekonomisini iyileştirmektedir.

Redüktör: Redüktör, motorun gücünü tekerleğe etkin bir şekilde iletmeye hizmet ettiği için bir tür şanzımandır [56]. Motor, içten yanmalı bir motordan çok daha yüksek bir devir/dakika değerine sahiptir. Bu nedenle şanzımanlar motor devrini sürüş durumuna uyacak şekilde değiştirirken, redüktör her zaman devir/dakikayı uygun bir değere indirmelidir. Düşük devir/dakika ile EA güç aktarma sistemi, sonuçta ortaya çıkan daha yüksek momentten faydalanabilir.

Yerleşik şarj cihazı (On-board Charger, OBC): Şarj portu aracılığıyla sağlanan AA elektriğini alır ve çekiş aküsünü şarj etmek için DA gücüne dönüştürür. Ayrıca şarj ekipmanı ile iletişim kurar ve şarj esnasında gerilim, akım, sıcaklık ve şarj durumu gibi batarya özelliklerini izler. Yerleşik şarj cihazı (OBC), AA'yı yavaş şarj cihazlarından veya ev prizlerinde kullanılan taşınabilir şarj cihazlarından DA'ya dönüştürmek için kullanılır. Bu, OBC'nin geleneksel konvertöre benzemesine neden olabilir. Ancak işlev açısından önemli ölçüde farklılık gösterirler. OBC şarj içindir ve konvertör hızlanma/yavaşlama içindir. Hızlı şarj cihazları elektriği doğru akımla sağladığından dolayı hızlı şarjda OBC'ye gerek yoktur.

Güç elektroniği denetleyicisi: Bu birim, çekiş aküsü tarafından sağlanan elektrik enerji akışını yöneterek, elektrikli çekiş motorunun hızını ve ürettiği momenti kontrol eder.

Termal sistem (Soğutma): Bu sistem, elektrik motoru, güç elektroniği elemanları ve diğer bileşenler için uygun bir çalışma sıcaklığı aralığı sağlar.

Çekiş pil takımı: Elektrik çekiş motoru tarafından kullanılmak üzere elektriği depolar.

Şanzıman (Elektrikli): Şanzıman, tekerlekleri sürmek için elektrikli çekiş motorundan gelen mekanik gücü aktarır.

Elektrik Güç Kontrol Ünitesi (Electric Power Control Unit, EPCU): Elektrik Güç Kontrol Ünitesi (EPCU), araçtaki elektrik gücünün akışını kontrol ederek tüm cihazların verimli şekilde çalışmasını sağlar. Evirici, Alçak Gerilim DA-DA Dönüştürücü (LDC) ve Araç Kontrol Ünitesinden (VCU) oluşur.

- **Evirici:** Evirici, akünün DA'sını AA'ya dönüştürür ve daha sonra motor hızını kontrol etmek için kullanılır. Cihaz, hızlanma ve yavaşlamadan sorumludur. Bu nedenle EA'nın sürülebilirliğini en üst düzeye çıkarmada önemli bir rol oynar.
- **Alçak Gerilim DA-DA Dönüştürücü:** EA'nın aküsünden gelen yüksek gerilimi düşük gerilime (12 V) dönüştürür ve aracın çeşitli elektronik sistemlerini besler. EA'daki tüm elektronik sistemler düşük gerilimde elektrik kullanır. Bu nedenle bu sistemlere faydalı olması için önce aküdeki yüksek gerilimin dönüştürülmesi gerekir.
- **Araç Kontrol Ünitesi:** Araçtaki tüm elektrik güç kontrol sistemlerinin kontrol kulesi olan VCU, EPCU'nun en önemli bileşenidir. Motor kontrolü, rejeneratif fren kontrolü, yük yönetimi ve elektronik sistemler için güç kaynağı dâhil olmak üzere aracın neredeyse tüm güç kontrol mekanizmalarını denetler.

2.4.1. Elektrikli Araçların Avantaj ve Dezavantajları

EA'ların önemli avantajları vardır. Bunlar aşağıdaki gibi sıralanabilir [57]:

- Zararlı emisyonların yayılımını ortadan kaldırdığı için çevreye katkısı büyüktür.
- EA'lar anlık moment üretirler. Bu yüzden tepki verme süreleri kısadır. Bu da diğer araçlara göre çok daha hızlı şekilde hızlanmasını sağlar. Dolayısıyla daha sessiz ve konforlu bir sürüş sağlanır.
- Araç yokuş aşağı inerken hızlanma ya da frenleme durumlarında elektrik üretimi sağlarlar.
- EA'larda marş motoru, enjektör, yağ ve yakıt filtreleri, benzin ve egzoz boruları gibi parçalar yer almaz. Dolayısıyla bakım masrafları azdır.
- EA'larda kullanılan pilin ömrü tükenebilir. Ancak gelişen teknolojiler sayesinde pilin yenilenmesi oldukça kolaydır.
- Bazı ülkelerde EA'ların vergileri düşüktür veya yoktur.

- EA’larda, içten yanmalı motorlu araçlarda olduğu gibi benzin ve mazot giderleri yoktur. Böylece benzinli araçlardaki dışa bağımlılığı ortadan kaldırarak maliyeti düşürür.
- Elektrik motoru ucuzdur, ömrü uzundur ve kolayca değiştirilebilir.

EA’ların avantajlarının yanı sıra bazı olumsuz tarafları da vardır. Bunların zaman içerisinde giderileceği düşünülmektedir. Bunlar şöyle sıralanabilir:

- Fiyatları diğer araçlara göre yüksektir.
- İklim şartlarına bağlı olarak batarya performansı etkilenebilir.
- Şarj istasyonlarının sayısı azdır. Bu yüzden uzun mesafelerde kullanmak sıkıntılıdır.
- Bataryaların şarj süreleri uzundur. Bu sorunu çözmek için çalışmalar devam etmektedir.
- Bataryaların maliyeti yüksektir ve ömürleri kısadır. Bu nedenle fiyat/performans oranı düşüktür.
- Yolun durumuna ve aracın kullanım şekline göre menzillerde farklılıklar olabilir. Bataryadaki gelişmeler ve pillerin ömrü arttıkça bu sorun da çözülecektir.

Bu bölümde EA’ların yapısı, çeşitleri ve çalışma prensibi detaylı olarak anlatılmıştır. Gelecek bölümde ise EA’da kullanılan motor çeşidi olan Sürekli Mıknatıslı Senkron Motorlar ele alınacaktır.

3. SÜREKLİ MİKNATISLI SENKRON MOTORLAR

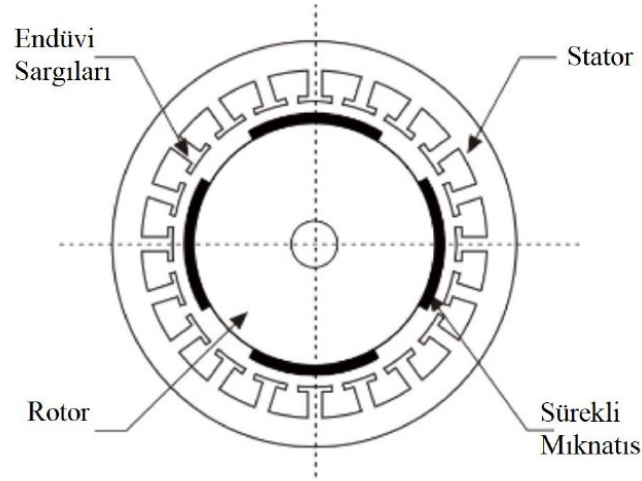
Sürekli Mıknatıslı Senkron Motorlar (SMSM), besleme frekansına bağlı olarak her zaman sabit bir hızda dönen makinalardır. AA gerilimler aracılığıyla elektronik olarak hızı ayarlanır ve sıralı olarak sürülürler. Bu tip motorlarda rotor manyetik akısı kalıcı mıknatıslarla elde edilir [58]. SMSM birçok özellik açısından sınıflandırılabilir. İç veya dış rotor yapıları mevcuttur. Çamaşır makinesi uygulamalarında kasnak-kayış tahrikli sistemde içten rotorlu yapı tercih edilirken, direkt tahrikli uygulamalarda rotoru dışında motor yapısı tercih edilmektedir. Stator yapısı, dağıtılmış veya konsantre sargılara sahip olabilir. Konsantre sargı tipleri, yüksek güç ve moment yoğunluğu, yüksek verim, akı zayıflatma kabiliyeti, kolay yapı, yüksek oluk doluluk faktörü ve sistem maliyeti gibi özelliklerine göre çeşitli avantajlara sahiptir.

SMSM, AA servo sürücüler için geleneksel olarak kullanılan diğer makinelerle göre sayısız avantajlar içerir. Bir asenkron motorun stator akımı, moment üreten bileşenin yanı sıra mıknatıslanma bileşeni de içerir. SMSM'nin rotorunda kalıcı mıknatıs kullanıldığından sabit hava aralığı akısı için stator üzerinden mıknatıslanma akımı sağlanmasına gerek yoktur. Stator akımı sadece moment üretmek için gereklidir. Dolayısıyla mıknatıslanma akımı olmadığı için SMSM daha yüksek bir güç faktöründe çalışır ve asenkron makinadan verimi daha yüksektir. Klasik sargılı rotorlu senkron makina, motor üzerinde genellikle fırça ve kolektörden sağlanan DA uyarıtımına sahip olmalıdır. Bu durum, rotor kayıpları ve düzenli fırça bakımı gerektirir. SMSM'nin geliştirilmesinin temel nedeninin, alan sargısını ve DA güç kaynağını kalıcı bir mıknatısla değiştirerek senkron makinanın belirtilen dezavantajlarını ortadan kaldırmaktır. Bu nedenle SMSM, sinüzoidal indüklenmiş bir Elektromotor Kuvvet'e (EMK) sahiptir ve tıpkı senkron makina gibi sabit moment üretmek için sinüzoidal akımlar gerektirir [59,60].

SMSM'ler, EA'lar, robotik uygulamalar, endüstriyel sürücüler, beyaz eşyalar ve servo sürücüler gibi pek çok alanda yaygın şekilde kullanılmaktadır [61].

3.1. Sürekli Mıknatıslı Senkron Motorların Yapısı

SMSM'ler, klasik senkron motorların uyarıtım sargılarının sürekli mıknatısla değiştirilmesi sonucunda oluşmuştur [61]. SMSM'ler, rotor, stator ve konum algılayıcılarından oluşurlar. Şekil 3.1'de bir SMSM'nin yapısı gösterilmiştir.



Şekil 3.1. Bir Yüzeye Monte Sürekli Mıknatıslı Senkron Motorun yapısı

SMSM'nin stator yapısı, asenkron motorlara oldukça benzerdir. Girdap akımlarını sınırlamak için bir veya iki yüzü yalıtılmış manyetik bakımdan kaliteli demir sacların paketlenmesi şeklinde yapılmıştır ve üç fazlı sargılardan oluşur [62].

SMSM'nin rotorunda, senkron motorda kullanılan rotor sargılarının yerine alan akısı oluşturmak için üzerine monte edilmiş sabit mıknatıslar vardır. Kalıcı mıknatıslar, hava aralığı içinde daha yüksek manyetik akı yoğunluğu sağlar. Böylece momenti ve dinamik performansını artırır.

Rotor yapısında çeşitli mıknatıslar kullanılır. Bunlardan ilki Aliminyum-Nikel-Kobalt'tan oluşan AlNiCo'dur. AlNiCo mıknatısların artık mıknatıslanma özelliği (B_r) yüksektir fakat artık mıknatısiyeti giderici kuvvet olan koersif kuvveti (H_c) düşüktür. Demanyetizasyon eğrisi lineer olmadığından küçük boyutlu motorlar dışında kullanımı tercih edilmez. Ancak yaklaşık 850°C gibi yüksek bir Curie sıcaklığına sahip olduğundan hala kullanımı devam etmektedir. Daha sonra Kobalt-Platin mıknatıslar ortaya çıkmıştır. Bunlarla daha başarılı sonuçlar elde edilmiştir. Fakat yüksek maliyetleri sebebiyle tercih edilmezler. Diğer bir mıknatıs çeşidi ise 1950'lerde ortaya çıkan Ferrit mıknatıslardır. Bunların artık mıknatıslanma özelliği düşüktür. Bu nedenle hava aralığında gerekli akıyı oluşturmak için daha fazla mıknatıs kullanılmalıdır. Düşük güç üretimine sahiptir. Orta derecede bir koersif kuvvete sahiptir. Maliyetleri daha düşüktür ve demanyetizasyon eğrisi lineerdir. Düşük maliyet gerektiren uygulamalarda tercih edilebilir.

1970'li yıllardan sonra nadir toprak mıknatısları olan Neodimyum ve Samaryum-Kobalt (SmCo) mıknatıslar geliştirilmiştir [62]. Lantanit alaşımlarından yapılırlar ve en güçlü mıknatıs malzemeleridir. SmCo mıknatısının artık mıknatıslanması ve koersif kuvveti yüksektir. Ancak maliyetleri çok yüksek olduğundan yüksek güç yoğunluğunun istenildiği uygulamalarda tercih edilirler. Ayrıca kırılgan bir yapıya sahip malzemelerdir.

SMSM'ler, mıknatısların rotordaki konumuna göre Yüzeye Monte Sürekli Mıknatıslı Senkron Motor (YMSMSM) ve Dâhili Sürekli Mıknatıslı Senkron Motorlar (DSMSM) olarak sınıflandırılabilir.

Mıknatıslar rotor yüzeyine monte edilmişse, Yüzeye Monte Sürekli Mıknatıslı Senkron Motor (YMSMSM) olarak adlandırılır. Mıknatıslar rotor yüzeyine yerleştirilirse hava aralığındaki manyetik alan yüksek olacaktır. Bu da mıknatıslama endüktansının düşük, yani endüvi reaksiyonunun ihmal edilebilir olmasını gösterir [63]. Makine yuvarlak kutup özelliği gösterir ve yüksek hızlı uygulamalarda tercih edilmezler. Bu topoloji, aynı zamanda düşük moment dalgalanması ve sorunsuz performans ile sonuçlanır.

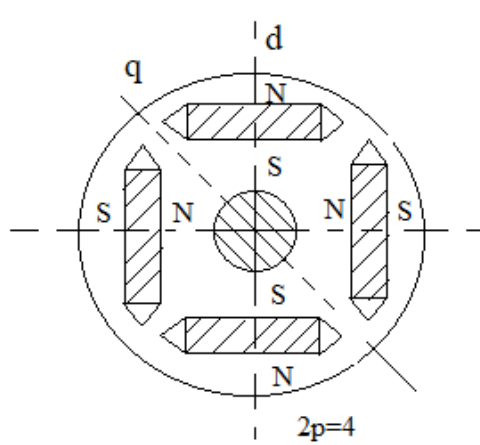
The diagram shows a central circle with six dashed lines extending to the vertices of a hexagon. The hexagon is divided into six regions, each containing a shaded rectangular area. The regions are labeled with 'N' and 'S' in a repeating pattern: top-left (N), top-right (S), right (N), bottom-right (S), bottom-left (N), and left (S). The shaded areas are labeled with 'd' and 'q' in a repeating pattern: top-left (d), top-right (q), right (d), bottom-right (q), bottom-left (d), and left (q). The label $2p=6$ is at the bottom.

29

3.1.2. Dâhili Sürekli Mıknatıslı Senkron Motorlar

SMSM'nin diğer çeşidi ise, kalıcı mıknatısların rotorun içine gömülü olduğu Dâhili Sürekli Mıknatıslı Senkron Motor'dur (DSMSM). Bu tip motorlarda relüktans momenti, elektromanyetik momenti artırma görevine sahiptir. DSMSM'nin yapısı daha dayanıklıdır. Bu yapıda, enine ve boyuna endüktanslar birbirinden farklı olduğundan relüktans momenti oluşur ve makinanın milinde indüklenen moment artar [62]. Böylece yüksek hızlara çıkmak mümkündür. Ancak makine çıkık kutup özelliği göstereceğinden hava aralığı azaldıkça endüvi reaksiyonunun etkisi de yükselecektir.

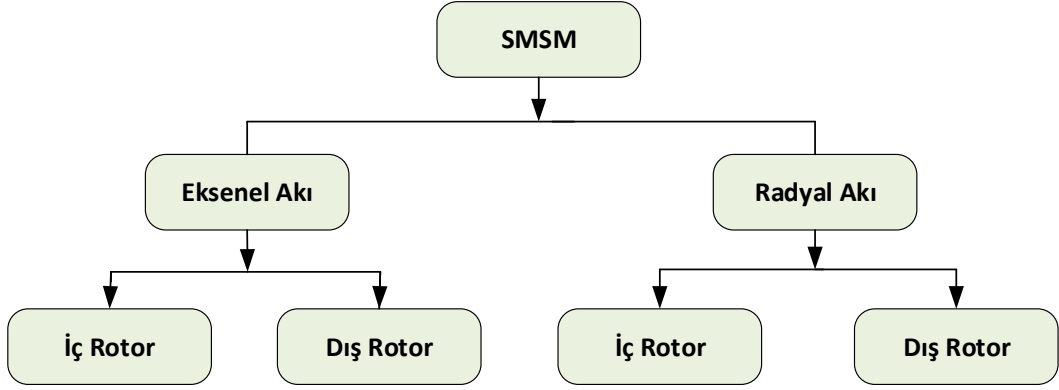
DSMSM, Şekil 3.3'te gösterildiği gibi radyal olarak mıknatıslanmış ve dönüşümlü olarak kutuplanmış mıknatıslara sahiptir. Mıknatıs kutup alanı rotor yüzeyindeki kutup alanından daha küçük olduğundan, açık devredeki hava aralığı akı yoğunluğu, mıknatıstaki akı yoğunluğundan daha azdır. d eksenindeki senkron reaktans, q eksenindekinden daha küçüktür. Çünkü q eksen manyetik akı, sürekli mıknatısları geçmeden çelik kutup parçalarından geçebilir. Mıknatıs merkezkaç kuvvetlerine karşı çok iyi korunur. Böyle bir tasarım, yüksek frekanslı yüksek hızlı motorlar için önerilir.



Şekil 3.3. Dâhili mıknatıslı rotor yapısı

Ekonomik olarak yüzeysel mıknatıslı yapıdaki motorlar tercih edilirler. Dâhili mıknatıslı yapıdaki motorlar ise daha yüksek güç yoğunluğuna sahiptir. Yüksek hızlarda çalışmaya izin verir ve daha yüksek moment sağlar.

SMSM'ler rotor ve statorun yerleşimine göre de Şekil 3.4'te gösterildiği gibi sınıflandırılabilir.



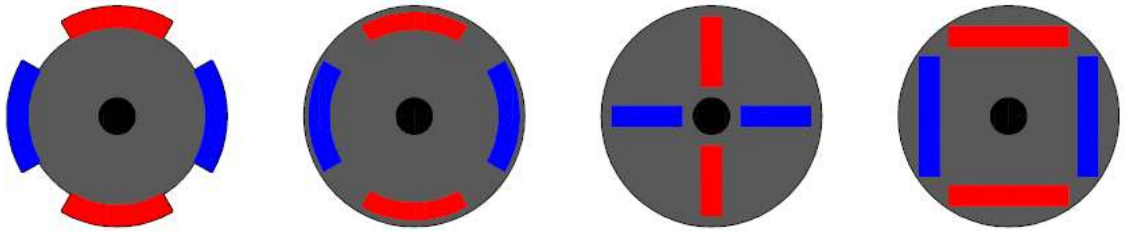
Şekil 3.4. Motor yapısına göre farklı SMSM çeşitleri

3.1.3. Radyal Akılı Motorlar

Radyal akı, makinenin alan akısının yönünün mile dik olduğu anlamına gelir. Elektrik motorlarının çoğu radyal akı yapısına sahiptir. Motorların yığın boyutunun arttırılmasıyla daha yüksek güç çıkışları elde edilebilir.

İç Rotor Yapısı

Rotoru içeride bir SMSM’de, rotor statorun içine yerleştirilir ve rotor mili iki kapak üzerindeki yataklara sabitlenir. Kalıcı mıknatıslar radyal olarak mıknatıslanır ve dönüşümlü olarak kutuplanır [64]. Mıknatıs kutup alanlarının rotor yüzeyindeki kutuplardan daha küçük olması nedeniyle açık devre akı yoğunluğu, mıknatısın kendi akı yoğunluğundan daha azdır. Bu yapıda mıknatıslar merkezkaç kuvvetlerine karşı korunur ve bu sayede makine, mıknatısların çıkma riski olmadan çok yüksek hızlarda çalışabilir. Rotorun küçük çapı nedeniyle ataletleri küçüktür. Bu durum, hızlanma ve geçici tepki açısından daha iyi dinamik performansa yol açar. Stator sargıları statorda açılmış oluklara yerleştirildiği için basit soğutma mekanizmaları ile makinenin soğuması sağlanır. Kalıcı mıknatısların rotora yerleştirilmesi makinenin özelliklerini etkiler. Böylece Şekil 3.5’te verilen farklı rotor tasarımları mevcuttur. En çok kullanılan çeşitler, yüzey, kollu ve gömülü sabit mıknatıslı rotorlardır.

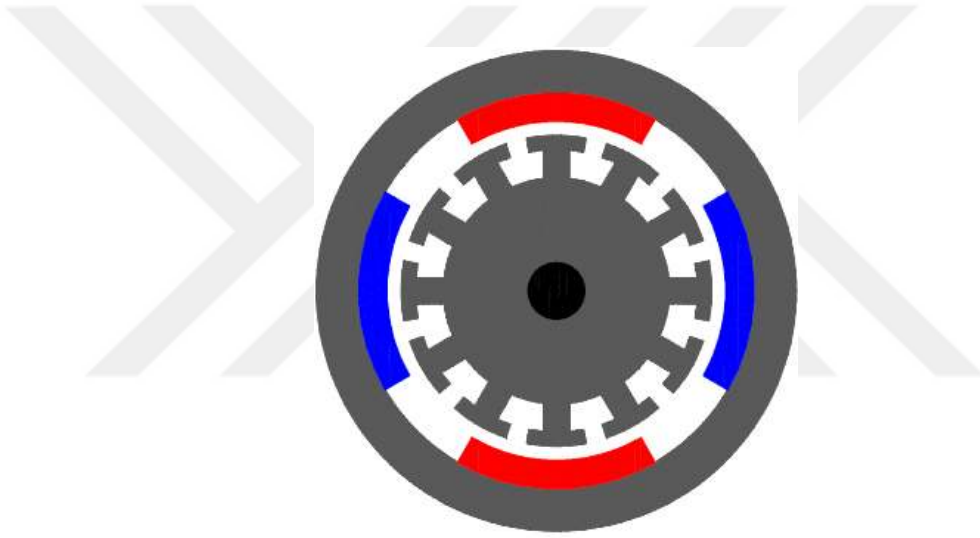


Şekil 3.5. a) YMSMSM b) DSMSM c) Spoke SMSM d) Bariyer SMSM rotor çeşitleri [64].

İç rotor SMSM genellikle yüksek çıkış gücünün, hızlı geçiş tepkisinin gerekli olduğu ve daha yüksek hızların istendiği uygulamalarda kullanılır. Bunlar endüstriyel servo sistemler, pompalar, çekiş sistemleri ve EA’larda tercih edilirler.

Dış Rotor Yapısı

Dış rotor yapıları motorlarda stator makinesinin içinde bulunurken rotor makinesinin dış yüzeyine sabitlenir. Rotor genellikle, mıknatısları demanyetizasyona ve merkezkaç kuvvetlerine karşı koruyan ve aynı zamanda asenkron bir başlangıç momentini sağlayan, harici yüksek iletkenliğe sahip ferromanyetik olmayan bir silindir kullanır. Bu silindir aynı zamanda bir damper görevi görür ve çıkışın salınımlarını iyileştirir. Bir dış rotor SMSM'nin yapısı Şekil 3.6'da verilmiştir.



Şekil 3.6. Bir dış rotor SMSM'nin yapısı [64].

Büyük rotor çapı nedeniyle bu tip motorlar, iç rotor tiplerine göre daha yüksek ataletle sahiptir. Bu özellik, hız açısından çok hassas kontrol ve sorunsuz çalışma sağlar. Dış rotor tipi SMSM, genellikle sabit disk ve kompakt disk sürücüler ve dronelar gibi küçük dalgalanmalarla hassas hız kontrolü gerektiren uygulamalarda kullanılır. Bu tip motorlar, sürücü sistemlerinin hacminin azaltıldığı, yükün doğrudan sürülmesinin tercih edildiği uygulamalarda da kullanılmaktadır. Bu uygulamalar elektrikli bisikletler, scooterlar ve yük taşıyıcı araçlar gibi hafif EA motorlarını içerir.

3.1.4. Eksenel Akı Motorlar

Eksenel akı makinaları, şaftın eksenine paralel olarak kalıcı mıknatıs akı yönüne sahiptir. Eksenel motorlar, radyal akı motorlara kıyasla daha yüksek moment yoğunluğuna ve daha

küçük hacime sahiptir. Düşük hızlarda yüksek moment sağlayabilirler. Esasen düşük hız ancak yüksek moment çalışması isteyen uygulamalarda redüktör kullanımını ortadan kaldırırlar. Bu uygulamalar arasında çekiş motorları ve EA'lar bulunmaktadır. Eksenel akılı motorlar da rotorun yerleşimine göre ikiye ayrılır. Şekil 3.7'de, iki rotor arasına yerleştirilmiş stator ile eksenel bir SMSM yapısını göstermektedir.



Şekil 3.7. Eksenel SMSM'nin yapısı [64].

3.2. Sürekli Mıknatıslı Senkron Motorların Çalışma Prensibi

Senkron motorlar, kaynağın frekansına ve motorun kutup sayısına bağlı olarak her zaman senkron hızda dönen sabit hızlı makinalardır. Statorlarına AA, rotorlarına ise DA uygulanır. Böylece iki ayrı manyetik alan üretildiğinden çift uyarmalıdır [62].

Fırçalı DA motorlarında, fırça kolektör yapısı ile rotor akımının yönü sürekli değiştirilerek düzgün bir moment elde edilir. Fırçasız DA motoru ve SMSM'lerde ise rotorda yer alan kalıcı mıknatıslar sayesinde manyetik alan oluşturulur. Bu da önemli avantajlar sağlar. Uyarma sargıları olmadığından kayıpları azaltır. Dolayısıyla verimleri yüksektir. Boyutları küçük olduğundan moment/hacim oranları daha yüksektir. Ayrıca, fırça kolektör bulunmadığı için de yapıları daha basittir ve bakım masrafları azdır.

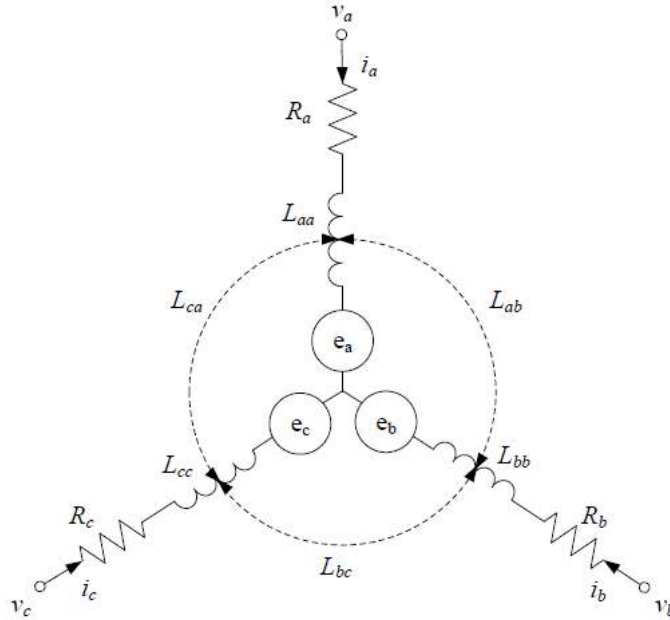
3.3. Sürekli Mıknatıslı Senkron Motorların Dinamik Modeli

SMSM'nin statoru ve rotoru sargılı senkron makinaya benzerdir. SMSM'de kullanılan kalıcı mıknatıslar, yüksek dirençli modern bir nadir toprak çeşididir. Bu nedenle rotorda indüklenen akımlar ihmal edilebilir. Ayrıca, kalıcı bir mıknatıs tarafından üretilen zıt EMK ile uyarılmış bir bobin tarafından üretilen zıt EMK arasında hiçbir fark yoktur. Dolayısıyla bir

SMSM'nin matematiksel modeli, rotoru sargılı senkron motora benzerdir ve aşağıdaki varsayımlar dikkate alınarak elde edilir [59,63]:

- 1) Doyma ihmal edilir.
- 2) Endüklenen EMK sinüzoidaldir.
- 3) Girdap akımları ve histerezis kayıpları ihmal edilebilir düzeydedir.
- 4) Alan akımı dinamiği yoktur.
- 5) Rotorda kafes yoktur.
- 6) Akımın deri etkisi ihmal edilmiştir.
- 7) Endüktans ve direnç değerleri sıcaklık ve frekansa göre değişiklik göstermez.

Y-bağlı üç fazlı bir SMSM'nin üç faz modeli Şekil 3.8'de verilen eşdeğer devresi kullanılarak çıkartılmıştır.



Şekil 3.8. Bir SMSM'nin üç faz eşdeğer devresi [63].

Şekil 3.8'deki eşdeğer devre kullanılarak, SMSM'nin üç faz gerilimleri Denklem 3.1'de gösterilmiştir.

$$\begin{bmatrix} V_a \\ V_b \\ V_c \end{bmatrix} = \begin{bmatrix} R_a & 0 & 0 \\ 0 & R_b & 0 \\ 0 & 0 & R_c \end{bmatrix} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} + \begin{bmatrix} L_{aa} & L_{ab} & L_{ca} \\ L_{ab} & L_{bb} & L_{bc} \\ L_{ca} & L_{bc} & L_{cc} \end{bmatrix} \frac{d}{dt} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} + \begin{bmatrix} e_a \\ e_b \\ e_c \end{bmatrix} \quad (3.1)$$

Burada, V_a, V_b, V_c , a, b, c fazlarına ilişkin stator gerilimleri, i_a, i_b ve i_c stator akımları, R_a, R_b ve R_c faz sargı dirençleri, e_a, e_b ve e_c statorda indüklenen zıt EMK'lar, L_{aa}, L_{bb} ve L_{cc} faz sargılarının toplam indüktansları, L_{ab}, L_{bc} ve L_{ca} fazları arasındaki karşılıklı indüktanslardır. R_a, R_b ve R_c faz dirençleri birbirine eşittir ve R_s ile ifade edilebilir.

Denklem 3.2'de verilen gerilim denklemi ψ_a, ψ_b, ψ_c manyetik akıları ile yeniden düzenlenirse;

$$\begin{bmatrix} V_a \\ V_b \\ V_c \end{bmatrix} = \begin{bmatrix} R_a & 0 & 0 \\ 0 & R_b & 0 \\ 0 & 0 & R_c \end{bmatrix} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} + \frac{d}{dt} \begin{bmatrix} \psi_a \\ \psi_b \\ \psi_c \end{bmatrix} \quad (3.2)$$

L_{ss} faz sargısı toplam endüktansı,

$$L_{ss} = L_{aa} = L_{bb} = L_{cc} = L_l + L_{ms} \quad (3.3)$$

$$M = L_{ab} = L_{bc} = L_{ca} = -\frac{1}{2} L_{ms} \quad (3.4)$$

Olarak bulunabilir. Burada L_l kaçak endüktans ve L_{ms} faz sargısı öz endüktansıdır. Fazlar arasındaki karşılıklı endüktanslar M ile ifade edilmiştir.

Bu indüktanslar kullanılarak a, b, c faz akıları Denklem 3.5'teki gibi yazılabilir.

$$\begin{bmatrix} \psi_a \\ \psi_b \\ \psi_c \end{bmatrix} = \begin{bmatrix} L_{ss} & M & M \\ M & L_{ss} & M \\ M & M & L_{ss} \end{bmatrix} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} + \begin{bmatrix} \lambda_f \cos \theta_r \\ \lambda_f \cos (\theta_r - \frac{2\pi}{3}) \\ \lambda_f \cos (\theta_r - \frac{4\pi}{3}) \end{bmatrix} \quad (3.5)$$

Burada λ_f zıt EMK sabitini, θ_r ise rotor pozisyonunu gösterir.

Gerilim denklemi manyetik akılara göre yeniden düzenlenirse;

$$\begin{bmatrix} V_a \\ V_b \\ V_c \end{bmatrix} = \begin{bmatrix} R_s & 0 & 0 \\ 0 & R_s & 0 \\ 0 & 0 & R_s \end{bmatrix} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} + \frac{d}{dt} \begin{bmatrix} L_{ss} & M & M \\ M & L_{ss} & M \\ M & M & L_{ss} \end{bmatrix} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} + \begin{bmatrix} \lambda_f \cos \theta_r \\ \lambda_f \cos (\theta_r - \frac{2\pi}{3}) \\ \lambda_f \cos (\theta_r - \frac{4\pi}{3}) \end{bmatrix} \quad (3.6)$$

Bu ifadelerin zamana göre türevi alındığında;

$$\begin{bmatrix} V_a \\ V_b \\ V_c \end{bmatrix} = \begin{bmatrix} R_s & 0 & 0 \\ 0 & R_s & 0 \\ 0 & 0 & R_s \end{bmatrix} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} + \begin{bmatrix} L_{ss} & M & M \\ M & L_{ss} & M \\ M & M & L_{ss} \end{bmatrix} \frac{d}{dt} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} - \omega_r \begin{bmatrix} \lambda_f \sin \theta_r \\ \lambda_f \sin (\theta_r - \frac{2\pi}{3}) \\ \lambda_f \sin (\theta_r - \frac{4\pi}{3}) \end{bmatrix} \quad (3.7)$$

Burada sinüs ifadeleri k_a, k_b, k_c katsayıları ile gösterilirse;

$$\begin{bmatrix} k_a \\ k_b \\ k_c \end{bmatrix} = - \begin{bmatrix} -\lambda_f \sin \theta_r \\ -\lambda_f \sin (\theta_r - \frac{2\pi}{3}) \\ -\lambda_f \sin (\theta_r - \frac{4\pi}{3}) \end{bmatrix} \quad (3.8)$$

Buna göre zıt EMK'ler;

$$\begin{bmatrix} e_a \\ e_b \\ e_c \end{bmatrix} = \omega_r \begin{bmatrix} k_a \\ k_b \\ k_c \end{bmatrix} \quad (3.9)$$

Zıt EMK denklemi kullanılarak faz gerilimleri tekrar düzenlenirse;

$$\begin{bmatrix} V_a \\ V_b \\ V_c \end{bmatrix} = \begin{bmatrix} R_s & 0 & 0 \\ 0 & R_s & 0 \\ 0 & 0 & R_s \end{bmatrix} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} + \begin{bmatrix} L_{ss} & M & M \\ M & L_{ss} & M \\ M & M & L_{ss} \end{bmatrix} \frac{d}{dt} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} + \omega_r \begin{bmatrix} k_a \\ k_b \\ k_c \end{bmatrix} \quad (3.10)$$

SMSM'in üç fazlı eşdeğer devresi dengeli bir dağılım gösterdiğinden akım denklemi Denklem 3.11'deki gibi yazılabilir.

$$i_a + i_b + i_c = 0 \quad (3.11)$$

Buna göre Denklem 3.10 akımlara göre yeniden düzenlenirse;

$$\begin{bmatrix} V_a \\ V_b \\ V_c \end{bmatrix} = \begin{bmatrix} R_s & 0 & 0 \\ 0 & R_s & 0 \\ 0 & 0 & R_s \end{bmatrix} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} + \begin{bmatrix} (L_{ss} - M) & 0 & 0 \\ 0 & (L_{ss} - M) & 0 \\ 0 & 0 & (L_{ss} - M) \end{bmatrix} \frac{d}{dt} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} + \omega_r \begin{bmatrix} k_a \\ k_b \\ k_c \end{bmatrix} \quad (3.12)$$

Burada L_s ,

$$L_s = L_{ss} - M = L_1 + L_{ms} - (-\frac{1}{2}L_{ms}) = L_1 + \frac{3}{2}L_{ms} \quad (3.13)$$

Buna göre a, b, c fazları için akım ifadeleri yazılırsa;

$$\frac{d}{dt} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} = \begin{bmatrix} L_s & 0 & 0 \\ 0 & L_s & 0 \\ 0 & 0 & L_s \end{bmatrix}^{-1} \left[\begin{bmatrix} -R_s & 0 & 0 \\ 0 & -R_s & 0 \\ 0 & 0 & -R_s \end{bmatrix} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} - \omega_r \begin{bmatrix} k_a \\ k_b \\ k_c \end{bmatrix} + \begin{bmatrix} V_a \\ V_b \\ V_c \end{bmatrix} \right] \quad (3.14)$$

Motor hızı;

$$\frac{d}{dt} \omega_r = \frac{p}{2} (T_e - T_L - B(\frac{2}{p}) \omega_r) / J \quad (3.15)$$

Rotor konumu;

$$\frac{d}{dt} \theta_r = \omega_r \quad (3.16)$$

Burada ω_r rotorun elektriksel hızı, p kutup sayısı, T_e elektriksel moment, T_L yük moment, J atalet moment ve B sürtünme katsayısıdır.

SMSM'nin rotor referans çerçevesindeki stator d-q denklemleri;

$$V_q = R i_q + p \lambda_q + \omega_s \lambda_d \quad (3.17)$$

$$V_d = R i_d + p \lambda_d + \omega_s \lambda_q \quad (3.18)$$

$$\lambda_q = L_q i_q \quad (3.19)$$

$$\lambda_d = L_d i_d + \lambda_{af} \quad (3.20)$$

Burada V_d , V_q d-q eksen gerilimleridir. i_d ve i_q d-q eksen stator akımları, L_d ve L_q d-q eksen indüktansları, λ_d ve λ_q d-q eksen stator akı bağıntıları, R ve ω_s sırasıyla stator direnci ve senkron hızdır. λ_{af} statoru halkalayan rotor mıknatısları nedeniyle oluşan akıdır.

Elektriksel moment;

$$T_e = 3p \left[\lambda_{af} i_q + (L_d - L_q) i_d i_q \right] / 2 \quad (3.21)$$

Motor dinamikleri için moment denklemi;

$$T_e = T_L + B \omega_r + j p \omega_r \quad (3.22)$$

p kutup çifti sayısı, T_L yük momenti, B sürtünme katsayısı, ω_r rotor açısal hızı ve j atalet momentidir. Senkron hız, rotor hızı ile Denklem 3.23'teki gibi ilişkilidir.

$$\omega_s = p \omega_r \quad (3.23)$$

Makine modeli, i_d ve i_q ile hız gibi terimleri içerdiğinden doğrusal değildir. ω_r , i_q ve i_d 'nin durum değişkenleridir. Dinamik simülasyon için SMSM denklemleri durum uzayı biçiminde Denklem 3.24-3.26 ile ifade edilmelidir.

$$p i_d = (V_d - R i_d + \omega_s L_q i_q) / L_d \quad (3.24)$$

$$p i_q = (V_q - R i_q - \omega_s L_d i_d - \omega_s \lambda_{af}) / L_q \quad (3.25)$$

$$p \omega_r = (T_e - T_L - B \omega_r) / j \quad (3.26)$$

d, q değişkenleri, Denklem 3.27'de tanımlanan Park dönüşümü ile a, b, c değişkenlerinden elde edilir.

$$\begin{bmatrix} V_q \\ V_d \\ V_0 \end{bmatrix} = 2/3 \begin{bmatrix} \cos \theta & \cos (\theta - \frac{2\pi}{3}) & \cos (\theta + \frac{2\pi}{3}) \\ \sin \theta & \sin (\theta - \frac{2\pi}{3}) & \sin (\theta + \frac{2\pi}{3}) \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix} \begin{bmatrix} V_a \\ V_b \\ V_c \end{bmatrix} \quad (3.27)$$

a, b, c değişkenleri, Denklem 3.28'de tanımlanan Park dönüşümünün tersi yoluyla d, q değişkenlerinden elde edilir.

$$\begin{bmatrix} V_a \\ V_b \\ V_c \end{bmatrix} = \begin{bmatrix} \cos \theta & \sin (\theta) & 1 \\ \cos (\theta - \frac{2\pi}{3}) & \sin (\theta - \frac{2\pi}{3}) & 1 \\ \cos (\theta + \frac{2\pi}{3}) & \sin (\theta + \frac{2\pi}{3}) & 1 \end{bmatrix} \begin{bmatrix} V_q \\ V_d \\ V_0 \end{bmatrix} \quad (3.28)$$

$$P_T = v_a i_a + v_b i_b + v_c i_c \quad (3.29)$$

d-q düzlemine göre güç;

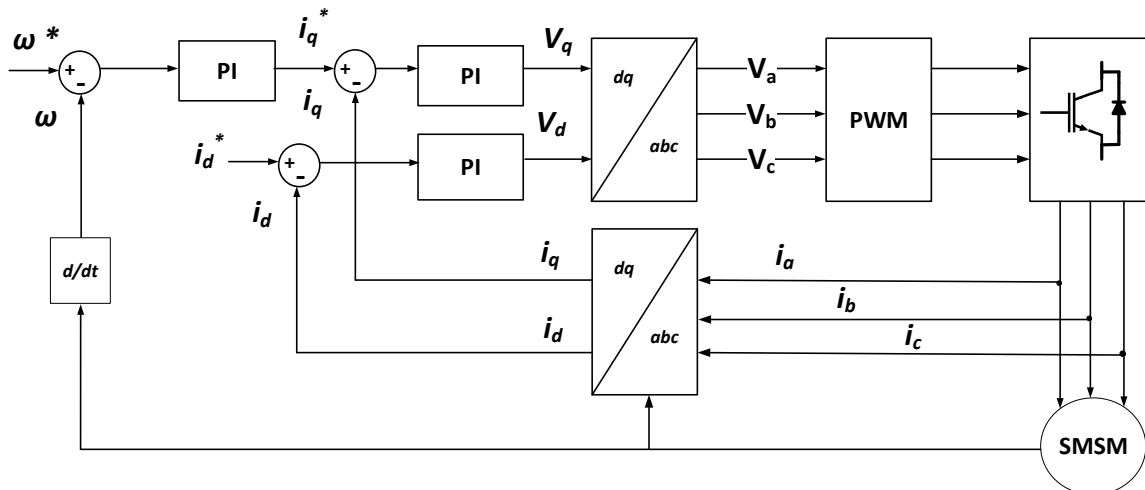
$$P_T = 3(v_d i_d + v_q i_q)/2 \quad (3.30)$$

ile hesaplanabilir.

3.4. Sürekli Mıknatıslı Senkron Motorların Kontrolü

Bir DSMSM'nin motor hız kontrolünün blok diyagramı Şekil 3.9'da gösterilmiştir. Burada referans hız işaretiyle (*) motorun dönmesi istenilen açısal hız değeri gösterilir. Dolayısıyla moment ayarı yapılabilmektedir.

Sistemde referans hız değeri gerçek hız ile karşılaştırılarak, hata değeri PI kontrolörden geçirilir ve i_q akımının referans değeri i_q^* bulunur. i_q^* , gerçek akım değeri olan i_q ile karşılaştırılır ve hata akımı PI kontrolöründen geçirilerek V_q gerilim sinyali elde edilir. Aynı şekilde referans akımı i_d^* , gerçek akım değeri olan i_d ile karşılaştırılarak hata akımı PI kontrolöründen geçirilir ve V_d gerilim sinyali elde edilir. Bulunan V_d ve V_q gerilimleri, motora gerekli olan üç faz gerilim sinyallerini elde etmek için Ters Park ve Clarke dönüşümleri ile a, b, c gerilimlerine dönüştürülür. Böylece, PWM sinyalleri üretilerek gerilim kaynaklı eviriciye referans olarak verilir ve motorun referans hızı takip etmesi sağlanır [65].



Şekil 3.9. Bir DSMSM'nin motor hız kontrol şeması

Bu bölümde, bu tez çalışmasında kullanılan SMSM'lerin yapısı, çeşitleri, çalışma prensibi ve dinamik modeli anlatılmıştır. Bir sonraki bölümde ise araçlarda ve endüstride kullanılan ağ sistemlerinden bahsedilecektir.



4. ARAÇLARDA VE ENDÜSTRİDE AĞ SİSTEMLERİ

Son yıllarda elektronik teknolojisindeki gelişmeler otomotiv alanına önemli şekilde yansımıştır. Artan bu gelişmelerle, araçlarda motor kontrol ünitesinin varlığı kaçınılmaz olmaktadır. Motor kontrol ünitesi, otomobillerde yüksek performanslı motorların kullanılmaya başlanmasını sağlayarak, daha ekonomik, çevreye duyarlı ve güvenli sürüş gibi olumlu etkiler doğurmuştur [66].

Günümüzde EA'larda kullanılan elektronik motor yönetim sistemleri verimi önemli ölçüde arttırmaktadır. Motor için gerekli yakıt miktarını ayarlayarak, zararlı gazların yayılımını ve gereksiz yakıt tüketimini de azaltır. Elektronik kontrol üniteleriyle aynı zamanda konfor ve güvenlik birimleri ile veri alışverişi de sağlanmaktadır.

4.1. Motor Yönetim Sistemleri

Motor yönetim sistemleri, aracın yakıt tüketimini azaltan ve motor performansını arttırarak verimli çalışmasını sağlayan sistemlerdir. Bu sistemin en önemli birimi Elektronik Kontrol Ünitesi'dir (EKÜ). EKÜ, bir araçtaki bir veya daha fazla elektrik sistemini veya alt sistemi kontrol eden bir elektronik birimdir. Bir mikroişlemci ile buna bağlı giriş ve çıkış sinyallerinin taşındığı bağlantılardan oluşur. Sensörlerden gelen bilgiler burada işlenir ve buna göre alt sistemlere iletilir. Sensörlerden aldığı yakıt, hava, soğutma suyu sıcaklıkları gibi çeşitli bilgileri değerlendirerek yakıt miktarını ayarlar [66]. Böylece verimi arttırarak egzoz gazlarının yayılımını azaltır. Ayrıca, arıza tespitinde de önemli rol oynar. Bu sistemlerin tek olumsuz yanı pahalı olmalarıdır.

EKÜ'nün, motor kontrol modülü, güç aktarım sistemi kontrol modülü, şanzıman kontrol modülü, fren kontrol modülü, merkezi kontrol modülü, merkezi zamanlama modülü, genel elektronik modül ve süspansiyon kontrol modülü gibi çeşitleri vardır. Bu sistemler aracın bilgisayarı olarak düşünülebilir. Bir araçta kullanılan EKÜ'lerin sayısı arttıkça buna bağlı olarak yazılımın karmaşıklığı da artmaktadır.

4.1.1. EKÜ'nün Yapısı ve Çalışması

EKÜ, bir mikroişlemci, giriş çıkış üniteleri, transistörler ve hafıza biriminden oluşmaktadır. Hafıza birimi, bilgilerin kaydedildiği bölümdür. Bu bellekler temelde iki kısma ayrılır:

- **Random Access Memory (RAM):** Rastgele erişimli belleklerdir. Bu bellekte enerji kesildiği zaman bilgi kaybolur. Programlama esnasında yazma ve okuma işlemlerinin yerine getirilmesinde kullanılır.

- **Read Only Memory (ROM):** Sadece okunabilir belleklerdir. Bu bellek de kendi içinde alt gruplara ayrılır:
 - **Erasable Programmable Read Only Memory (EPROM):** Silinebilir programlanabilir salt okunur bellektir. Sensörlerden gelen anlık veriler EPROM'a kaydedilir.
 - **Electronically Erasable and Programmable Read Only Memory (EEPROM):** Elektronik olarak silinebilir ve programlanabilir salt okunur bellektir. EKÜ'ye yazılım yüklenmesi bu belleğe yapılır.

Boyutları oldukça küçük olan EKÜ'ler mikrobilgisayarlardır ve EPROM içerisinde bulunan programları çalıştırmaktadır. Sensörlerden basınç, sıcaklık, hız, hava yoğunluğu gibi veriler sürekli olarak alınmaktadır. Alınan bu analog bilgiler sayısala dönüştürülerek EKÜ'ye iletilir. EKÜ, EPROM içerisinden ilgili değerleri bulur ve motorun çalışmasını denetleyerek sistemi kontrol eder. Daha sonra EKÜ gerekli bilgileri, sayısalan analoğa dönüştürerek ilgili kısımlara sinyal gönderir. Tüm bu işlemler çok kısa sürede gerçekleşir. Sensörlerin çalışmadığı durumda, EKÜ kayıtlı olan yedek bilgilerle görevini yapar. Herhangi bir arızada ise sürücüyü uyararak arıza tespitinde de önemli rol oynar.

4.2. Veri İletim Sistemleri

Sayısal bilgilerin bir noktadan başka bir noktaya aktarılmasına veri iletimi denir. Veri iletim sistemleri, bilgisayarlar, bilgisayarlar ile terminaller veya alıcılar arasında veri iletimini gerçekleştirir [67]. Veri iletiminde sayısal tekniklerin kullanılması, yüksek verim ve düşük maliyet gibi önemli avantajlar sağlar. Sayısal verilerin iletiminde seri ve paralel olmak üzere iki adet veri iletim yöntemi kullanılmaktadır. Bu iletim yöntemlerinde aşağıdaki parametreler önemlidir [68]:

- Doğru hedef: Verinin sadece doğru hedefe ulaşmasıdır.
- Doğruluk: Verinin kaynağından çıktığı haliyle iletilmesidir.
- Zaman: Verinin zamanında hedefe ulaşmasıdır. Gerçek zamanlı iletişimde oldukça önemlidir.
- Gecikme değişimi: Veri paketlerinin hedefe ulaşma süresindeki değişimdir.

Veri iletim sistemi beş birimden oluşur. Bunlar:

- Mesaj: Ses, görüntü gibi iletilen bilgidir.
- Gönderici: Video kamera gibi veriyi ileten cihazlardır.
- Alıcı: Televizyon, bilgisayar gibi veriyi alan cihazlardır.
- İletim ortamı: Alıcı ve gönderici arasında verinin fiber optik kablo, burgulu tel çifti, koaksiyel kablo gibi çeşitli yollarla iletilmesini sağlarlar.
- Protokol: Veri iletim kurallarını tanımlayan kısımdır.

4.2.1. Seri Haberleşme

Seri haberleşmede bilgi aktarılırken aslında paralel veri yongaları yani bitler olarak üretilir. Ancak bu bilgi tek bir veri hattı üzerinden iletilir. Bilginin bu dönüşüm işlemine paralel-seri veri dönüşümü denir. Bu dönüşümde veri iletim hızı yani bit/s oranı (baud-rate) önemlidir.

Seri veri iletiminde kontrol bilgileri aynı hat üzerinden taşınır. Böylece birden fazla kablo kullanımı önlenmiş olur. Veriler taşınırken verinin kodlanması, gönderilmesi, çözülmesi ve kontrol işlevinin yerine getirilmesi belirli bir süre alır [68]. Bu sebepten ötürü seri haberleşme, paralel haberleşmeye göre daha yavaş gibi görünebilir. Ancak bu gecikme, milisaniyeler mertebesinde olduğu için pratikte ihmal edilebilir.

4.2.2. Paralel Haberleşme

Paralel haberleşmede, alıcı ve verici tarafında 8, 16, 32 ve 64 adet gibi farklı bağlantı noktaları bulunur. 8 adet bağlantı ile 1 byte yani 8 bitlik bir veri aktarımı gerçekleşir. Bir PLC ünitesinin 8 adetlik çıkış modülü vericiyi oluşturur. Bu 8 adet vericiye karşılık 8 adet alıcı bulunur. Konvansiyonel paralel kablolama kullanılır ve PLC ünitesinin çıkışları ayrı ayrı bir selenoid valfi kumanda eden röle kontaklarını kumanda eder [68]. Bu durum, çıkış sayısı arttığında bir sorun oluşturacaktır. Bir verici ucunda sinyal olduğunda kumanda edilen selenoid valf, 24 V DA enerji ile çalıştırılarak istenilen işlevi gerçekleştirecektir. Ayrıca programın içeriğine, gönderilen sinyalin tipine ve süresine göre valflerin enerjide kalma süresi değiştirilebilecektir.

4.3. Haberleşme Protokolleri

Haberleşme protokolü, alıcı ve verici arasında hızlı, güvenilir ve anlaşılabilir bir veri iletişimi gerçekleştirebilmek için haberleşme ile ilgili tüm kuralların detaylı tanımlanmasıdır. Alıcı ve verici cihazların belirli bir protokol üzerinden haberleşecek şekilde üretilmiş olması gerekir. Protokol, tüm cihazların bir sistem üzerinden haberleştiği ve ona göre anlaşarak işlevlerini yerine getirdiği ortak bir konuşma dilidir.

Haberleşme protokolleri Tablo 4.1’de gösterildiği gibi, açık ve kapalı sistem olmak üzere iki kategoriden oluşur. Açık sistem haberleşme protokolü, markaya bağlı olmadan aynı hedef için üretilmiş tüm cihazların birbiriyle haberleşebilmesidir. Örneğin, Omron ve Phonix firmasının Profibus ürünleri birbirleriyle sıkıntısız haberleşebilir. Kapalı sistem haberleşme protokolü ise her üreticinin sadece kendi ürünleri arasında haberleşebildiği protokoldür. Mitsubishi Melsec, Omron Phoenix Contact Interbus-S Compobus ürünleri bunlara örnek verilebilir.

Tablo 4.1. Açık ve kapalı haberleşme sistemleri [69].

Protokol	Tanımlama	Üretici Firma
Açık Haberleşme Sistemi	Interbus-S	Phoenix
	Profibus	Pno/Siemens
	Canopen	CIA
	Devicenet	Allen-Bradley
	As-ara yüzü	ASI Verein/Siemens
	FIP	AEG-Schneider
	CAN	Automation
	LON	Intel/Motorolla/Phillips Echelon/Motorolla
Kapalı Haberleşme Sistemi	Beckhoff I/O	Beckhoff
	Melsec	Mitshubishi
	Suconet K	Klockner Moeller
	ET 100	Siemens
	CS 31	ABB
	Remote I/O Modnet	Alen Bradley

Bu haberleşme sistemlerinin dışında SPI, I²C, Evrensel Asenkron Alıcı/Verici (Universal Asynchronous Receiver/Transmitter, UART), USB gibi pek çok seri haberleşme protokolleri bulunur [70]. Bu protokoller özellikleri bakımından Tablo 4.2’de karşılaştırılmıştır.

Tablo 4.2. Seri haberleşme protokollerinin özellikleri [71].

BUS	Pozitif Yönleri	Negatif Yönleri
UART	Yaygın kullanım, Düşük maliyetli ve basit.	Sınırlı işlevsellik, Noktadan noktaya haberleşme yapar.
CAN	Güvenli, hızlı, az kablo kullanımı.	Kompleks yapıya sahiptir.
USB	Hızlı, basit ve düşük maliyetli.	Güçlü master ve özel sürücüler gerektirir.
SPI	Hızlı, evrensel olarak kabul edilmiş ve düşük maliyetli.	Belirlenmiş bir standardı yoktur.
I ² C	Basit, yaygın kullanım, düşük maliyet.	Sınırlı hıza sahiptir.

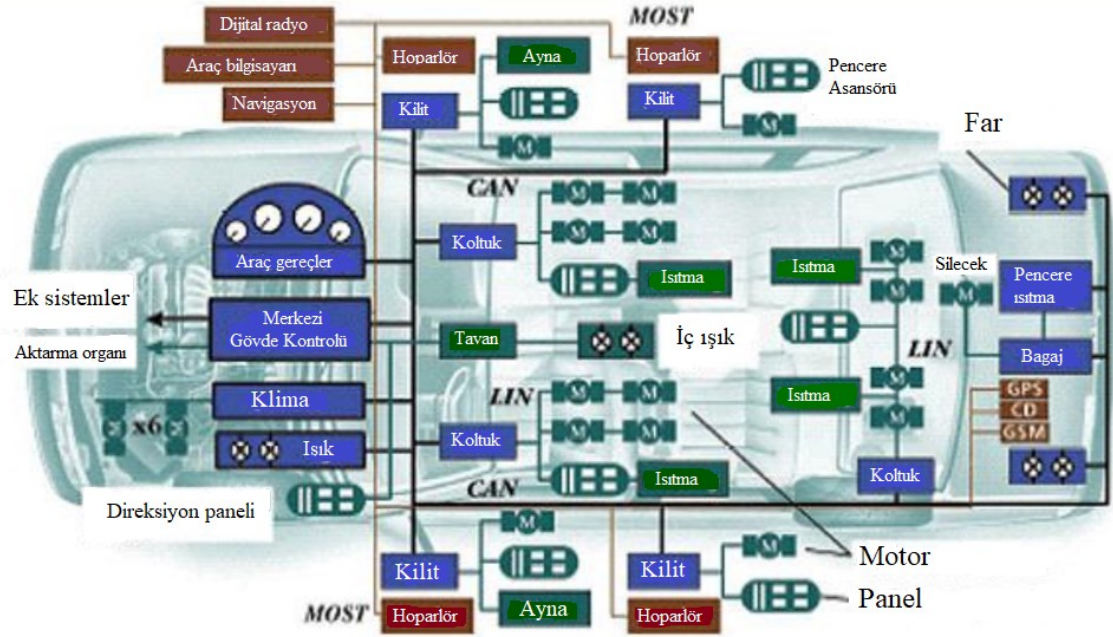
En yaygın kullanılan seri haberleşme protokolleri CAN ve UART’tır. UART protokolü, noktadan noktaya haberleşme yapar. Düşük maliyetli olduğu için tercih edilir. En çok kullanılan çeşitleri RS232 ve RS485’tir. Bu protokolün bazı dezavantajları vardır. Bunlar:

- Byte tabanlı haberleşme yaptığından aynı anda birden fazla byte gönderilemez.
- Noktadan noktaya haberleşme yapması kablo kullanımını arttırır. Bu da araç ağırlığının ve dolayısıyla maliyetin artmasına neden olur.
- Hata kontrolü yoktur. Bunun için ayrı bir yazılım kullanılması gerekir. Bu da sistemin yavaşlamasına sebep olmaktadır.

UART protokolünün getirdiği bu olumsuzlukları ortadan kaldırmak için CAN haberleşme protokolü geliştirilmiştir. CAN protokolü, çoklu master özelliğiyle özellikle otomotiv ve endüstriyel uygulamalarda EKÜ'leri ve sensörleri haberleştirirken yaygın şekilde kullanılırlar.

4.4. Elektronik Kontrol Üniteleri Arasında Haberleşme

Araçlarda kullanılan EKÜ'ler, sensörlerden aldığı bilgilere göre yönettikleri sistemlerin çalışmasını düzenlerken birbirleri ile koordineli şekilde çalışmalıdır. EKÜ'lerin birbirleriyle haberleşmelerini sağlamak amacıyla üç çeşit haberleşme sistemi kullanılır. Bunlar; Kontrol Alan Ağı (Controller Area Network, CAN), Yerel Bağlantı Ağı (Local Interconnect Network, LIN) ve Medya Odaklı Sistemleri Taşıma (Media Oriented Systems Transport, MOST) Ağıdır. Şekil 4.1'de görüldüğü gibi motor kısmında CAN, konfor donanımında LIN, eğlence ve bilgilendirme kısmında ise MOST hattı kullanılır.



Şekil 4.1. Haberleşme sistemlerinin kullanım yerleri [72].

4.4.1. CAN-Bus Haberleşme Protokolü

Açılımı “Controller Area Network-Bus (CAN-Bus)” olan Kontrol Alan Ağı, 1983 yılında Robert Bosch GmbH tarafından otomotiv endüstrisi için geliştirilmiş bir haberleşme protokolüdür. Başlangıçta Avrupa'da üretilen otomobiller için geliştirilmiş olsa da mobil otomasyon ve daha birçok alanda kullanılmıştır. Bosch, CAN-Bus için çeşitli standartlarını yayınlamıştır. Bunlardan en sonuncusu 1991 yılında yayınlanan CAN 2.0'dır. CAN 2.0'ın A ve B olmak üzere iki versiyonu vardır. CAN 2.0A, düşük hızlı haberleşme olup ISO 11519 standardındadır. A bölümü, 11 bitlik veri tanımlayıcıyı ifade eder. CAN 2.0B ise yüksek hızlı

olup, ISO 11898 standardında belirlenmiştir. Burada B, 29 bitlik veri tanımlanabilen genişletilmiş versiyondur.

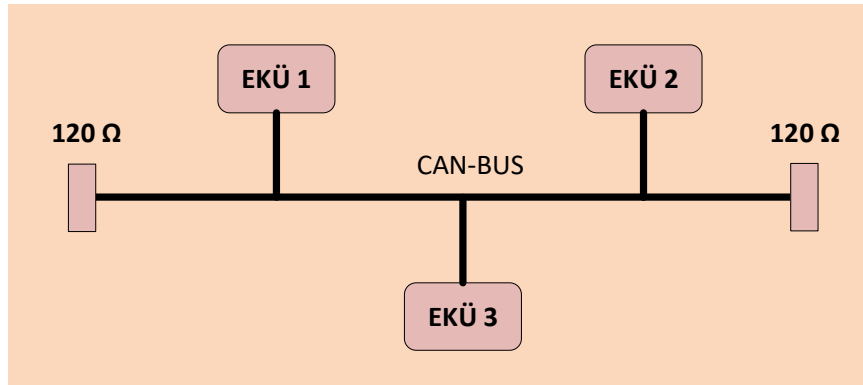
CAN-Bus, otomobiller, sensörler, asansörler, üretim ve dokuma makineleri, medikal cihazlar, makine kontrol birimleri, bina otomasyonu, elektrikli sandalyeler gibi pek çok alanda yaygın şekilde kullanılırlar. CAN-Bus protokolü, üstün özelliklerinden dolayı özellikle otomotiv alanında tercih edilirler. Bu özellikler aşağıdaki gibi sıralanabilir:

- **Mesaj Yönlendirme:** Hatta mesajlar gönderilirken her modülün bir ID numarası vardır. Böylece hangi modülden mesaj geldiği bellidir ve ona göre yönlendirilir.
- **Mesaj Önceliği:** Birden fazla ünite CAN-Bus hattına aynı anda mesaj gönderirse yüksek öncelikli mesaj önce gönderilir. ID numarasına göre hatta mesaj gönderme önceliği vardır. ID numarası düşük mesajlar daha önceliklidir. Daha düşük öncelikli mesaj geri çekilerek hattı bekler. Hat boşaldıktan sonra mesajını gönderir. Böylece veri iletimi hızlı ve kayıplar olmadan yapılabilir.
- **Haberleşme Hızı:** CAN-Bus hattının haberleşme hızı 5 kbit/s ile 1 Mbit/s arasında değişmektedir. 1 Mbit/s hızına çıkabildiğinden hızlı bir haberleşme protokolüdür. Ağın büyüklüğüne ve mesafelere göre haberleşme hızı değişir. Aynı hatta bağlı tüm üniteler aynı haberleşme hızına sahiptirler. Veri uzunlukları kısa olduğundan, diğer haberleşme sistemlerine göre gecikme süresi de düşüktür.
- **Çoklu Gönderim Yapılması (Multicast):** Mesajlar filtreledikten sonra bir mesaj aynı anda birden çok birim tarafından alınıp değerlendirilebilir.
- **Veri Kararlılığı:** Tüm birimler hatta gelen mesajları aynı anda değerlendirirler. Buna göre gelen mesajı kabul edip etmemeye karar verilir. Bu işlem, çoklu gönderim ve hata bulma sistemleriyle birlikte gerçekleştirilir.
- **Güvenilirlik:** CAN-Bus sisteminde oluşan hataları bulmak için pek çok yöntem kullanılır. Bir hata bulunduğunda bu hataya neden olan mesajlar iptal edilir. Bu nedenle güvenilir sistemlerdir.
- **Sistem Esnekliği:** CAN-Bus hattına istenildiği kadar düğüm eklenip çıkartılabilir. Bu durum yazılımını, diğer düğümlerin donanımını ya da uygulama alanlarını değiştirmez. Bu da sisteme kolayca müdahale edilmesine izin verir ve sistemin esnekliğini sağlar.
- **Az Kablo Kullanımı ve Düşük Maliyet:** Veri iletişimi, farklı elektriksel gürültülerden etkilenmemesi için CAN-High ve CAN-Low hatlarından oluşur. Tüm sensörler ve üniteler bu hatta bağlanarak aynı hat üzerinden haberleşme sağlanır. Böylece her bir elemanı ayrı ayrı kablolama önlenmiş olur. Az kablo kullanımı, daha düşük maliyet demektir. Aynı zamanda hata tespitinin kolay olmasını sağlayarak sistemin güvenilirliğini artırır.
- **Yüksek Veri İletimi:** Saniyede 10000 mesaj gönderimi mümkündür.

CAN-Bus Hattının Yapısı

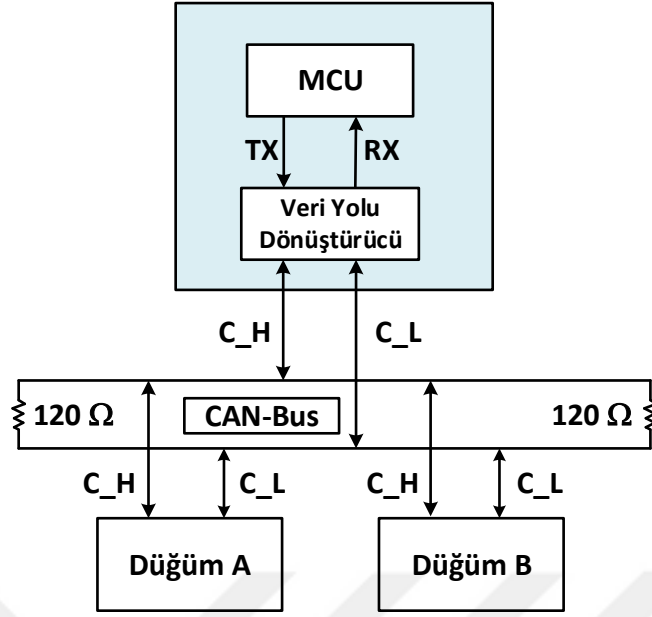
CAN-Bus veri iletim hattı CAN-High (CAN-H) ve CAN-Low (CAN-L) olarak adlandırılan kablolardan oluşur. Bu kablolar, elektromanyetik etkilere karşı dirençli olmasını sağlamak için saç örgüsü şeklinde yapılmış çift yönlü iletim kablolarıdır. Her iki kablodaki gerilimler zıt yönlüdür. Bir kablodaki gerilim yaklaşık 0 V ise diğer kablodaki gerilim 5 V ya da tam tersidir. Bu kabloların kullanılmasının önemli avantajları vardır. Bunlar, parazite karşı yüksek direnç sağlamaları, her sensör için ayrı kablo bağlantısının önlenerek aynı hat üzerinden haberleşmenin sağlanması, böylece kablo ağırlığının azalması, arıza tespitinin kolaylaşması ve daha güvenilir bir sistem elde edilmesidir.

CAN-Bus haberleşmesi birbirinin tersi olan iki hattaki potansiyel farkın işlemciler tarafından anlamlandırılmasına göre çalışır. Hattın başında ve sonunda 120 ohm'luk sonlandırma dirençleri bulunur. Bu dirençler sinyallerin bozulmasını önler ve hatta bir kopukluk olduğunda bunu tespit etmek için kullanılır. Ayrıca bilgilerin geri dönüşünü engeller. Böylece hatta bilgiler karışmaz. CAN-Bus kablolarının kontrolünde iki yöntem kullanılır: Birincisi, hatta gerilim yokken eşdeğer direncin 60 ohm olması gerekir. Eğer değilse, hatta kopukluk olduğu anlamına gelir. İkinci yöntemde ise, gerilim ölçümü yapılır. CAN-H ve CAN-L ile şasi arasında gerilimin değeri yaklaşık 2.5 V olmalıdır. Buna göre Şekil 4.2'de bir CAN-Bus hattı gösterilmiştir. Şekilde görüldüğü gibi, tüm sensörler, EKÜ'ler ve üniteler bu dirençlerin arasına bağlanır.



Şekil 4.2. CAN-Bus haberleşme hattı [73].

CAN hattına bağlı her bir işlemcinin üzerinde bir veri dönüştürme işlemi yapan alıcı/verici ünitesi bulunur. Bu ünite Şekil 4.3'te görüldüğü gibi bir alıcı (Receiver, RX) ve vericiden (Transmitter, TX) oluşur. CAN-Bus hattının potansiyel farkını işlemcinin anlamlandırabileceği hale dönüştürür. İşlemciden çıkan bitleri de CAN-Bus'a uygun hale dönüştürür.



Şekil 4.3. CAN-Bus veri yolu dönüştürücüsü [73].

CAN-Bus Haberleşmesinin Çalışması

CAN-Bus haberleşme sistemi “Multimaster Çalışma” mantığına göre çalışır. Tüm üniteler verilerini eşit öncelikli yollama hakkına sahiptir. Mesajlar eşit erişimli yollanırken, bir taraftan öncelikli mesajların daha erken gönderilmesine de dikkat edilir. Hatta bağlı tüm ünitelere bir ID numarası verilir ve mesajlar ilgili ID'ye gönderilir. ID numarası düşük olanlar yüksek öncelikli demektir. Böylece istenilen mesajların daha erken iletilmesi sağlanmış olur.

CAN-Bus'ta üniteler veri gönderirken, çakışmalar ve hatalar olmaması için hattın boş olduğu anda gönderim yaparlar. Bu çalışma şekli “Çarpışma Algılayıcıyla Taşıyıcı Dinleyen Çoklu Erişim” (Carrier Sense Multiple Access with Collision Detection (CSMA/CD)) olarak geçer. Bunun için önce hat kontrol edilir. Eğer bir ünite veri gönderirken, uzak mesafelerden dolayı başka bir ünite mesajın hatta olduğunu fark edemeyip veri gönderirse çarpışma gerçekleşir. Bu durumda iki ünite de veri göndermeyi bir süre bekletir ve hat boş olduğunda tekrar gönderirler. Böylece gönderilen verilerin çakışmaması sağlanır ve veri ağının uzunluğu sınırlandırılır. 40 m'lik veri ağı uzunluğu ile 1 Mbit/s haberleşme hızında veri iletimi mümkündür. Tablo 4.3'te görüldüğü gibi, haberleşme hızıyla veri ağı uzunluğu arasında ters ilişki vardır. Ağ uzunluğu düşüktüğü haberleşme hızı artar. Böylece 40 kbit/s'den 1 Mbit/s'ye hıza kadar haberleşmek mümkündür.

Tablo 4.3. Veri ağı uzunluğu ile haberleşme hızı arasındaki ilişki

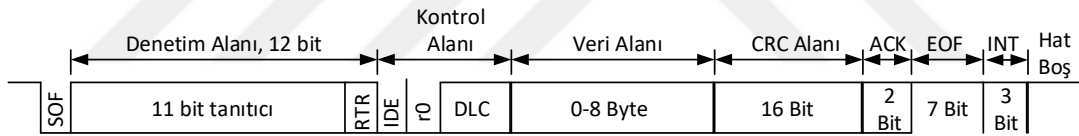
Veri Ağı Uzunluğu	Haberleşme Hızı
40 m	1 Mbit/s
100 m	500 kbit/s
200 m	250 kbit/s
500 m	125 kbit/s
1000 m	40 kbit/s

CAN-Bus Çerçeve Çeşitleri

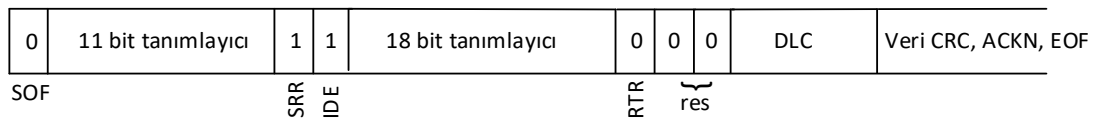
CAN-Bus protokolünde mesajlar dört çeşit çerçeve ile iletilir. Bunlar, veri çerçevesi (data frame), istek çerçevesi (remote frame), hata çerçevesi (error frame) ve aşırı yük çerçevesidir (overload frame).

Veri Çerçevesi

Veri paketlerinin tanımlandığı çerçevedir. Standart ve genişletilmiş veri çerçevesi olmak üzere iki çeşitten oluşur. Standart çerçeve, 11 bitlik mesaj tanımlamada kullanılırken, genişletilmiş çerçeve ile 29 bitlik mesaj tanımlanabilir. Bu yapılar Şekil 4.4 ve 4.5'te ayrı ayrı gösterilmiştir.



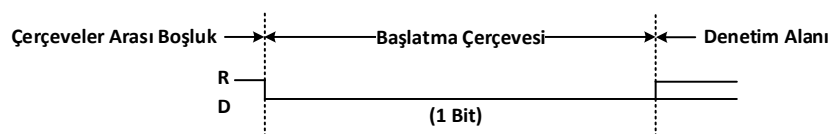
Şekil 4.4. Standart veri çerçevesi



Şekil 4.5. Genişletilmiş veri çerçevesi

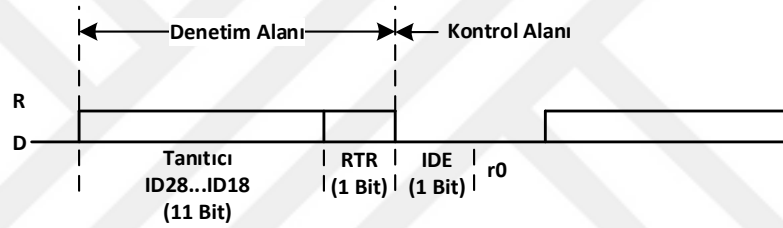
Bu çerçevelerde geçen alanlar şöyledir:

- **Çerçeve Başlangıcı (Start of Frame, SOF):** Şekil 4.6'da gösterildiği gibi bir bitlik çerçeve başlangıç bitidir. Tüm düğümler bu bitin yükselen kenarı ile senkronize olmalıdır.

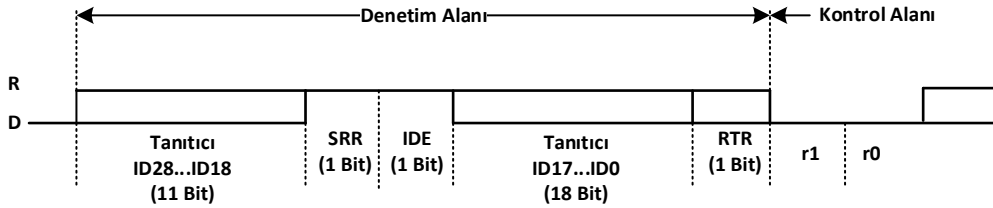


Şekil 4.6. SOF biti [73].

- **Denetim Alanı (Arbitration Field):** Denetim alanı, standart veri çerçevesinde 12 bittir. Bunun 11 biti mesaj tanımlamada kullanılırken, 1 biti Uzak Gönderim İstek bitidir (Remote Transmit Request, RTR). RTR biti lojik 1 olduğunda istek çerçevesi, lojik 0 olduğunda veri çerçevesi anlamına gelir. Bu bit, veri çerçevesinin uzaktaki bir üniteden istekte bulunmasında kullanılır. Genişletilmiş veri çerçevesinde ise 29 bitlik mesaj tanımlama alanı vardır. Bunun yanında 1 bit RTR biti, 1 bit Yedek Uzak İstek Biti (Substitute Remote Request, SRR) ve 1 bit de Tanımlayıcı (Identifier, IDE) bit olmak üzere toplam 32 bitten oluşur. RTR bitinin yedeği olan SRR biti, sadece genişletilmiş çerçevede bulunur ve her zaman lojik 1 değerindedir. IDE biti ise, veri çerçevesinin standart mı yoksa genişletilmiş çerçeve mi olduğunu gösterir. Şekil 4.7 ve 4.8’de görüldüğü gibi standart çerçeve için değeri 0, genişletilmiş çerçeve için de 1’dir. Buna göre standart çerçevede $2^{11} = 2048$ mesaj tanımlanabilirken, genişletilmiş çerçevede $2^{29} = 536.870.912$ mesaj tanımlanabilir. Standart ve genişletilmiş veri çerçevelerinin denetim alanları sırasıyla Şekil 4.7 ve 4.8’de gösterilmiştir.

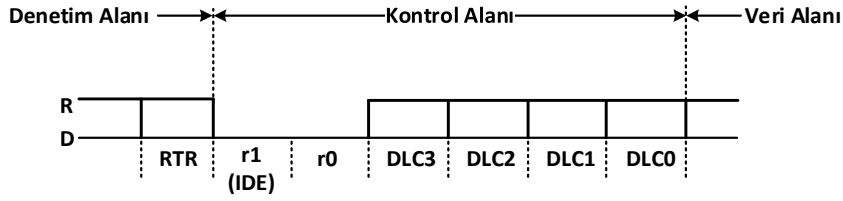


Şekil 4.7. Standart çerçeve denetim alanı



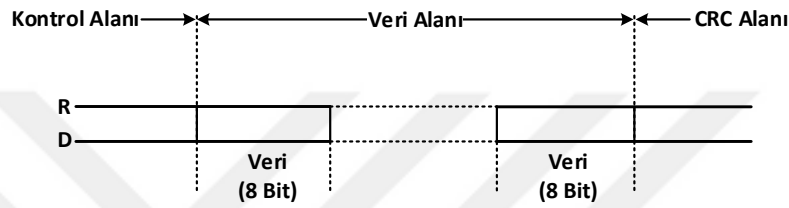
Şekil 4.8. Genişletilmiş çerçeve denetim alanı

- **Kontrol Alanı (Control Field):** Kontrol alanı, 1 bitlik IDE, 1 bit daha sonra kullanılmak üzere ayrılmış r0 biti ve 4 bit uzunluğunda veri uzunluk kodundan (Data Length Code, DLC) oluşur. Genişletilmiş veri çerçevesinde ise tek fark, daha sonra kullanılmak üzere 2 bitlik alan ayrılmıştır. Kontrol alanı Şekil 4.9’da gösterilmiştir.



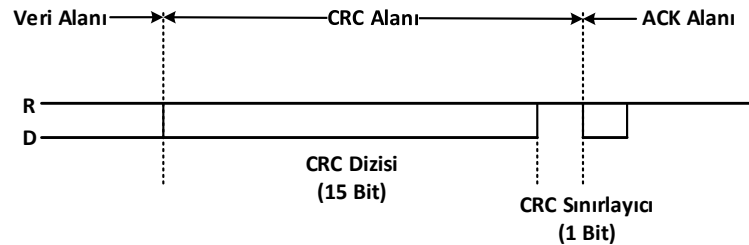
Şekil 4.9. Kontrol alanı

- **Veri Alanı (Data Field):** Veri alanı Şekil 4.10’da gösterildiği gibi, 1 byte’tan 8 byte’a kadar olan verinin taşındığı kısımdır.



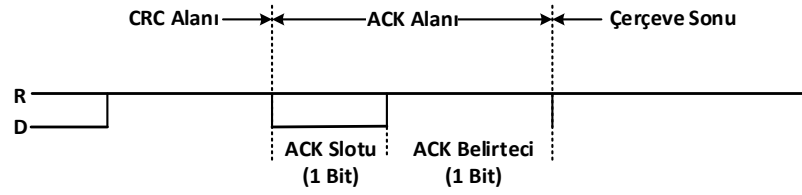
Şekil 4.10. Veri alanı

- **Dönüşsel Fazlalık Kontrol Alanı (Cyclic Redundancy Check Field, CRC):** CRC alanı Şekil 4.11’de görüldüğü gibi, 15 bitlik CRC dizisi (CRC Sequence) ve 1 bitlik CRC sınırlayıcısından (CRC Delimiter) oluşur ve hata denetlemede kullanılır. CRC, verilerin kablodan bozulmadan iletebilmesi için veri üzerinde bir hesaplama yapar. Hesaplama sonucunda bir CRC kodu üretir. Alıcı kısmı da başlama biti, denetim alanı, kontrol alanı, veri alanı ve CRC alanı üzerinde bir hesaplama yapar. Alınan veri CRC kodu ile aynı ise gelen veri doğrudur ve göndericiye alındı bilgisi (Acknowledgement Field, ACK) gönderilir. Eğer doğru değilse, bir hata çerçevesi gönderilerek veri tekrar gönderilir.



Şekil 4.11. CRC alanı

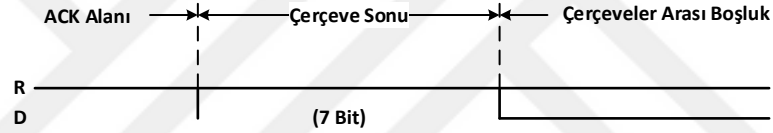
- **Alındı Bilgisi Alanı (Acknowledgement Field, ACK):** ACK, bir bitlik ACK slotu ve bir bitlik ACK belirtecinden oluşur. Şekil 4.12’de bu alan gösterilmiştir.



Şekil 4.12. ACK alanı

ACK, gönderici düğüm tarafından lojik 1 olarak gönderilir. Öncelikle başlama biti gönderilerek hatta gönderici belli olur. Daha sonra denetim alanı, kontrol alanı, veri alanı ve CRC alanı gönderilir. ACK'da, bilginin alınıp alınmadığı tespit edilir. Gönderici ünite mesajı aldıysa bu alanı lojik 0 yapar. Eğer herhangi bir üniteden mesaj alındı bilgisi gelmezse hata oluşmuş demektir ve hata çerçevesi üretilir. Böylece bilgi tekrar yollanır.

- **Çerçeve Sonu (End of Frame, EOF):** Veri çerçevesinin tamamlandığını gösteren 7 bitlik kısımdır. Bu alan Şekil 4.13'te gösterilmiştir.

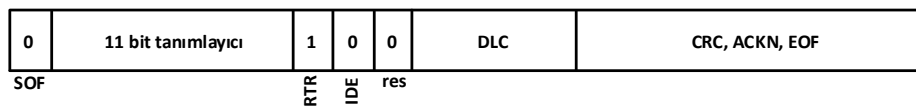


Şekil 4.13. Çerçeve sonu alanı

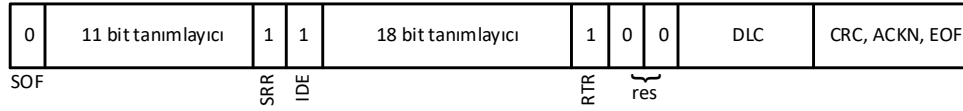
- **Çerçeveler Arası Boşluk (Intermission Field, INT):** Veri ya da istek çerçevesinin arka arkaya gönderilmesi durumunda çerçeveleri birbirinden ayırmak için kullanılan üç bitlik boşluktur. Lojik 1 değerindedir. Eğer lojik 0 değerini almışsa yeni bir çerçeve başlamış demektir.

İstek Çerçevesi (Remote Frame)

Bir ünite başka bir üniteden veri isterken istek çerçevesini kullanır. Veri çerçevesindeki RTR bitinin lojik 1 yapılması durumunda istek gönderilir. Eğer RTR biti lojik 0 ise veri çerçevesini ifade eder. İstenen verinin bilgisi mesaj ID alanında bulunduğundan, istek çerçevesi yollanıyorsa veri alanı yok demektir. Standart ve genişletilmiş istek çerçevesi Şekil 4.14 ve Şekil 4.15'te sırasıyla gösterilmiştir.



Şekil 4.14. Standart istek çerçevesi

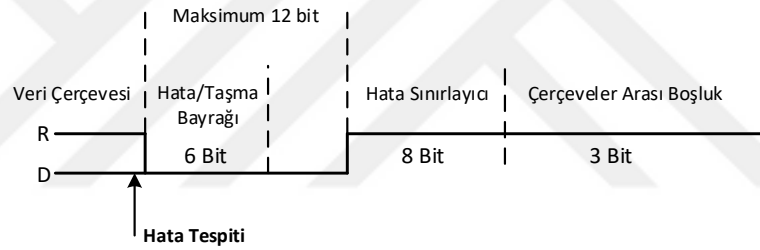


Şekil 4.15. Genişletilmiş istek çerçevesi

Hata Çerçevesi (Error Frame)

CAN-Bus'ta veri alma ve gönderme işlemlerinde bir hata oluştuğunda ne tür bir hata olduğunu hata çerçevesi gösterir. Bu çerçevenin yapısı Şekil 4.16'da verilmiştir. Bir hata çerçevesi, 6'dan 12 bite kadar değişebilen bir hata bayrağından ve 8 bitlik bir hata sınırlayıcıdan oluşur.

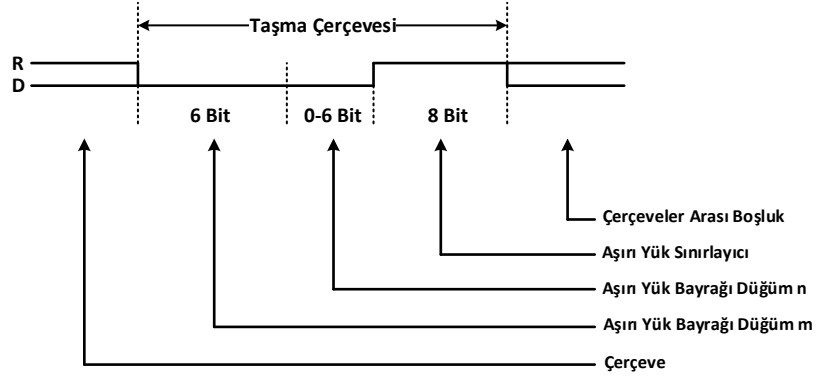
CAN-Bus hattında beş defa art arda aynı seviyede bit geldiğinde 6. bit dolgu biti (stuff bit) olarak adlandırılır ve ters lojik seviyeli bit gelir. 6 bit art arda aynı seviyede geldiğinde hata olarak algılanır ve hata bayrağı gönderilir. Hata bayrağı gönderildikten sonra 8 bit lojik 1 olarak gönderilerek hata çerçevesi tamamlanır.



Şekil 4.16. Hata çerçevesinin yapısı.

Taşma Çerçevesi (Overload Frame)

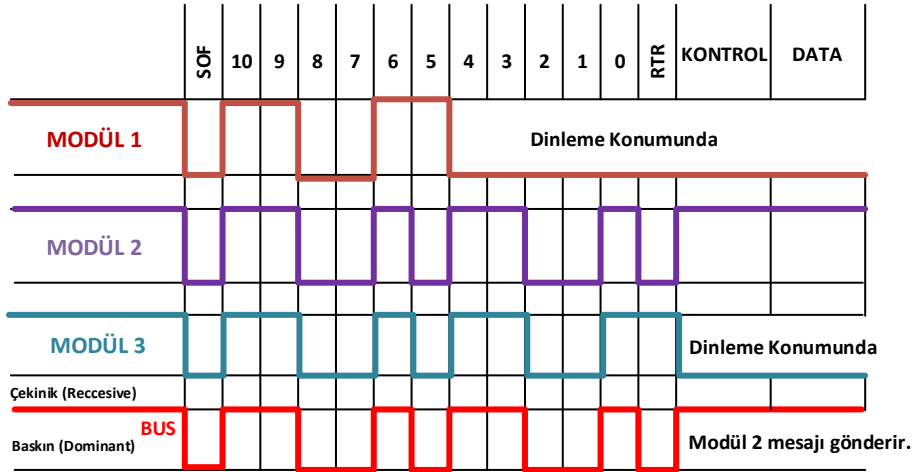
Taşma çerçevesi, alıcı düğümün bir sonraki alacağı veri için dâhili çalışma zamanının yetmemesi durumunda, çerçeveler arasındaki boşlukta baskın bir bitin veya hata sınırlayıcının 8. bitinde baskın bitin olması durumunda gönderilen çerçevedir. Taşma çerçevesinin çalışması hata çerçevesi ile aynı mantıktadır. Tek fark, taşma çerçevesi çerçeveler arasında başlar. Şekil 4.17'de yapısı detaylı gösterilmiştir.



Şekil 4.17. Taşma çerçevesinin yapısı.

Bit Gönderme İşlemi (Arbitration)

CAN-Bus'ta bit gönderme işleminde mesajların öncelik sırası önemlidir. Hangi mesajın daha önce gönderilmesi gerekiyorsa bu belirlenir. Bit gönderme işlemi gerçekleşirken lojik durumların baskın ve çekinik olarak tanımlanması gerekir. Lojik 0 baskın, lojik 1 ise çekinik bittir. Baskın bit çekinik bitten üstündür. CAN hattına iki düğümden mesajlar gönderildiğinde mesajların içerisindeki çekinik ve baskın bitler sırasıyla karşılaştırılır. Eğer karşılaştırılan bitler aynıysa sonraki bite geçilir. Farklıysa baskın bite sahip olan mesaj üstün olacağından bu düğümden veri gönderilecektir. Çekinik biti olan düğüm boş işlem periyodunu bekleyerek mesajını tekrar gönderir. Şekil 4.18'de bit gönderme işlemi detaylı gösterilmiştir.



Şekil 4.18. Bit gönderme işlemi

Şekil 4.18'de görüldüğü gibi, üç modül CAN hattına mesajlarını gönderir. Sırasıyla her modülün bitleri karşılaştırılır. İlk durumda Modül 1 çekinik bite sahip olduğundan dinlenme konumuna geçer ve bit gönderme işlemini kaybeder. 2. durumda ise Modül 2 ve Modül 3'ün

bitleri karşılaştırılmaya devam edilir. Modül 3 çekinik bite sahip olduğundan dinlenme konumuna geçer ve baskın biti olan Modül 2 mesajını gönderir.

4.4.2. Yerel Bağlantı Ağı (LIN)

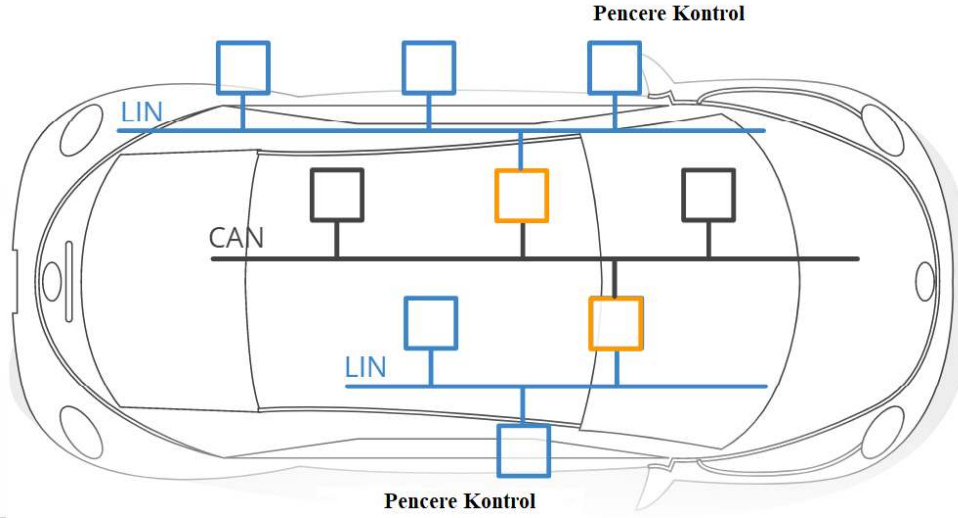
CAN haberleşme protokolü araçtaki her bileşenin haberleşmesinde kullanılmak için pahalı bir protokoldür. Bunun için 1998 yılında LIN Consortium, Volcano Automotive Group ve Motorola'dan sağlanan teknolojilerle aralarında BMW, Volkswagen Group, Audi, Volvo Cars, Mercedes-Benz olan beş otomobil üreticisi tarafından açılımı “Local Interconnect Network (LIN)” olan “Yerel Bağlantı Ağı” kurulmuştur. LIN araçlarda, elektrikli cam, klima, merkezi kilit, sunroof gibi konforla ilgili birçok birimde iletişimi sağlayan bir seri ağ protokolüdür. Araçta kullanılan her donanım için ayrı kablolama gerektirmesi kablo sayısını arttırdığı ve arıza tespitini zorlaştırdığı için bu protokolle bunun önüne geçilmiştir. LIN protokolünün temel özellikleri şunlardır:

- Tek bir ana (master) ünite, 16 adete kadar da köle (slave) ünite kullanılır. CAN-Bus'ta ise birden fazla ana ünite olabilir.
- 20 kbit/s'ye kadar haberleşme hızı sağlar ve veri yolu uzunluğu 40 m'dir. CAN'de ise 1 Mbit/s hızına kadar haberleşmek mümkündür.
- Garantili gecikme süreleri,
- 2, 4 ve 8 byte olmak üzere değişken uzunluktaki veri çerçeveleri,
- Konfigürasyon esnekliği,
- Veri kontrolü ile hata tespiti ve
- Arızalı düğümlerin tespitini sağlar.
- Çalışma gerilimi 12 V'tur.

Bu protokolün avantajları ise şöyle sıralanabilir [74]:

- Kullanımı kolaydır.
- CAN ve diğer haberleşme protokollerinden daha az maliyetlidir.
- Kablolamayı azaltır. Bu da maliyeti etkiler ve arıza tespitini kolaylaştırır.
- Güvenilirdir.
- Uygulaması kolaydır.
- Protokol lisans ücreti yoktur.

Bir araç üzerinde CAN ve LIN hatları Şekil 4.19'da gösterilmiştir.



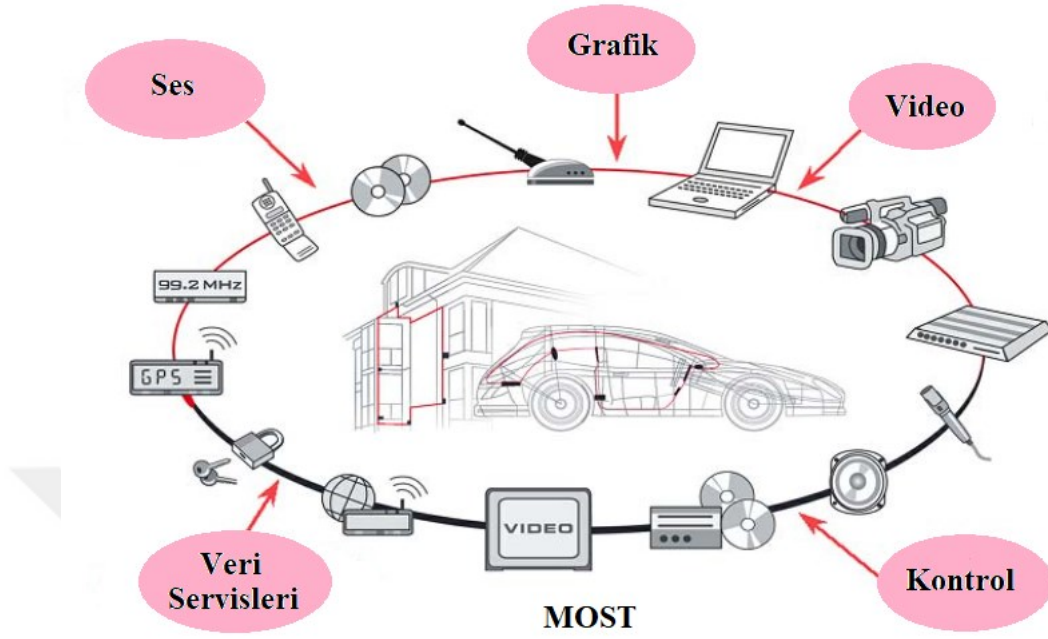
Şekil 4.19. Bir araçta CAN ve LIN hatlarının gösterimi [75].

4.4.3. Medya Odaklı Sistemleri Taşıma Ağı (MOST)

Açılımı “Media Oriented Systems Transport (MOST)” olan Medya Odaklı Sistemleri Taşıma Ağı, otomotiv endüstrisinde kullanılan yüksek hızlı bir multimedia ağ teknolojisidir. Oasis SiliconSystems AG (SMSC) tarafından BMW, Becker Radio ve DaimlerChrysler ile birlikte tasarlanmıştır. Araç içi veya dışı uygulamalar için kullanılabilir. Araçlarda radyo, televizyon, telefon, navigasyon gibi eğlence donanımları için geliştirilmiştir. MOST ağı, 21.2 Mbit/s haberleşme hızına sahiptir. Audi, BMW, General Motors, Honda, Hyundai, Jaguar, Lancia, Land Rover, Mercedes-Benz, Porsche, Toyota, Volkswagen, SKODA, SEAT ve Volvo olmak üzere hemen hemen her otomobil markasında kullanılmaktadır. MOST hattının avantajları aşağıdaki gibi sıralanabilir:

- Yüksek veri hızı,
- Daha az kablo kullanımı,
- Korumalı elektrik veri hatlarına kıyasla daha hafif ve esnek olması,
- Elektromanyetik uyumluluğu,
- Herhangi bir elektromanyetik parazite neden olmaması,
- Elektromanyetik girişim ışınlanmasına duyarlı olmasıdır.

MOST hattında kullanılan sistemler Şekil 4.20'de gösterilmiştir.



Şekil 4.20. MOST hattında kullanılan sistemler [76].

Bu bölümde, araçlarda ve endüstride kullanılan ağ sistemleri, haberleşme protokolleri ve bu tez çalışmasında kullanılan CAN-Bus haberleşme protokolü detaylı olarak anlatılmıştır. Gelecek bölümde ise sayısal görüntü işleme ve algoritmaları ele alınacaktır.

5. SAYISAL GÖRÜNTÜ İŞLEME

5.1. Görüntü İşlemede Temel Kavramlar

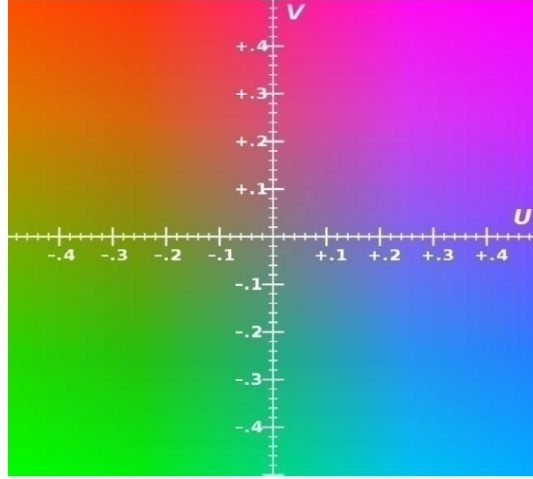
5.1.1. Sayısal Görüntü

Sayısal görüntü, özel elektronik cihazlar kullanılarak görüntüden yansıyan ışığı mercek ya da objektiften yararlanarak bir düzlemde toplanan ışık enerjisinin, ışığa duyarlı devre elemanları sayesinde elektriksel sinyallere dönüştürülmesiyle oluşturulmuş sayısal verilerdir [77]. İki boyutlu görüntünün örneklenmesi sonucu oluşan sayısal bir görüntü, A satır ve B sütun sayısı olmak üzere $A \times B$ boyutlu bir matris olarak ifade edilir. Sayısal bir görüntüdeki en küçük birim piksel olarak isimlendirilir. Her piksel 0-255 arasında bir değer alır. Görüntü f ile ifade edilirse bir pikselin değeri koordinat sistemi üzerinde $f(x, y)$ şeklinde gösterilebilir.

5.1.2. Renk Uzayları

İnsan gözünün yapısından dolayı bütün renkler ana renk olarak adlandırılan kırmızı, yeşil ve mavinin değişik kombinasyonlarından oluşmuş olarak algılanmaktadır. Temel renklerin bir araya gelmesiyle ikincil renkler meydana gelmektedir. Mor (kırmızı+mavi), camgöbeği rengi (yeşil+mavi), sarı (kırmızı+yeşil) renkleri ikincil renklerdir. Doğru oranlarda temel renkler bir araya getirilirse beyaz renk elde edilebilir. Aynı şekilde ikincil renklere bir tanesi ve bu rengin tersi bir araya gelirse beyaz renk elde edilebilir. Örneğin mor ile yeşil bir araya gelirse beyazı oluşturur. Buna göre sayısal görüntüyü tanımlamak için kullanılan renk uzayları şöyledir:

- **RGB Renk Uzayı:** Bu renk uzayı kırmızı (Red), yeşil (Green) ve mavi (Blue) ana renkleri kullanarak oluşturulmuştur. En yaygın kullanılan renk uzayıdır. Bu renk uzayının kullanıldığı iki boyutlu görüntülerde her piksel kırmızı, mavi ve yeşil renk yoğunluk değerine sahiptir. Ayrıca bir parlaklık bilgisi yoktur. RGB renk uzayına sahip görüntüler tanımlanırken üç boyutlu matrisler kullanılır. İlk ve ikinci boyut, görüntüdeki piksellerin konumunu, üçüncü boyut ise ilgili pikselin renk yoğunluğunu gösterir. R, G, B'nin her biri 8 bitlik yani 0 ile 255 arasında bir değer saklayabilir.
- **YUV (YCbCr) Renk Uzayı:** Açılımında Y parlaklığı (Luminance), U ve V renk bilgilerini ifade eder. U, mavi renk yoğunluğu (Chrominance1) ve V kırmızı renk yoğunluğudur (Chrominance2). RGB pek çok uygulama için yeterli olsa da birçok uygulama için sınırlı kalmaktadır. Bu yüzden, parlaklık ve yoğunluk bilgisini renk bilgisinden ayırmak için YUV renk uzayı tercih edilir. YUV uzayında renk yoğunlukları ve bunlara karşılık gelen değerler Şekil 5.1'de gösterilmiştir. Buradaki Y parlaklık değeri, RGB uzayında gri görüntüye karşılık gelir.



Şekil 5.1. YUV renk yoğunlukları ve değerleri [77].

Şekil 5.1’de görüldüğü gibi U ve V renk kanalları -0.5 ile +0.5 aralığında değişen, sırasıyla mavi ve kırmızı renk tonlarını gösteren bir uzaydır. Bu değerler 8 bitlik görüntü olarak ifade edilirse -0.5 değeri 0 değerine, 0.5 değeri ise 255 değerine karşılık gelir [77].

R, G, B değerleri 0 ile 255 arasındaki değerler olduğundan bu değerleri kullanarak görüntüdeki her piksel için YUV renk dönüşümü Denklem 5.1, 5.2 ve 5.3’teki gibi yapılabilir [78].

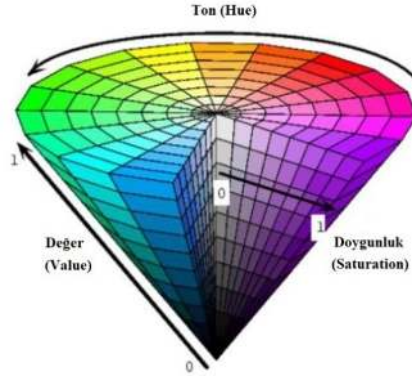
$$Y = (0.2215 * R) + (0.7154 * G) + (0.0721 * B) \quad (5.1)$$

$$U = -(0.1145 * R) - (0.3855 * G) + (0.5000 * B) \quad (5.2)$$

$$V = (0.5016 * R) - (0.4556 * G) - (0.0459 * B) \quad (5.3)$$

Bu renk uzayında Y, (0-255), U (-112 ile +112), V ise (-157 ile +157) aralığında değerler alır.

- **HSV Renk Uzayı:** Açılımı “Hue Saturation Value” olan ton, doygunluk ve parlaklık değeri anlamına gelen renk uzayıdır. YUV renk uzayı da bazı uygulamalar için çok yeterli olmadığından HSV renk uzayı ortaya çıkmıştır. İnsanların renkleri nasıl algıladıklarıyla ilgili sorgulamalarıyla oluşmuş daha gelişmiş bir versiyondur. Bu renk uzayı Şekil 5.2’de gösterilmiştir.



Şekil 5.2. HSV renk uzayı [79].

- **CMYK Renk Uzayı:** Camgöbeği (Cyan), eflatun (Magenta), sarı (Yellow), siyah (Key) renklerinden oluşan renk uzayıdır. Ana renkler olan bu renkler, matbaa alanında renkli baskılama işlemlerinde kullanılır. Burada “Key” olarak adlandırılan siyah renk anahtar renktir.

5.2. Görüntü İşleme

Görüntü işleme, görüntüdeki yararlı bilgileri çıkarmak, gelişmiş bir görüntü oluşturmak, nesneleri, desenleri ayırt etmek ve görüntüyü sayısal forma dönüştürmek amacıyla kullanılan bir yöntemdir [80]. Bir görüntü, $f(x,y)$ ile gösterilen iki boyutlu bir fonksiyon olarak tanımlanabilir. Burada x ve y düzlemsel koordinatlarıdır ve herhangi bir koordinat çiftinde (x,y) f'nin genliği, o noktada görüntünün gri seviyesi veya yoğunluğu olarak adlandırılır [81]. f'nin x , y ve genlik değerleri sonlu ve ayrık nicelikler olduğunda, bu görüntüye sayısal görüntü denir. Sayısal görüntüler, her biri piksel olarak ifade edilen, belirli konum ve değere sahip sonlu sayıda elemandan oluşur.

Sayısal görüntü işleme, görüntülerin analiz ve düzenleme işlemleridir. Bu işlemler gerçekleştirilirken genel olarak kullanılan yöntemler üç kısımdan oluşur. Bunlar; ön işleme, geliştirme ve bilgi çıkarımıdır [80]. Ön işleme, düşük seviye görüntüler üzerinde kullanılan bir yöntemdir. Görüntüdeki istenmeyen bozulmaları azaltmak ve yararlı bilgileri ortaya çıkartmak için kullanılır. İyileştirme, görüntünün görsel algılığını arttırmak ve görüntü içerisindeki bilgileri daha iyi anlamlandırabilmek için kullanılır. Bilgi çıkarımı ise, görüntünün özelliklerini çıkartmak ve sınıflandırmak amacıyla kullanılan son yöntemdir. Bu yöntemler biyomedikal, biyometrik kimlik doğrulama, endüstriyel otomasyon, güvenlik gibi alanlarda önemli bir yere sahiptir.

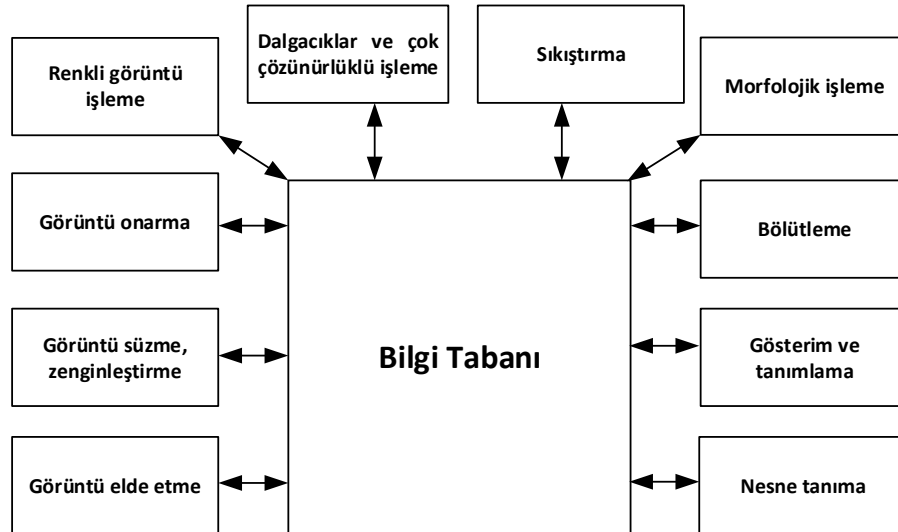
5.3. Görüntü İşleme Kullanım Alanları

Sayısal görüntü işleme, günümüzde pek çok alanda yaygın şekilde kullanılmaktadır. Bu alanlardan bazıları aşağıdaki gibi sıralanabilir [80,81]:

- Kriminoloji/Adli Tıp
- Meteoroloji
- Otonom
- Navigasyon
- Radar
- Astronomi
- Tıbbi Uygulamalar
- Askeri Sanayi
- Radyoloji
- Kalite Kontrol
- Sayısal Kamera Görüntüleri
- Morfoloji
- Bilgisayarla Görme
- Ultrasonik Görüntüleme

5.4. Sayısal Görüntü İşlemede Temel Adımlar

Sayısal görüntü işlemenin adımları detaylı olarak Şekil 5.3'te gösterilmiştir.



Şekil 5.3. Sayısal görüntü işleme adımları [77].

- 1. Görüntü elde etme:** Görüntüler bir sensör tarafından yakalanır. Kamera veya sensör çıkışı dijital formda değilse analog dijital dönüştürücü kullanılarak sayısala dönüştürülür.
- 2. Görüntü zenginleştirme:** Belirli uygulamalar için görüntüyü işleyerek orijinalinden daha iyi bir sonuç elde etme işlemidir. İyileştirme tekniklerinin amacı, gizli ayrıntıları ortaya çıkarmak veya bir görüntüdeki ilgi çekici belirli özellikleri vurgulamaktır.
- 3. Görüntü onarma:** Bir görüntünün görünümünü bozucu etkisine uygun bir süzgeçten geçirerek iyileştirme işlemidir.
- 4. Renkli görüntü işleme:** Bir görüntüdeki ilgi çekici özellikleri çıkarmak için görüntünün rengini kullanma işlemidir. Renk bilgisine göre nesne tanıma yapılabilmektedir.
- 5. Dalgacıklar:** Çeşitli çözünürlük derecelerindeki görüntüleri temsil etmenin temelidir. Görüntü verilerini sıkıştırmak için kullanılır.
- 6. Sıkıştırma:** Bir görüntüyü kaydetmek için gereken depolamayı veya onu iletmek için gereken bant genişliğini azaltmak için kullanılan tekniklerdir.
- 7. Morfolojik işlemler:** Görüntüyü belirleyen sınırları ve yararlı olan görüntü bileşenlerini ayıklamak için kullanılan işlemlerdir.
- 8. Bölütleme yöntemleri:** Bir görüntüyü oluşturan bölge veya nesnelerin alt bölümlere ayrılmasında kullanılan yöntemlerdir. Böylece daha alt seviyeden sorunlar çözülebilir.
- 9. Gösterim ve tanımlama:** Bu bölümde, verilerin bir sınır olarak mı yoksa tam bir bölge olarak mı temsil edileceğine karar verilir. Bölütlenmiş görüntüdeki her bir bölümü tanımlamada kullanılır.
- 10. Nesne tanıma:** Anlatılan görüntü işleme yöntemleri sonucunda hedef nesne için görüntüden elde edilen bazı verilerden yola çıkarak hedef nesneye benzer nesneleri tanıma işlemleridir.

5.5. Görüntü İşleme Algoritmaları

5.5.1. Renkli Görüntülerin Griye Dönüştürülmesi

Renkli olan RGB (Red, Green, Blue) görüntüleri griye dönüştürme yöntemi görüntü işleme uygulamalarında yaygın şekilde kullanılmaktadır. Böylece gri seviye görüntü ile hesaplama gereksinimleri azaltılır ve daha iyi performans elde edilir.

Renkli görüntülerin griye dönüştürülmesi işleminde bazı yöntemler kullanılır. İlk yöntemde, Denklem 5.4'te görüldüğü gibi renkli görüntünün kırmızı, yeşil ve mavi bileşenlerinin aritmetik ortalaması alınarak hesaplanır. Denklem 5.5'te ise RGB renklerinin ağırlıklı bir kombinasyonunu kullanarak insan gözünün parlaklık algısına göre oluşturulur. Bu yöntem bilgisayarla görme alanında sıkça kullanılır. Denklem 5.6'da verilen son yöntemde de denklem 5.5'teki katsayılarla yakın değerlerle dönüştürme yapılır. Bu yöntem yüksek çözünürlüklü televizyonlarda kullanılmaktadır [80].

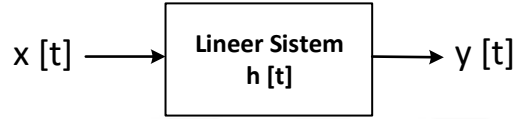
$$Y = \frac{1}{3}(R + G + B) \quad (5.4)$$

$$Y = (0.2989 * R) + (0.5870 * G) + (0.1140 * B) \quad (5.5)$$

$$Y = (0.2126 * R) + (0.7152 * G) + (0.0722 * B) \quad (5.6)$$

5.5.2. Impuls Cevabı ve Konvolüsyon

Konvolüsyon, Şekil 5.4'te gösterildiği gibi bir sistemin $x(t)$ girişine karşılık üreteceği $y(t)$ çıkışını, birim dürtü cevabı $h(t)$ ile bulma işlemidir.



Şekil 5.4. Konvolüsyon işlemi

Bu işlem Denklem 5.7'deki gibi gösterilebilir.

$$x[t] * h[t] = y[t] \quad (5.7)$$

Matematikte konvolüsyon, bir fonksiyonun şeklinin bir integral alma işlemiyle başka bir fonksiyon tarafından değiştirilmesidir [82]. Sinyal işleme, görüntü işleme, olasılık ve istatistik gibi pek çok alanda kullanılır. Konvolüsyon işlemi, ayrık ve sürekli zamandaki sinyallerin incelenmesinde önemlidir. Bir f ile g fonksiyonunun konvolüsyonu Denklem 5.8'teki gibi ifade edilir.

$$(f * g)(t) = \int_{-\infty}^{\infty} f(\tau) g(t - \tau) d\tau \quad (5.8)$$

5.5.3. İki Boyutlu Lineer Konvolüsyon

Konvolüsyon görüntü işlemede yaygın olarak kullanılan önemli algoritmalarından biridir. Bu işlem görüntüye düzlemsel alçak ve yüksek geçiren filtre uygulamak için kullanılır ve görüntü matris değerleriyle çekirdek matris değerlerinin çarpıp toplanmasıyla gerçekleştirilir. İki boyutlu konvolüsyon işlemi Denklem 5.9'da gösterilmiştir.

$$y(m, n) = x(m, n) * h(m, n) = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} x[k, l] h[m - k, n - l] \quad (5.9)$$

Denklemden x giriş görüntüsünü, y çıkış görüntüsünü, h da çekirdek matrisini ifade eder. Konvolüsyon işlemine ait özellikler aşağıda sıralanmıştır:

- **Yer Değiştirme İşlemi:** Denklem 5.10'dan görüldüğü gibi giriş görüntüsü ve çekirdek matrisinin yer değiştirilmesi sonucu değiştirmez.

$$Yer\ deg\ istance : X * H = H * X \quad (5.10)$$

- **Lineerlik:** Denklem 5.11'de görüldüğü gibi, giriş görüntüsünün skaler bir sabit ile çarpıldıktan sonra konvolüsyon işlemine tabi tutulmasıyla giriş görüntüsünün, skaler bir sabit ile çarpılmış çekirdek matrisi ile konvolüsyon işlemine tabi tutulması aynı sonucu vermektedir.

$$Lineerlik : (s \cdot X) * H = X * (s \cdot H) = s \cdot (X * H) \quad (5.11)$$

Denklem 5.12'de gösterildiği gibi piksel piksel eklenen iki girişin birlikte konvolüsyon işlemine tabi tutulmasıyla, iki girişin ayrı ayrı konvolüsyon işlemine tabi tutulup toplanmasında aynı sonuç elde edilir.

$$(X1 + X2) * H = (X1 * H) + (X2 * H) \quad (5.12)$$

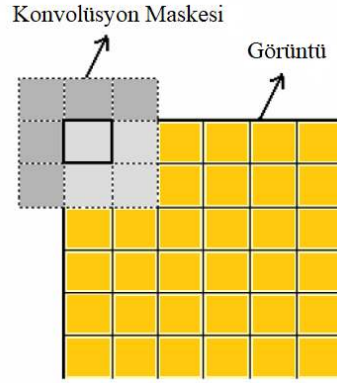
Bu özelliklerden farklı olarak Denklem 5.13'te gösterildiği gibi, giriş görüntüsünün skaler bir sabit ile toplanıp, konvolüsyon işlemine tabi tutulmasıyla skaler bir sabit ile konvolüsyon işleminin sonucunun toplanması birbirine eşit değildir.

$$(S + X) * H \neq s + (X * H) \quad (5.13)$$

Denklem 5.14'te görüldüğü gibi konvolüsyon işleminin birleşme özelliği vardır.

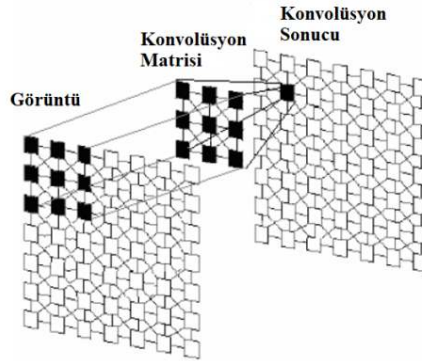
$$Birlesme : A * (X * H) = (A * X) * H \quad (5.14)$$

İki boyutlu konvolüsyon işleminde Şekil 5.5'te görüldüğü gibi, piksellerin komşuluğundan yararlanarak konvolüsyon maskesi görüntü üzerinde kaydırılır [83]. Bu işlem sonucunda görüntüdeki bir pikselin yeni değeri, kendisinin ve komşu piksellerdeki değerlerin ağırlıklı ortalaması alınarak bulunur. Konvolüsyon şablonu (kernel) denilen bir matris ile piksellerin ağırlıkları belirlenir.



Şekil 5.5. 2 boyutlu konvolüsyon

Konvolüsyon şablonu, değişken boyutlarda olabilen bir matristir. Örneğin, 3x3 boyutunda bir matris için konvolüsyon işlemi Şekil 5.6’da gösterilmiştir. Buna göre konvolüsyon matrisi görüntü üzerine getirildiğinde elde edilen yeni piksel değeri çıkış görüntüsünde görülmektedir. Bu şablon görüntü üzerinde gezdirilerek tüm piksellere uygulanır. Böylece iki boyutlu konvolüsyon işlemi gerçekleştirilmiş olunur.

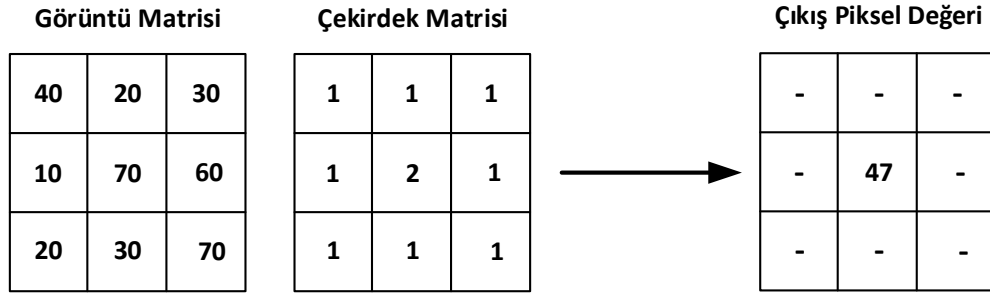


Şekil 5.6. Görüntüye uygulanan konvolüsyon işlemi ve sonucu

Konvolüsyon işleminde çıkış pikselinin hesaplanması Şekil 5.7’de detaylı gösterilmiştir. Çekirdek matrisi gerçekleştirilecek uygulamaya göre değişiklik gösterir. Genellikle 3x3’lük matrisler tercih edilir. Girişteki görüntü ile çekirdek matrisi aynı boyutta olmalıdır [80]. Buna göre görüntüde kenar çıkarma, keskinleştirme, bulanıklık giderme gibi farklı işlemler yapılabilir. Şekil 5.7’de görüldüğü gibi girişteki görüntünün her bir pikseli, çekirdek matrisine karşılık gelen değerle çarpılıp toplanarak ağırlıklı ortalaması alınır. Bulunan değer en yakın tam sayıya yuvarlanır.

Buna göre çıkış piksel değeri;

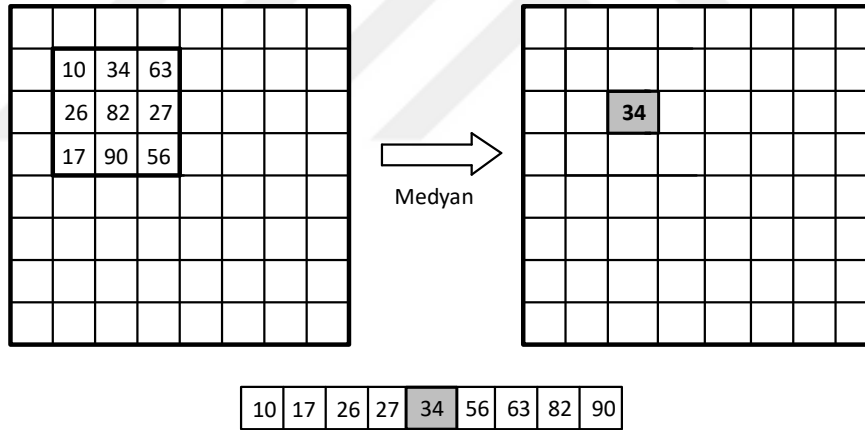
$$(40*1+20*1+30*1+10*1+70*2+60*1+20*1+30*1+70*1)/9=46.6666=>47 \text{ olarak bulunur.}$$



Şekil 5.7. Konvolüsyon işleminde çıkış pikselinin hesaplanması

5.5.4. Medyan Filtre

Medyan filtre, bir görüntünün üzerinde kayan bir penceredir ve merkez piksel değeri, penceredeki piksellerin medyan değeri ile değiştirilir [81]. Pencere, 3x3 kare veya herhangi bir şekilde olabilir. Medyan filtrelemede, penceredeki merkezi ve komşu pikseller sıralanır ve orta değer, merkez pikselle değiştirilir. Buna göre medyan filtrenin çıkışı Şekil 5.8'de bir örnek üzerinde gösterilmiştir.



Şekil 5.8. Medyan filtrenin çıkışı

Bu filtre, görüntülerdeki gürültüleri gidermek için kullanılan başarılı bir yöntemdir. Diğer dijital filtrelerle gürültü azaltma sırasında keskin kenarlar korunamaz. Medyan filtrenin diğer filtreler göre avantajı, gürültüleri giderirken keskin kenarları da korumasıdır.

5.5.5. Sobel Kenar Çıkarma Filtresi

Sobel kenar filtresi görüntü işleme tekniklerinde, görüntüdeki renk değişiminin fazla olduğu yerleri bulmak amacıyla kullanılmaktadır. Yüksek frekans alanlarını vurgulamak için iki konvolüsyon maskesi kullanır ve görüntü yoğunluğunun gradyanını hesaplar [81]. Konvolüsyon

maskeleri Denklem 5.15'te gösterilmektedir. X ve Y koordinatlarındaki kenarları tespit etmek için maskeler ayrı ayrı uygulanabilir. Bu değerler, gradyan yönünün büyüklüğünü ve açısını bulmak için birleştirilebilir. Denklem 5.16 ve 5.17'de görüldüğü gibi, $|G|$ gradyan büyüklüğü ve Q gradyan yönünü ifade eder. G_x ve G_y değerleri yüksek ise kenar mukavemeti de yüksek olacaktır. Bu yüksek değerler kenarların varlığını gösterir.

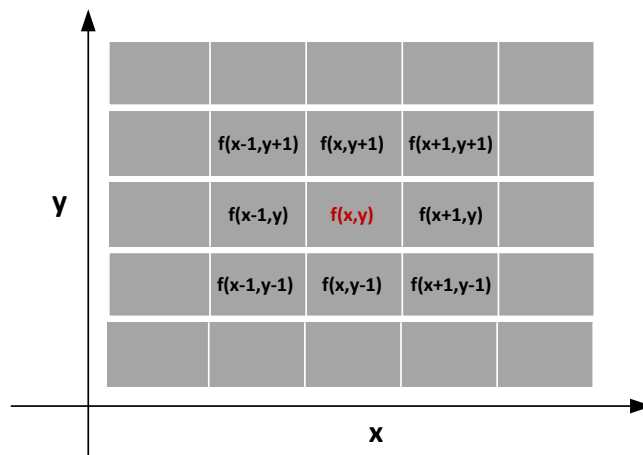
$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} * A \quad G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} * A \quad (5.15)$$

$$|G| = \sqrt{G_x^2 + G_y^2} \quad (5.16)$$

$$Q = \arctan \left(\frac{G_y}{G_x} \right) \quad (5.17)$$

Böylece dikey ve yatay eksenle kullandığı farklı iki maske ile tasarımcının girdiği eşik değerin altında ya da üstünde olmasına göre renk değişimleri bulunur. Genelde gri tonlu görüntülerde kullanılır. Verilen değerlerle oluşturulan yeni görüntüde değişimin olduğu yerleri siyah ya da beyaz renkten birine, değişimin olmadığı yerleri de diğer renge dönüştürür. Bu sayede kenarlar yani keskin yerler renk değişiminden belirlenmiş olur.

Şekil 5.9'da bir görüntü Kartezyen koordinat sistemi üzerinde gösterilmiştir. Buna göre her bir pikselin değeri koordinatlara göre verilmiştir. Ortadaki piksel değeri $f(x,y)$ olarak alınırsa, diğer piksel değerleri de buna göre yazılabilir.



Şekil 5.9. Bir görüntünün Kartezyen koordinat sisteminde gösterimi [84].

G_x yatay eksen matrisiyle elde edilen görüntü Denklem 5.18 ve 5.19 ile ifade edilebilir.

$$G_x = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} * A \quad (5.18)$$

Burada A orijinal görüntüdür.

$$G_x = f(x-1, y-1) * 1 + f(x-1, y) * 2 + f(x-1, y+1) * 1 + f(x+1, y-1) * (-1) + f(x+1, y) * (-2) + f(x+1, y+1) * (-1) \quad (5.19)$$

G_y düşey eksen matrisiyle elde edilen görüntü ise Denklem 5.20 ve 5.21 ile gösterilir.

$$G_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} * A \quad (5.20)$$

$$G_y = f(x-1, y+1) * 1 + f(x, y+1) * 2 + f(x+1, y+1) * 1 + f(x-1, y-1) * (-1) + f(x, y-1) * (-2) + f(x+1, y-1) * (-1) \quad (5.21)$$

Buna göre bulunan bu iki yeni görüntü matrisinin aynı koordinattaki piksel değerlerinin karelerinin toplamının karekökü alınarak, hedef görüntünün o koordinattaki piksel değeri bulunmuş olur.

G_x ve G_y yatay ve düşey eksen matrisleri kullanılarak hesaplanan $|G|$ gradyan büyüklüğü Denklem 5.22 ve 5.23'te gösterilmiştir.

$$G = \sqrt{G_x^2 + G_y^2} \quad (5.22)$$

$$|G| = |G_x| + |G_y| \quad (5.23)$$

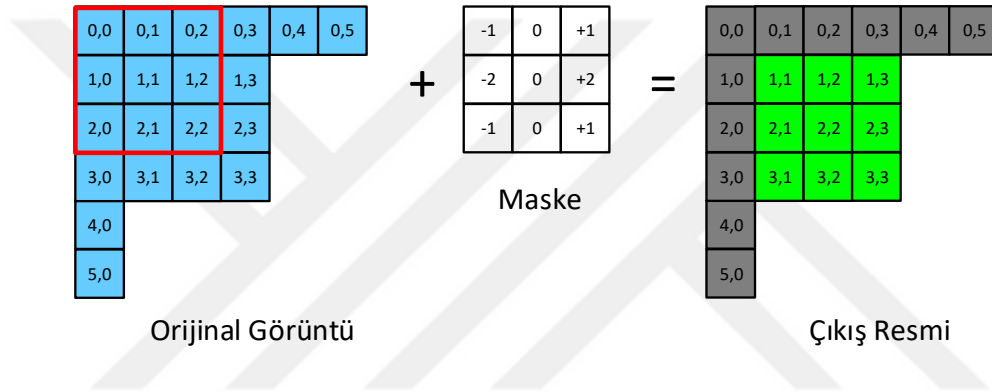
G 'nin hesabı kök ifadesi olduğu için işlemcilerde hesaplanmaya uygun değildir. Bunun için bu ifade mutlak değer içindeki ifadeye dönüştürülür. Şekil 5.10'da Sobel filtresinin yatay ve düşey matrisleri gösterilmiştir. Burada düşey eksen maskesi, düşey doğrultudaki kenarları belirlerken yatay doğrultudaki gürültünün azalmasını sağlar. Yatay eksen maskesi ise yatay doğrultudaki kenarları belirlemede ve düşey doğrultudaki gürültünün azalmasında etkilidir [85].

-1	0	1
-2	0	2
-1	0	1

-1	-2	-1
0	0	0
1	2	1

Şekil 5.10. Sobel filtre matrisleri G_x ve G_y

Şekil 5.11’de gösterildiği gibi, Sobel filtresinde orjinal görüntüden, hedef piksel merkezde olmak üzere 3x3’lük piksel komşuluğu seçilir ve 3x3’lük maske ile çarpılıp, çıkan 9 çarpım sonucu toplanır. Elde edilen toplam, oluşturulan yeni matriste hedef pikselin yerine yazılır.



Şekil 5.11. Sobel filtresi ve çıkışı [86].

Sobel filtrenin hesaplaması daha yavaş olsa da daha büyük konvolüsyon çekirdeği, giriş görüntüsünü daha fazla yumuşatır ve böylece filtreyi gürültüye karşı daha az duyarlı hale getirir [81]. Maskenin genişliği ne kadar büyükse, gürültüye karşı duyarlılığı o kadar düşük olur ve filtre aynı zamanda benzer kenarlar için önemli ölçüde daha yüksek çıktı değerleri üretir. Ayrıca yumuşatma etkisi nedeniyle orijinal kenarlardan daha kalın kenarlar oluşturur.

5.5.6. Prewitt Kenar Çıkarma Filtresi

Gradyantını alarak görüntünün birinci türevindeki en yüksek ve en düşük olduğu yerleri bularak kenarları çıkarmaya yarayan bir diğer yöntem de Prewitt filtresidir [87]. Bu filtrede de Sobel’deki gibi 3x3’lük iki adet konvolüsyon matrisi kullanılır. Şekil 5.12’de Prewitt filtresinin yatay ve düşey matrisleri gösterilmiştir.

-1	-1	-1
0	0	0
1	1	1

-1	0	1
-1	0	1
-1	0	1

Şekil 5.12. Prewitt filtre matrisleri G_y ve G_x

5.5.7. Robert Kenar Çıkarma Filtresi

Robert kenar çıkarma filtresinde 2x2'lik iki adet konvolüsyon matrisi kullanılır. Gürültüye karşı hassas filtrelerdir [87]. Robert filtresinin yatay ve düşey matrisleri Şekil 5.13'te gösterilmiştir.

1	0
0	-1

0	1
-1	0

Şekil 5.13. Robert filtre matrisleri G_y ve G_x

5.5.8. Canny Kenar Çıkarma Filtresi

Canny kenar çıkarma algoritmasının ilk adımı, görüntüyü filtreleyerek pürüzsüz hale getirmektir. Optimal kenar algılama operatörünün en iyi yaklaşımı olan Gauss fonksiyonunun ilk türevini alır. Daha sonra görüntüyü sırasıyla satır ve sütuna göre düzleştirmek için uygun 1-d Gauss fonksiyonu seçilir. Yani görüntü matrisine konvolüsyon işlemi uygulanır. Konvolüsyon işleminde Canny algoritması, genellikle görüntüyü yumuşatmak ve gürültüden kurtulmak için Denklem 5.24'teki iki boyutlu Gauss fonksiyonunu kullanır [88].

$$G(x, y) = \exp \left[-\frac{(x^2 + y^2)}{2\sigma^2} \right] / 2\pi\sigma^2 \quad (5.24)$$

Burada σ , Gauss filtre parametresini gösterir ve yumuşatılmış görüntüyü kontrol eder.

Canny algoritmasının ikinci adımı, görüntünün gradyan hesabıdır. Görüntünün gradyanının büyüklüğü ve yönü hesaplanır. Bunun için 2x2 komşu alanın sınırlı farkını kullanır. X ve Y yönlerinde, birinci mertebeden kısmi türev yaklaşımı Denklem 5.25 ve 5.26'da verilmiştir.

$$E_x[i, j] = (I[i+1, j] - I[i, j] + I[i+1, j+1] - I[i, j+1]) / 2 \quad (5.25)$$

$$E_y[i, j] = (I[i, j+1] - I[i, j] + I[i+1, j+1] - I[i+1, j]) / 2 \quad (5.26)$$

Görüntü gradyan hesaplama operatörleri Denklem 5.27 ve 5.28’de gösterilmiştir.

$$G_x = \begin{pmatrix} -1 & 1 \\ -1 & 1 \end{pmatrix} \quad (5.27)$$

$$G_y = \begin{pmatrix} 1 & 1 \\ -1 & -1 \end{pmatrix} \quad (5.28)$$

Buna göre gradyanın büyüklüğü ve yönü hesaplanabilir. Görüntü gradyan büyüklüğü;

$$||M(i, j)|| = \sqrt{E_x[i, j]^2 + E_y[i, j]^2} \quad (5.29)$$

Görüntü gradyanının yönü ise Denklem 5.30 ile gösterilir.

$$\theta(i, j) = \arctan\left(\frac{E_y[i, j]}{E_x[i, j]}\right) \quad (5.30)$$

$M[i, j]$ gradyan büyüklüğü görüntüsünü elde ettikten sonra, kenarları doğru bir şekilde konumlandırmak için görüntü üzerinde Maksimum Olmayan Bastırma İşlemi (Non-Maximum Suppression (NMS)) gerçekleştirmek gerekir. NMS süreci, her kenarın bir piksel genişliğinde olmasını sağlar. Canny algoritmasında, gradyan yönü boyunca gradyan büyüklüğüne enterpolasyon yapmak için sekiz yönden oluşan 3x3’lük komşu alanı kullanılır. $M[i, j]$ büyüklüğü gradyan yönündeki iki enterpolasyon sonucundan daha büyükse, kenar noktası olarak işaretlenir, aksi halde kenar olmayan nokta olarak işaretlenir. Böylece kenarlar elde edilir. Bu yöntem, son kenar görüntüsünün kenarında gürültünün etkisini azaltır.

Bu tez çalışmasında, EA’nın güvenli sürüşü üzerine çalışmalar yapılarak çeşitli görüntü işleme algoritmaları kullanılmıştır. Bu yüzden bu bölümde sayısal görüntü işleme tekniği ve algoritmaları anlatılmıştır. Bir sonraki bölümde ise tekerlek içi motorlu EA için EDS tasarımı, oluşturulan deneysel sistem ve alınan sonuçlar detaylı olarak verilecektir.

6. TEKERLEK İÇİ MOTORLU ELEKTRİKLİ ARAÇLAR İÇİN ELEKTRONİK DİFERANSİYEL SİSTEM TASARIMI

6.1. Giriş

Bu tez çalışmasında, dört tekerden tahrikli bir EA için Elektronik Diferansiyel Sistem (EDS) tasarlanmış ve EA'nın dinamik modeli oluşturulmuştur. Aracın direksiyon açısına ve hızına bağlı olarak, dört tekerin dönmesi gereken hız değerleri Ackermann-Jeantand geometrisinden çıkartılan matematiksel denklemlere göre hesaplanmıştır. Bunun için CoDeSys V2.3 yazılımı kullanılmıştır.

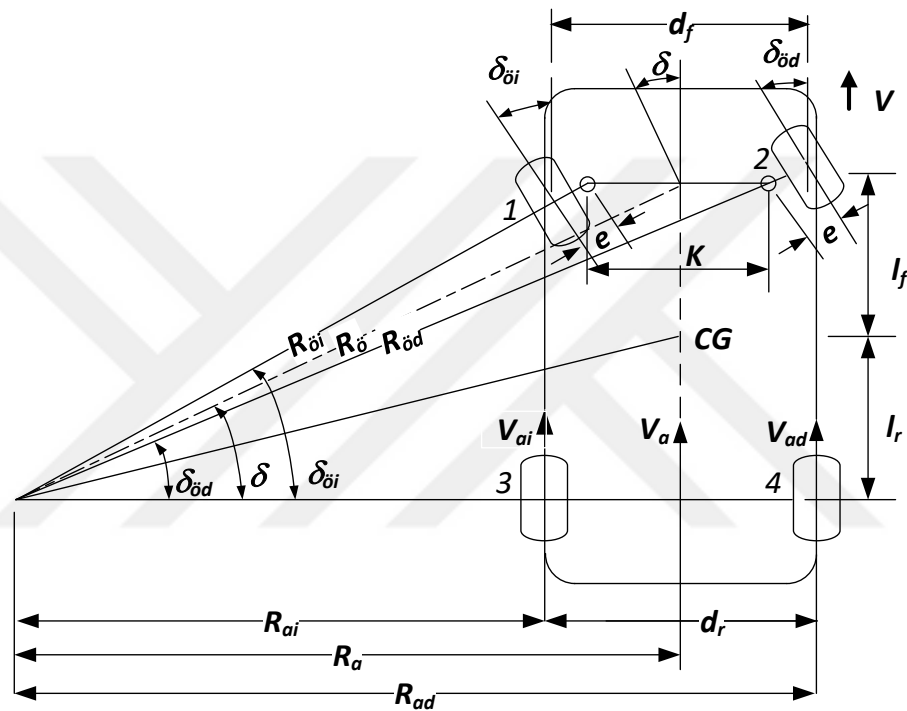
6.2. Elektronik Diferansiyel Sistemler

Araç seyir dinamikleri, yol boyunca maruz kalan araçların ve kuvvetlerin hareketleriyle ilgilendir. Araç dinamiklerini etkileyen ana kuvvetler, lastiklerin yolla temas ettiği yüzeyler ile bu kuvvetler arasındaki ilişkiler sonucunda ortaya çıkar ve bu kuvvetler aracın performansını doğrudan etkiler. Tekerleklerin uyguladığı kuvvet, aracın çekiş gücünü oluşturur. Günümüzde önden ve arkadan çekişli sistemlerin üstün özelliklerine rağmen dezavantajları araç performansını olumsuz etkilemektedir. Dört tekerlekten çekiş sistemleri bu sorunların üstesinden gelmek için tasarlanmıştır. Bu nedenle motorlar tekerleklerin içine yerleştirilmiştir. Böylece gelişen motor tasarımları sayesinde EA'ların ağırlığı azaltılmaktadır. Aynı zamanda hareket eden bir araçta her zaman daha yüksek derecede bir çekiş sürekliliği sağlanır. Özellikle zorlu yol koşullarında dört tekerlekten çekiş sistemini çamurlu, ıslak, karlı ve buzlu yollarda kullanmak uygundur. Dolayısıyla, her bir tekerlek içi SMSM bağımsız olarak kontrol edilebilir. EA, bu motorlar aracılığıyla sistemin sürüş dinamiklerinden elde edilen bilgilerle de kontrol edilir. Bu kontrol yapısı, EA'ların yol tutuş kabiliyetini arttırdığı için virajlarda savrulmayı azaltarak üstünlük sağlamaktadır.

Bu çalışmada, tekerleklere ayrı ayrı dört adet DSMSM yerleştirilmiştir. Böylece, moment dört motora eşit olarak dağıtılır ve her bir DSMSM bağımsız olarak kontrol edilir. Araçlarda kullanılan geleneksel mekanik diferansiyeller ağır sistemler olduklarından EA'nın ağırlığını azaltmak için elektronik diferansiyeller tercih edilir. Bunun getirdiği önemli avantajlar vardır. EA'nın performansını arttırarak mekanik kayıpları azaltır. EDS'de mekanik aktarma organları olmadığı için her bir tekerlek bağımsız olarak kontrol edilebilmektedir. Yol durumu değiştiğinde tekerlek hızları da farklı olmalıdır. Buna göre dış tekerlek hızı, dönüş yönüne göre iç tekerleğin hızından fazla olmalıdır.

6.3. Ackermann Jeantand Modeli

Bu çalışmada, dört tekerden tahrikli bir EA'nın EDS'sini tasarlamak için Ackermann-Jeantand geometrisi kullanılmıştır. Direksiyon açısına ve aracın hız bilgisine göre EA'nın dört tekerlek içi motor hızları bu modelden çıkartılan matematiksel denklemlere göre elde edilmiştir. Buna göre araç viraja girerken güvenli sürüşün sağlanabilmesi için, dış tekerlek hızı iç tekerleğin hızından daha yüksek olmalıdır. Tasarlanan EDS'nin Ackermann-Jeantand modeli Şekil 6.1'de gösterilmiştir.



Şekil 6.1. Ackermann-Jeantand modeli [1].

EA düz bir yolda giderken, tüm tekerleklerin hızları birbirine eşittir ve dönüş açısı sıfırdır. Dönüş açısı sıfırdan farklıysa, araç viraja giriyor demektir. Buna göre tekerlek hızları farklı olacaktır. EDS'nin Şekil 6.1'de gösterilen Ackermann-Jeantand geometrisinden çıkartılan matematiksel denklemleri sırasıyla Denklem 6.1-6.11'de verilmiştir.

Buna göre ön iç tekerlek açısı;

$$\delta_{\delta i} = \arctan \left[\frac{(l_f + l_r) \cdot \tan(\delta)}{(l_f + l_r) - ((K/2) \cdot \tan(\delta))} \right] \quad (6.1)$$

Burada, δ , K , l_f ve l_r sırasıyla direksiyon açısı, sağ ve sol king pim arasındaki mesafe ve ön ve arka tekerleklerin ağırlık merkezine olan mesafeleridir.

Ön dış tekerlek açısı;

$$\delta_{od} = \arctan \left[\frac{(l_f + l_r) \cdot \tan(\delta)}{(l_f + l_r) + ((K / 2) \cdot \tan(\delta))} \right] \quad (6.2)$$

Ön iç ve dış tekerleklerin dönme yarıçapları;

$$R_{oi} = \frac{l_f + l_r}{\sin(\delta_{oi})} \quad (6.3)$$

$$R_{od} = \frac{l_f + l_r}{\sin(\delta_{od})} \quad (6.4)$$

Arka iç ve dış tekerleklerin dönüş yarıçapları;

$$R_{ai} = \frac{(l_f + l_r)}{\tan(\delta)} - \frac{d_r}{2} \quad (6.5)$$

$$R_{ad} = \frac{(l_f + l_r)}{\tan(\delta)} + \frac{d_r}{2} \quad (6.6)$$

ile hesaplanabilir.

Burada d_r arka tekerlekler arasındaki mesafedir.

EA'nın ağırlık merkezinin yarıçapı Denklem 6.7 ile ifade edilir.

$$R_{cg} = \sqrt{(R_{ai} + (d_r / 2))^2 + (l_r)^2} \quad (6.7)$$

Ön ve arka iç ve dış tekerleklerin açısal hızları sırasıyla Denklem 6.8-6.11'den bulunabilir.

$$\omega_{oi} = \frac{V \cdot R_{oi}}{(R_{cg}) \cdot r} \quad (6.8)$$

$$\omega_{od} = \frac{V \cdot R_{od}}{(R_{cg}) \cdot r} \quad (6.9)$$

$$\omega_{ai} = \frac{V \cdot R_{ai}}{(R_{cg}) \cdot r} \quad (6.10)$$

$$\omega_{ad} = \frac{V \cdot R_{ad}}{(R_{cg}) \cdot r} \quad (6.11)$$

Burada V ve r sırasıyla aracın hızı ve tekerleğin yarıçapıdır.

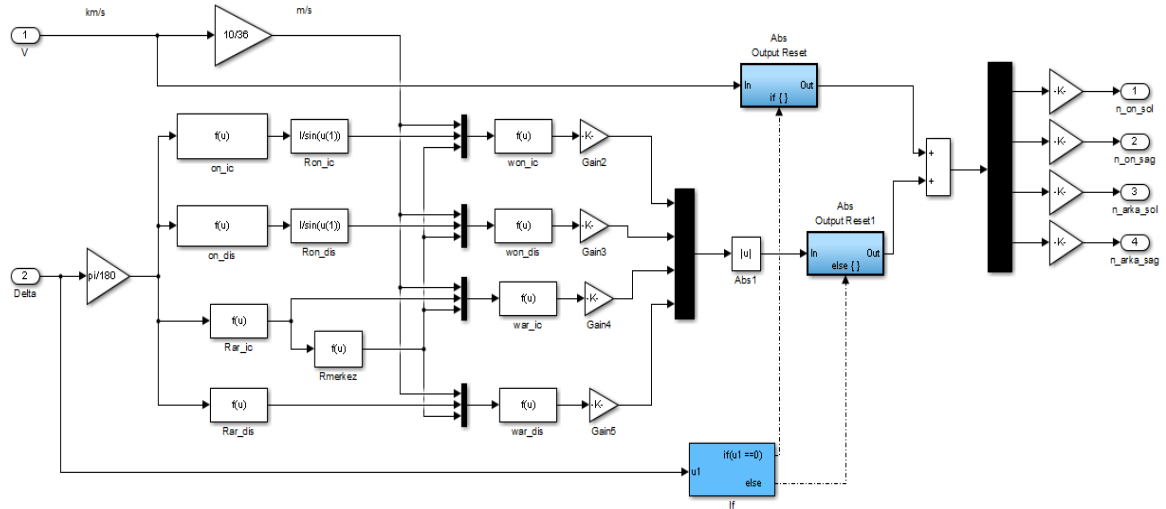
Bu denklemler Codesys yazılımında kullanılarak EA'nın dört tekerlek hızları hesaplanmaktadır. Denklemlerde yer alan bazı parametreler bir araçtan alınan sabit parametrelerdir ve değerleri Tablo 6.1'de gösterilmiştir.

Tablo 6.1. EDS tasarımında kullanılan parametreler

Parametreler	Değerleri (m)
l_f	1.45
l_r	0.835
d_r	1.35
r	0.395
K	1.219

6.4. Elektronik Diferansiyel Sistemin Modellenmesi

Bu tez çalışmasında, EDS'nin modellenmesi öncelikle Matlab/Simulink'te gerçekleştirilmiştir. Direksiyon açısı ve aracın hızına göre, Ackermann-Jeantand geometrisinden çıkartılan denklemler kullanılarak dört tekerlek içi motorun hızları bu modelden hesaplanmıştır. Tasarlanan Simulink model Şekil 6.2'de gösterilmektedir.

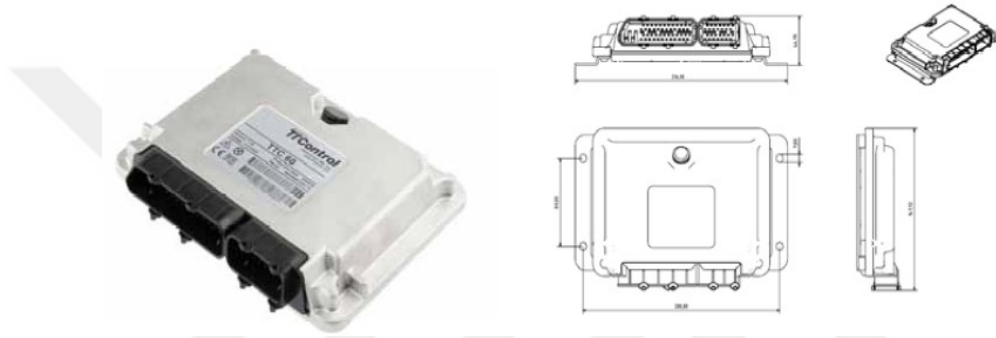


Şekil 6.2. EDS'nin Matlab/Simulink modeli

Şekil 6.2'deki modele göre, araç düz yolda giderken yani direksiyon açısı sıfır derece iken dört tekerlek hızı birbirine eşittir. Direksiyon açısı sıfırdan farklı olursa araç viraja giriyor demektir. Bu durumda dört tekerleğin hızları birbirinden farklı olacak ve dıştaki tekerleklerin hızı, içteki tekerleklerin hızından yüksek olacaktır. Böylece sistemin stabilitesi sağlanacaktır.

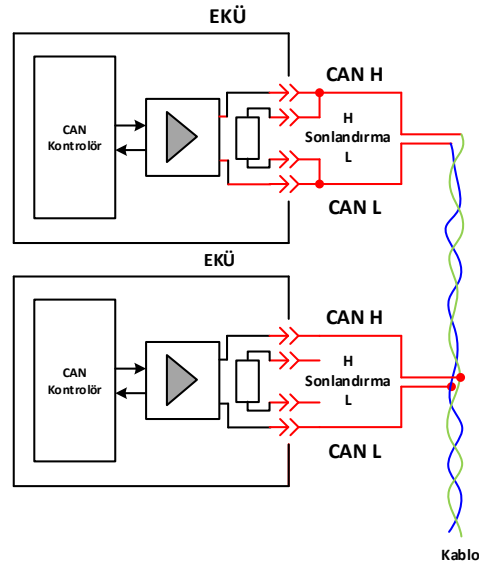
6.5. HY-TTC60 İşlemci

Bu tez çalışmasında 16/32-bit Infineon XC2287 mikrodenetleyicisi olan HY-TTC60 işlemci kullanılmıştır. Bu işlemci, TTControl tarafından üretilen, güç ve kontrol kartlarından oluşan 16 bit HY-TTC 50 kontrolör ailesinin gelişmiş bir modelidir [59]. TTControl, TTTech ve HYDAC Uluslararası iki şirketin ortaklığıyla oluşmuş, operatör ara yüzleri ve elektronik kontrol sistemleri geliştiren bir firmadır. Çok sayıda analog ve dijital giriş, 594 kb RAM, RS-232 ve LIN seri ara yüzleri ve iki CAN veri hattına sahiptir. CoDeSys 2.3 veya C/C++ ile programlanır. Bu nedenle endüstriyel uygulamalarda ve araçlarda yaygın olarak kullanılan HY-TTC60 bu çalışmada da tercih edilmiştir. Bu işlemci Şekil 6.3'te gösterilmiştir.



Şekil 6.3. HY-TTC 60 işlemci [89].

HY-TTC60'ın CAN-Bus veri hattı Şekil 6.4'te görülmektedir.



Şekil 6.4. CAN veri hattı [90].

HY-TTC60'ın bağlantı uçları ise Tablo 6.2'de verilmiştir. CAN-Bus hattında hat başında ve sonundaki 120 Ω 'luk dirençleri devreye almak için bazı pinler kısa devre edilmelidir. Bu çalışmada CAN 0 hattı kullanıldığı için 273 numaralı pin 280 ile 279 numaralı pin de 272 ile kısa devre edilir.

Tablo 6.2. HY-TTC60 bağlantı pinleri [91].

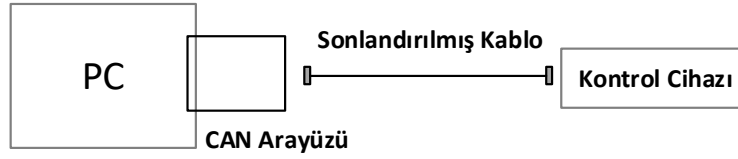
	Pin	Görevi	Açıklama
CAN 0	273	CAN L (Siyah)	Hat başı olduğu için 280 ile kısa devre edilir.
CAN 0	279	CAN H (Kırmızı)	Hat sonu olduğu için 272 ile kısa devre edilir.
CAN 0	272	Sonlandırma direnci	
CAN 0	280	Sonlandırma direnci	
CAN 1	265	CAN H (Siyah)	Hat sonu olduğu için 258 ile kısa devre edilir.
CAN 1	259	CAN L (Kırmızı)	Hat başı olduğu için 266 ile kısa devre edilir.
CAN 1	258	Sonlandırma direnci	
CAN 1	266	Sonlandırma direnci	
Besleme	101	24 V Besleme	
Besleme	127	24 V Besleme	
Besleme	128	24 V Besleme	
Besleme	102	GND	
Besleme	271	24 V Besleme	

6.6. PCAN-USB CAN Hattı

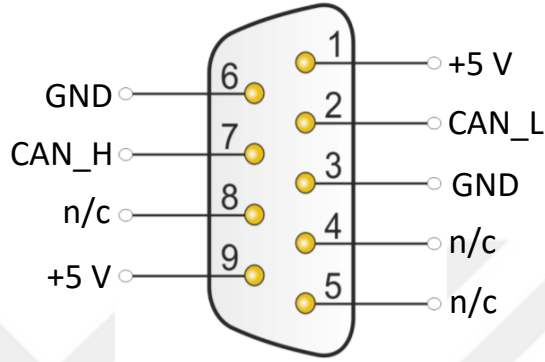
Bu tez çalışmasında kullanılan PCAN-USB adaptörü CAN-Bus hattını oluşturur. Bu adaptör, Şekil 6.5'te görüldüğü gibi USB konektörü ile bilgisayarı bir CAN veri yoluna bağlar. Böylece USB üzerinden gerilim beslemesi yapılır. Bir ucu bilgisayarın USB girişine bağlı, diğer ucu CAN-Bus'ın CAN-H ve CAN-L uçlarını oluşturur. Bağlantı şeması Şekil 6.6'da gösterilmiştir. CAN, CiA standardı Şekil 6.7'de görüldüğü gibi DS 102-1'e göre 9 pinli bir SUB-D soketi ile bağlanır. CAN sinyalleri ile ana bilgisayar arasında 500 V'a kadar galvanik izolasyon, entegre bir DA/DA dönüştürücü yardımıyla elde edilir. CAN hattına alınan ve hattan gönderilen mesajlar PCAN-View programı üzerinden görülmektedir.



Şekil 6.5. PCAN-USB kablosu



Şekil 6.6. PCAN-USB bağlantı uçları [92].



Şekil 6.7. PCAN-USB bağlantı şeması [92].

Özellikleri aşağıdaki gibi sıralanabilir:

- Üzerine entegre edilmiş bir led bulunur. Bu da PCAN-USB'nin çalışma durumunu gösterir.
- CAN özellikleri 2.0A (11-bit ID) ve 2.0B (29-bit ID) ile uyumlu 9 pinli soket üzerinden CAN veri yolu bağlantısı sağlanır.
- 5 kbit/s'den 1 Mbit/s'ye kadar yüksek hızlarda veri iletimi mümkündür.
- 16 MHz saat frekansı vardır.
- Donanım, yazılım aracılığıyla sıfırlanabilir.
- Harici bir alıcı-verici için CAN bağlantısına 5 V besleme bir bağlantı teli ile bağlanabilir.

6.7. CoDeSys Yazılımı

Açılımı “Controlled Development System” olan CoDeSys, “Kontrollü Geliştirme Sistemi” anlamına gelir. Uluslararası IEC 61131-3 standartlarına uygun, kontrol uygulamaları geliştirmek için kullanılan, Programlanabilir Mantıksal Denetleyiciler (Programmable Logic Controller, PLC) için eksiksiz bir geliştirme ortamıdır. Bu tez çalışmasında kullanılan HY-TTC60 işlemcisi CoDeSys ile programlanır.

Düzenleyicilerin ve hata ayıklama işlevlerinin kullanımı, Visual C++ gibi gelişmiş programlama dillerinin kanıtlanmış geliştirme programı ortamlarına dayanır. CoDeSys'te beş programlama dili mevcuttur. Bunlar:

- Instruction List (IL): Düşük seviyeli bir dildir ve Assembly'i andırır.
- Structured Text (ST): Blok yapılı ve Pascal'a benzeyen yüksek seviyeli bir programlama dildir.
- Ladder Diagram (LD): Merdiven mantığı, röle lojik donanımının devre şemalarına dayanan grafiksel bir diyagramla bir programı temsil eden bir programlama diline dönüşmüştür. Bu mantık, endüstriyel kontrol uygulamalarında PLC'ler için yazılım geliştirmede kullanılır.
- Function Block Diagram (FBD): Giriş ve çıkış değişkenleri arasındaki fonksiyonu tanımlayabilen PLC tasarımı için geliştirilmiş bir grafik dilidir.
- Sequential Function Chart (SFC): Sıralı fonksiyon tablosu (SFC), sıralı uygulamalar ve akış tabloları içeren PLC'ler için kullanılan bir grafik programlama dilidir. SFC standardı, kontrol sistemleri için fonksiyon çizelgelerinin hazırlanması olarak tanımlanır.

6.7.1. CoDeSys Fonksiyonları

CoDeSys, Program Organizasyon Birimi (Program Organization Unit, POU), veri türleri (data types), görüntüleme öğeleri (visualizations) ve kaynaklar (resources) olmak üzere farklı nesnelerden oluşur. POU'lar, yazılımın oluşturulduğu kısımdır. Bir projede tanımlanan ilk POU PLC_PRG olarak geçer. Oluşturulan tüm POU'lar PLC_PRG içinde çağırılır. Tüm alt fonksiyonlar burada çalıştırılır.

Projede kullanılan adreslerin doğruluğunu kontrol etmek için öncelikle PLC konfigüre edilir. Daha sonra, problemi çözmek için gereken POU'lar istenilen programlama dilinde oluşturulabilir. Programlama tamamlandıktan sonra, proje derlenir ve varsa hataları kaldırılır. Tüm hatalar giderildikten sonra simülasyon etkinleştirilir. Simüle edilmiş PLC'de oturum açılır ve proje PLC'ye yüklenir. Böylece çevrimiçi moda geçilir. Daha sonra PLC konfigürasyon kısmından proje test edilir. Bunun için, giriş değişkenleri manuel olarak girilir ve çıkışların istenildiği gibi olup olmadığı gözlemlenir. POU'lardan da değişkenler gözlemlenebilir. Programlama hatası durumunda kesme noktaları ayarlanabilir. Süreç böyle bir kesme noktasında durursa, tüm proje değişkenlerinin değerlerini zamanın bu noktasında incelemek mümkündür. Sırasıyla çalıştırılarak programın mantıksal doğruluğu kontrol edilebilir.

6.7.2. CoDeSys Proje Bileşenleri

Proje

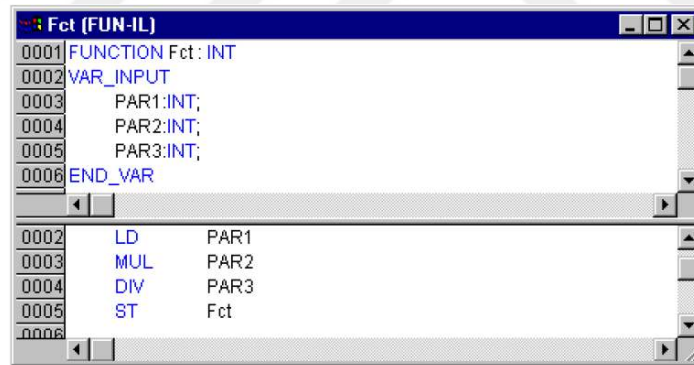
Bir proje, bir PLC programındaki tüm nesneleri içerir. İlk olarak bir isim verilir ve proje kaydedilir. Bir projeye dâhil edilen birimler; POU'lar, veri türleri, görselleştirmeler, kaynaklar ve kütüphanelerdir.

Program Organizasyon Birimi (POU)

Program Organizasyon Birimleri (POU), fonksiyonlar, fonksiyon blokları ve programları içerir. Bir POU, açıklama ve gövde olmak üzere iki kısımdan oluşur. Açıklama kısmında kullanılan değişkenler tanımlanır. Gövde ise, yazılımın oluşturduğu kısımdır. Yazılımda, IL, ST, CFC, FBD, SFC, LD gibi tüm IEC standardındaki programlama dilleri kullanılabilir. Bu POU'ları kullanabilmek için "standard.lib" kütüphanesi eklenmelidir. Bir POU, diğer POU'ları çağırabilir.

Fonksiyon

Bir fonksiyon, işlendiğinde tam olarak bir veri ögesi veren ve metin dillerinde ifade edilebilen bir POU'dur. Codesys'te fonksiyon tanımlanırken, "FUNCTION Fct: INT" şeklinde yazılır. Bir fonksiyon, FUNCTION sözcüğüyle başlar. Fonksiyonunun adından sonra iki nokta üst üste konulur ve ardından veri tipi belirtilir. Bu örnekte "Integer (INT)" olarak tanımlanmıştır. Ek olarak, fonksiyona bir sonuç atanmalıdır. Bu, fonksiyon adının bir çıkış değişkeni olarak kullanıldığı anlamına gelir. IL'de bir fonksiyon, yalnızca bir adımın içinde veya bir geçiş içinde kullanılabilir. ST'de, bir ifadede işlenen olarak yer alabilir. Üç giriş değişkenini alan ve ilk ikisinin çarpımını üçüncüye bölen bir fonksiyonun IL dilindeki örneği Şekil 6.8'de verilmiştir.



Şekil 6.8. IL dilinde bir fonksiyon örneği [93].

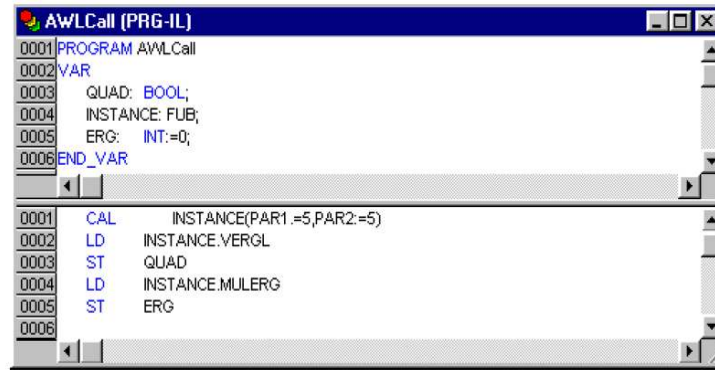
Fonksiyon Bloğu

Fonksiyon bloğu, bir işlemde bir veya daha fazla değer veren bir POU'dur. Bir fonksiyon bloğu, "FUNCTION_BLOCK" ile başlar. Fonksiyon bloğunun kopyaları veya örnekleri oluşturulabilir. Her bir örnek kendi tanımlayıcısına (örnek adı) ve girişlerini, çıkışlarını ve dâhili değişkenlerini içeren bir veri yapısına sahiptir. Örnekler yerel veya global değişkenler olarak bildirilirken, fonksiyon bloğunun adı bir tanımlayıcının türü olarak belirtilir.

FUB fonksiyon bloğunun INSTANCE adında bir örneği:

INSTANCE: FUB;

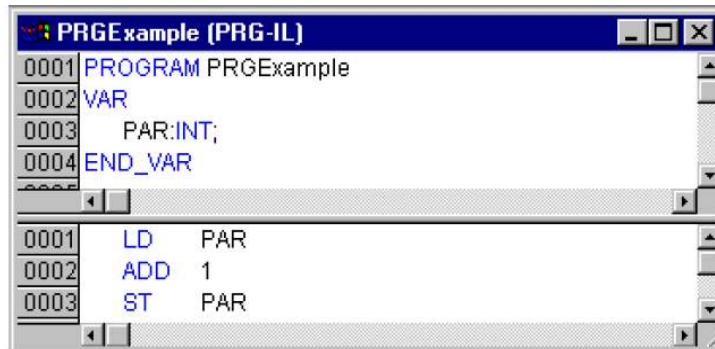
Fonksiyon blokları her zaman yukarıda açıklanan örnekler aracılığıyla çağrılır. İki giriş ve iki çıkış değişkenli bir fonksiyon bloğunun IL dilindeki örneği Şekil 6.9’da gösterilmiştir.



Şekil 6.9. IL dilinde bir örnek fonksiyon bloğu.

Program

Bir program, işlem sırasında birkaç değer döndüren bir POU'dur. Program çalıştırdıktan sonra alınan değerler tekrar çalıştırılana kadar korunur. Programlar çağırılabilir. Bir POU bir programı çağırırsa ve bu şekilde programın değerleri değiştirilirse, program başka bir POU içinden çağırılmış olsa bile bu değişiklikler programın bir sonraki çağırılışına kadar korunur. Bu, bir fonksiyon bloğu çağırmaktan farklıdır. Bir fonksiyon bloğunun sadece verilen örneğindeki değerler değiştirilir. Bu nedenle, bu değişiklikler yalnızca aynı durum çağırıldığında bir rol oynar. Bir program açıklaması “PROGRAM” sözcüğü ile başlar ve “END_PROGRAM” ile biter. Programda girişler “:=” işaretinden sonra, çıkışlar ise “=>” işaretinden sonra IL ve ST dillerinde yazılarak atanabilir. Bir program örneği Şekil 6.10’da gösterilmiştir.



Şekil 6.10. IL dilinde örnek bir program.

PLC Program

PLC_PRG her programda olması gereken bir POU'dur. Bu POU, kontrol döngüsü başına bir kez çağırılır. Codesys'te yeni bir proje oluşturulduktan sonra 'Proje' ve 'Nesne Ekle' komutları kullanılarak PLC_PRG adında bir POU oluşturulur. Oluşturulan tüm POU'lar PLC Program içinde tanımlanır. Bir PLC_PRG örneği;

```
PROGRAM PLC_PRG
```

```
VAR
```

```
Inst : Counter;
```

```
END_VAR
```

Kaynaklar

Projeyi yapılandırmak, düzenlemek ve değişkenlerin değerlerini izlemek için kaynaklar kısmı kullanılır. Yapılan işlemlerin detayları şunlardır:

- Proje veya ağ genelinde kullanılabilen Global Değişkenler (Global Variables)
- Projeye kütüphane eklemek için kütüphane yöneticisi
- Donanımı yapılandırmak ve giriş/çıkışları tanımlamak için PLC Konfigürasyon kısmı (PLC Configuration)
- Programın çalışması boyunca programa rehberlik etmek için İş Konfigürasyonu (Task Configuration)
- Başlangıç değişken değerlerini ayarlamak ve görüntülemek için İzleme ve Alma Yöneticisi (Watch and Receipt Manager)
- Hedeflenen sistem için son bir konfigürasyon gerekli olduğunda, hedef sistem ayarları (Target system settings) yapılabilir.
- Hedef sisteme ve CoDeSys'te yapılan hedef ayarlarına bağlı olarak aşağıdaki kaynaklar da projede mevcut olabilir:
 - Örneklem İzleme İzni (Sampling Trace): Değişken değerlerin grafiksel gösterimi
 - Parametre Yöneticisi (Parameter Manager): Bir ağdaki diğer kontrolörler ile veri alışverişi için kullanılır.
 - PLC-Tarayıcı (PLC-Browser): Kontrolör monitörüdür.
 - Araçlar (Tools): CoDeSys içinden harici araç programlarını çağırmak için hedefe bağlı kullanılabilen kısımdır.

Kütüphaneler

POU'ları, veri türlerini ve global değişkenleri kullanıcı tanımlı değişkenler gibi kullanabilmek için projeye pek çok kütüphane dahil edebilir. Standard.lib ve util.lib, Codesys'te kullanılan ana kütüphanelerdir.

Veri Tipleri

Standart veri tiplerinin yanı sıra, kullanıcı kendi veri tiplerini de tanımlayabilir. Codesys'te kullanılan standart veri tipleri şunlardır:

- **BOOL:** BOOL tipi değişkenlere TRUE ve FALSE değerleri verilebilir. 8 bit hafıza alanı vardır.
- **Integer:** Integer değişkenler, BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT ve UDINT'tir. Farklı sayı türlerinin her biri farklı bir değer aralığını kapsar. Integer veri tipleri için Tablo 6.3'teki limit aralıkları geçerlidir.

Tablo 6.3. Integer veri tipleri

Veri Tipi	Alt limit	Üst Limit	Hafıza Alanı
BYTE	0	255	8 Bit
WORD	0	65535	16 Bit
DWORD	0	4294967295	32 Bit
SINT	-128	127	8 Bit
USINT	0	255	8 Bit
INT	-32768	32767	16 Bit
UINT	0	65535	16 Bit
DINT	-2147483648	2147483647	32 Bit
UDINT	0	4294967295	32 Bit

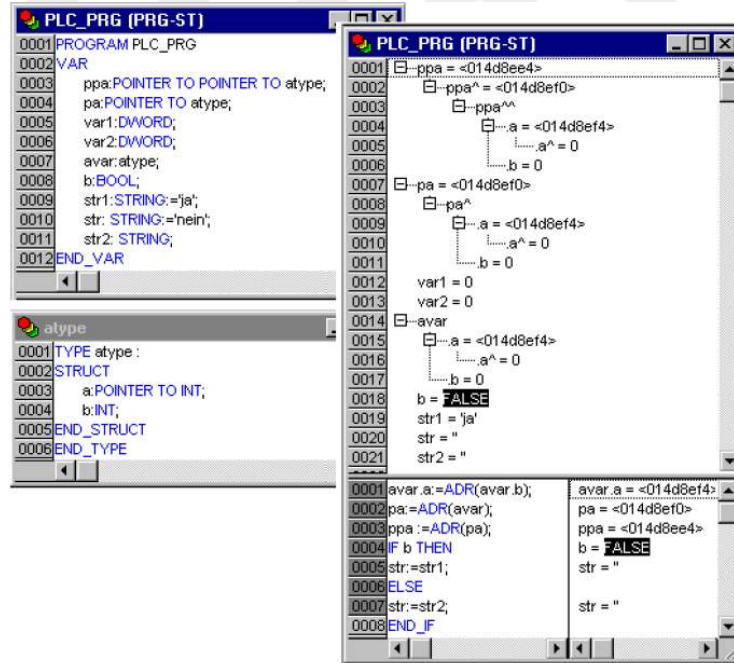
- **REAL/LREAL:** REAL ve LREAL kayan nokta sayı tipleridir. Ondalık sayılar için kullanılır. 32 bit bellek alanı REAL için ve 64 bit LREAL için ayrılmıştır.
- **STRING:** STRING türü bir değişken herhangi bir karakter dizisi içerebilir. Açıklama kısmındaki boyut girişi, değişken için ne kadar bellek alanı ayrılması gerektiğini belirler. Dizideki karakter sayısını ifade eder ve parantez veya köşeli parantez içine yerleştirilebilir. Herhangi bir boyut belirtilmezse varsayılan boyut olan 80 karakter kullanılır.

Görüntüleme

CoDeSys, proje değişkenlerini görüntüleyebilmek için görselleştirmeler sağlar. Görselleştirme yardımıyla geometrik öğeler çevrimiçi olarak çizilebilir. Çevrimiçi modda, bunlar daha sonra belirtilen değişken değerlere cevap olarak form/reng/metin çıktılarını değiştirebilir.

CoDeSys'te Değişkenleri İzleme

Codesys'te hatta bağlandıktan sonra çevrimiçi modda, tüm değişkenlerin değerleri gerçek zamanlı olarak okunur. Ayrıca, İzleme ve Alma Yöneticisindeki ve bir görselleştirmedeki değişkenlerin mevcut değerleri okunabilir. Fonksiyon bloklarının örneklerinden gelen değişkenler izlenecekse, önce karşılık gelen örnek açılmalıdır. VAR_IN_OUT değişkenlerinin izlenmesinde, referansı kaldırılan değer çıkıştır. Göstergelerin izlenmesinde, hem gösterge hem de referansta bulunmayan değer açıklama kısmında çıkış olarak alınır. PLC_PRG'de değerlerin okunması Şekil 6.11'de gösterilmiştir.



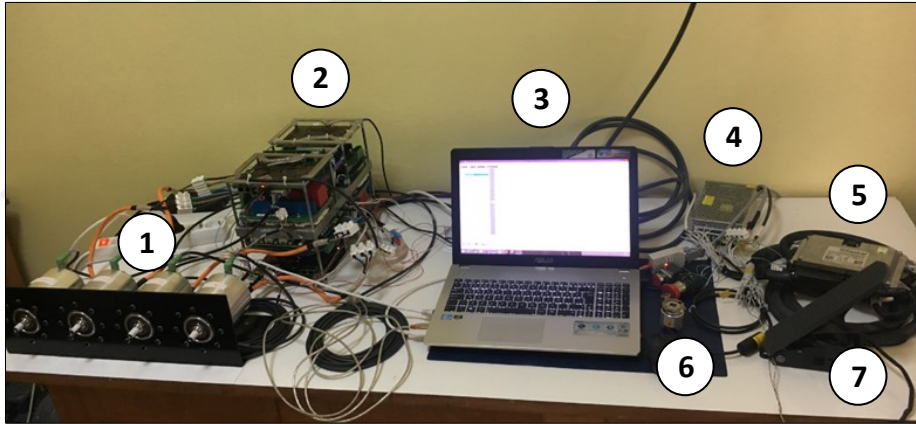
Şekil 6.11. Değişkenlerin izlenmesi

Simülasyon

Simülasyon sırasında, oluşturulan PLC programı PLC'de değil, CoDeSys'in üzerinde çalıştığı hesap makinesinde işlenir. Çevrimiçi işlemlerin tümü gerçekleştirilebilir. Bu da programın mantıksal doğruluğunu PLC donanımı olmadan test etmeyi sağlamaktadır.

6.8. Deney Düzeneği

Bu tezde, dört tekerlek içi motorlu bir EDS gerçekleştirmek için Şekil 6.12'deki deney düzeneği kurulmuştur. Sistemde dört adet DSMSM ve sürücüleri, HY-TTC60 ARM işlemci, CANopen enkoder, gaz pedalı, PC ve 24 V DA besleme kaynağı bulunmaktadır. DSMSM'ler ve sürücüleri Şekil 6.13'te ayrıntılı olarak gösterilmektedir. Sistemde kullanılan PCAN-USB kablolu CAN-Bus hattını oluşturmaktadır. Tüm sürücüler ve enkoder CAN uyumlu olduklarından doğrudan CAN-Bus hattına bağlanır. Gaz pedalı ise analog bir sensör olduğundan HY-TTC60 işlemci üzerinden hatta bağlanarak 1 Mbit/s'de haberleşme sağlanır.



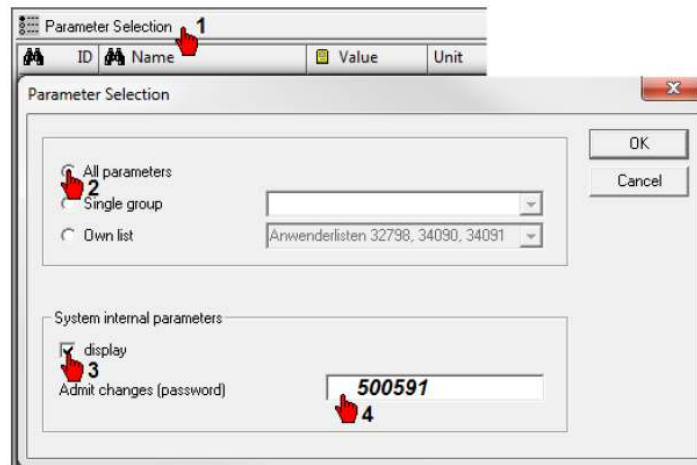
Şekil 6.12. Deney düzeneği (1. Dört adet DSMSM'ler, 2. Sürücüleri, 3. PC, 4. 24 V DC kaynak, 5. HY-TTC60 işlemci, 6. Enkoder, 7. Gaz pedalı).



Şekil 6.13. Dört adet DSMSM ve sürücüleri

Bu tezde kullanılan DSMSM sürücülerinin CAN-Bus hattında haberleşmesini sağlamak için her sürücü için parametre ayarları yapılır. Bunun için Aipex Lite Programı kullanılır. Bu program AMK Firması tarafından sağlanan motor sürücülerinin parametre ayarlarının yapıldığı ve çalışma durumunun gözlemlendiği bir yazılımdır. Burada yapılan ayarlar sırasıyla şöyledir:

- 1. ID34023:** CAN-Bus haberleşmesi için her sürücüye verilen ID numarası buradan ayarlanır. Programda sürücüler için 1, 2, 5 ve 6 ID numaraları kullanılmıştır.
- 2. ID34024:** Bus haberleşme hızıdır. Tüm sistem en yüksek hız değeri olan 1 Mbit/s ile haberleşmektedir.
- 3. ID34025:** Bus mod ayarıdır. Giriş değeri olan 4 H girilir.
- 4. ID34028:** Bus çıkış oranıdır. Giriş değeri olan 0X105 H girilir. Bu sistem dâhili parametresidir. Bunu aktif etmek için Şekil 6.14'te görüldüğü gibi "parameter selection"a tıklanır ve all parameters seçilir. System internal parameters'da display işaretlenerek şifre kısmına 500591 yazılır. Böylece diğer parametreler de açılmış olur. 105 H'ta 01'in anlamı döngü çıkışıdır. 5 ise ms cinsinden çıkış oranını ifade eder. Yani 5 ms'ye denk gelir.



Şekil 6.14. Tüm parametreleri aktif etme [94].

SMSM Sürücülerinin Diğer Parametre Ayarları

SMSM'lerin moment sınırlamasını aktif etmek için bu parametreler kullanılır. Bunun için Tablo 6.4'te gösterilen tüm ayarlar sırasıyla yapılmaktadır.

Tablo 6.4. SMSM sürücü parametre ayarları [94].

ID	Decimal değeri	Açıklaması
32798-2	1	Çalışma durumu (FSE=1)
32798-3	720	Maksimum batarya gerilimi
32798-4	250	Minimum batarya gerilimi
32798-5	500	Tam momentte inverter sıcaklığı
32798-6	600	Moment yokken inverter sıcaklığı
32798-7	670	Şarj sonu gerilimi için azalmanın başladığı değer
32798-8	1150	Tam momentte güç kaynağının sıcaklığı
32798-9	1250	Moment yokken güç kaynağının sıcaklığı
32798-10	300	Derin deşarj koruması için azalmanın başladığı değer

Default Parametreleri

Motor çalışma modunu ayarlamak için aşağıdaki parametreler kullanılır:

ID32800: AMK ana çalışma modudur. Giriş değeri olan 480003 H ayarlanır.

ID32796: Kontrolör RF kaynak donanımını aktif etmek için sistem parametreleri kullanılır. Motor kontrolü bu parametreyle aktif edilir. Oldukça önemlidir. Giriş değeri olan 5 dec girilir. Motor kontrolünü aktif etmek için RF donanım sinyali (X140 BE1) ve CAN sinyalleri üzerindeki motorları aktif etmek gerekir.

ID32901: Global servis bitleridir. Sürücü sistemini aktif etmek için bu parametre oldukça önemlidir. Parametre hataları olduğunda öncelikle bu kontrol edilmelidir. Giriş değeri olan 10240 girilmelidir.

ID32773: Motoru aktif olarak izlemek için genel parametreler kullanılır. Giriş değeri olan 5005 H girilmelidir.

ID113: Maksimum hızı ayarlamak için motor parametreler grubu kullanılır. Giriş değeri kullanıcı tarafından belirlenir. 20000 değeri girilmiştir.

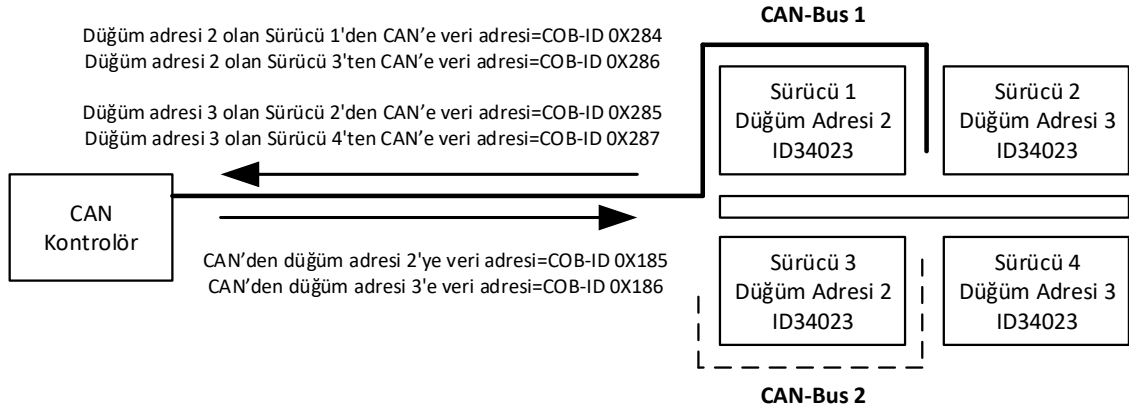
ID38 ve ID39: Pozitif ve negatif hız değerleridir. +20000 ve -20000 değerleri girilmiştir.

ID32837: DA Bus gerilimini izleme komutudur. Bunun için de 250 V değeri girilmiştir.

Parametre ayarlarında özellikle sürücülerin veri adresleri ve haberleşme hızları oldukça önemlidir. Aipex Pro'da ayarlanan değerlere göre PCAN-View programından mesajlar gönderilir.

Data Telegramlar

SMSM sürücülerine gönderilen ve sürücülerden alınan mesajları ayarlamak için data telegramlar kullanılır. Hangi veri adreslerinin kullanıldığı Şekil 6.15'te gösterilmiştir. Sürücü sisteminde görüldüğü gibi iki adet CAN hattı bulunmaktadır. CAN-Bus 1 hattına sürücü 1 ve 2 bağlıken, CAN-Bus 2 hattına ise sürücü 3 ve 4 bağlıdır. Sürücü 1 ve 2'den hatta mesajlar 282H+Node ID'den alınmakta, sürücü 3 ve 4'ten ise 284H+Node ID'den alınmaktadır.



Şekil 6.15. Veri alma ve gönderme adresleri [94].

Tablo 6.5'te, sürücüler ve CAN hattı arasındaki veri adresleri gösterilmiştir. Görüldüğü gibi CAN-Bus'tan sürücülere 183H+Node ID ile mesajlar gönderilmektedir. Sürücülerden de 282H+Node ID ile mesajlar hatta alınır. Sürücülere düğüm ID'leri olarak 1, 2, 5 ve 6 verilmiştir. Bu yüzden 184H, 185H, 188H ve 189H'tan mesajlar gönderilmiştir. Sürücülerden de 283H, 284H, 287H ve 288H'tan mesajlar alınmıştır. Motorun mili döndürüldüğünde bu mesajlardan gelen byte'ların değiştiği görülebilmektedir.

Tablo 6.5. Veri alma ve gönderme adresleri [94].

Veri adresi	İsmi	Haberleşmenin yönü
0x282	Sürücü 1	İnverter 1'den CAN kontrolörüne
0x284	Sürücü 2	İnverter 2'den CAN kontrolörüne
0x183	CAN hattı	CAN kontrolörden inverterlere

SMSM'leri CAN-Bus'tan haberleştirip döndürebilmek için Tablo 6.6'da gösterilen mesajlar her bir sürücü için 184H, 185H, 188H ve 189H adreslerinden ayrı ayrı gönderilmelidir.

Tablo 6.6. Sürücülere gönderilecek mesajlar [94].

İsmi	Bitlerin Uzunluğu	Veri Tipi	Açıklaması
AMK sürücü durumu	16	Unsigned	Çalışma durumu biti
AMK sürücü gerçek hızı	16	Signed	Gerçek hız değeri
AMK sürücü moment akımı	16	Signed	Gerçek moment akımı I_q 'nın hesabı
AMK sürücü mıknatıslanma akımı	16	Signed	Gerçek mıknatıslanma akımı I_d 'nin hesabı
AMK sürücü kontrol	16	Unsigned	Kontrol biti
AMK sürücü hedef hızı	16	Signed	Hız değeri (dev/dak)
AMK sürücü pozitif moment	16	Signed	Pozitif moment sınırı
AMK sürücü negatif moment	16	Signed	Negatif moment sınırı
AMK sürücü motor sıcaklığı	16	Signed	Motor sıcaklığı (0.1 °C)
AMK sürücü inverter sıcaklığı	16	Signed	Soğuk plaka sıcaklığı (0.1 °C)
AMK sürücü hatası	16	Unsigned	Hata numarası
AMK sürücü IGBT sıcaklığı	16	Signed	IGBT sıcaklığı (0.1 °C)

Tablo 6.5 ve 6.6'ya göre gönderilecek mesajlar sırasıyla aşağıdaki gibidir:

1. Sürücülerde başlangıçta oluşabilecek hataları resetlemek için öncelikle 184H'tan 8 byte'lık 00 08 00 00 00 00 00 00 mesajı gönderilir.
2. Daha sonra sürücüleri aktif etmek için 184H'tan 8 byte'lık 00 07 00 00 00 00 00 RF mesajı gönderilir. Bunu devreye aldıktan sonra 1. mesaj devreden çıkartılır.
3. 2. mesaj devredeyken 184H'tan en son 8 byte'lık 00 07 2C 01 2C 01 00 00 gönderilir. Bu mesajın 2. byte'ı RF mesajı (07), 3. ve 4. byte'ı hız bilgisini (2C 01), 5. ve 6. byte'ı pozitif momentini (2C 01), 7. ve 8. byte'ı da negatif momentini ifade eder.
4. Aynı işlemler diğer sürücüler için de 185H, 188H ve 189H adreslerinden gönderilen mesajlarla tekrarlanır. Cycle time'lar hepsi için 10 ms'dir. Yani tüm mesajlar 10 ms'de bir gönderilir.

CAN-Bus hattına gelen ve hattan gönderilen mesajları görebilmek için PCAN-View programı kullanılır. Buna göre PCAN View'den gönderilen ve alınan bu mesajlar Şekil 6.16'da gösterildiği gibidir.

PCAN-View

File CAN Edit Transmit View Trace Help

Receive / Transmit Trace PCAN-USB

Message	DLC	Data	Cycle Time	Count
283h	8	00 81 00 00 83 00 BC FF	5	7917431
284h	8	00 81 00 00 87 00 E3 FF	5	7917431
285h	8	37 01 89 01 00 00 00 00	5	7917431
286h	8	0A 01 8C 01 00 00 8C 01	5	7917431
287h	8	00 81 00 00 19 00 D2 FF	5	7917345
288h	8	00 81 00 00 07 FF C1 FF	5	7917420
289h	8	5C 01 8E 01 00 00 8E 01	5	7917343
28Ah	8	1B 01 8B 01 00 00 8B 01	5	7917421

Receive

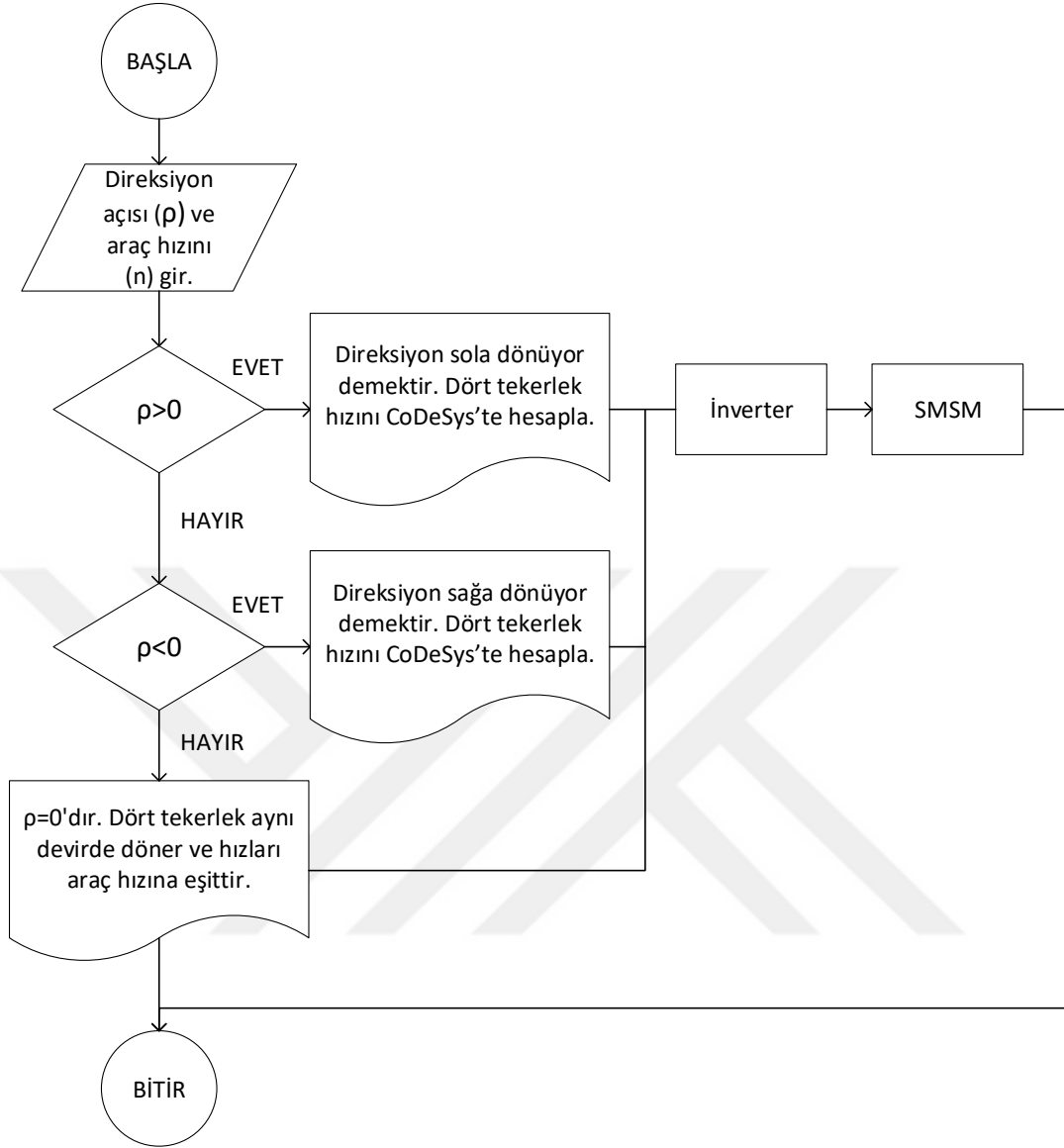
Message	DLC	Data	Cycle Time	Count	Trigger	Comment
184h	8	00 08 00 00 00 00 00 00	10	403	Time	
184h	8	00 07 2C 01 2C 01 D4 F4	10	37405	Time	
184h	8	00 07 00 00 00 00 00 00	10	268	Time	
185h	8	00 08 00 00 00 00 00 00	10	262	Time	
185h	8	00 07 00 00 00 00 00 00	10	259	Time	
185h	8	00 07 2C 01 2C 01 D4 F4	10	36795	Time	
188h	8	00 07 00 00 00 00 00 00	10	224	Time	
188h	8	00 08 00 00 00 00 00 00	10	1040	Time	
188h	8	00 07 2C 01 2C 01 D4 F4	10	32649	Time	
189h	8	00 07 00 00 00 00 00 00	10	212	Time	
189h	8	00 07 2C 01 2C 01 D4 F4	10	32097	Time	
189h	8	00 08 00 00 00 00 00 00	10	228	Time	

Transmit

Connected to PCAN-USB (500 kBit/s) Overruns: 0 QXmtFull: 0

Şekil 6.16. PCAN-View'den alınan ve gönderilen mesajlar

Bu çalışmada, dört tekerleğin hızları ilk olarak Codesys Yazılım'ı ile direksiyon açısı ve aracın hızına bağlı olarak hesaplanmıştır. Ackermann geometrisi kullanılarak matematiksel denklemler elde edilmiş ve Codesys V2.3 yazılımında kullanılmıştır. Tasarlanan EDS'nin akış diyagramı Şekil 6.17'de gösterildiği gibidir.



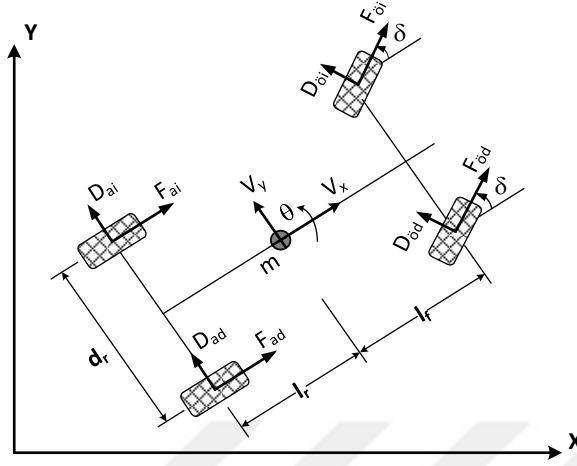
Şekil 6.17. EDS'nin akış diyagramı

Şekil 6.17'de görüldüğü gibi, öncelikle direksiyon açısı ve aracın hız değerleri bir pedal ve enkoder yardımıyla alınmıştır. Direksiyon açısı pozitif değerdeyse araç sola, negatif değerdeyse sağa dönecektir. Buna göre motorların hızları hesaplanarak sürücülere gönderilmektedir. Eğer direksiyon açısı sıfır ise, araç düz yolda gidiyor demektir. Dolayısıyla dört tekerlek hızları araç hızına eşit olmalıdır.

6.9. Dört Tekerlek İçi Motorlu Bir Elektrikli Aracın Dinamik Modeli

Bu bölümde gerçekleştirilen dört SMSM'li bir EA'nın dinamik modeli Şekil 6.18'de gösterilmiştir. Düzlemsel hareket edebildiği düşünülen araç dört ayrı tekerden F kuvvetleri ile

bağımsız tahrik edilmektedir. EA'nın dönüş hareketi sırasında direksiyon açısına bağlı olarak tekerleklere dört yanal kayma kuvveti (D) etki etmektedir.



Şekil 6.18. Elektrikli aracın fiziksel modeli

EA'nın matematiksel modeli oluşturulurken Newton'un ikinci hareket kanunu kullanılmıştır. Buna göre x eksenini doğrultusunda araca etki eden kuvvetler altında birinci hareket denklemi Denklem 6.12'den elde edilir.

$$m\ddot{x} = (F_{\ddot{o}d} + F_{\ddot{o}i})\cos(\delta) + (F_{ad} + F_{ai}) - (D_{\ddot{o}d} + D_{\ddot{o}i})\sin(\delta) + m y \ddot{\theta} \quad (6.12)$$

Burada $F_{\ddot{o}i}$, $F_{\ddot{o}d}$, F_{ai} , F_{ad} , $D_{\ddot{o}i}$, $D_{\ddot{o}d}$, D_{ai} , D_{ad} , sırasıyla iç ve dış ön ve arka tekerleklerin boyuna ve yanal kuvvetleridir. Ayrıca, m EA'nın kütlesidir ve δ , θ sırasıyla EA'nın ağırlık merkezi etrafındaki direksiyon açısı ve yana kayma açısıdır.

Benzer şekilde y eksenini doğrultusunda araca etki eden kuvvetler altında ikinci hareket denklemi Denklem 6.13'te verilmiştir.

$$m\ddot{y} = (F_{\ddot{o}d} + F_{\ddot{o}i})\sin(\delta) + (D_{\ddot{o}d} + D_{\ddot{o}i})\cos(\delta) + D_{ad} + D_{ai} - m x \ddot{\theta} \quad (6.13)$$

Aracın ağırlık merkezi etrafında yaptığı dönme hareketinde araca etki eden momentler göz önüne alınarak üçüncü hareket denklemi şu şekilde gösterilebilir.

$$I_z \ddot{\theta} = l_f ((F_{\ddot{o}d} + F_{\ddot{o}i})\sin(\delta) + (D_{\ddot{o}d} + D_{\ddot{o}i})\cos(\delta)) - l_r (D_{ad} + D_{ai}) + \frac{d_r}{2} ((F_{\ddot{o}d} - F_{\ddot{o}i})\cos(\delta) + F_{ad} - F_{ai}) \quad (6.14)$$

Burada I_z , z eksenine göre EA'nın kütle atalet momentini ifade eder. l_f , l_r ve d_r sırasıyla ön ve arka tekerleklerin ağırlık merkezine olan mesafeleri ve arka tekerlekler arasındaki mesafedir.

Elde edilen hareket denklemleri sayısal çözüm için gerekli olan birinci mertebeden durum denklemlerine dönüştürüldüğünde Denklem 6.15-6.20 elde edilir.

$$\dot{q}_1 = q_2 \quad (6.15)$$

$$\dot{q}_2 = \frac{(F_{\ddot{o}d} + F_{\ddot{o}i}) \cos(\delta) + (F_{ad} + F_{ai}) - (D_{\ddot{o}d} + D_{\ddot{o}i}) \sin(\delta) + m q_4 q_6}{m} \quad (6.16)$$

$$\dot{q}_3 = q_4 \quad (6.17)$$

$$\dot{q}_4 = \frac{(F_{\ddot{o}d} + F_{\ddot{o}i}) \sin(\delta) + (D_{ad} + D_{ai}) + (D_{\ddot{o}d} + D_{\ddot{o}i}) \cos(\delta) - m q_2 q_6}{m} \quad (6.18)$$

$$\dot{q}_5 = q_6 \quad (6.19)$$

$$\dot{q}_6 = \frac{l_f ((F_{\ddot{o}d} + F_{\ddot{o}i}) \sin(\delta) + (D_{\ddot{o}d} + D_{\ddot{o}i}) \cos(\delta)) - l_r (D_{ad} + D_{ai})}{I_z} + \frac{d_r ((F_{\ddot{o}d} - F_{\ddot{o}i}) \cos(\delta) + F_{ad} - F_{ai})}{2 I_z} \quad (6.20)$$

Bu ifadelerde q_1 aracın x-eksenindeki ötelenmesini, q_2 ise x-eksenindeki hızını, q_3 aracın y-eksenindeki ötelenmesini, q_4 ise y-eksenindeki hızını, q_5 aracın z-ekseni etrafındaki dönme açısını ve q_6 ise z-eksenindeki açısal hızı temsil etmektedir.

Esas koordinatlardaki gösterim için homojen dönüşüm matrisi Denklem 6.21'de gösterilmiştir.

$$H = [\cos(\theta) \quad -\sin(\theta) ; \sin(\theta) \quad \cos(\theta)] \quad (6.21)$$

Denklem 6.21 kullanılarak Denklem 6.22'deki ifadede faydalanılabilir.

$$[x;y] = H.[dx;dy] + [x0;y0] \quad (6.22)$$

Burada;

dx : Hareketli koordinat sisteminde x eksenindeki bir döngülük yol farkı

dy : Hareketli koordinat sisteminde y eksenindeki bir döngülük yol farkı

$x0$: Esas koordinat sisteminde her döngü için başlangıç x değeri; her döngüde $x0=x$ olur.

$y0$: Esas koordinat sisteminde her döngü için başlangıç y değeri; her döngüde $y0=y$ olur.

Hareket denklemlerinde yer alan D yanal kayma kuvvetleri Denklem 6.23 ile hesaplanır.

$$D = -\alpha C \quad (6.23)$$

Burada α değeri tekerin kayma açısını vermektedir. Her bir teker için hesaplanırken tekerlerin hız vektörü bileşenleri göz önüne alınır. Burada yer alan C katsayısı teker tipine, aracın hızına, iklime göre kuru, ıslak, karlı, buzlu yollarda teker ile yer arasındaki dirence bağlı olarak değişen ve yapılan kabullere göre belirlenen sabit bir sayıdır.

Ön ve arka, iç ve dış tekerleklerin yana kayma açıları sırasıyla Denklem 6.24-6.27'de verilmiştir.

$$\alpha_{\ddot{o}i} = \tan^{-1} \left(\frac{q_4 + q_6^l f}{q_2 - \frac{d_r}{2} q_6} \right) - \delta \quad (6.24)$$

$$\alpha_{\ddot{o}d} = \tan^{-1} \left(\frac{q_4 + q_6^l f}{q_2 + \frac{d_r}{2} q_6} \right) - \delta \quad (6.25)$$

$$\alpha_{ai} = \tan^{-1} \left(\frac{q_4 - q_6^l r}{q_2 - \frac{d_r}{2} q_6} \right) \quad (6.26)$$

$$\alpha_{ad} = \tan^{-1} \left(\frac{q_4 - q_6^l r}{q_2 + \frac{d_r}{2} q_6} \right) \quad (6.27)$$

6.9.1. Hava Sürtünme Direnci

Düşük hızlarda hava direnci diğer kayıpların yanında oldukça düşük seviyededir. Ancak hız 30-40 km/s değerine ulaşınca hava direnci önem kazanır. Bunun sebebi hava direncinin hızın karesiyle doğru orantılı olarak artmasıdır.

Geometrik boyutları araç dış formuna bağlı direnç katsayısı (C_w) belli olan bir araca herhangi bir hızda etkiyen direnç kuvveti Denklem 6.28'den hesaplanabilir.

$$F_s = 0.5 \cdot d \cdot C_w \cdot A \cdot V^2 \quad (6.28)$$

Burada d , C_w , A ve V sırasıyla havanın yoğunluğu, direnç katsayısı, aracın dikey kesit alanı ve araç hızıdır.

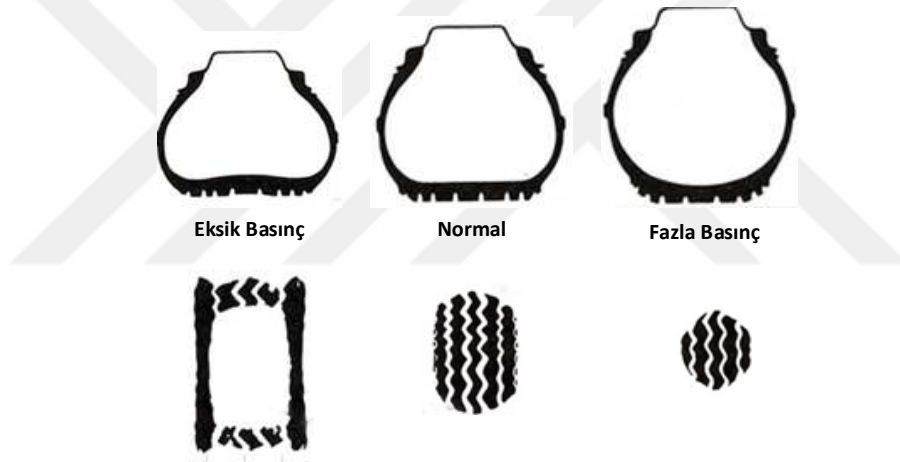
Günümüzde direnç katsayıları, binek araçlarında 0,25'e, otobüslerde 0,5'e, motosikletlerde 0,4'e, kamyonlarda ise 0,65'e kadar düşürülebilmektedir. Buna göre bu çalışmada kullanılan sabit parametrelerin değerleri Tablo 6.7'de verilmiştir.

Tablo 6.7. EA'nın dinamik modelinde kullanılan sabit parametreler ve değerleri

Parametreler	Değerleri
d	1.255 KG/M ³
A	3 M ²
C_w	0.4
C	3678.75
m	1000 KG

6.9.2. Lastik Teker Dinamiği

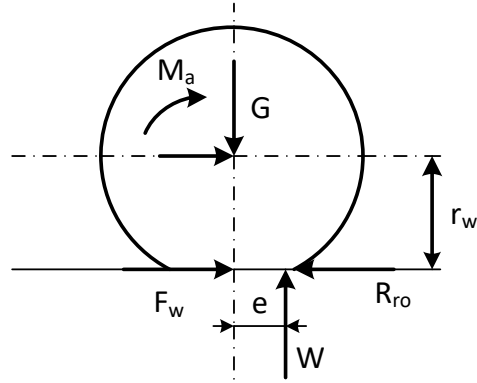
Düşey bir yükün altında bulunan sabit durumdaki tekerleğin zemine temas ettiği kısımda bir basınç alanı oluşur. Temas yüzeyi taşıt ağırlığının lastik basıncına oranıdır. Dolayısıyla fazla şişirilmiş bir lastiğin yüzey ile temas alanı, az şişirilmiş bir lastiğin yüzey temas alanından daha küçük olacaktır. Şekil 6.19'da farklı lastik basınçları altındaki lastik temas yüzeyinde oluşan alanların değişimi gösterilmiştir.



Şekil 6.19. Lastik temas yüzeyindeki alanın basınca göre değişimi

Basınç merkezi, bileşke temas kuvvetinin geçtiği nokta olarak tanımlanır. Doğal olarak sabit duran bir tekerlekte bu merkez, temas yüzeyinin geometrik merkezidir. Ancak hareket halindeki bir tekerlekte basınç dağılımı simetrik değildir ve basınç merkezi hareket yönünde olmak üzere geometrik merkezin önüne geçer.

Şekil 6.20'de görüldüğü gibi, tekerleğe M_a torku ve dolayısıyla da F_w tahrik kuvveti etki ettirildiğinde yukarıda belirtildiği gibi hareket eden bir tekerlekte temas yüzeyindeki basınç dağılımının bileşkesi, temas noktasının geometrik merkezinin ilerisine geçmektedir.



Şekil 6.20. Dinamik tekerleğin serbest cisim diyagramı

Basınç merkezi ile geometrik merkez arasındaki mesafeye e denilirse, düz bir yolda ve sabit bir hızda hareket halindeki bir tekerleğin yuvarlanma direnci, tekerleğin merkezine göre momentler toplamından yola çıkarak Denklem 6.29'dan hesaplanabilir.

$$M_a + R_{ro} r_w = F_w r_w + W e \quad (6.29)$$

Burada M_a tahrik momenti (Nm), F_w uygulanan tahrik kuvveti (N), R_{ro} yuvarlanma direnci (N), W tekerleğe gelen tepki kuvveti (N), r_w tekerlek dinamik yarıçapı (m), e tekerlek basınç merkezi ile geometrik merkez (temas noktası) arasındaki mesafedir.

Denklem 6.30'da e/r_w oranı, yuvarlanma direnci katsayısı olarak tanımlanmakta ve f_{ro} ile gösterilmektedir.

$$R_{ro} = W \left(\frac{e}{r_w} \right) \quad (6.30)$$

Dolayısıyla tekerin hareket yönünde yere uygulayabileceği maksimum çekiş kuvveti, tahrik kuvveti ve yuvarlanma direnci kuvvetinin farkına bağlı olarak Denklem 6.31'den bulunabilir.

$$F_{net} = F_w - R_{ro} \quad (6.31)$$

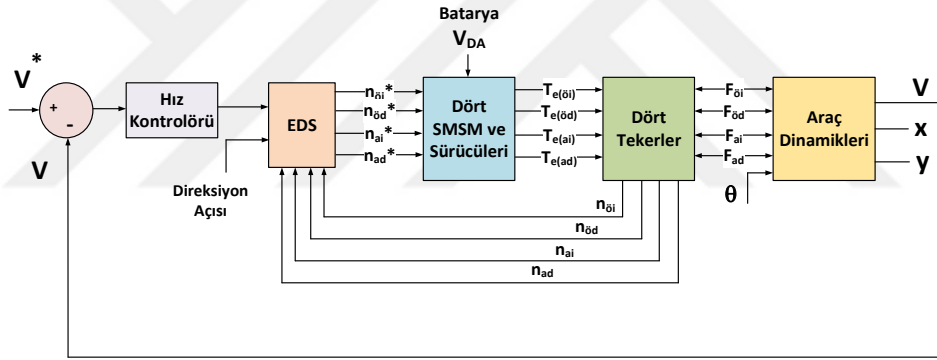
Lastiklerin yuvarlanma direnci katsayıları çeşitli durumlara göre değişmektedir. Bu faktörler taşıt hızı, lastik yapısı, şişirme basıncı, malzeme ve yol yüzeyinin durumudur. Buna göre farklı yol yüzeyleri için yuvarlanma direnci katsayıları Tablo 6.8'de verilmiştir.

Tablo 6.8. Yuvarlanma direnci katsayıları

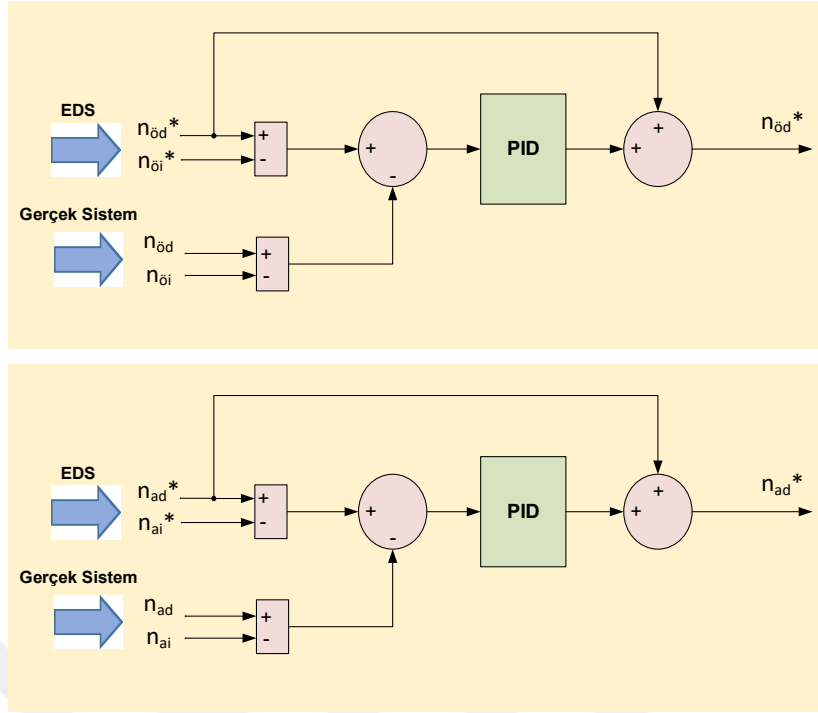
Yolun Durumu	f_{ro}
<i>Düzgün asfalt</i>	0,0015
<i>Küçük taş döşenmiş</i>	0,0015
<i>İri taş döşenmiş</i>	0,0015
<i>Çamurlu yol</i>	0,05
<i>Gevşek toprak, kum</i>	0,1...0,35

6.10. Dört Tekerlek İçi Motorlu Bir Elektrikli Aracın Kontrolü

Dört tekerlekten tahrikli bir EA'nın her bir tekerleğini bağımsız olarak kontrol etmek büyük bir avantajdır. Her tekerleğin hızı ve momentini değiştirilebilir. Böylece istenildiğinde sadece önden çekişli veya arkadan çekişli bir EA elde etmek mümkündür. Ancak araç gövdesinin dengesini sağlamak için bu dört tekerleğin birbiriyle uyumlu çalışması gerekir. Bu gereksinimi karşılamak için, tasarlanan EDS kullanılarak dört tekerlek içi SMSM'li bir EA'nın kontrolü Şekil 6.21'deki gibi gerçekleştirilmiştir. Buradaki * sembolü referans değerleri göstermektedir.

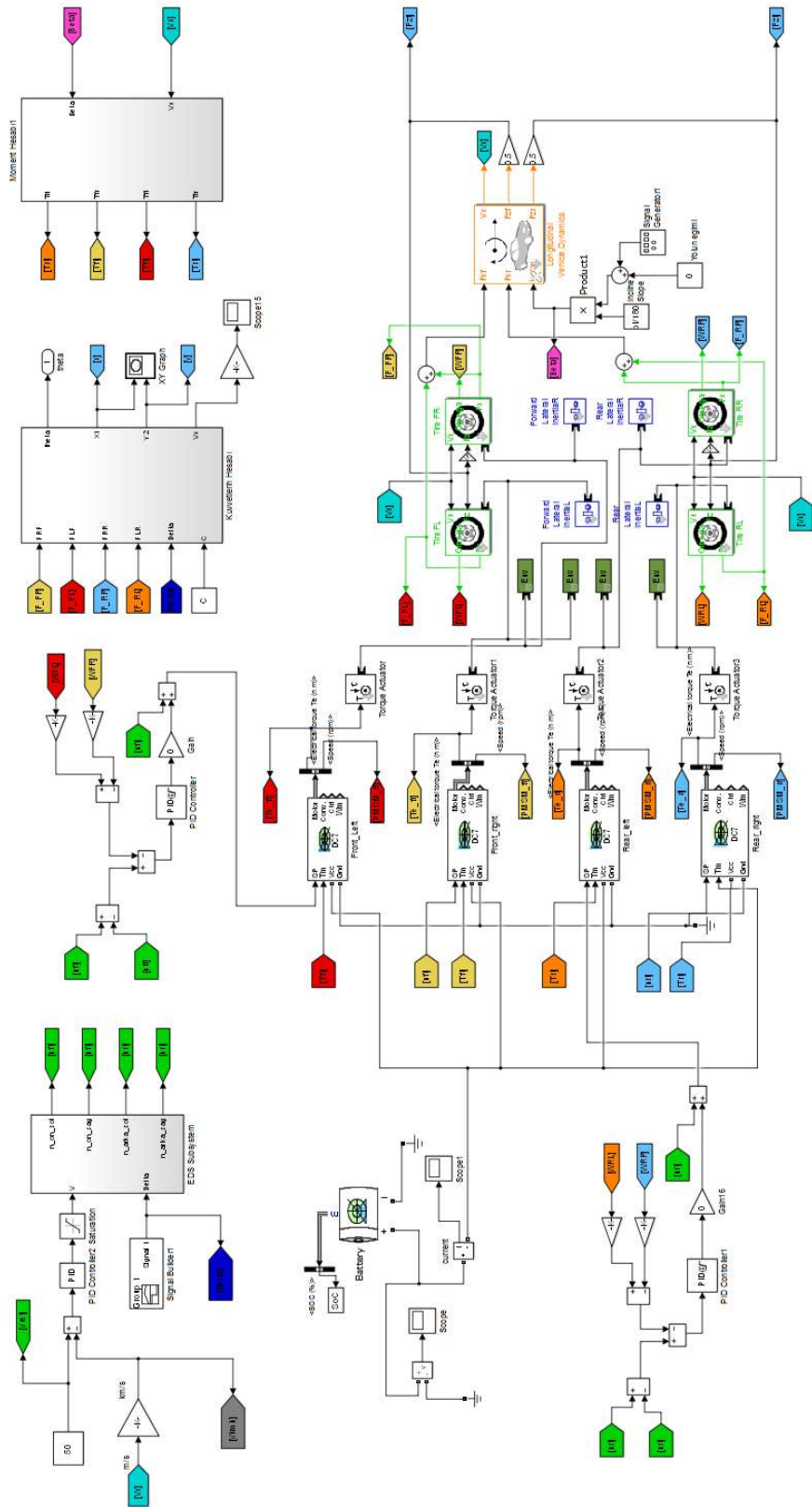
**Şekil 6.21.** Dört tekerlek içi motorlu SMSM'li bir EA'nın dinamik modeli

Şekil 6.21'de gösterilen sistemde araç dinamik modeli ve EA'nın referans hız hatası hız kontrol cihazına uygulanmaktadır. Hız kontrolörü olarak PID kontrolör kullanılmaktadır. PID çıkışı, EDS için bir hız değeri üretir. EDS, direksiyon açısına göre dört tekerleğin her birine bir hız komutu gönderir. Direksiyon açısı sıfırdan farklı olduğunda, EDS'nin sağ ve sol tekerlere gönderdiği hız değerleri farklı olacaktır. EDS çıkışındaki referans hızlar arasındaki fark, EA'nın tekerlek hızları arasındaki farka eşit olmalıdır. Dört tekerlekten birinin hızı istenilen hız değerinden farklı olduğunda tekerlekler arasındaki hız farkını değiştirir. Bu durum EA'nın dengesini bozar ve güvenli sürüşü engeller. Bu çalışmada, EDS'nin sağ ve sol tekerlek referans hız farkları gerçek sisteminkine eşit olacak şekilde bir kontrolör tasarlanmıştır. Sistemin çalışması Şekil 6.22'de ön ve arka tekerlekler için detaylı olarak gösterilmiştir.



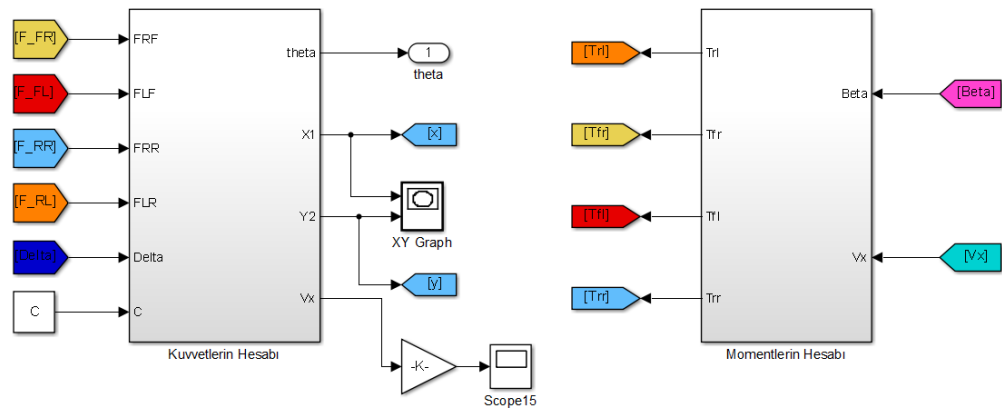
Şekil 6.22. EA'nın ön ve arka tekerlekleri için gerçek sistem ve EDS'nin çıkışından alınan hız farklarının PID kontrolörden geçirilmesi

Şekil 6.21'deki blok şemasına göre dört tekerlek içi motorlu SMSM'li bir EA'nın dinamik modeli Matlab/Simulink'te oluşturulmuştur. Tasarlanan model Şekil 6.23'te gösterilmektedir.



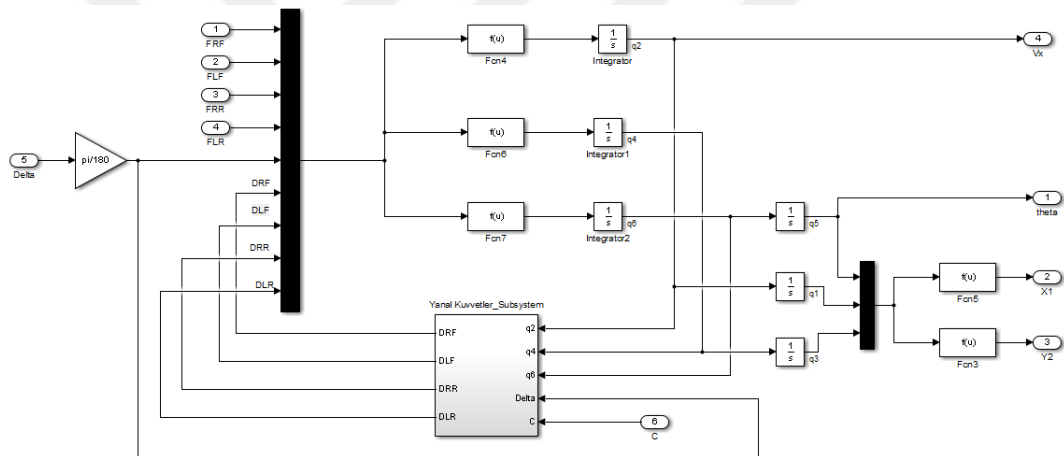
Şekil 6.23. Dört tekerden tahrikli bir EA'nın dinamik modeli

Bu modelde EA'nın dört ayrı tekerleği için kuvvetlerin ve momentlerin hesabı Şekil 6.24'deki Simulink modelinden elde edilmiştir.



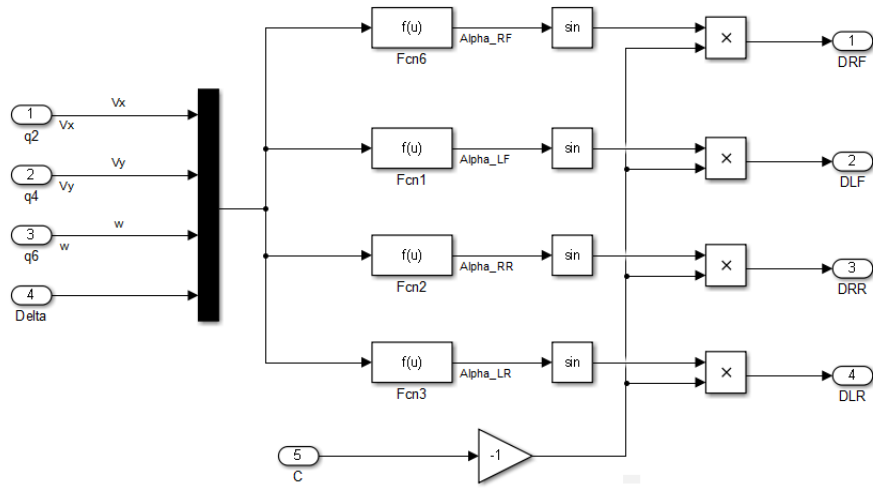
Şekil 6.24. Kuvvet ve momentlerin hesabı

Buna göre aracın matematiksel denklemleri kullanılarak tahrik kuvvetlerinin hesabı Şekil 6.25'te detaylı olarak gösterilmiştir.



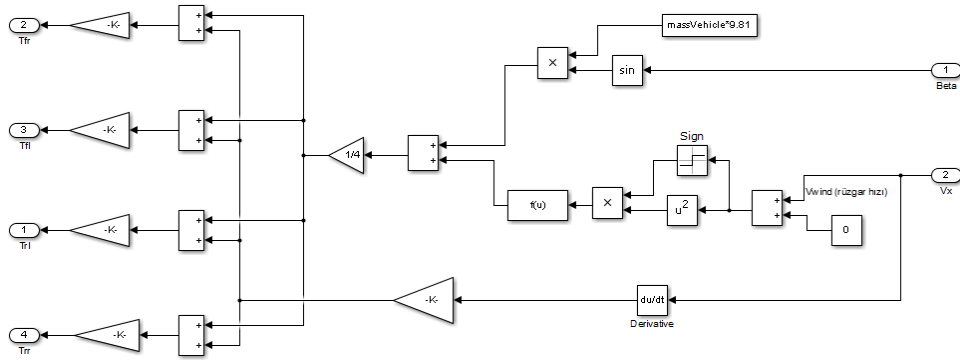
Şekil 6.25. Tahrik kuvvetlerinin alt sistemi

Şekil 6.25'te tasarlanan modelde direksiyon açısının değişimine ve araç dinamiklerinden elde edilen boylamsal kuvvetlere göre yolun xy yol eğrisi çizdirilmiştir. Buna göre direksiyon açısı değiştikçe aracın çizdiği yol da değişecektir. Burada yer alan yanıl kuvvetler bloğunun modellenmesi ise Şekil 6.26'da gösterilmiştir.



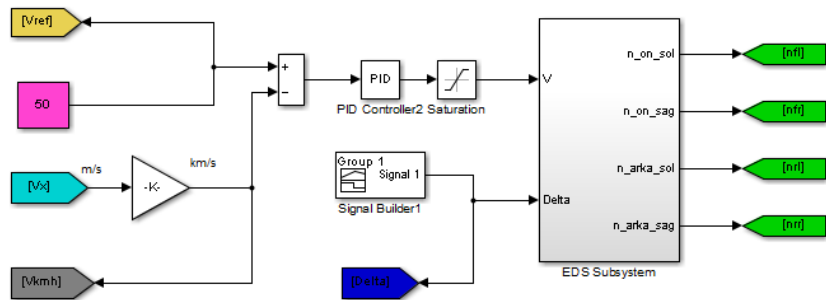
Şekil 6.26. Yanal kuvvetlerin hesabı

Şekil 6.24'te gösterilen moment bloğunun içyapısı ve hesaplamaları detaylı olarak Şekil 6.27'de verilmiştir.



Şekil 6.27. Momentlerin hesabı

Tasarlanan dinamik modelde, araç dinamikleri bloğunun çıkışından alınan gerçek sistemin hız değeri V_x , 50 km/saat olan referans hız değeri ile karşılaştırılarak hızlar arasındaki fark Şekil 6.28'deki gibi bir PID kontrolörden geçirilmiş ve EDS'ye uygulanmıştır.



Şekil 6.28. Gerçek sistem hızı ile referans hız farkının PID kontrolörü

Şekil 6.28’de görüldüğü gibi, direksiyon açısı ve araç hızına bağlı olarak EA’nın dört tekerlek hızları tasarlanan EDS ile hesaplanmıştır. Buradan hesaplanan tekerlek hızları referans değerlerdir. Motorlar, sürücüler ve araç dinamiklerinden oluşan gerçek sistemde ise, motorlardan alınan hız değerleri değiştikçe sistemin stabilitesi bozulacaktır. Bunu dengelemek için bir PID kontrolör tasarlanmıştır. Böylece SMSM’lerin yük koşulları da yolun eğimine ve viraja göre değerlendirilmektedir. EA’nın boylamsal ve yanal kuvvetleri hesaplanmaktadır ve bir hız kontrol cihazı kullanılarak EA’nın hız eğrisi elde edilir. Bu kontrol yöntemi, literatürdeki bazı karmaşık yöntemlerden daha basit bir yaklaşım sunmakta ve böylece farklı direksiyon açıları ve EA hızları için EA’nın stabilitesi sağlanmaktadır.

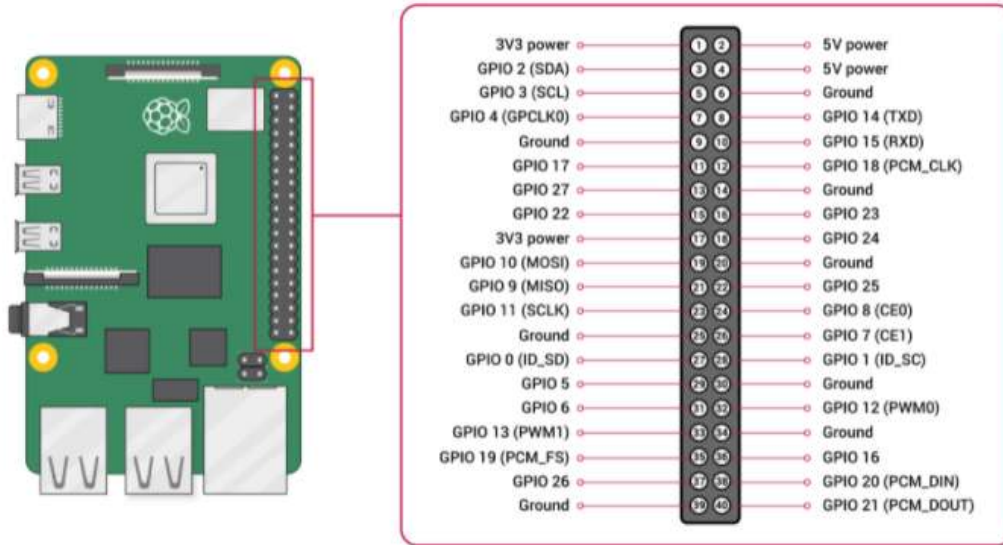
Görüldüğü gibi bu bölümde, bu tez çalışmasında gerçekleştirilen dört tekerlekten tahrikli bir EA’nın EDS tasarımı, matematiksel denklemleri ve dinamik modeli anlatılmıştır. Tasarlanan modellerden ve deneysel sistemden alınan sonuçlar ise detaylı olarak sonuçlar bölümünde verilecektir. Gelecek bölümde de EA’nın güvenli sürüşü üzerine gerçekleştirilen uygulamalar anlatılacaktır.

7. FPGA VE RASPBERRY Pi İLE GÖRÜNTÜ İŞLEME VE GÜVENLİ SÜRÜŞ UYGULAMALARI

7.1. Kullanılan Donanımlar

7.1.1. Raspberry Pi

Bu tez çalışmasında, görüntü işleme uygulamalarını gerçekleştirmek için Şekil 7.1’de gösterilen Raspberry Pi 4 kartı kullanılmıştır. Kartta iki adet 5 V pin, iki adet 3.3 V’luk pin ve topraklama pinleri (0V) bulunur. Kalan pinlerin tamamı genel amaçlı 3.3 V’luk pinlerdir. Yani çıkışlar 3.3 V’a ayarlanmıştır ve girişler 3.3 V toleranslıdır. Bu kartın güçlü bir özelliği, üst kenarı boyunca sıralanmış genel amaçlı kullanılan 40 pinli giriş/çıkış (GPIO) pinleridir. GPIO pinlerinden herhangi biri yazılımda giriş veya çıkış pini olarak belirlenebilir ve çok çeşitli amaçlar için kullanılabilir. Giriş ya da çıkış pini olarak belirlenmiş bir GPIO pini, yüksek (3.3 V) veya düşük (0 V) olarak ayarlanabilir. Basit giriş ve çıkış cihazlarının yanı sıra GPIO pinleri, PWM, SPI, I2C, Seri haberleşme gibi çeşitli amaçlar için de kullanılabilir [95]. Anaconda Navigator Programı ile Python programlama dilinde programlanır.

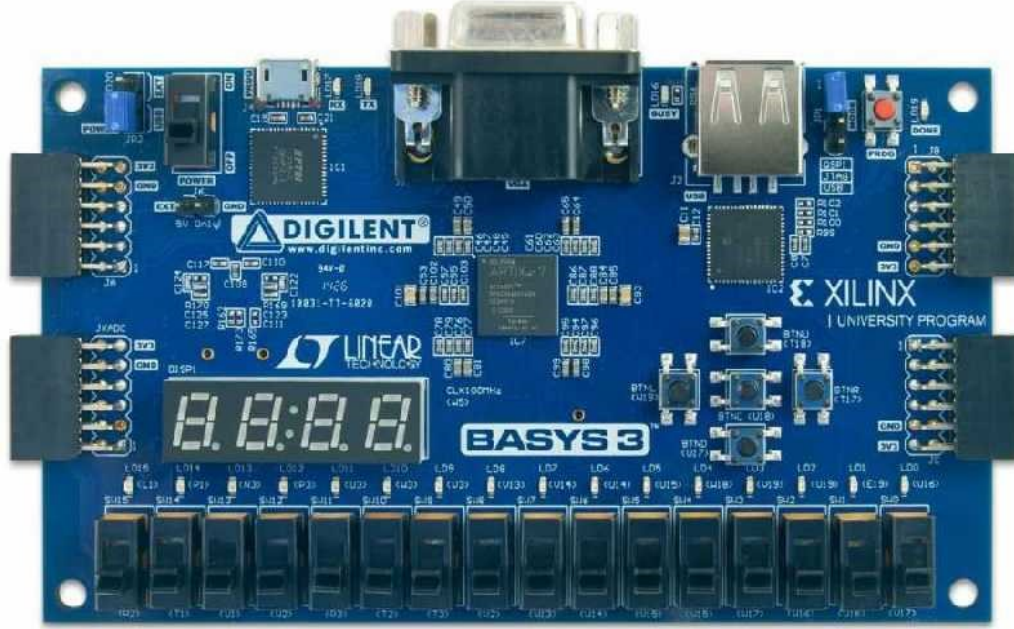


Şekil 7.1. Raspberry Pi GPIO pinleri

7.1.2. Alanda Programlanabilir Kapı Dizileri (FPGA)

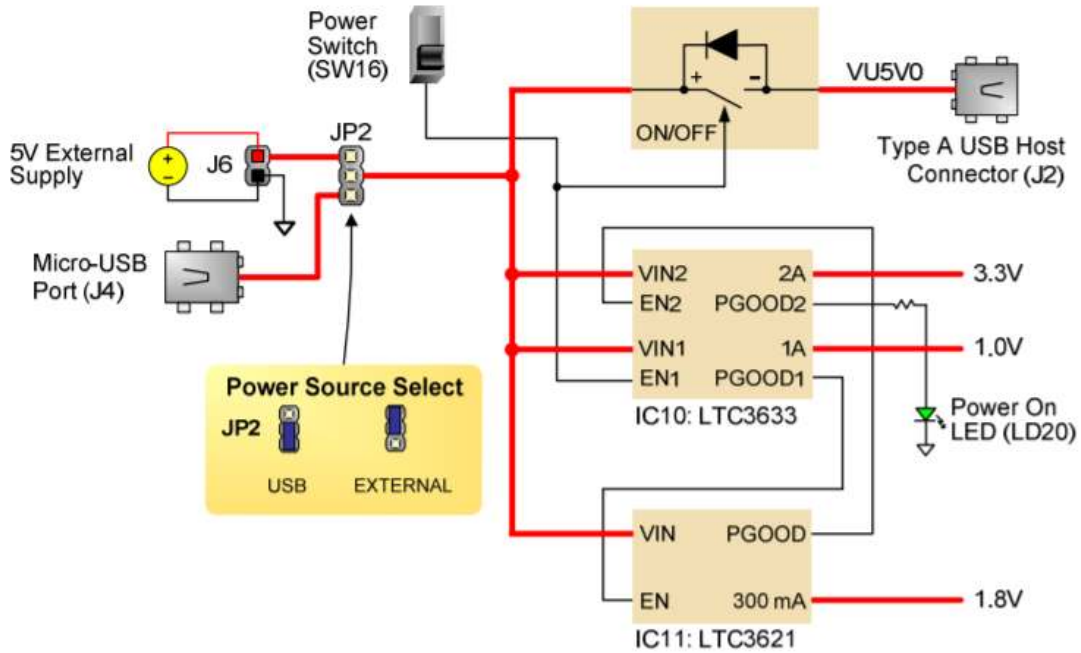
Bu tez çalışmasında, Alanda Programlanabilir Kapı Dizileri (Field Programmable Gate Array, FPGA)’nden biri olan Basys 3 kartı kullanılmıştır. Xilinx’in en son Artix-7™ ailesinden olan tam donanımlı ve hazır bir devre geliştirme platformudur. Yüksek kapasiteli FPGA (Xilinx XC7A35T-1CPG236C), USB, VGA ve diğer bağlantı noktaları ile giriş/çıkışlardan oluşur [96].

Kartta, 16 switch, 16 led, 5 buton, 4 haneli 7 segmentli ekran, 12 bit VGA çıkışı, FPGA programlama ve haberleşme için Digilent USB-JTAG bağlantısı ve USB-UART bağlantısı bulunmaktadır. Basys 3 FPGA kartı Şekil 7.2’de gösterilmiştir.



Şekil 7.2. Basys 3 FPGA kartı

Basys 3 güç devresi ise Şekil 7.3’te verilmiştir.



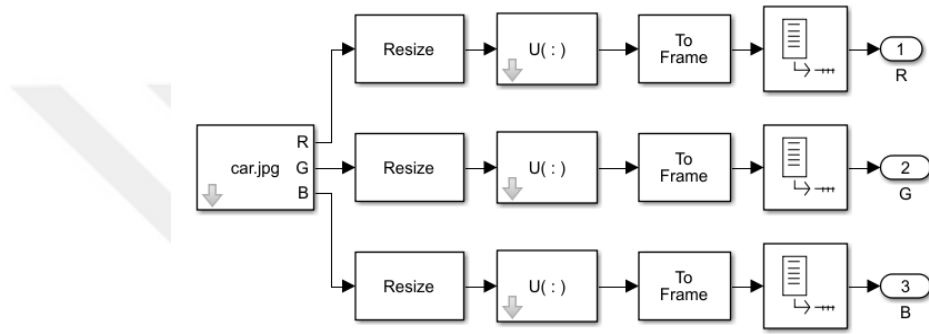
Şekil 7.3. Basys 3 güç devresi

7.2. Gerçekleştirilen Görüntü İşleme Uygulamaları

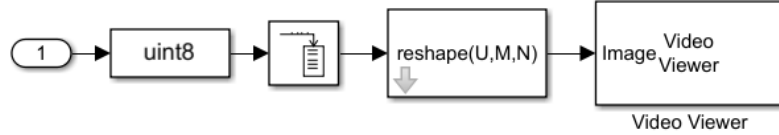
Bu kısımda, Raspberry Pi ve FPGA kullanılarak gerçekleştirilen görüntü işleme algoritmaları ve alınan sonuçlar yer almaktadır.

7.2.1. Xilinx System Generator Kullanılarak Görüntü ve Video İşleme Uygulamaları

Xilinx System Generator (XSG) kullanılarak çeşitli görüntü ve video işleme algoritmaları Matlab/Simulink'te gerçekleştirilmiştir. Bu algoritmalar sırasıyla Şekil 7.4 ve Şekil 7.5'te gösterildiği gibi ön işlem ve son işlem arasında gerçekleştirilmektedir.



Şekil 7.4. Simulink'te gerçekleştirilen ön işleme alt bloğu



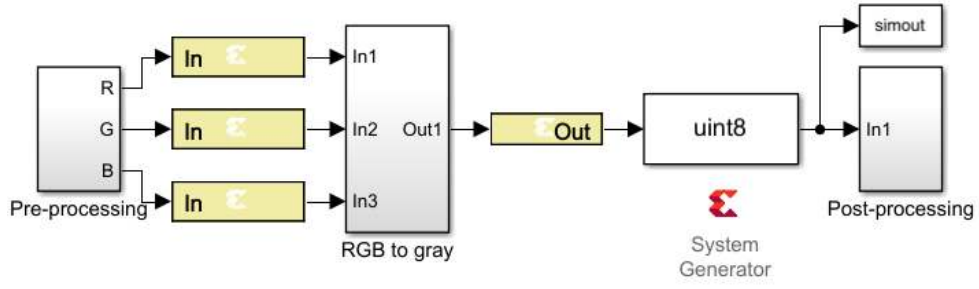
Şekil 7.5. Simulink'te gerçekleştirilen son işleme alt bloğu

7.2.2. RGB Görüntülerin Gri Seviyeye Dönüştürülmesi

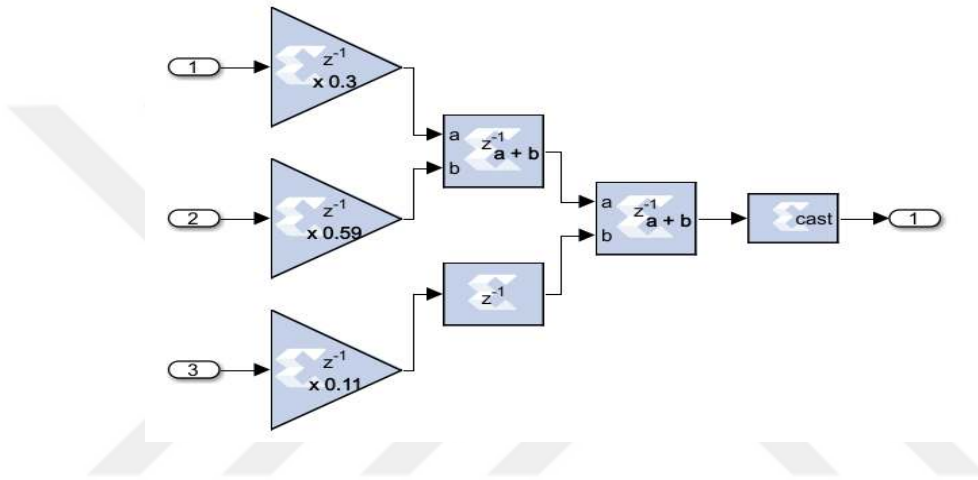
Kırmızı (R), Yeşil (G) ve Mavi (B) olmak üzere her piksel rengine göre, renkli görüntüler işlem süresini en aza indirmek için gri tonlama düzeyine dönüştürülür. Bu dönüşüm Denklem 7.1 ile gerçekleştirilir.

$$Y = 0.3 R + 0.59 G + 0.11 B \quad (7.1)$$

Buna göre RGB'den griye dönüşümü MATLAB/Simulink'te gerçekleştirilmiştir. Simulink modelleri sırasıyla Şekil 7.6 ve Şekil 7.7'de gösterilmektedir.



Şekil 7.6. RGB'den griye dönüşüm Simulink modeli



Şekil 7.7. RGB'den griye dönüşüm işlemi Simulink alt bloğu

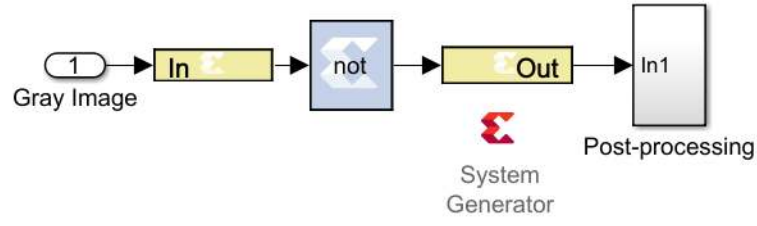
RGB'den griye dönüşümden elde edilen sonuç Şekil 7.8'deki gibidir.



Şekil 7.8. RGB'den griye dönüşümden elde edilen sonuç

7.2.3. Görüntüyü Tersine Çevirme

Simulink'te gri görüntüye ters çevirme algoritması uygulanır. Şekil 7.9'da gösterildiği gibi bu algorithmada, görüntünün piksel değeri 255 ise “Not” bloğu kullanılarak değer sıfıra dönüştürülür. Negatif görüntü basitçe görüntünün piksel değeri ile ters çevrilir.



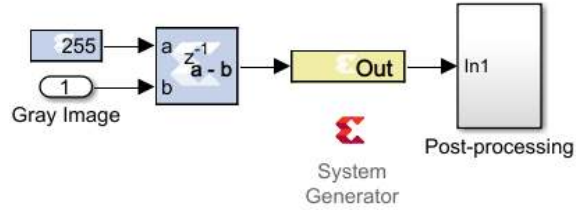
Şekil 7.9. Görüntünün tersini alma Simulink modeli

Tersi alınan görüntünün çıkışı Şekil 7.10’da gösterilmiştir.



Şekil 7.10. Tersi alınan görüntünün çıkışı

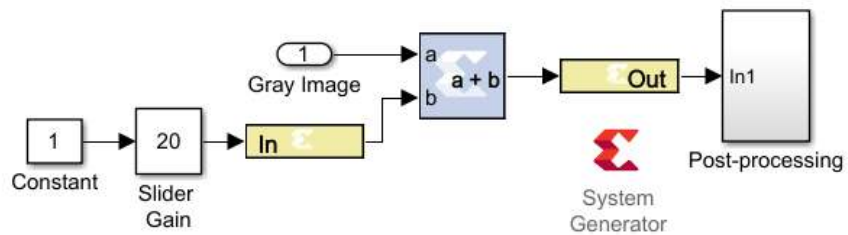
Şekil 7.11’de mantıksal değil operatörü kullanılmadan farklı yöntemler de gösterilebilir.



Şekil 7.11. Tersi alınan görüntü için Simulink’te oluşturulan farklı bir algoritma

7.2.4. Görüntü Geliştirme

Gri tonlamalı görüntü iyileştirme, Şekil 7.12’de görüldüğü gibi görüntünün her pikselini iyileştirmek için kullanılır.



Şekil 7.12. Görüntü geliştirmenin Simulink modeli

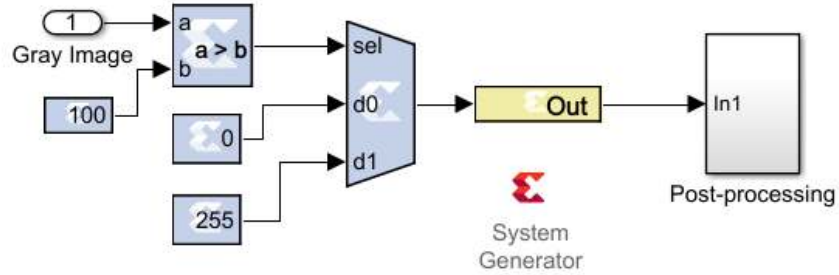
Gri seviye görüntünün geliştirilmesinden elde edilen sonuç ise Şekil 7.13'te gösterilmiştir.



Şekil 7.13. Gri seviye görüntü ve geliştirilmiş görüntü

7.2.5. Eşikleme (Thresholding)

Eşikleme, görüntü bölütleme için önemli yöntemlerden biridir. Sabit bir değerden veya beyaz pikselden küçük olduğunda görüntüdeki her pikseli görüntü yoğunluğunda siyah bir pikselle değiştirmek için kullanılır. Bu sabit değerın altındaki tüm gri seviye değerleri siyah olarak sınıflandırılacak ve diğer değerler beyaz olacaktır. Gri tonlamalı görüntünün eşikleme algoritması ve sonucu sırasıyla Şekil 7.14 ve Şekil 7.15'de gösterilmiştir.



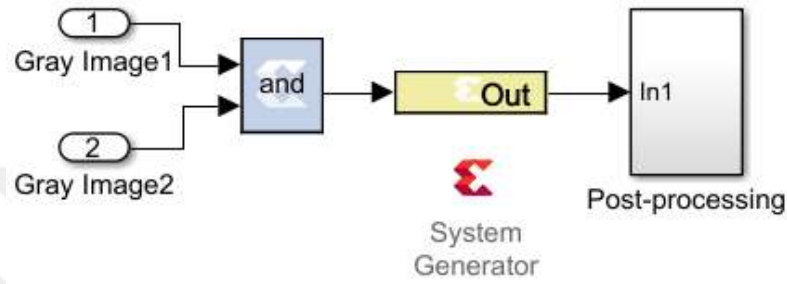
Şekil 7.14. Gri seviye görüntünün eşikleme işleminin Simulink modeli



Şekil 7.15. Gri seviye görüntünün eşikleme işlemi sonucu elde edilen yeni görüntü

7.2.6. Erozyon ve Geniřletme

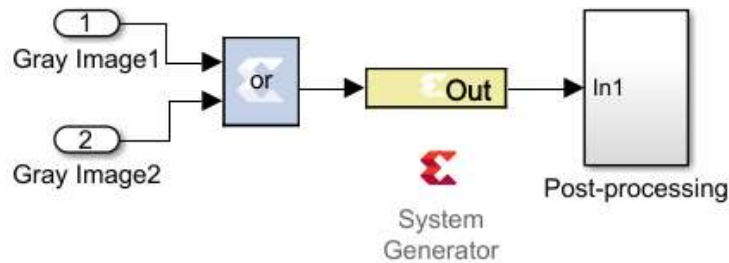
Erozyon ve geniřletme en temel morfolojik iřlemlerdir. Geniřletmede, bir grntdeki her piksel, ıktı grnts iindeki yapılandırma elemanının řekli ile deėiřtirilir. Mantıksal “VEYA” bloėunu kullanarak bir grntdeki nesnelerin sınırlarına pikseller ekler. te yandan, nesne sınırlarındaki pikseller, mantıksal “AND” bloėu kullanılarak ařındırma yoluyla kaldırılır. Bu piksellerin sayısı, yapılandırma elemanının boyutuna ve řekline baėlıdır. Erozyon ve geniřletme algoritmaları ve ikili grntnn sonuları sırasıyla řekil 7.16, 7.17, 7.18 ve 7.19’da verilmiřtir.



řekil 7.16. İgili tabanda grntnn erozyon algoritmasının Simulink modeli



řekil 7.17. İgili tabanda grntnn erozyon algoritması sonucu elde edilen grnt



řekil 7.18. İgili tabanda grntnn geniřletme algoritmasının Simulink modeli

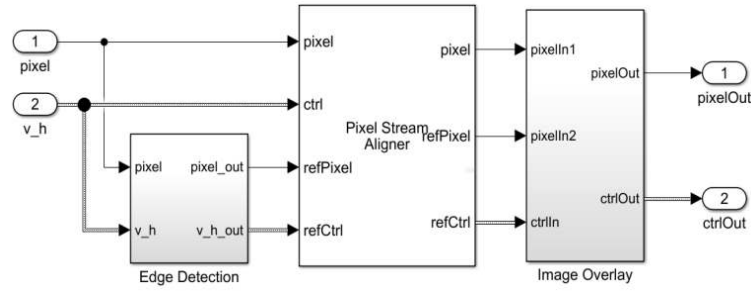


Şekil 7.19. İkili tabanda görüntünün genişletme algoritması sonucu elde edilen görüntü

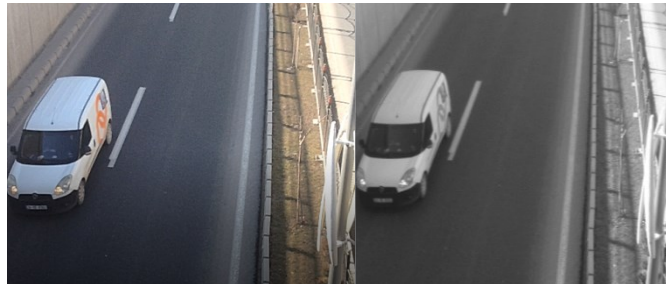
7.2.7. Kenar Çıkarma

Kenar çıkarma, bir nesnenin kenarlarını algılamak için kullanılan önemli bir algoritmadır. Ortak kenar algılama operatörleri Sobel, Canny, Prewitt, Roberts, Sharpen ve Gauss filtrelemedir. Tüm bu operatörler, görüntü piksel değerleri ile operatör matrisi arasında meydana gelen bir konvolüsyon ile çalışır. Kenarları tespit etmek için algoritmada filtreleme, iyileştirme ve köşelerin tespiti olmak üzere üç adım vardır. Filtreleme ve maskeleme işlemi, uygun filtre maskesi kullanılarak gerçekleştirilir.

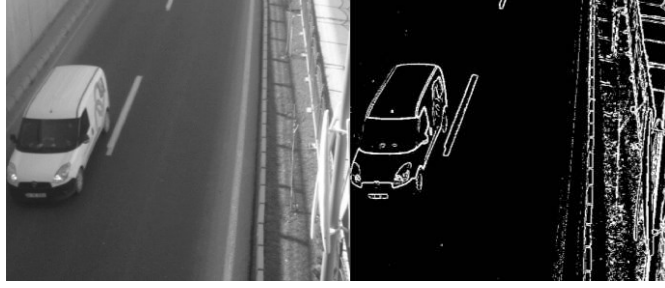
Şekil 7.20'de gösterilen kenar algılama algoritması, kamera ile çekilen bir video üzerinde uygulanmıştır. İlk olarak Şekil 7.21'de RGB video griye dönüştürülür ve ardından Şekil 7.22'de görüldüğü gibi videoda nesnelerin kenarları çıkartılır.



Şekil 7.20. Video kenar çıkarma algoritması Simulink modeli



Şekil 7.21. RGB video görüntüsünü gri seviyeye dönüştürme



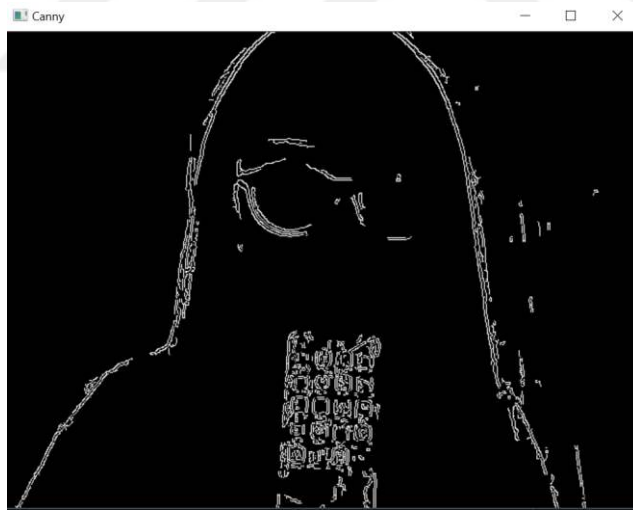
Şekil 7.22. Gri seviye video görüntüsüne kenar çıkarma metodu uygulama

7.3. Raspberry Pi’de Gerçekleştirilen Görüntü İşleme Uygulamaları

Bu bölümde, Raspberry Pi 4 kartı kullanılarak Anaconda Spyder programında Python programlama dili ile gerçekleştirilen uygulamalar anlatılmaktadır.

7.3.1. Canny Kenar Çıkarma Algoritması

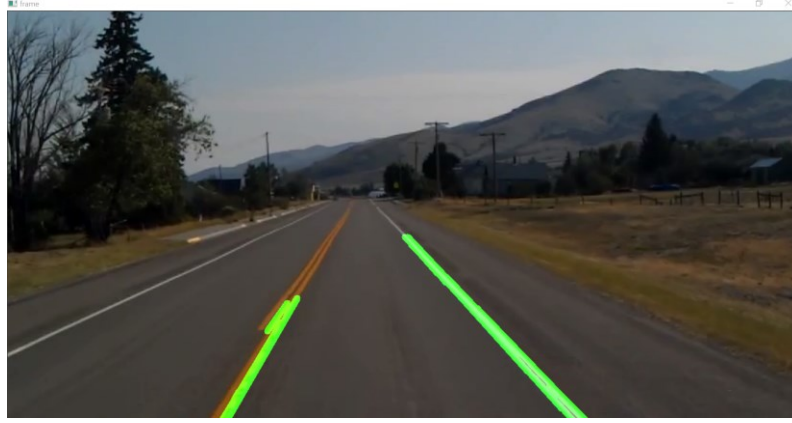
Kenar çıkarma metotlarından biri olan ve yaygın kullanılan Canny kenar çıkarma algoritması ile kameradan alınan görüntünün kenarları Şekil 7.23’deki gibi elde edilmiştir.



Şekil 7.23. Canny kenar çıkarma algoritmasının sonucu

7.3.2. Yol Çizgilerinin Takibi

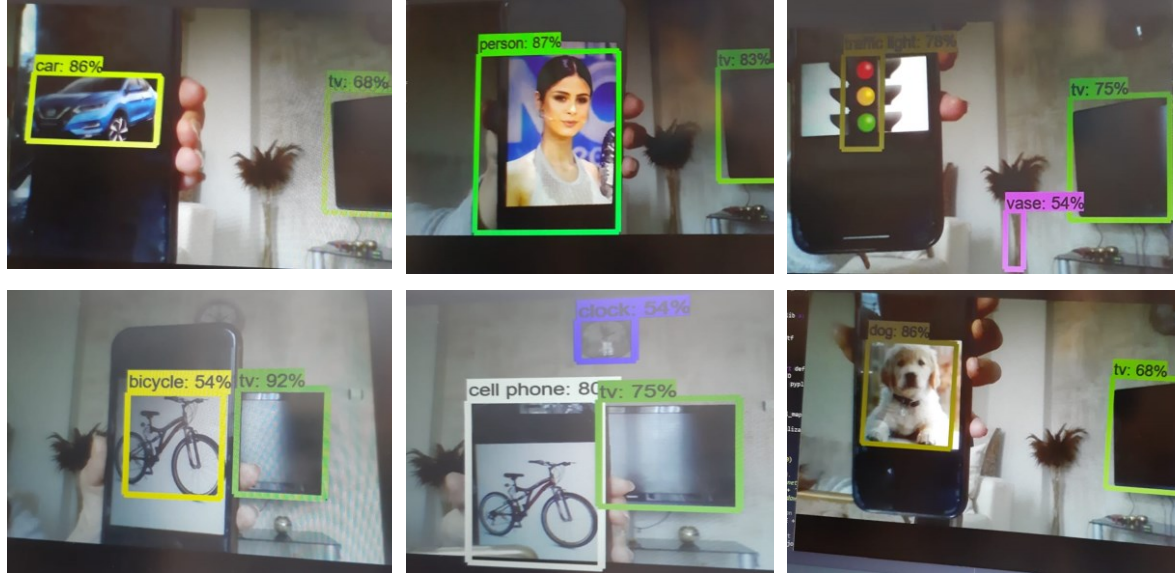
Aracın yolda düzgün ilerlemesi için yol çizgilerinin takibi üzerine, Anaconda Spyder Programı’nda bir algoritma geliştirilmiştir. Buna göre bir video görüntüsü üzerinde yol çizgilerinin takibi yapılmıştır. Elde edilen sonuç Şekil 7.24’te verilmiştir.



Şekil 7.24. Yol çizgilerinin takibi

7.3.3. Nesne Tanıma

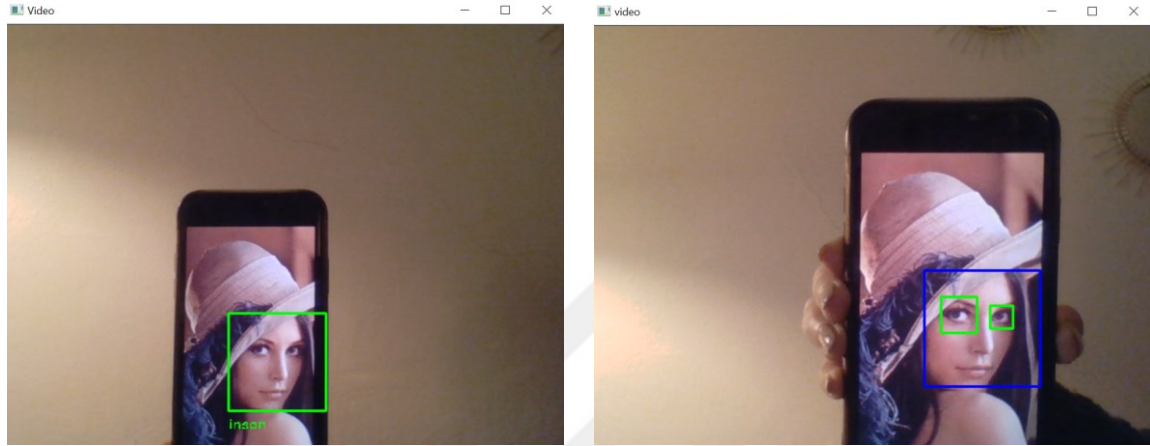
Raspberry Pi ile TensorFlow Kütüphanesi kullanılarak nesne tanıma işlemi gerçekleştirilmiştir. TensorFlow, açık kaynak kodlu bir derin öğrenme kütüphanesidir. Bu kütüphane ile çok çeşitli yapay zekâ uygulamaları gerçekleştirilebilir. Yapısı oldukça esnektir. Esasen Python kullanılarak geliştirilmiş olsa da bunun yanı sıra C++, Java, Javascript, C# gibi birçok programlama dilini de destekler. Spyder’da gerçekleştirilen yazılımda datalar eğitilmiş ve trafik ışığı, araba, köpek, televizyon, cep telefonu, insan, bisiklet gibi toplam doksan adet nesne tanımlanmıştır. Elde edilen bazı sonuçlar Şekil 7.25’te gösterilmiştir.



Şekil 7.25. Çeşitli nesneleri tanıma

7.3.4. Yüz ve Göz Tanıma

Python ile OpenCV Kütüphanesi kullanılarak yüz ve göz tanıma uygulamaları gerçekleştirilmiştir. OpenCV'nin açılımı "Open Source Computer Vision Library" yani Açık Kaynak Bilgisayar Görme Kütüphanesi'dir. Gerçek zamanlı uygulamalarda yaygın şekilde kullanılır. Buna göre yüz ve göz tanıma uygulamalarının sonuçları Şekil 7.26'da gösterilmiştir.



Şekil 7.26. Yüz ve göz tanıma

7.3.5. Hareket Algılama

Bu uygulamada belirli büyüklükteki piksel blokları önceki zaman ile sonraki zaman arasında değişiklik olması durumunu sürekli olarak kontrol eder. Eğer bu iki zamandaki piksel renk değerlerinde farklılık varsa hareket vardır yoksa hareket yoktur.

T0 : Önceki zaman

T1 : Sonraki zaman olmak üzere;

$$Piksel\ deęeri = \begin{cases} T0 = T1 & \text{ise} & \text{Orijinal Görüntü} \\ T0 \neq T1 & \text{ise} & \text{Kırmızı} \end{cases}$$

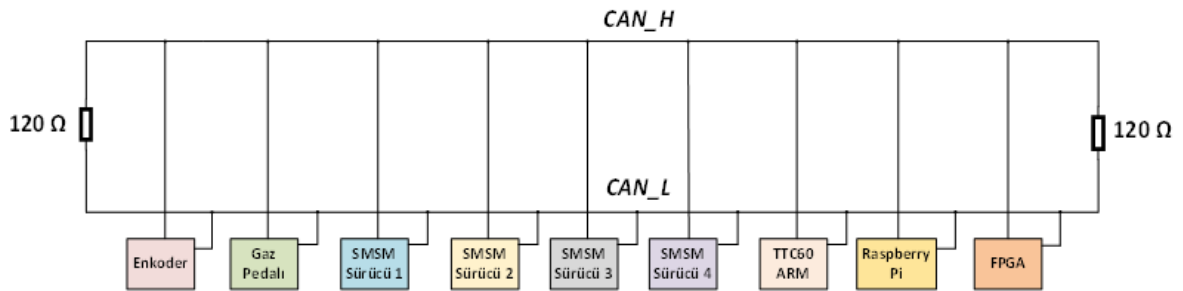
Şekil 7.27'de hareket algılama yazılımının sonucu gösterilmiştir. Hareketi algıladığı an yeşil çerçeve ile gösterilir. Böyle bir algorithmada kullanılan mevcut bir görüntü algılayıcı kamera aynı zamanda hareket tespitini anlamak amacıyla kullanılabilir.



Şekil 7.27. Hareket algılama

7.4. Dört Tekerlek İçi Motorlu Elektrikli Araçta Güvenli Sürüş Uygulamaları

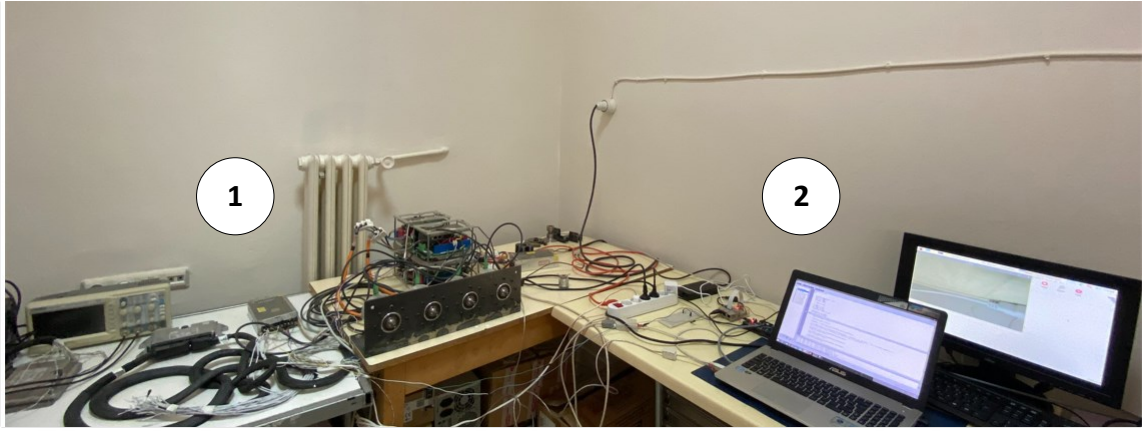
Bu tez çalışmasında, öncelikle dört tekerlek içi motorlu bir EA için EDS tasarımı gerçekleştirilmiştir. Direksiyon açısına ve aracın hızına bağlı olarak tekerleklerin dönmesi gereken hız değerleri CoDeSys, Matlab/Simulink ve deneysel sonuçlara göre ayrı ayrı elde edilmiştir. Gerçekleştirilen deneysel düzenek detaylı olarak Bölüm 6'da anlatılmıştır. EDS tasarlandıktan ve motorlar istenilen hızda döndükten sonra güvenli sürüş uygulamaları gerçekleştirilmiştir. Bunun için CAN-Bus hattına Raspberry Pi 4 kartı bağlanmıştır. Görüntü işleme uygulamaları bu kart üzerinde gerçekleştirilmiştir. Basys 3 FPGA kartı da Raspberry Pi'ye bağlanarak CAN hattına bağlanmıştır. Buna göre CAN-Bus hattının son hali Şekil 7.28'de gösterilmektedir.



Şekil 7.28. CAN-Bus hattına bağlı üniteler

Şekil 7.28'de görüldüğü gibi, CAN-Bus hattına dört adet SMSM sürücüsü, HY-TTC60 işlemci, enkoder, FPGA, Raspberry Pi ve gaz pedalı bağlanmıştır. Hat başında ve sonunda 120 Ω 'luk hat başı ve sonu dirençler kullanılır. Burada ana ünite (master) HY-TTC60 işlemcisidir.

Diğer işlemciler ve sensörler yardımcı (slave) ünitelerdir. Gerçekleştirilen tüm deneysel düzenek Şekil 7.29’da gösterilmiştir.

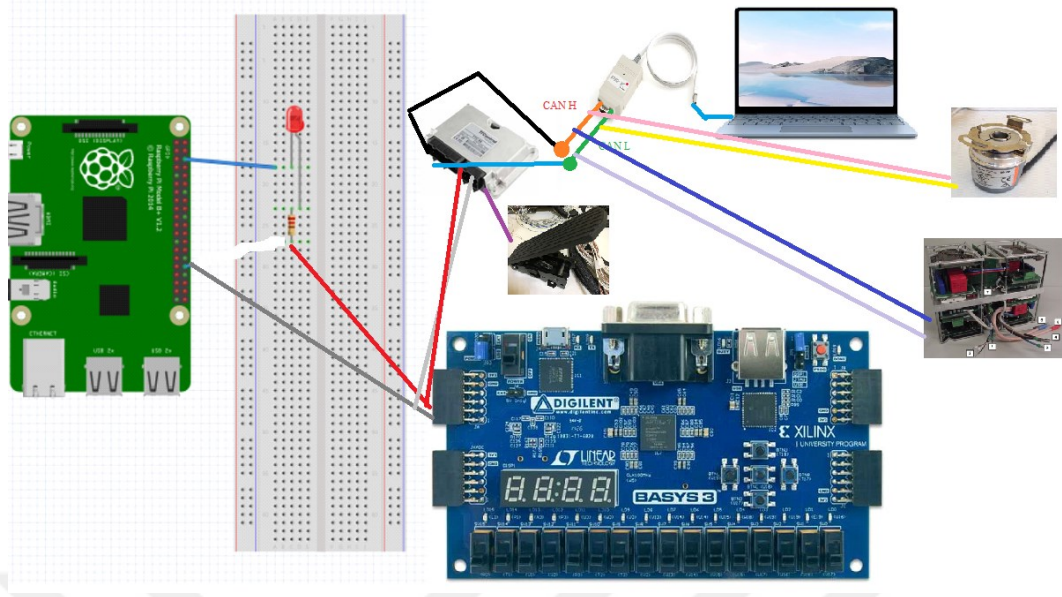


Şekil 7.29. Tüm deney düzeneği

Şekil 7.29’da görüldüğü gibi deney düzeneği iki kısımdan oluşur. 1. bölümde, EDS’nin gerçekleştirildiği dört adet tekerlek içi SMSM motor ve sürücü sistemi, enkoder, pedal ve HY-TTC60 işlemci yer alır. 2. bölümde ise, görüntü işleme uygulamaları için Raspberry Pi 4, kamera, Basys3 FPGA ve VGA monitör yer alır. Şekil 7.30’da gösterilen Raspberry Pi üzerine bağlı kamera modülünden alınan görüntüler gerçek zamanlı olarak işlenir ve FPGA’ya gönderilir. 2. kısmın bağlantı şeması da Şekil 7.31’de gösterilmiştir.

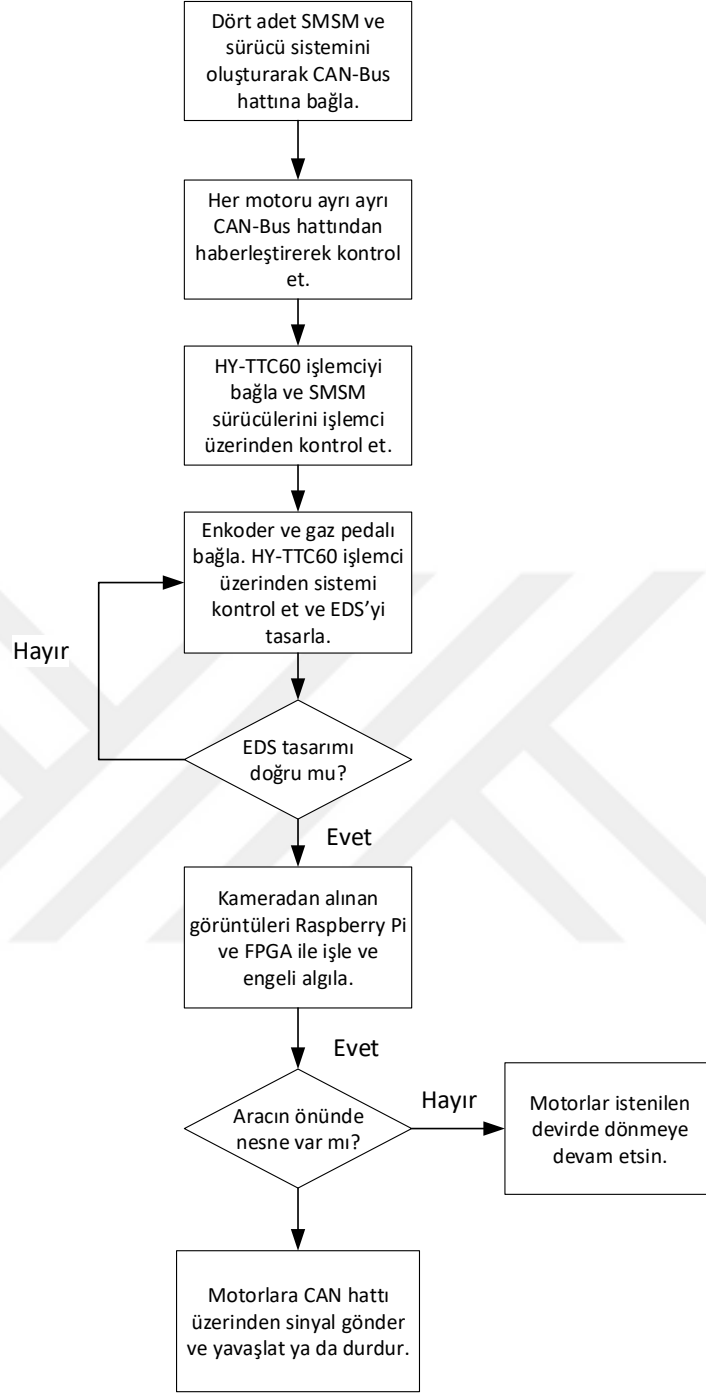


Şekil 7.30. Raspberry Pi ve kamera modülü



Şekil 7.31. Raspberry Pi ve FPGA'nın deney düzeneğine bağlantı şeması

Raspberry Pi 4 kartı kullanılarak Python yazılımı ile hareket algılama uygulaması gerçekleştirilmiştir. Bu uygulamayla kameranın önüne bir engel çıktığında bunu kare içine alarak ekrana “Engel var” yazdırılır. Daha sonra Raspberry GPIO'larına bağlı olan FPGA çıkışına lojik 1 gönderilir ve Şekil 7.31'de görüldüğü gibi led yanar. Tüm üniteler CAN-Bus hattına bağlı olduğundan hatta lojik 1 sinyali gelir ve engel olduğu anlaşılır. Bu durumda ana işlemci olan HY-TTC60 ile CoDeSys üzerinden programlanarak motorların hızları düşürülür ya da durdurulur. Eğer engel yoksa FPGA çıkışına lojik 0 gönderilir ve ekrana “Engel yok” yazdırılır. Motorlar tasarlanan EDS'ye göre istenilen devirde dönmeye devam eder. Böylece kameradan alınan bilgilere göre araç için bir güvenlik tedbiri alınmış olunur. Gerçekleştirilen sistemin akış diyagramı Şekil 7.32'de verilmiştir.

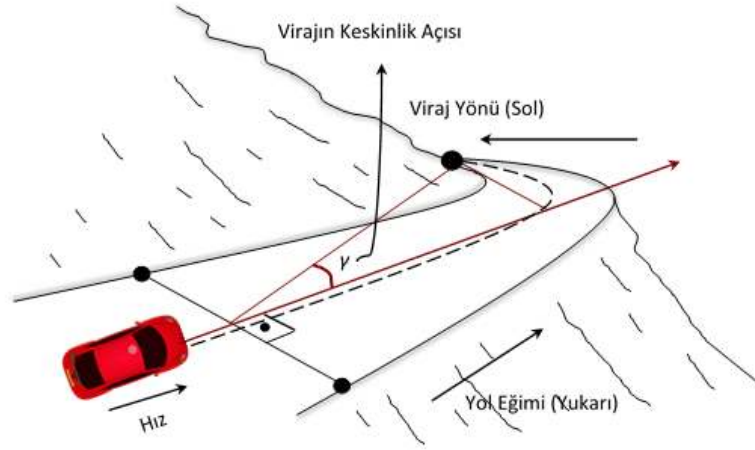


Şekil 7.32. Tüm sistemin akış diyagramı

7.5. Viraj Yönü ve Açısının Tespit Edilmesi ve Risk Hesabı

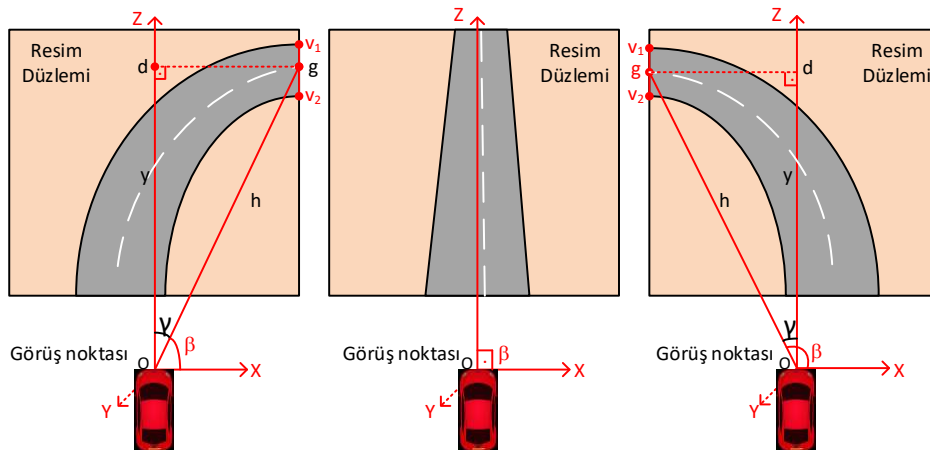
Şehirlerarası yollarda aracın güvenli şekilde ilerleyebilmesi için viraj yönünün tespiti büyük önem taşımaktadır. Özellikle uzun yollarda öngörülemeyen keskin virajlara yüksek hızda girmek büyük risk oluşturur. Bu yüzden bu çalışmada, bu riskleri önlemek ve önceden sürücüyü uyarmak için bir risk tahmini yapılmıştır. Aracın önüne yerleştirilen bir kameradan alınan yol

görüntüsü üzerinde viraj yönü, viraj açısı ve aracın direksiyon açısı tespit edilmiştir. Bunun için ufuk noktası, yol sınırları ve şeritleri belirlenerek viraj geometrisinden hesaplamalar yapılmaktadır. Şekil 7.33'te şehirlerarası yolda karşılaşılabilen bir virajın geometrik yapısı gösterilmiştir.



Şekil 7.33. Şehirlerarası bir yolda viraj geometrisi

Yolun geometrik yapısı güvenli sürüşü etkileyen en temel özelliklerdendir. Zorlu hava şartlarında yol durumuna bağlı olarak kaza riski artabilir. Yolun geometrisi sağa, sola kıvrılan ve düz yol olarak değişebilir. Trafiğin sağdan aktığı ülkelerde sağa dönen virajları almak sola dönen virajlara göre daha kolaydır. Buna göre viraj yönü yolun ufuk noktasına göre kolayca tespit edilebilir. Viraj ufuk noktası yolun sağında ise viraj yönü sağa doğru, solundaysa viraj yönü sola doğrudur. Şekil 7.34'te farklı yol durumları gösterilmiştir. Şekilde gösterilen β açısı, x eksenini viraj ufuk noktası arasında kalan açıdır ve viraj yönünü tespit etmek için kullanılır. Eğer β açısı doksan dereceden küçükse viraj sağa doğrudur, geniş açı ise yani doksan dereceden büyükse viraj sola doğrudur. β açısı yaklaşık eşit doksan derece ise düz yoldur; yani viraj yok demektir.



Şekil 7.34. (a) Viraj sağa doğru, (b) Düz yol, (c) Viraj sola doğru [97].

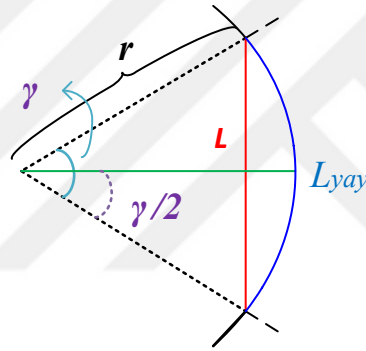
Viraj açısını hesaplamak için Şekil 7.34'te gösterilen uzunluklar kullanılır. Bunlar, aracın görüş noktasının viraj ufuk noktasına olan dik uzaklığı (y) ve görüş noktasının virajın ufuk noktasına olan mesafesidir (h). Burada gösterilen γ açısı viraj açısıdır ve g noktası v_1 ve v_2 noktalarının ortasıdır. Eğer γ açısı sıfır ise düz yol demektir.

Buna göre γ viraj açısı Denklem 7.2'den bulunabilir.

$$\cos(\gamma) = \frac{y}{h} \Rightarrow \gamma = \arccos\left(\frac{y}{h}\right) \quad (7.2)$$

Denklem 7.2'den görüldüğü gibi y uzunluğu ve h mesafesi ne kadar kısa olursa viraj açısı o kadar fazla olmaktadır.

Viraj geometrisi detaylı olarak Şekil 7.35'te gösterilmiştir.



Şekil 7.35. Viraj geometrisi

Buna göre viraj yarıçapı Denklem 7.3'ten hesaplanabilir.

$$r = \frac{180^\circ L_{yay}}{\pi \gamma} \quad (7.3)$$

Burada, L_{yay} yay uzunluğu, r eğrinin yarıçapı, γ viraj açısı, L ise uzunluktur. Viraj yarıçapı Denklem 7.4'teki gibi düzenlenirse;

$$r = \frac{L}{2 \sin(\gamma/2)} \quad (7.4)$$

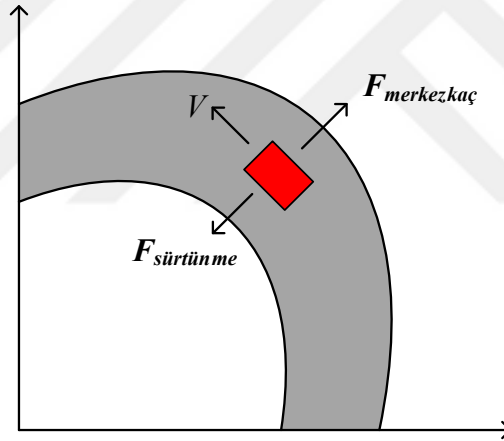
Viraj açısı γ , Denklem 7.5'ten hesaplanabilir.

$$\gamma = 5729.58/r \quad (7.5)$$

Kuzey Amerika yol çalışmalarında eğrilik derecesini hesaplamak için kullanılan olağan mesafe 100 fit (30,5 m) yaydır. SI birim sistemi tercih edilirse uzunluk olarak 100 metre (330 fit) kullanılabilir. Eğrilik derecesinin 100 birim yay uzunluğuna dayandığı durumlarda, eğrilik derecesi yani viraj açısı ile yarıçap arasındaki dönüşüm $\varphi = (18000/\pi)/r \approx 5729.57795/r$ olarak alınmaktadır [98].

7.6. Araca Etki Eden Kuvvetlerin İncelenmesi ve Risk Hesabı

Araç virajlı yolda giderken araca etki eden kuvvetler Şekil 7.36’da gösterilmiştir. Bunlar sürtünme kuvveti ve merkezkaç kuvvetidir. Sürtünme kuvveti, birbirine temas eden ve kayan iki yüzey tarafından üretilen kuvveti ifade eder ve aracın savrulmasını önleyen kuvettir. Aracın dönme hareketiyle var olan ivmesinden dolayı oluşan kuvvet ise merkezkaç kuvvetidir. Dönüş sırasında merkezkaç kuvvet, aracı dönme ekseninin dışına itmeye çalışır. Eğer merkezkaç kuvveti sürtünme kuvvetinden büyük olursa araç dışarı doğru savrulur. Bunu dengelemek için Şekil 7.36’da görüldüğü gibi ters yönde bir sürtünme kuvveti uygulanır.



Şekil 7.36. Virajlı yolda araca etki eden kuvvetler

Merkezkaç kuvveti Denklem 7.6’dan hesaplanabilir.

$$F_{\text{merkezkaç}} = \frac{mV^2}{r} \quad (7.6)$$

Burada m araç kütlesi, V araç hızıdır.

Sürtünme kuvveti ise Denklem 7.7’den hesaplanmaktadır.

$$F_{\text{sürtünme}} = k m g \quad (7.7)$$

Denklem 7.7’de verilen k sürtünme katsayısı, g ise yerçekimi ivmesidir. k , kuru yüzeyler için 0.9 alınmıştır. g ’nin değeri ise 9.80665 m/s^2 ’dir.

Denklem 7.6’dan görüldüğü gibi merkezkaç kuvveti, aracın hızına bağlı olduğundan araç hızlandıkça değeri artacak ve virajlı yollarda risk oluşturacaktır. Merkezkaç kuvveti sürtünme kuvvetinden büyük olduğu durumda araç virajdan çıkacak ve savrulacaktır. Bu nedenle bu kuvvetlerin hesabı da büyük önem taşımaktadır.

7.7. Metodun Gerçekleştirilmesi

7.7.1. Şeritler ve Hedef Noktaların Belirlenmesi

Kaynak noktalarını ve hedef noktalarını belirlemek için, paralel şeritleri olduğu bilinen kavisli olmayan bir yolun görüntüsü alınmaktadır. Dolayısıyla, şeritlerin paralel olması gerektiği için dört hedef noktası bir dikdörtgen oluşturacaktır. Kaynak noktaları, görüntünün şeritlerindeki pikseller yuvarlak yapılarak Şekil 7.37’deki gibi gösterilmektedir.



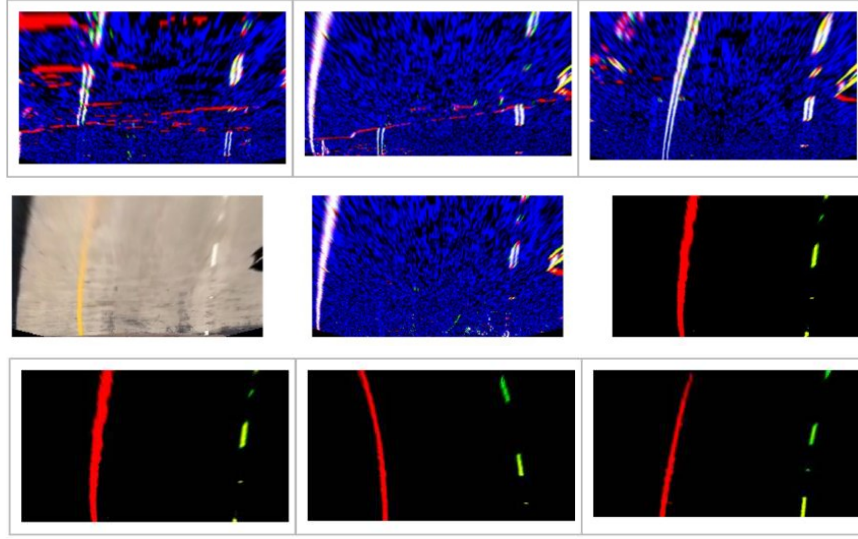
Şekil 7.37. Kaynak noktaları ve şerit çizgilerinin belirlenmesi

7.7.2. Şerit Piksellerini Filtreleme: Gradyant ve Renk Eşikleme

Yol şeritlerinde istenmeyen pikselleri filtrelemek için gradyant eşikleme ve renk eşiklemenin bir kombinasyonu kullanılır. Renk eşiklemesi için sarılar ve beyazlar dışında her şey maskelenir. Renkler HSL formatında temsil edilir. Ton değeri, kırmızı, yeşil ve mavi kombinasyonlarına dayalı olarak algılanan renk numarasıdır. Doygunluk değeri, rengin ne kadar renkli veya ne kadar mat olduğunun ölçüsüdür. Açıklık, rengin beyaza ne kadar yakın olduğunun ölçüsüdür. Sarı şeritler, belirli bir değerin üzerindeki hafiflik ve doygunluk kombinasyonu ile güzel bir şekilde ayırt edilir [99]. Beyaz şeritler, doygunluk ve renk tonundan bağımsız olarak gerçekten yüksek bir hafiflik değerine sahip olarak ayırt edilir.

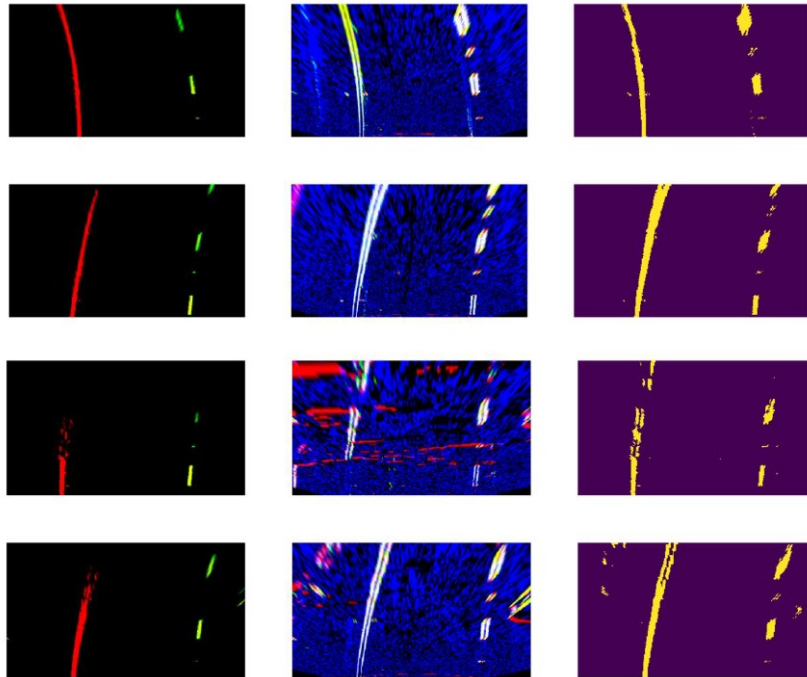
Gradyan eşikleme yapmak için Sobel operatörü kullanılır. Sobel operatörünü bir görüntüye uygulamak, görüntünün türevini yani gradyanını x ve y yönlerinde alma yöntemidir. Bu türev,

belirli bir değerin bir piksel konumundan diğerine ne kadar hızlı değiştiğini ölçer. Resmin açıklık değerine Şekil 7.38'deki gibi Sobel filtresi uygulanmıştır.



Şekil 7.38. Şerit piksellerini filtreleme

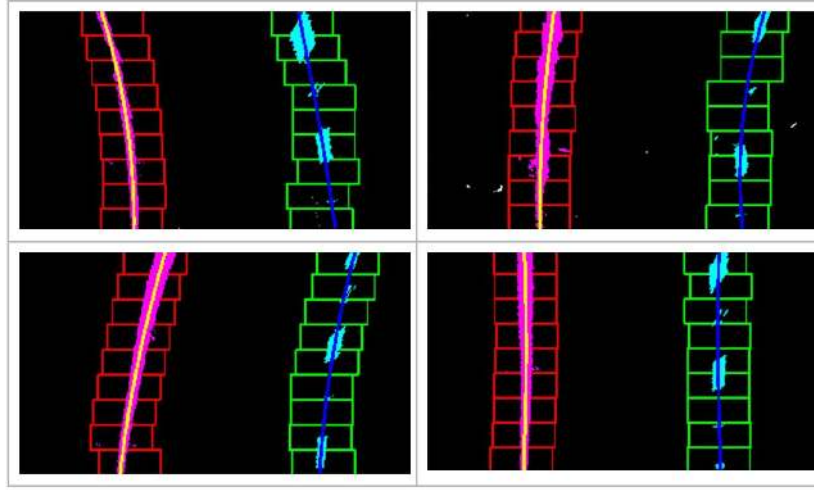
Hem renk hem de gradyan eşikleme sonuçlarını eklemek, şerit çizgileri üzerinde güzel bir filtreleme işlemi yapılmasını sağlar. Gölgeler nedeniyle görüntünün kenarlarından bazı yabancı değerler alınsa da daha sonra maskeleme işlemi uygulanarak bunlar Şekil 7.39'daki gibi düzeltilmektedir.



Şekil 7.39. Şerit piksellerini maskeleme

7.7.3. Polinom Uydurma ve Eğrilik Ölçümü

İkinci derece polinom fonksiyonu $x = Ay^2 + By + C$ olan her şerit için bir eğri uydurulabilir ve $[A, B, C]$ katsayıları bulunabilir. Bunun için `polyfit()` fonksiyonu kullanılmaktadır. Noktalarla tanımlanır ve bu noktalara en iyi uyan belirli bir eğri derecesi polinom katsayılarını verir. Hangi piksellerin bir şeridin parçası olduğuna karar vermek için temel bir algoritma uygulanır. "Sürgülü pencere algoritması" olarak adlandırılan bu algoritma kullanılarak çerçevedeki şeritleri takip eden bir pencere oluşturulmuştur. Bir "pencere" içindeki pikseller "ilgilenen pikseller" olarak işaretlenir ve şeritteki noktalar listesine eklenir. Daha sonra bu piksellerin x değerlerinin ortalaması alınır. Şeridin tepe noktasına çıkana kadar bu işlem tekrarlanır. Bu şekilde, 2. derece polinomun katsayılarını yayan "polyfit()" fonksiyonuna tüm pikseller toplanır. Bu işlem Şekil 7.40'ta gösterilmiştir.



Şekil 7.40. Polinom uydurma ve eğrilik ölçümü

Denklem 7.8'deki gibi 2. derece polinomun katsayıları kullanılarak, her bir şeridin eğrilik yarıçapı Denklem 7.9'dan hesaplanmaktadır.

$$f(y) = Ay^2 + By + C \quad (7.8)$$

$$r = \frac{\left[1 + \left(\frac{dx}{dy} \right)^2 \right]^{3/2}}{\left| \frac{d^2x}{dy^2} \right|} \quad (7.9)$$

7.7.4. Segmentasyon (Bölütleme)

Dijital görüntü işleme ve bilgisayarla görmede, görüntü segmentasyonu diğer bir adıyla bölütleme, dijital bir görüntüyü birden çok bölüme yani görüntü nesneleri olarak da bilinen piksel kümelerine ayırma işlemidir. Segmentasyon genellikle görüntü analizinin ilk aşamasıdır ve amacı, bir görüntüyü daha basit, anlamlı ve analiz edilmesi daha kolay hale dönüştürmek ve görüntülerdeki nesneleri ve sınırları konumlandırmaktır [100–102]. Yoldan kaydedilen görüntüler genellikle şerit tespiti göz önüne alındığında gökyüzü, ağaçlar, araçlar, binalar vb. gibi pek çok işe yaramaz bilgi, yani şerit dışı bilgiler içerir. Bu tür şerit dışı bilgilerin etkisini azaltmak için, kaplama piksellerinin aynı düzlemde oldukları gibi aynı normal vektöre sahip olduğu bilgisine dayanarak yolu segmentlere ayırmak gerekir. Böylece görüntü bölütlemenin sonucu, toplu olarak tüm görüntüyü kaplayan bir dizi parça veya görüntüden çıkarılan kenar algılama gibi bir dizi konturdur. Bir bölgedeki piksellerin her biri, renk, yoğunluk veya doku gibi bazı karakteristik veya hesaplanmış özelliklere göre benzerdir. Bitişik bölgeler, aynı karakteristiklere göre önemli ölçüde farklı renktedir.

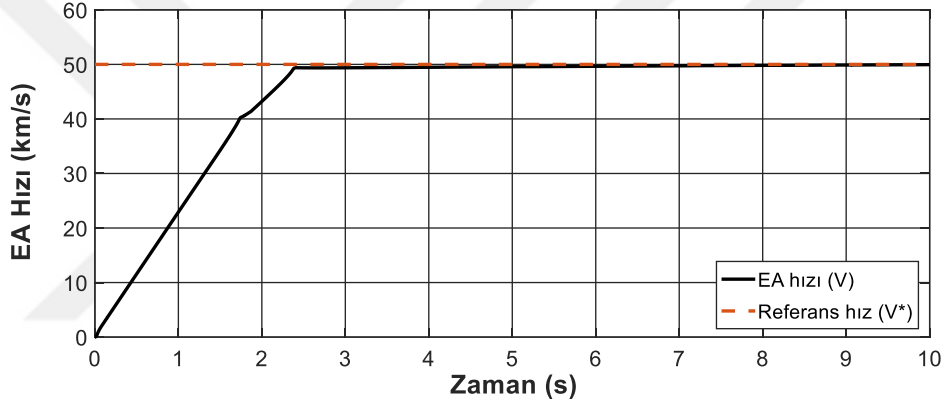
Bu bölümde, EA'nın güvenli sürüşü üzerine farklı platformlarda gerçekleştirilen çeşitli görüntü işleme ve güvenli sürüş uygulamaları anlatılmıştır. Bu uygulamaların sonuçları da detaylı olarak gelecek bölümde verilecektir.

8. SONUÇLAR

Bu bölümde, bu tez çalışması kapsamında alınan sonuçlar üç ana başlık altında verilmiştir.

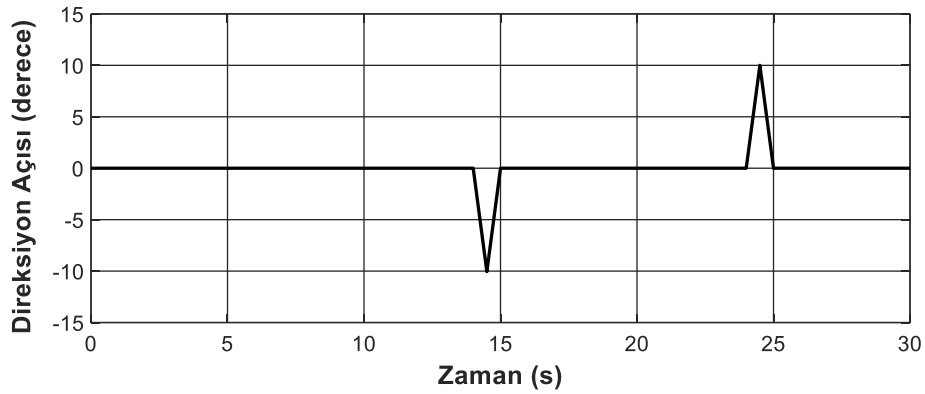
8.1. Elektrikli Aracın Dinamik Modelinden Alınan Sonuçlar

Dört tekerlekten tahrikli bir EA'nın dinamik modeli Matlab/Simulink'te gerçekleştirilmiştir. Bu model Bölüm 6'da kapsamlı olarak anlatılmıştı. Bu bölümde ise tasarlanan modelden alınan sonuçlar verilmektedir. Modelde aracın referans hız değeri 50 km/saat olarak alınmıştır. Bu hız değeri gerçek sistemden alınan hız değeri ile karşılaştırılarak bir PID kontrolörden geçirilmiştir. Buna göre EA'nın gerçek hız değerinin, referans hızı yakaladığı Şekil 8.1'de görülmektedir.



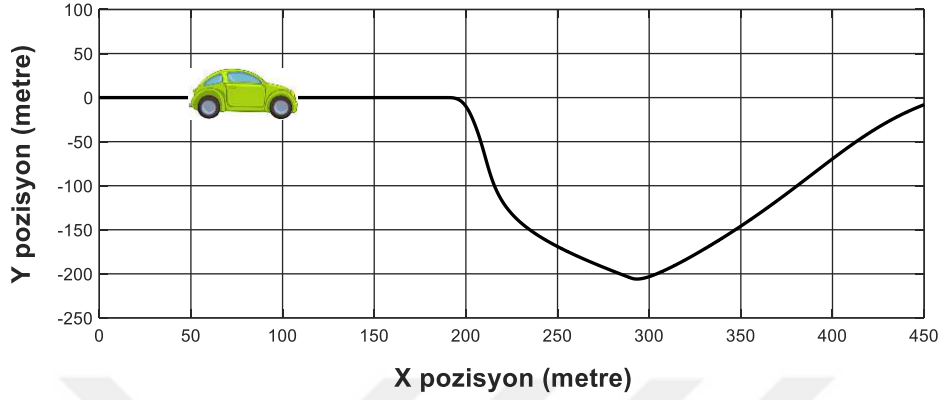
Şekil 8.1. EA'nın gerçek hız ve referans hız eğrileri

Modelde direksiyon açısı Şekil 8.2'deki gibi değişken pozitif ve negatif değerlerde alınmaktadır. Böylece aracın düz yolda, sağa ve sola eğimli yollarda gittiği durumlar incelenmektedir.



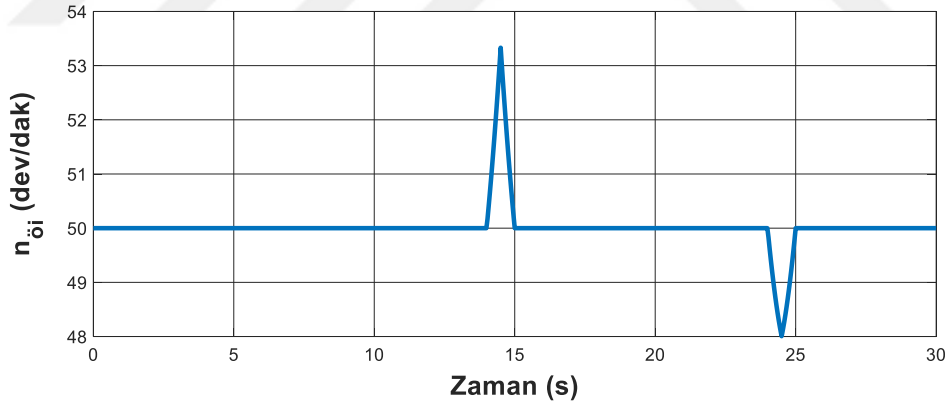
Şekil 8.2. Dinamik modelde kullanılan direksiyon açısının eğrisi

Şekil 8.2'deki direksiyon açı değerleri ve EA'nın referans hız değeri için aracın yol eğrisi çizdirilmiştir. Direksiyon açısı değiştikçe aracın çizdiği yol da değişecektir. Buna göre EA'nın verilen yol profili üzerinde güvenli bir şekilde hareket ettiği Şekil 8.3'te gözlemlenmiştir.

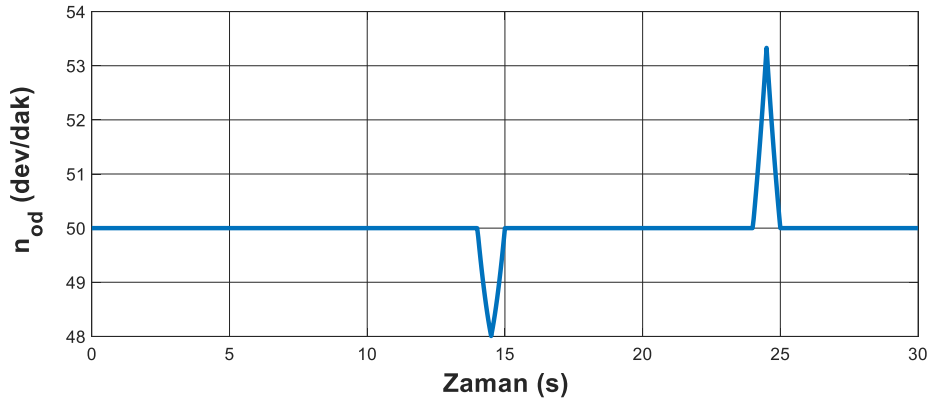


Şekil 8.3. Farklı direksiyon açıları için EA'nın yol eğrisi

Daha sonra tasarlanan EDS modeli ile direksiyon açısı ve araç hızına bağlı olarak EA'nın dört tekerlek hızları hesaplanmıştır. Buradan hesaplanan hızlar EA'nın referans hız değerleridir. Buna göre ön iç ve dış tekerlek hızları sırasıyla Şekil 8.4 ve Şekil 8.5'te gösterilmiştir.

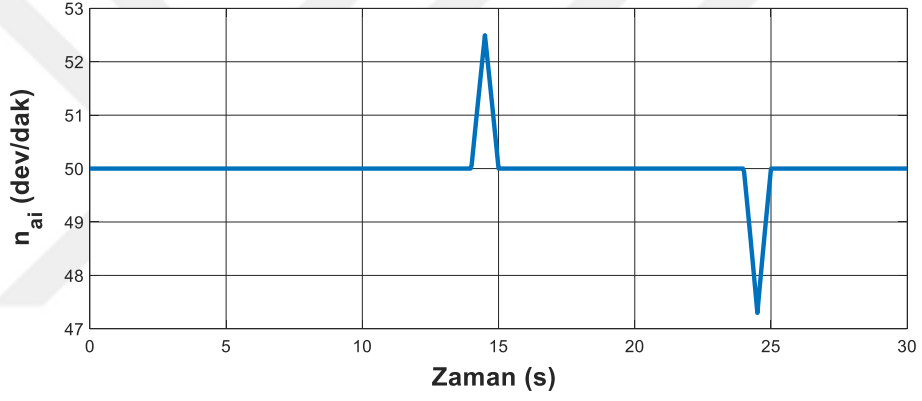


Şekil 8.4. EDS'den hesaplanan ön iç tekerlek hızı

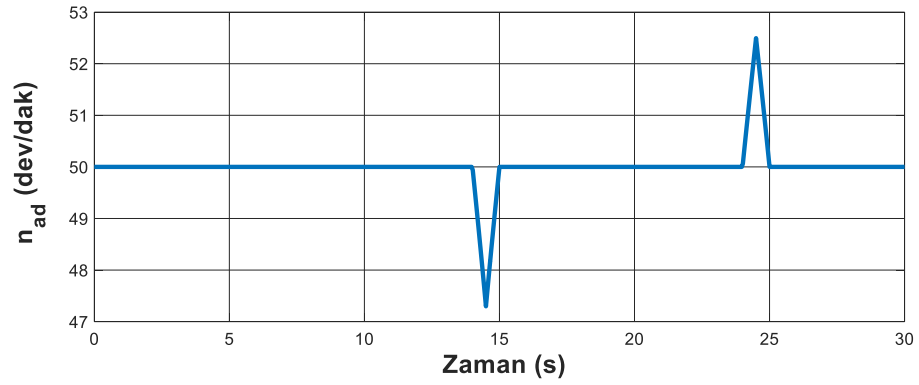


Şekil 8.5. EDS'den hesaplanan ön dış tekerlek hızı

Arka iç ve dış tekerlek hızları ise sırasıyla Şekil 8.6 ve Şekil 8.7'de gösterilmiştir.



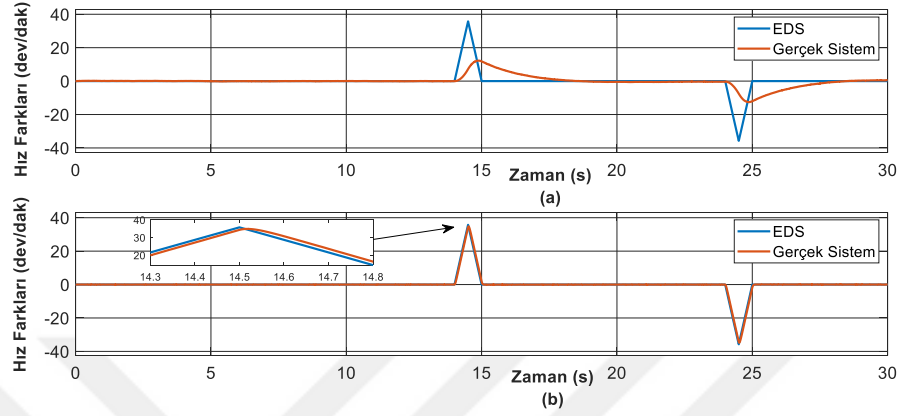
Şekil 8.6. EDS'den hesaplanan arka iç tekerlek hızı



Şekil 8.7. EDS'den hesaplanan arka dış tekerlek hızı

EDS tasarlandıktan sonra, araç dinamiklerinden oluşan EA'nın gerçek sistemi modellenmiştir. Gerçek sistemden alınan hız değerleri değiştikçe sistemin stabilitesi bozulacaktır.

Bunu dengelemek için bir PID kontrolör tasarlanmıştır. Tasarlanan kontrolörde EDS tarafından hesaplanan referans hızlar ile gerçek sistem tarafından kontrolörsüz ve kontrolörlü olarak elde edilen tekerlek hızlarının karşılaştırılması yapılmıştır. Bu karşılaştırma EA'nın ön tekerlekleri için sırasıyla Şekil 8.8'de gösterilmiştir.



Şekil 8.8. a) Kontrolörsüz sistem b) Kontrolörlü sistem için EDS ile gerçek sistemin sağ ve sol ön tekerlek hız farkları

Şekil 8.8'de görüldüğü gibi, kontrolörsüz sistemde EDS çıkışından alınan ön tekerlek hızları arasındaki fark, gerçek sistemin çıkışından alınan hız farklarından oldukça farklı iken, kontrolörlü sistemde bu fark eşitlenerek EA'nın stabilitesi sağlanmıştır.

8.2. Codesys, Matlab/Simulink Simulasyonları ve Deneysel Sonuçlar

EA'nın dinamik modeli ve EDS tasarımı Matlab/Simulink'te gerçekleştirildikten sonra deney düzeneği kurulmuştur. Codesys'ten ve Simulink simulasyonlarından elde edilen motor hız değerleri, motor sürücülerine gönderilmiş ve tasarlanan EDS'ye göre dört motor istenilen devirde döndürülmüştür. Simülasyonlarda direksiyon açısının değerleri 1°'den 10°'ye kadar 1° aralıklarla değiştirilmiştir ve aracın hız değerleri 50 km/s, 100 km/s ve 150 km/s olarak alınmıştır. Bu hız değerlerine karşılık gelen dev/dak değerleri sırasıyla 335.769, 671.539, 1007.309 dev/dak'tır. Bu değerler kullanılarak ön iç (n_{oi}) ve dış tekerlekler ($n_{öd}$), arka iç (n_{ai}) ve dış tekerlekler (n_{ad}) olmak üzere tekerlek hızları simülasyonlarda dev/dak değerleri olarak ayrı ayrı hesaplanmıştır. Ayrıca bir takometre kullanılarak deneysel sonuçlar da elde edilmiştir. Buna göre tasarlanan EDS'yi doğrulamak için, EA'nın farklı hız değerlerinde Codesys, Matlab/Simulink simulasyon sonuçları, deneysel sonuçlarla Tablo 8.1, 8.2 ve 8.3'te karşılaştırılmıştır.

Tablo 8.1. 50 km/s araç hızı ve farklı direksiyon açıları için Codesys, Simulink ve deneysel sonuçlar

δ ($^{\circ}$)	V (km/s)	$n_{\delta i}$ (dev/dak)	$n_{\delta d}$ (dev/dak)	$n_{a i}$ (dev/dak)	$n_{a d}$ (dev/dak)
Codesys Sonuçları					
1	50	334.2512	337.3773	334.0318	337.4944
2	50	332.8218	339.0727	332.2791	339.2061
3	50	331.4827	340.8557	330.5111	340.9056
4	50	330.2347	342.7259	328.7267	342.5940
5	50	329.0792	344.683	326.9251	344.2719
6	50	328.0172	346.7271	325.1051	345.9399
7	50	327.0502	348.8581	323.2659	347.5989
8	50	326.1797	351.0762	321.4063	349.2495
9	50	325.4075	353.3816	319.5253	350.8925
10	50	324.7354	355.7749	317.6217	352.5284
Matlab/Simulink Sonuçları					
1	50	334.3	337.4	334.0	337.5
2	50	332.8	339.1	332.3	339.2
3	50	331.5	340.9	330.5	340.9
4	50	330.2	342.7	328.7	342.6
5	50	329.1	344.7	326.9	344.3
6	50	328.0	346.7	325.1	345.9
7	50	327.1	348.6	323.3	347.6
8	50	326.2	351.1	321.4	349.2
9	50	325.4	353.4	319.5	350.9
10	50	324.7	355.8	317.6	352.5
Deneysel Sonuçlar					
1	50	334.1	336.8	333.7	337.1
2	50	332.5	338.6	331.9	338.8
3	50	331.2	340.1	329.8	340.5
4	50	329.9	341.9	328.0	341.9
5	50	328.8	344.0	326.2	343.7
6	50	327.8	345.9	324.5	345.5
7	50	326.8	347.8	322.8	346.9
8	50	325.9	350.5	320.9	348.8
9	50	325.2	352.8	319.0	350.1
10	50	324.5	355.2	316.8	351.9
Codesys ve Deneysel Sonuçlara göre Bağıl Hata (%)					
1	50	0.045	0,171	0,099	0,117
2	50	0.096	0,139	0,114	0,120
3	50	0.085	0,222	0,215	0,119
4	50	0.101	0,241	0,221	0,203
5	50	0.084	0,198	0,222	0,166
6	50	0.066	0,239	0,186	0,127
7	50	0.076	0,303	0,144	0,201
8	50	0.085	0,164	0,158	0,129
9	50	0.063	0,165	0,164	0,226
10	50	0.072	0,162	0,259	0,178

Tablo 8.2. 100 km/s araç hızı ve farklı direksiyon açıları için Codesys, Simulink ve deneysel sonuçlar

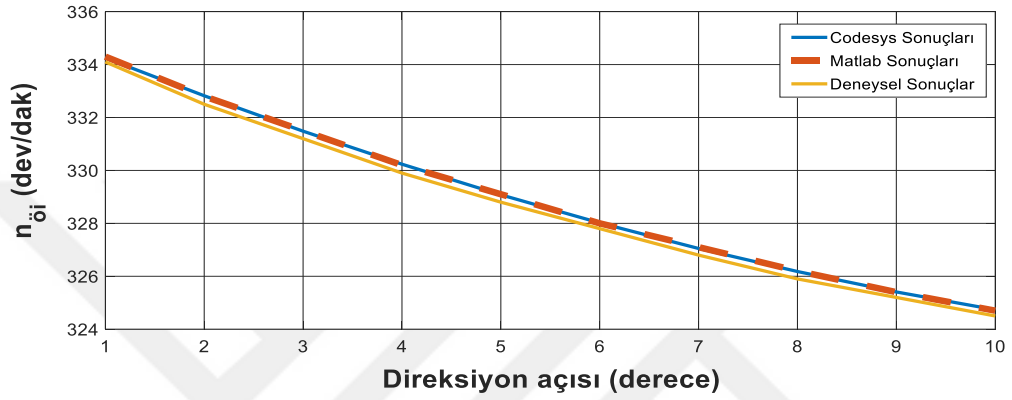
δ (°)	V (km/s)	$n_{\delta i}$ (dev/dak)	$n_{\delta d}$ (dev/dak)	n_{ai} (dev/dak)	n_{ad} (dev/dak)
Codesys Sonuçları					
1	100	668.5023	674.7545	668.0636	674.9888
2	100	665.6436	678.1454	664.5582	678.4121
3	100	662.9653	681.7114	661.0222	681.8112
4	100	660.4694	685.4517	657.4534	685.1880
5	100	658.1584	689.366	653.8502	688.5437
6	100	656.0343	693.4542	650.2103	691.8798
7	100	654.1003	697.7161	646.5318	695.1978
8	100	652.3595	702.1523	642.8126	698.4991
9	100	650.8151	706.7632	639.0506	701.7850
10	100	649.4708	711.5498	635.2434	705.0568
Matlab/Simulink Sonuçları					
1	100	668.5	674.8	668.1	675.0
2	100	665.6	678.1	664.6	678.4
3	100	663.0	681.7	661.0	681.8
4	100	660.5	685.5	657.5	685.2
5	100	658.2	689.4	653.9	688.5
6	100	656.0	693.5	650.2	691.9
7	100	654.1	697.7	646.5	695.2
8	100	652.4	702.2	642.8	698.5
9	100	650.8	706.8	639.1	701.8
10	100	649.5	711.5	635.2	705.1
Deneysel Sonuçlar					
1	100	667.8	674.1	667.7	674.5
2	100	664.9	677.7	663.8	677.6
3	100	662.6	680.8	660.7	681.0
4	100	659.9	684.7	656.6	684.4
5	100	657.5	688.9	653.1	687.6
6	100	655.5	692.8	649.3	691.2
7	100	653.7	697.1	645.6	694.6
8	100	651.8	701.5	642.2	697.7
9	100	650.2	705.6	638.5	700.3
10	100	648.7	711.0	634.8	704.3
Codesys ve Deneysel Sonuçlara göre Bağlı Hata (%)					
1	100	0,105	0,097	0,054	0,072
2	100	0,112	0,066	0,114	0,120
3	100	0,055	0,134	0,049	0,119
4	100	0,086	0,110	0,130	0,115
5	100	0,100	99,900	0,115	0,137
6	100	0,081	0,094	0,140	0,098
7	100	0,061	0,088	0,144	0,086
8	100	0,086	0,093	0,095	0,114
9	100	0,095	0,165	0,086	0,212
10	100	0,119	0,077	0,070	0,107

Tablo 8.3. 150 km/s araç hızı ve farklı direksiyon açıları için Codesys, Simulink ve deneysel sonuçlar

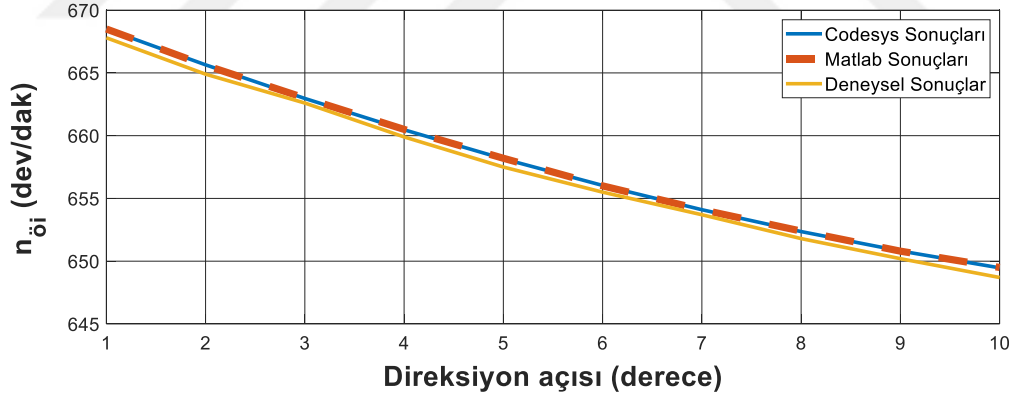
δ (°)	V (km/s)	$n_{\delta i}$ (dev/dak)	$n_{\delta d}$ (dev/dak)	n_{ai} (dev/dak)	n_{ad} (dev/dak)
Codesys Sonuçları					
1	150	1002.753	1012.132	1002.095	1012.483
2	150	998.4655	1017.218	996.8373	1017.618
3	150	994.4479	1022.567	991.5332	1022.717
4	150	990.7042	1028.178	986.1802	1027.782
5	150	987.2375	1034.049	980.7751	1032.815
6	150	984.0514	1040.181	975.3155	1037.820
7	150	981.1505	1046.574	969.7977	1042.797
8	150	978.5392	1053.228	964.2191	1047.749
9	150	976.2225	1060.145	958.5759	1052.677
10	150	974.2062	1067.325	952.8652	1057.585
Matlab/Simulink Sonuçları					
1	150	1003.0	1012.0	1002.0	1012.0
2	150	998.5	1017.0	996.8	1018.0
3	150	994.4	1023.0	991.5	1023.0
4	150	990.7	1028.0	986.2	1028.0
5	150	987.2	1034.0	980.8	1033.0
6	150	984.1	1040.0	975.3	1038.0
7	150	981.2	1047.0	969.8	1043.0
8	150	978.5	1053.0	964.2	1048.0
9	150	976.2	1060.0	958.6	1053.0
10	150	974.2	1067.0	952.9	1058.0
Deneysel Sonuçlar					
1	150	1002.7	1010.7	1000.4	1011.2
2	150	997.8	1015.8	995.1	1017.0
3	150	993.5	1022.3	990.2	1021.9
4	150	990.0	1027.4	985.2	1026.8
5	150	986.6	1032.5	979.3	1031.5
6	150	983.5	1039.0	974.1	1037.1
7	150	980.8	1046.1	968.2	1041.8
8	150	977.9	1052.2	963.5	1046.5
9	150	975.3	1058.6	957.4	1051.9
10	150	973.5	1066.5	951.2	1057.0
Codesys ve Deneysel Sonuçlara göre Bağlı Hata (%)					
1	150	0,005	0,141	0,169	0,127
2	150	0,067	0,139	0,174	0,061
3	150	0,095	0,026	0,134	0,080
4	150	0,071	0,076	0,099	0,096
5	150	0,065	0,150	0,150	0,127
6	150	0,056	0,114	0,125	0,069
7	150	0,036	0,045	0,165	0,096
8	150	0,065	0,098	0,075	0,119
9	150	0,094	0,146	0,123	0,074
10	150	0,072	0,077	0,175	0,055

Tablo 8.1, 8.2 ve 8.3'te görüldüğü gibi, EA'nın dört tekerlek hızlarının Codesys sonuçları Matlab/Simulink ve deneysel sonuçlarla karşılaştırılarak doğrulanmıştır. Aynı zamanda dış tekerlek hızları iç tekerlek hızlarından daha yüksektir. Buna göre, EDS optimal ve verimli bir şekilde tasarlanmıştır.

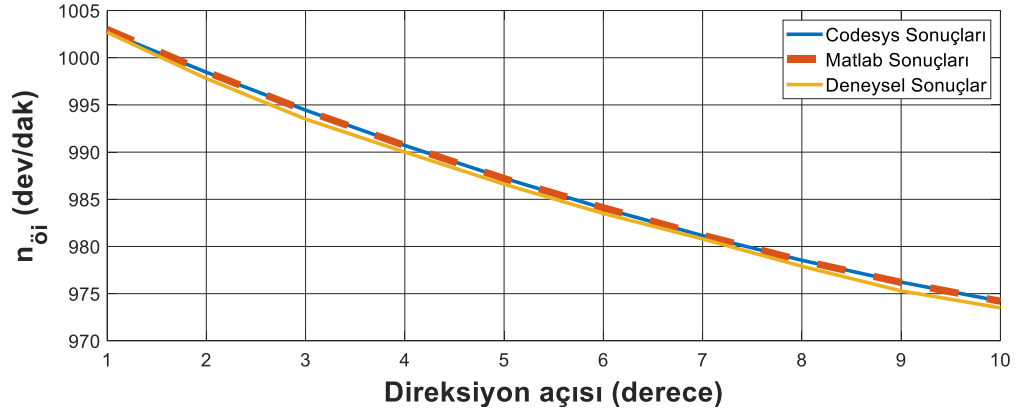
Elde edilen bu sonuçlara göre dört tekerlek hızlarının Codesys, Matlab/Simulink ve deneysel sonuçları ayrı ayrı çizdirilmiştir. Ön iç tekerlek hızı, Şekil 8.9, 8.10, 8.11'de gösterildiği gibi 50 km/s, 100 km/s ve 150 km/s araç hızları için sırasıyla hesaplanmıştır.



Şekil 8.9. 50 km/s araç hızı için ön iç tekerlek hızının Codesys, Matlab ve deneysel sonuçları

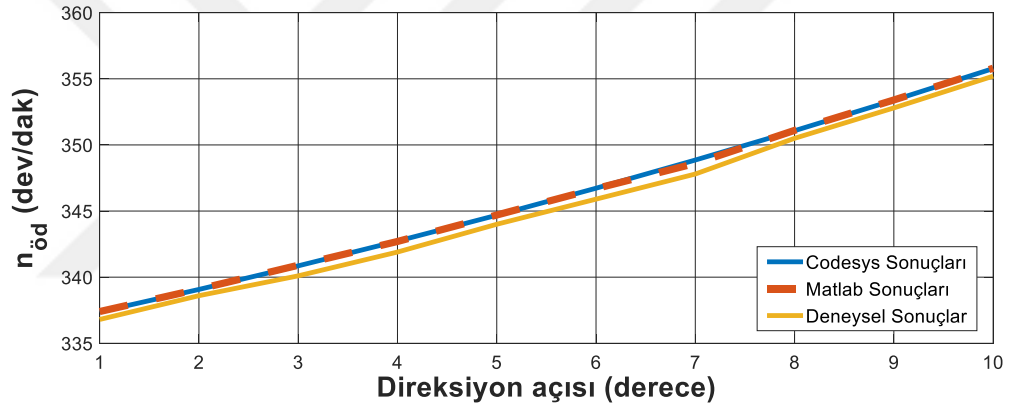


Şekil 8.10. 100 km/s araç hızı için ön iç tekerlek hızının Codesys, Matlab ve deneysel sonuçları

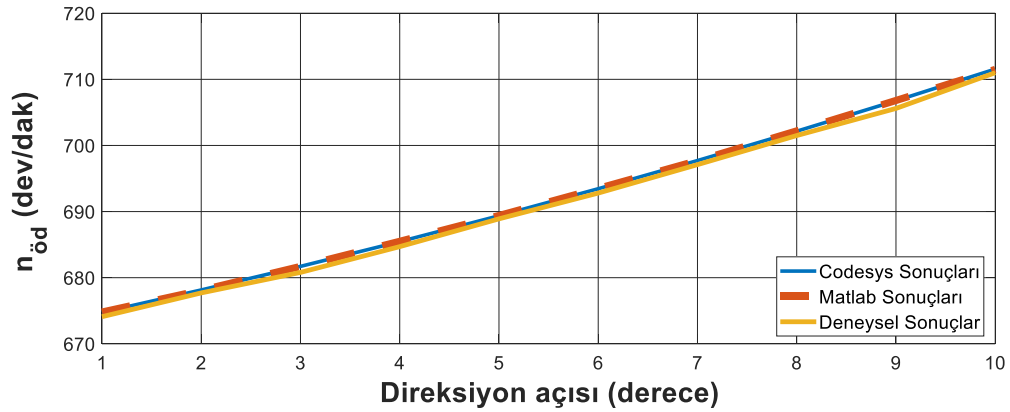


Şekil 8.11. 150 km/s araç hızı için ön iç tekerlek hızının Codesys, Matlab ve deneysel sonuçları

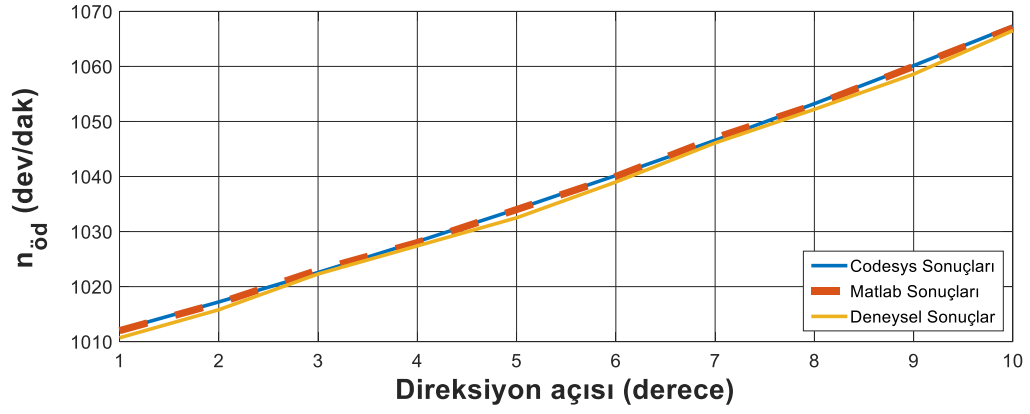
$n_{öd}$, Şekil 8.12, 8.13, 8.14'te gösterildiği gibi 50 km/s, 100 km/s ve 150 km/s araç hızları için sırasıyla çizdirilmiştir.



Şekil 8.12. 50 km/s araç hızı için ön dış tekerlek hızının Codesys, Matlab ve deneysel sonuçları

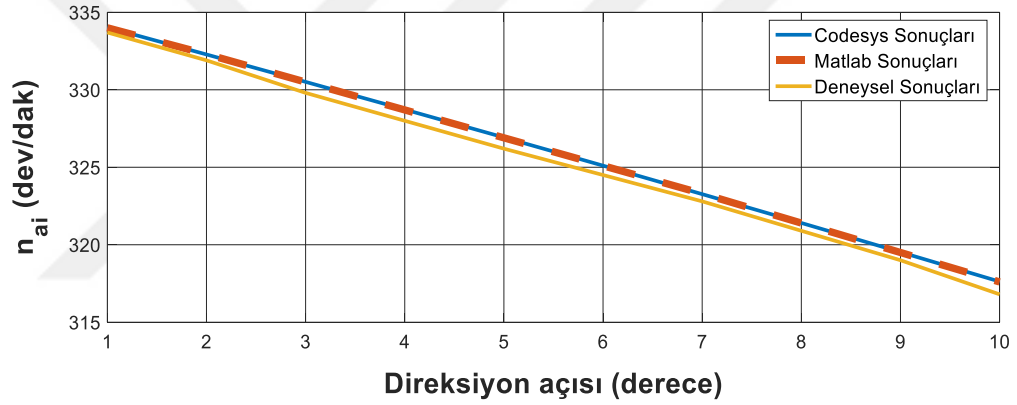


Şekil 8.13. 100 km/s araç hızı için ön dış tekerlek hızının Codesys, Matlab ve deneysel sonuçları

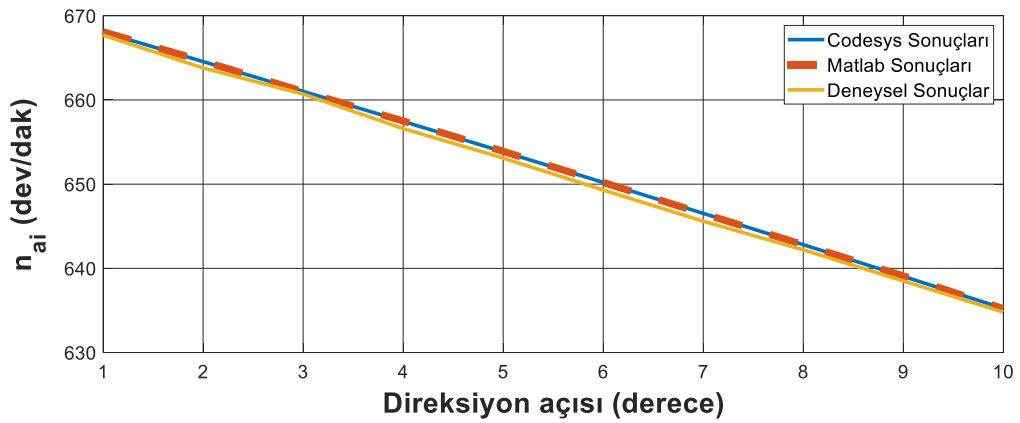


Şekil 8.14. 150 km/s araç hızı için ön dış tekerlek hızının Codesys, Matlab ve deneysel sonuçları

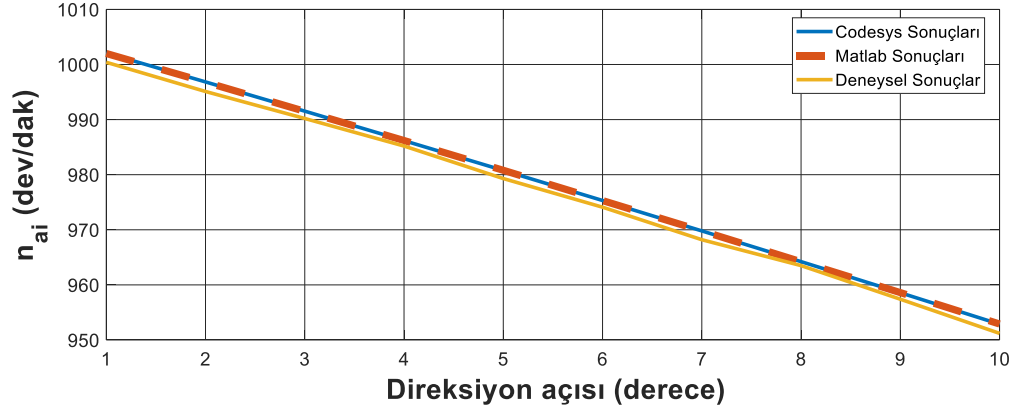
n_{ai} , Şekil 8.15, 8.16, 8.17'de gösterildiği gibi 50 km/s, 100 km/s ve 150 km/s araç hızları için sırasıyla çizdirilmiştir.



Şekil 8.15. 50 km/s araç hızı için arka iç tekerlek hızının Codesys, Matlab ve deneysel sonuçları

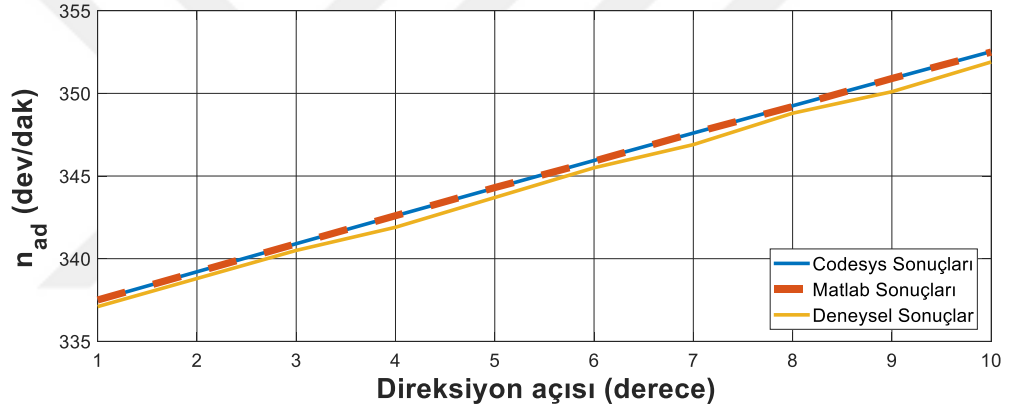


Şekil 8.16. 100 km/s araç hızı için arka iç tekerlek hızının Codesys, Matlab ve deneysel sonuçları

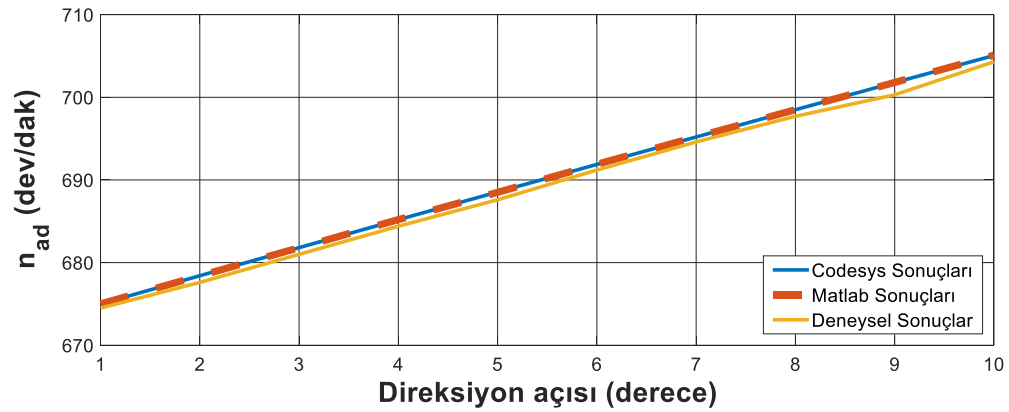


Şekil 8.17. 150 km/s araç hızı için arka iç tekerlek hızının Codesys, Matlab ve deneysel sonuçları

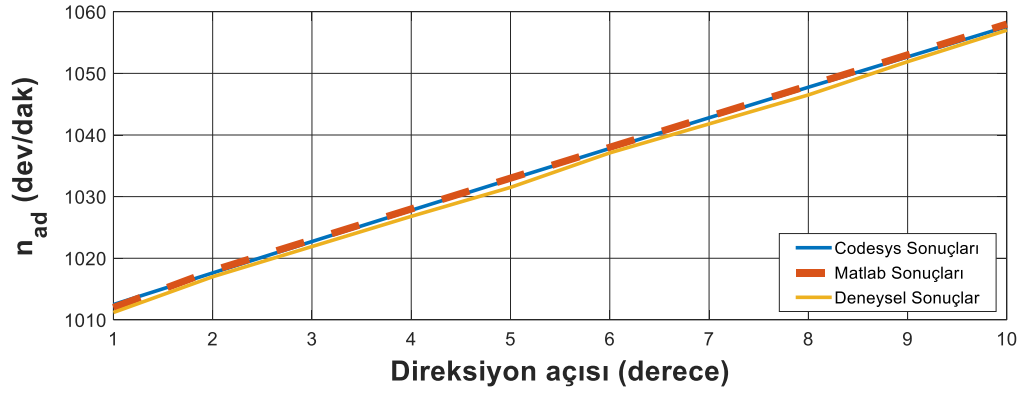
n_{ad} , Şekil 8.18, 8.19 ve 8.20’de gösterildiği gibi 50 km/s, 100 km/s ve 150 km/s araç hızları için sırasıyla çizdirilmiştir.



Şekil 8.18. 50 km/s araç hızı için arka dış tekerlek hızının Codesys, Matlab ve deneysel sonuçları



Şekil 8.19. 100 km/s araç hızı için arka dış tekerlek hızının Codesys, Matlab ve deneysel sonuçları



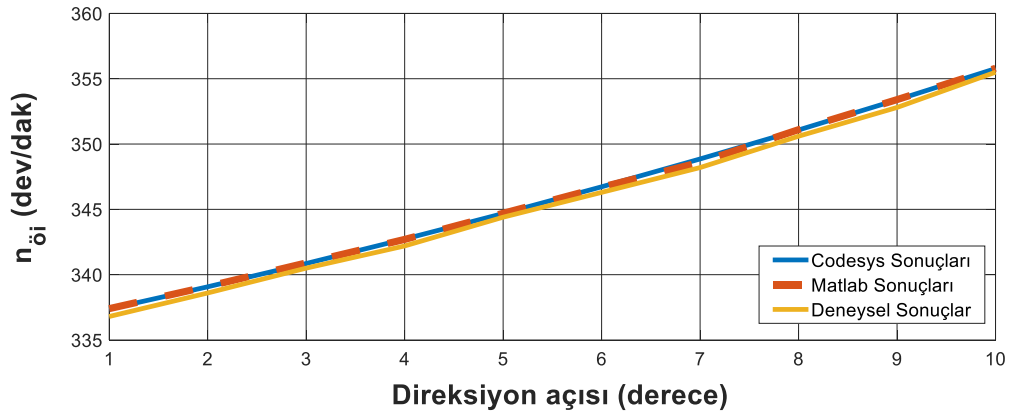
Şekil 8.20. 150 km/s araç hızı için arka dış tekerlek hızının Codesys, Matlab ve deneysel sonuçları

Araç 50 km/saat hızla giderken direksiyonun tersi yönü için EA'nın dört tekerlek hızlarının Codesys, Matlab/Simulink ve deneysel sonuçları Tablo 8.4'te karşılaştırılmıştır.

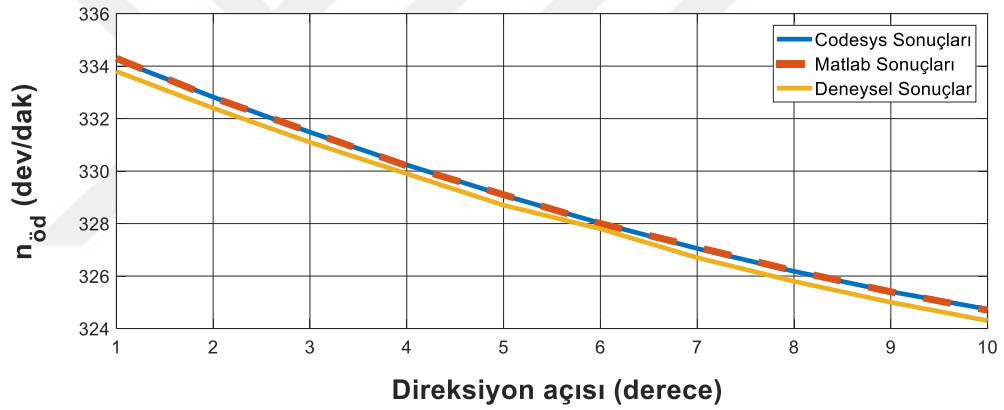
Tablo 8.4. 50 km/s araç hızı ve direksiyonun tersi yönü için Codesys, Simulink ve deneysel sonuçlar

$\delta (^{\circ})$	V (km/s)	$n_{\ddot{o}i}$ (dev/dak)	$n_{\ddot{o}d}$ (dev/dak)	n_{ai} (dev/dak)	n_{ad} (dev/dak)
Ters Yön		Codesys Sonuçları			
1	50	337.3773	334.2512	337.4944	334.0318
2	50	339.0727	332.8218	339.2061	332.2791
3	50	340.8557	331.4827	340.9056	330.5111
4	50	342.7259	330.2347	342.594	328.7267
5	50	344.683	329.0792	344.2719	326.9251
6	50	346.7271	328.0172	345.9399	325.1051
7	50	348.8581	327.0502	347.5989	323.2659
8	50	351.0762	326.1797	349.2495	321.4063
9	50	353.3816	325.4075	350.8925	319.5253
10	50	355.7749	324.7354	352.5284	317.6217
		Matlab/Simulink Sonuçları			
1	50	337.4	334.3	337.5	334.0
2	50	339.1	332.8	339.2	332.3
3	50	340.9	331.5	340.9	330.5
4	50	342.7	330.2	342.6	328.7
5	50	344.7	329.1	344.3	326.9
6	50	346.7	328.0	345.9	325.1
7	50	348.6	327.1	347.6	323.3
8	50	351.1	326.2	349.2	321.4
9	50	353.4	325.4	350.9	319.5
10	50	355.8	324.7	352.5	317.6
		Deneyisel Sonuçlar			
1	50	336.8	333.8	337.0	333.7
2	50	338.6	332.4	338.7	331.8
3	50	340.5	331.1	340.4	329.8
4	50	342.2	329.9	342.2	328.1
5	50	344.4	328.7	343.9	326.4
6	50	346.3	327.8	345.4	324.8
7	50	348.2	326.7	347.3	322.7
8	50	350.6	325.8	348.8	320.8
9	50	352.8	325.0	350.5	319.1
10	50	355.5	324.3	352.2	317.0
Codesys ve Deneyisel Sonuçlara göre Bağlı Hata (%)					
1	50	0,171	0,135	0,146	0,099
2	50	0,139	0,127	0,149	0,144
3	50	0,104	0,115	0,148	0,215
4	50	0,153	0,101	0,115	0,191
5	50	0,082	0,115	0,108	0,161
6	50	0,123	0,066	0,156	0,094
7	50	0,189	0,107	0,086	0,175
8	50	0,136	0,116	0,129	0,189
9	50	0,165	0,125	0,112	0,133
10	50	0,077	0,134	0,093	0,196

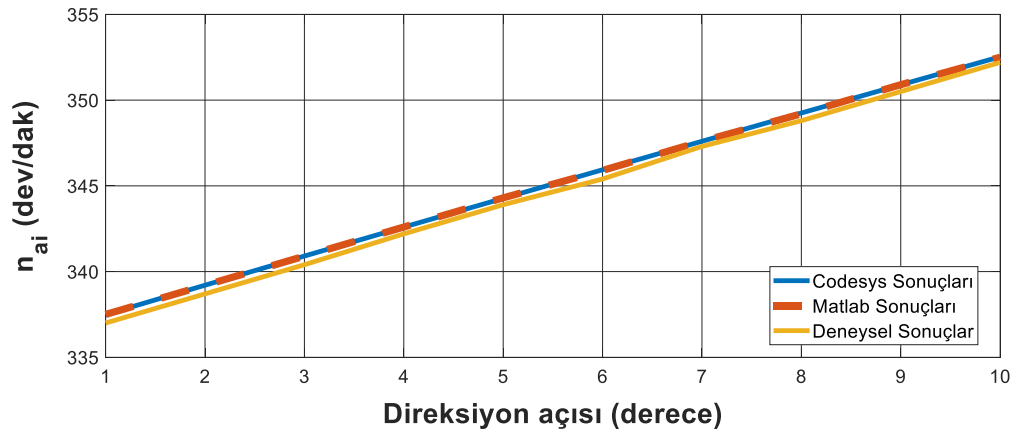
Tablo 8.4'teki sonuçlara göre dört tekerlek hızları sırasıyla Şekil 8.21, 8.22, 8.23 ve 8.24'te çizdirilmiştir.



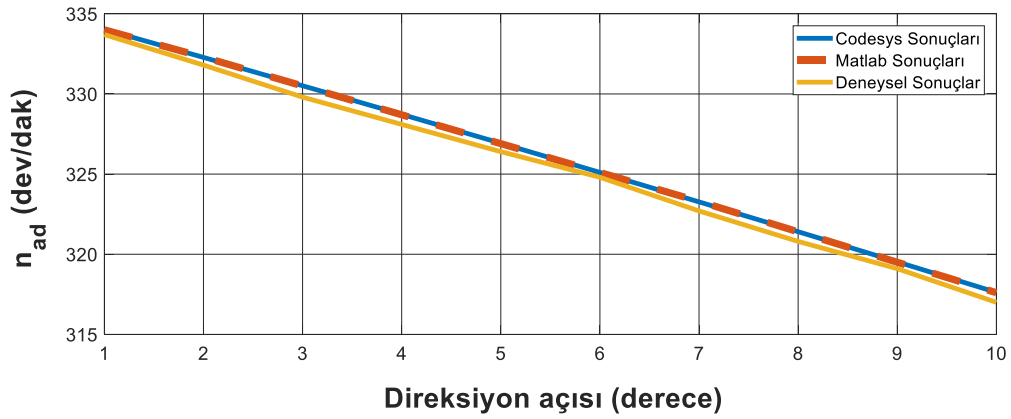
Şekil 8.21. 50 km/s araç hızında direksiyonun tersi yönü için ön iç tekerlek hızının Codesys, Matlab ve deneysel sonuçları



Şekil 8.22. 50 km/s araç hızında direksiyonun tersi yönü için ön dış tekerlek hızının Codesys, Matlab ve deneysel sonuçları



Şekil 8.23. 50 km/s araç hızında direksiyonun tersi yönü için arka iç tekerlek hızının Codesys, Matlab ve deneysel sonuçları



Şekil 8.24. 50 km/s araç hızında direksiyonun tersi yönü için arka dış tekerlek hızının Codesys, Matlab ve deneysel sonuçları

Yukarıdaki şekillerde görüldüğü gibi, Matlab/Simulink ve Codesys sonuçları hemen hemen aynıdır ve deneysel sonuçlar da küçük bir farkla onlara oldukça yakındır. Bu farkın nedeni, takometrenin kullanımı ve çözünürlüğünden kaynaklanmaktadır. Böylece tüm sonuçlar birbirleriyle karşılaştırılarak doğrulanmıştır.

8.3. Elektrikli Aracın Güvenli Sürüşü Üzerine Yapılan Çalışmaların Sonuçları

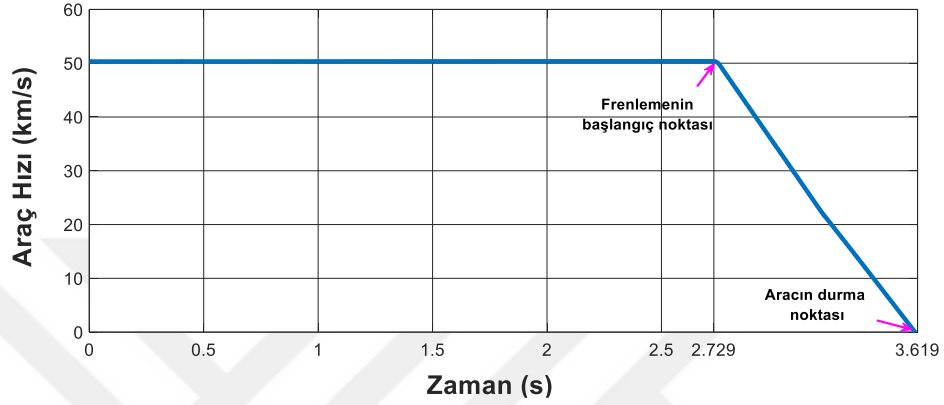
8.3.1. Kameradan Alınan Görüntülere Göre Engel Tespiti

Bu bölümde, EA'nın güvenli sürüşü üzerine yapılan çalışmaların sonuçları verilmiştir. İlk olarak, Raspberry Pi'ye bağlı bir kameradan alınan görüntülere göre engel tespiti yapılmıştır. Aracın önüne engel geldiğinde SMSM sürücülerine CAN-Bus üzerinden bilgi gönderilerek motorlar durdurulmaktadır.

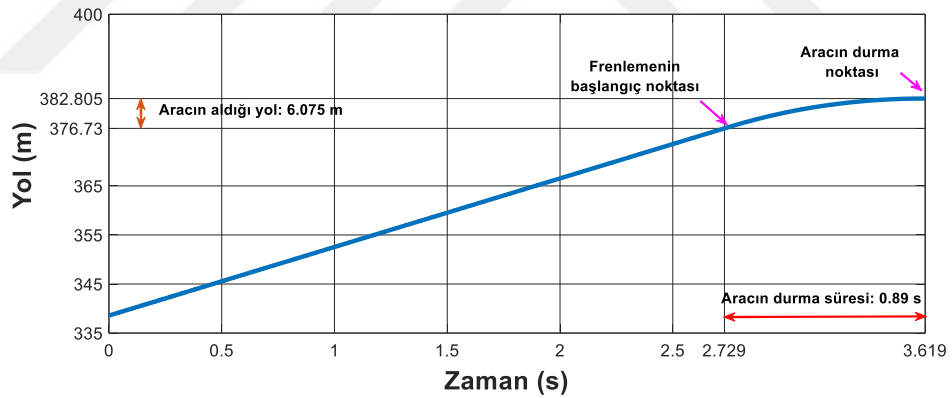
Son yıllarda sensör tabanlı uygulamalar çeşitli sensörleri kullanarak basit uygulamalar gerçekleştirirken, görüntü tabanlı teknolojiler kameralar ile Bilgisayarlı Görme ve Makine Öğrenimini kullanarak ileri düzey uygulamaların yapılmasını sağlamıştır. Bu nedenle uzun mesafelerde engel tespiti için kamera ile görüntüler tanımlanarak motorlara gereken sinyaller gönderilmektedir. Engel tespitinde bir engel olduğu bilinmeden nesne tespiti (detection) işlemi yapılarak engeller belirlenir ve buna göre konumu ve pozisyonu öğrenilir. Ayrıca pozisyon bilgisi alınarak lokalizasyon işlemi de yapılabilir. Mesafe sensörü gibi sensörler ise nesne belirleme işlemi yapamaz ve sadece yakın mesafede etkindir. Yalnızca varlığı bilinen bir cismin mesafesini ölçer. Kameralar ise hem nesne belirlemede hem de tanımda kullanılır. Kamera ve sensörden hangisi engeli tespit ederse sinyal görevini o üstlenir. Böylece engelin kamera ile tespit edilmesi ileriye dönük çalışmalara basamak oluşturarak, tanıma (recognition) ve olayları yorumlamaları sağlamaktadır. Bu çalışmada gerçekleştirilen yazılım ile kameranın önüne gelen engel tespit

edilerek o andaki hız ve mesafe ölçümüne göre motorlara bir fren sinyali gönderilir. Böylece motorlar durdurularak güvenli sürüş sağlanmaktadır.

Buna göre ilk olarak EA, 50 km/s hızla giderken frenleme durumunda hızların değişimi çizdirilerek aracın aldığı yol uzunlukları ve durma süreleri bulunmuştur. EA, 50 km/s hızla giderken frenleme yapılarak hızı sıfıra düşürülmüştür. Bu durumda hızın değişimi ve yol eğrisi Şekil 8.25 ve 8.26'da sırasıyla gösterilmiştir.



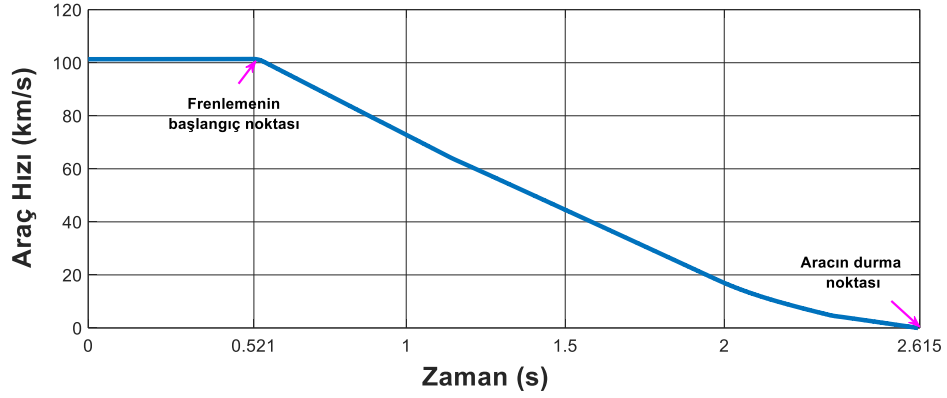
Şekil 8.25. EA 50 km/s hızda giderken frenleme yapılması durumunda hızın değişimi



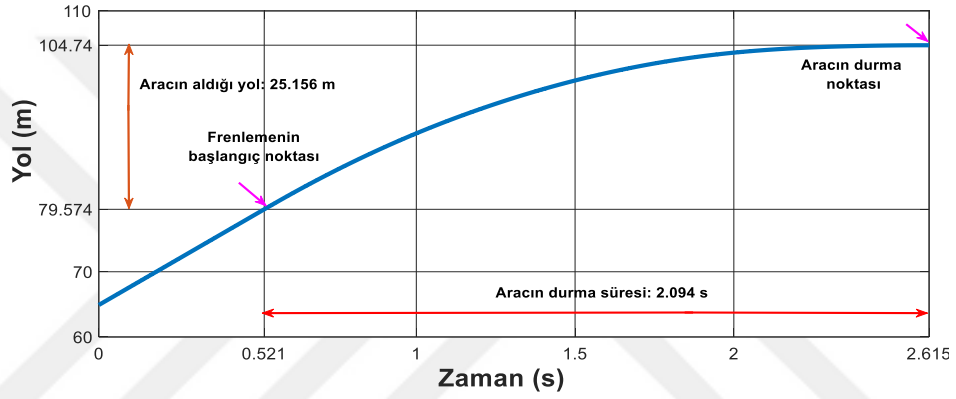
Şekil 8.26. EA 50 km/s hızda giderken frenleme yapılması durumunda aldığı yol eğrisi

Şekil 8.26'da görüldüğü gibi 50 km/s araç hızında, 2,729. saniyede frene basılmış ve EA 3,619. saniyede durmuştur. Bu durumda EA'nın durma süresi 0,89 s ve aldığı yol 6,075 m'dir.

Daha sonra EA 100 km/s hızla giderken frenleme yapılması durumunda sonuçlar tekrar alınmıştır. Buna göre araç hızının değişimi ve yol eğrisi Şekil 8.27 ve 8.28'de sırasıyla gösterilmiştir.



Şekil 8.27. EA 100 km/s hızda giderken frenleme yapılması durumunda hızın değişimi

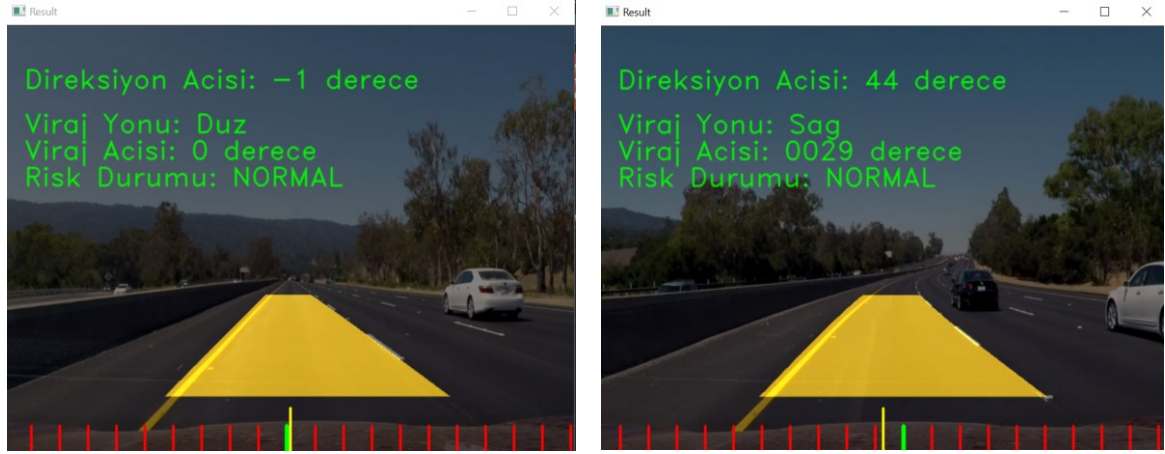


Şekil 8.28. EA 100 km/s hızda giderken frenleme yapılması durumunda aldığı yol eğrisi

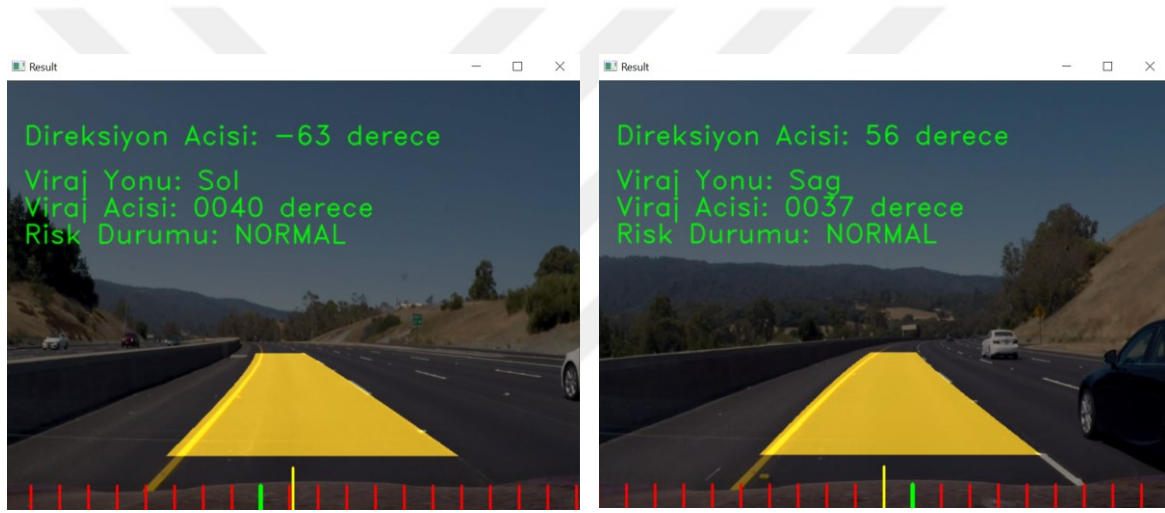
Şekil 8.27 ve 8.28'de görüldüğü gibi 0,521. saniyede frene basılmış ve EA 2,615. saniyede durmuştur. Buna göre 100 km/s araç hızı için durma süresi 2,094 s'ye çıkmış ve araç durma anında 25,156 m yol kat etmiştir.

8.3.2. Viraj Yönü ve Açısının Tespiti ile Risk Hesabı

Segmentasyon işlemi ile aracın viraj yönü ve açısı tespit edilerek risk hesabı yapılmıştır. Buna göre piksellerin yoğunlaştığı bölgelere bakılarak yolun eğimi çıkartılmıştır. Ayrıca direksiyon açısı hesaplanmış, aracın düz ya da eğimli yolda giderken risk oluşturduğu açı değerleri ve durumlar ekranda gösterilmiştir. Bu durumlar düz yol, sağa ve sola eğimli yol olarak belirlenmiştir. Sürüş esnasında riskli bir durum yok ise ekrana “normal”, tehlikeli durum olduğunda ise “yüksek risk” yazdırılmıştır. Bunun yanı sıra merkezkaç ve sürtünme kuvvetleri hesaplanarak aracın savrulma durumları da incelenmiştir. Buna göre 100 km/saat araç hızı için direksiyon açısı, viraj yönü, viraj açısı ve risk durumları sırasıyla Şekil 8.29 ve Şekil 8.30'da gösterilmiştir.



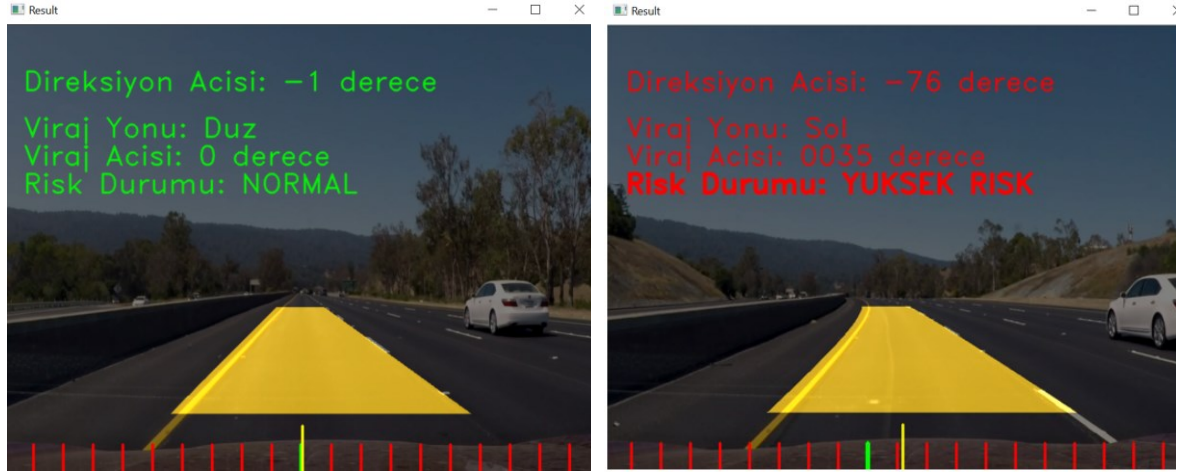
Şekil 8.29. 100 km/s araç hızında sağ ve düz yol için viraj bilgilerinin gösterimi



Şekil 8.30. 100 km/s araç hızında sağa ve sola kıvrılan yol için viraj bilgilerinin gösterimi

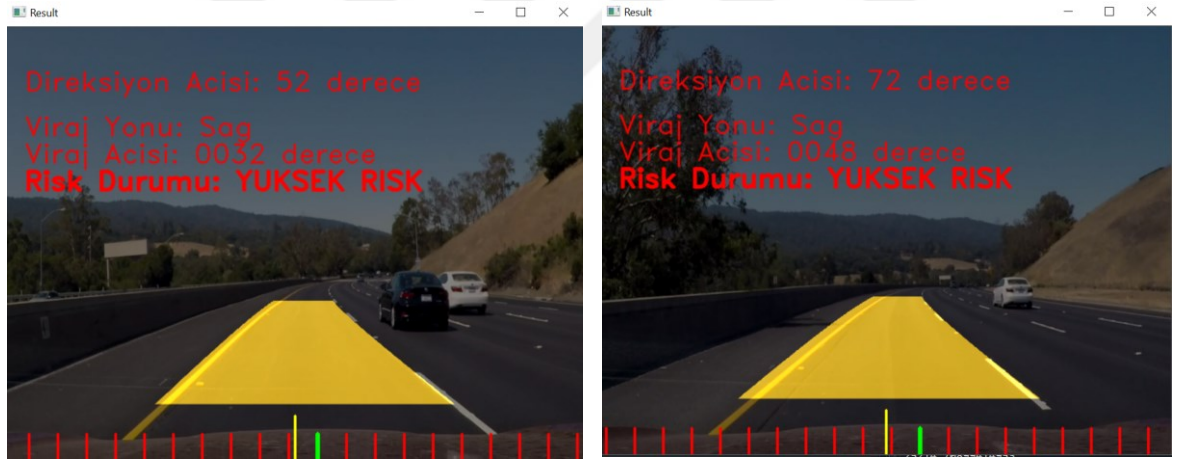
Araç hızı 100 km/saat'ten 200 km/saat'e çıkartılarak sonuçlar Şekil 8.31'deki gibi tekrar alınmıştır. Viraj açısı 0^0 iken, direksiyon açısı -1^0 olduğunda araç düz yolda yüksek hızda gidiyor demektir. Yani riskli bir durum yoktur. Bu yüzden ekrana "normal" yazdırılmıştır.

Araç hızı 200 km/saat için, viraj açısı 35^0 ve direksiyon açısı -76^0 olduğunda araç sola doğru gidiyor demektir. Yüksek hızdan dolayı merkezkaç kuvveti yüksektir ve sürtünme kuvvetini geçtiği an araç yoldan savrulur. Bu durumda ekrana "yüksek risk" yazdırılmıştır.



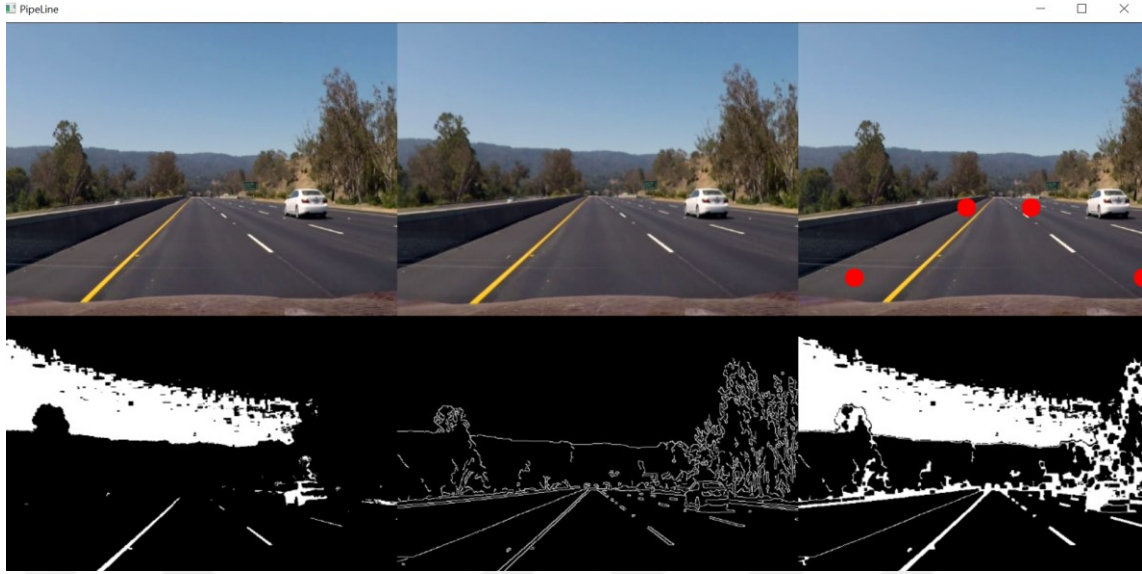
Şekil 8.31. 200 km/s araç hızında sol ve düz yol için viraj bilgilerinin gösterimi

Araç hızı 200 km/saat için, viraj açıları sırasıyla 32^0 ve 48^0 , direksiyon açıları 52^0 ve 72^0 olduğunda araç sağa doğru gidiyor demektir. Burada da araç hızı yüksek olduğundan dolayı merkezkaç kuvveti yüksektir ve sürtünme kuvvetini geçtiği an araç yoldan savrulur. Bu durumda Şekil 8.32’de görüldüğü gibi ekrana “yüksek risk” yazdırılmıştır.



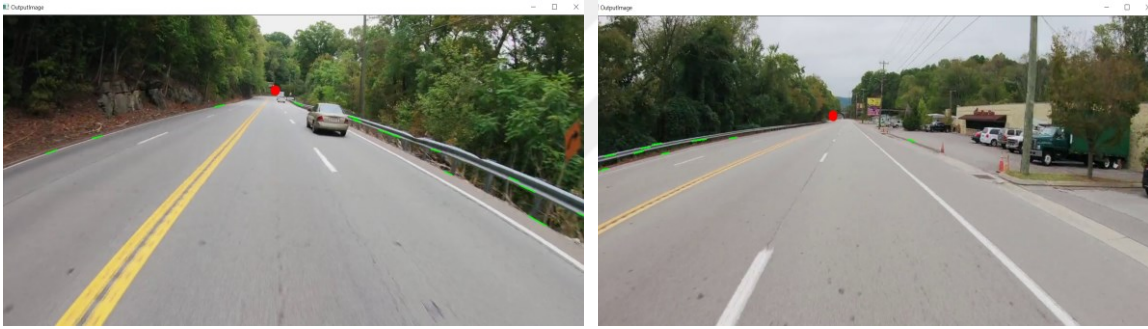
Şekil 8.32. 200 km/s araç hızında sağa kıvrılan yol için viraj bilgilerinin gösterimi

Daha sonra görüntü üzerinde çeşitli morfolojik işlemler gerçekleştirilerek yolun ufuk noktası hesaplanmıştır. Buna göre yol çizgileri, yolun ufuk noktaları, eşikleme işlemi ve Canny kenar filtresinin sonuçları Şekil 8.33’te gösterilmiştir.



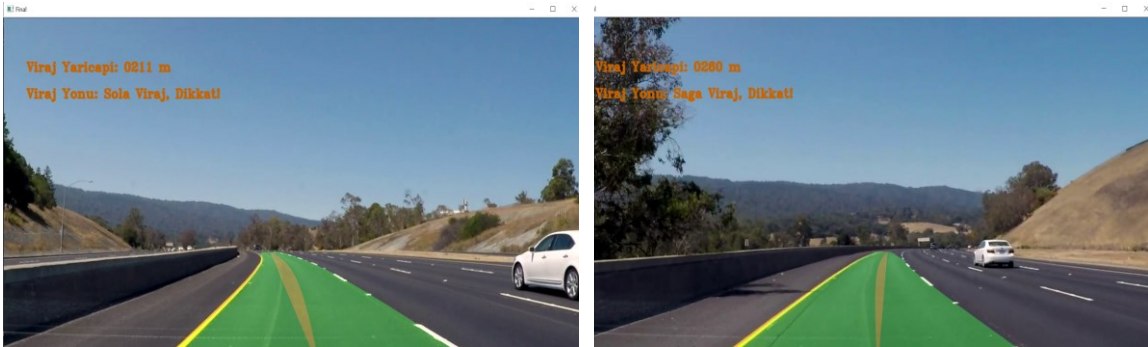
Şekil 8.33. Morfolojik işlemler sonucu

Şekil 8.34’te ise yolun kenar çizgileri çıkartılarak ufuk noktası hesabı gerçekleştirilmiştir.



Şekil 8.34. Yol kenar çizgileri ve ufuk noktasının hesabı

Şekil 8.35’te de farklı bir yol görüntüsü üzerinde segmentasyon işlemi yapılarak viraj yarıçapı hesaplanmıştır. Buna göre viraj yönü tespit edilerek sürücüyü uyarar bir sistem geliştirilmiştir.



Şekil 8.35. Segmentasyon işlemi ve yol viraj yönünün belirlenmesi

Böylece EA'nın güvenli sürüşü üzerine yapılan bu çalışmalar ile özellikle uzun yolda yorgunluk, uykusuzluk ve dikkatsizlik gibi sebeplerle virajlara hızlı giren sürücüler uyarılabilir ve otonom sürüş hedefinin önemli bir bölümü olan ileri sürücü destek sistemi tasarlanmış olmaktadır.

8.4. Sonuçlar ve Değerlendirme

Bu tez çalışmasında, EA'lar için güvenlik önlemleri ele alınmış ve buna göre bir aktif güvenlik sistem tasarlanmıştır. Çalışma, temel olarak iki kısımdan oluşmaktadır. İlk kısımda, dört tekerlek içi SMSM'den oluşan bir EA için EDS tasarlanmıştır. Literatürdeki çalışmalara bakıldığında EDS konusunda çok fazla çalışmanın olmadığı ve EDS'nin genellikle EA'nın sadece ön veya arka tekerlekleri için çalışıldığı görülmektedir. Bu tez çalışmasında ise farklı olarak, bir EDS'nin simülasyonu ve tasarımı EA'nın dört tekerleği için gerçekleştirilmiştir. Dört adet SMSM, her bir tekerleğin içine yerleştirilir. Böylece, moment dört SMSM için eşit olarak dağıtılır ve her bir SMSM bağımsız olarak kontrol edilebilir. Ayrıca bu durum, mekanik aktarma organlarının kullanımını ortadan kaldırarak aracın kütlesini azaltır ve bakım gereksinimlerini de ortadan kaldırır. Tasarlanan EDS'de gerçek zamanlı bir güç yönetimi sağlanmış ve dört tekerlek için farklı moment referansları üretilmiştir. Böylece, farklı direksiyon açıları ve araç hızı için her tekerleğin hızı ayarlanarak optimal bir EDS tasarımı gerçekleştirilmiştir. EDS tasarlandıktan sonra, dört tekerlektten tahrikli EA'nın Matlab/Simulink'te dinamik modeli oluşturulmuştur. Aracın boylamsal ve yanal kuvvetleri ile tekerleklerden oluşan tüm araç dinamikleri sisteme eklenmiştir. Böylece dört adet SMSM, sürücüler ve araç dinamiklerinden oluşan gerçek bir sistem oluşturulmuştur. Bu modelde gerçek sistemin tekerlek hız farkları, EDS'den alınan referans hız farklarına eşit olacak şekilde bir PID kontrolör tasarlanmıştır. Tasarlanan bu kontrol yöntemi, literatürdeki karmaşık yöntemlerin aksine daha basit bir yaklaşım sunmakta ve işlem kolaylığı sağlamaktadır. Böylece farklı direksiyon açıları için EA'nın yol eğrisi çizdirilerek eğimli yolda sistemin stabilitesi ve güvenliği sağlanmış, moment ve güç dört tekerleğe dengeli olarak dağıtılmıştır.

Dinamik model tasarımından sonra tekerlek hızları, direksiyon açısına ve aracın hız bilgilerine bağlı olarak CoDeSys V2.3 yazılım paketi kullanılarak hesaplanmıştır. Bu çalışmada, literatürde olmayan ve pek çok avantajı olan endüstriyel bir ARM işlemci kullanılmıştır. Kullanılan tüm sensörler, işlemci ve dört SMSM sürücüsü CAN-Bus haberleşme protokolü ile haberleştirilmiştir. Bu protokol, hızlı iletişim, mesaj önceliği, daha az kablo kullanımı ve güvenilir olması gibi önemli özelliklerinden dolayı özellikle otomobil endüstrisinde yaygın şekilde kullanılmaktadır. Bu yüzden bu çalışmada da tercih edilmiştir. Daha sonra, CoDeSys tarafından hesaplanan hız değerleri CAN-Bus üzerinden dört SMSM sürücüsüne gönderilmiş ve tüm sistem 1 Mbit/s olan en yüksek veri hızında haberleştirilmiştir. Bu çok hızlı iletişim ve farklı

bir endüstriyel işlemci kullanılması bu çalışmayı endüstriyel hale getirmekte ve literatürdeki diğer çalışmalardan ayırmaktadır. Son olarak, deney düzeneği kurulmuş ve takometre kullanılarak deneysel sonuçlar alınmıştır. Elde edilen simülasyon sonuçları, deneysel sonuçlarla karşılaştırılarak tasarlanan EDS doğrulanmıştır.

Bu tez çalışmasının ikinci kısmında ise, Raspberry Pi ve FPGA gibi gömülü sistem platformları kullanılarak EA'nın güvenli sürüşü üzerine çalışmalar gerçekleştirilmiştir. Bu amaçla yüz ve göz tanıma, Canny kenar çıkarma yöntemi, nesne tanıma, yol çizgilerinin takibi, viraj yönü ve açısının tespiti, risk hesabı, hareket algılama gibi çeşitli görüntü işleme uygulamaları yapılmıştır. Raspberry Pi'ye bağlı bir kameradan alınan görüntülerle gerçek zamanlı görüntü işleme gerçekleştirilmiştir. Hareket algılama yazılımında, kameranın önüne bir nesne geldiğinde bunu algılayarak nesne işaretleme işlemi yapılmış ve ekrana engel tespit edildiğiyle ilgili uyarı verilmiştir. Daha sonra, Raspberry Pi'nin genel amaçlı giriş/çıkış portlarına FPGA bağlanmıştır. Kameranın önüne engel geldiği anda FPGA'nın çıkışından HY-TTC60 işlemcisine bir sinyal gönderilmiş ve bu bilgi CAN-Bus hattı üzerinden motorlara iletilmiştir. Burada amaç, araç yolda giderken önüne bir engel çıktığında motorları yavaşlatarak ya da durdurarak güvenli sürüşü sağlamaktır. Engel olmadığı durumda SMSM'ler tasarlanan EDS'ye göre istenilen devirde dönmeye devam etmektedir. Sistemde, tüm işlemciler, motor sürücüler ve sensörler CAN-Bus hattı üzerinden 1 Mbit/s veri hızı ile haberleştirilmiştir. Aynı zamanda CAN hattında hangi verilerin önce iletilmesini istiyorsak, hatta bilgiler karışmadan istenilen ünitelerin öncelikli veri göndermesi sağlanmıştır. Bunun yanı sıra, eğimli yollar için viraj yönü ve açısını hesaplamaya yönelik bir çalışma gerçekleştirilmiştir. Bu çalışmayla araca etki eden merkezkaç ve sürtünme kuvvetleri hesaplanarak EA'nın hızına ve viraj yönüne bağlı olarak risk hesabı yapılmıştır. Riskli durum oluştuğunda sürücüye uyarı veren bir sistem tasarlanmış ve otonom sürüşün önemli bir basamağını oluşturan bir SDS geliştirilmiştir. Böylece bu tez çalışmasıyla, tüm üniteler arasında oldukça hızlı, sağlam ve güvenli bir haberleşme sağlanarak, engel geldiği anda motorlara tepki veren ve çalışmalarını durduran güvenli bir sistem tasarlanmıştır ve EA'nın güvenli sürüşü üzerine çalışmalar gerçekleştirilmiştir. Ayrıca elektrik makinelerinde, haberleşme sistemlerinin ve görüntü işleme uygulamalarının kullanılması, bu çalışmayı disiplinler arası bir çalışma haline getirerek özgünlük katmıştır.

ÖNERİLER

Gelecekte, bu çalışmada gerçekleştirilen EA'nın dinamik modeline kameradan alınan görüntüler eklenerek gerçek sistemden alınan geri beslemeye göre bir kontrol yöntemi önerilmektedir. Örneğin, EA belirli bir hızda giderken aracı durdurduğumuzda motorlardan alınan hız değerleri sıfır olmasına rağmen gerçek sistemde tekerlekler kayabilir. Bu durum güvenli sürüşü engeller. Bu nedenle kamera görüntüleri kullanılarak geri besleme alınabilir ve EA'nın güvenli sürüşü üzerine uygulamalar gerçekleştirilebilir.

Günümüzde otonom araçlar üzerine yapılan çalışmalar önem kazanırken, teknolojinin gelişmesiyle birlikte bu araçlardan beklenen özellikler de artmaya başlamıştır. Bu özellikler, sadece kendi kendine güvenli şekilde hareket etmesi değil, aynı zamanda insan davranışlarını da algılayabilecek, akıllı iletişim teknolojilerinin araçlarda kullanılmasıdır. Araç içinde kullanılan çeşitli sensörlerle sürücüye destek veren sistemler tam güvenli sürüş için yetersiz olabilmektedir. Bunun nedeni trafikte pek çok çevresel faktörün yer alması ve diğer araçlardan kaynaklanan hataların da kazalara önemli ölçüde neden olmasıdır. Bunları göz önüne almak için Bağlantılı Araçlar (Connected Vehicles) kavramı ortaya çıkmıştır. Burada amaç, aracın sürüş esnasında diğer araçlarla ve çevreyle haberleşmesini sağlayarak kaza risklerini önlemek ve güvenli sürüşü sağlamaktır. Bu yüzden bu alanda yapılacak çalışmalar önem kazanmıştır.

Bağlantılı Araçlar, farklı iletişim çeşitlerinden oluşur. Bunlardan biri araç-çevre haberleşmesidir. Bu haberleşmede araçlar, trafik ışıkları, şerit çizgileri, trafik levhaları gibi yollardaki çevre birimleriyle sürekli iletişim halindedirler. Bu veriler akıllı ulaşım sistemleri teknolojisi için anlık trafik bilgisi, yol durumu, kaza bilgisi, park bilgisi sağlayabilir. Böylece bu verilere göre yolların hız limitleri, trafik ışıklarının süresi gibi parametreler anlık olarak değiştirilerek ayarlanabilmektedir. Diğer bir haberleşme olan araç-arac haberleşmesinde ise, araçlar trafikte iken bir haberleşme ağı üzerinden anlık hız ve konum verilerini birbirlerine aktarabilirler. Dolayısıyla herhangi bir kaza anında sürücüye uyarı verebilir ve gereken tedbirler alınarak güvenli sürüşü sağlamaya yönelik çalışmalar gerçekleştirilebilir. Böylece gelişen bu teknolojilerle gelecekte, ileri sürücü destek sistemleri iletişim teknolojileriyle birleştirilerek tam güvenli ve verimli sürüşü sağlamaya yönelik çalışmaların yapılması önerilebilmektedir.

KAYNAKLAR

- [1] Y.E. Zhao, J.W. Zhang, X.Q. Guan, Modeling and simulation of electronic differential system for an electric vehicle with two-motor-wheel drive, in: IEEE Intell. Veh. Symp. Proc., 2009: pp. 1209–1214. <https://doi.org/10.1109/IVS.2009.5164454>.
- [2] S. Gair, A. Cruden, J. McDonald, B. Hredzak, Electronic differential with sliding mode controller for a direct wheel drive electric vehicle, in: Proc. IEEE Int. Conf. Mechatronics 2004, ICM'04, 2004: pp. 98–103. <https://doi.org/10.1109/icmech.2004.1364419>.
- [3] Y. Zhou, S. Li, X. Zhou, Z. Fang, The control strategy of electronic differential for EV with four in-wheel motors, in: 2010 Chinese Control Decis. Conf. CCDC 2010, 2010: pp. 4190–4195. <https://doi.org/10.1109/CCDC.2010.5498381>.
- [4] L. Zhai, S. Dong, Electronic differential speed steering control for four in-wheel motors independent drive vehicle, in: Proc. World Congr. Intell. Control Autom., 2011: pp. 780–783. <https://doi.org/10.1109/WCICA.2011.5970621>.
- [5] A. Draou, Electronic differential speed control for two in-wheels motors drive vehicle, in: Int. Conf. Power Eng. Energy Electr. Drives, 2013: pp. 764–769.
- [6] H. Kahveci, H.I. Okumus, M. Ekici, An electronic differential system using fuzzy logic speed controlled in-wheel brushless DC motors, in: Int. Conf. Power Eng. Energy Electr. Drives, 2013: pp. 881–885. <https://doi.org/10.1109/PowerEng.2013.6635726>.
- [7] R. Wan, G. Liu, D. Zhang, W. Gong, A fault-tolerant electronic differential system of electric vehicles, in: 2013 9th IEEE Veh. Power Propuls. Conf. IEEE VPPC 2013, IEEE Computer Society, 2013: pp. 6–10. <https://doi.org/10.1109/VPPC.2013.6671654>.
- [8] A. Bahri, A. Gupta, F. Ji, Performance enhancement of electronic differential in electric vehicles using a novel wavelet controller, in: 2014 IEEE Int. Conf. Veh. Electron. Safety, ICVES 2014, Institute of Electrical and Electronics Engineers Inc., 2014: pp. 152–157. <https://doi.org/10.1109/ICVES.2014.7063740>.
- [9] S. Sharma, R. Pegu, P. Barman, Electronic differential for electric vehicle with single wheel reference, in: 2015 1st Conf. Power, Dielectr. Energy Manag. NERIST, ICPDEN 2015, Institute of Electrical and Electronics Engineers Inc., 2015. <https://doi.org/10.1109/ICPDEN.2015.7084489>.
- [10] W. Khan-Ngern, W. Keyoonwong, N. Chatsiriwech, P. Sangnopparat, P. Mattayaboon, P. Worawalai, Embedded Electronic Differential System on Two Brushless DC Motor Drives for Electric Vehicle Steering Control, in: 2018 Int. Conf. Embed. Syst. Intell. Technol. Int. Conf. Inf. Commun. Technol. Embed. Syst. ICESIT-ICICTES 2018, Institute of Electrical and Electronics Engineers Inc., 2018. <https://doi.org/10.1109/ICESIT-ICICTES.2018.8442060>.
- [11] K. Hartani, M. Bourahla, Y. Miloud, M. Sekour, Electronic Differential with Direct Torque Fuzzy Control for Vehicle Propulsion System, Turkish J. Electr. Eng. Comput. Sci. 17 (2009) 21–38. <https://doi.org/10.3906/elk-0801-1>.
- [12] E. Ozkop, I.H. Altas, H.I. Okumus, A.M. Sharaf, A fuzzy logic sliding mode controlled electronic differential for a direct wheel drive EV, Int. J. Electron. 102 (2015) 1919–1942. <https://doi.org/10.1080/00207217.2015.1010183>.
- [13] D. Yin, D. Shan, J.S. Hu, A study on the control performance of electronic differential system for four-wheel drive electric vehicles, Appl. Sci. 7 (2017) 74. <https://doi.org/10.3390/app7010074>.
- [14] Y.E. Zhao, J. Zhang, Modelling and simulation of the electronic differential system for an electric vehicle with two-motor-wheel drive, Int. J. Veh. Syst. Model. Test. 4 (2009) 117–131. <https://doi.org/10.1504/IJVSMT.2009.029180>.
- [15] K. Hartani, Y. Miloudi, A. Miloudi, Electric Vehicle Stability with Rear Electronic Differential Traction, in: Int. Symp. Environ. Friendly Energies Electr. Appl., 2010: pp. 1–5.
- [16] Y.E. Zhao, J.W. Zhang, Study on electronic differential control system of independent in-wheel motor drive electric vehicle, J. Syst. Simul. 18 (2008) 4767–4771.
- [17] Y.P. Yang, X.Y. Xing, Design of electric differential system for an electric vehicle with dual wheel motors, in: Proc. IEEE Conf. Decis. Control, 2008: pp. 4414–4419. <https://doi.org/10.1109/CDC.2008.4739021>.
- [18] C. Fu, R. Hoseinnezhad, A. Bab-Hadiashar, R.N. Jazar, S. Watkins, Electronic differential for high-performance electric vehicles with independent driving motors, Int. J. Electr. Hybrid Veh. 6

- (2014) 108–132. <https://doi.org/10.1504/IJEHV.2014.065720>.
- [19] B. Tabbache, A. Kheloui, M.E.H. Benbouzid, An adaptive electric differential for electric vehicles motion stabilization, *IEEE Trans. Veh. Technol.* 60 (2011) 104–110.
 - [20] A. UYSAL, E. SOYLU, Embedded System Design and Implementation of an Intelligent Electronic Differential System for Electric Vehicles, *Int. J. Adv. Comput. Sci. Appl.* 8 (2017) 129–134. <https://doi.org/10.14569/ijacsa.2017.080918>.
 - [21] Y.C. Chang, C.Y. Lu, W.C. Chen, J.S. Hu, J.F. Jiang, T.K. Chang, H.Y. Wei, A new control strategy for an electronic differential system for urban electric vehicles, *Inventions.* 3 (2018). <https://doi.org/10.3390/inventions3010019>.
 - [22] X. Zhao, Q. Yu, M. Yu, Z. Tang, Research on an equal power allocation electronic differential system for electric vehicle with dual-wheeled-motor front drive based on a wavelet controller, *Adv. Mech. Eng.* 10 (2018). <https://doi.org/10.1177/1687814018760039>.
 - [23] F.J.L. Daya, P. Sanjeevikumar, F. Blaabjerg, P.W. Wheeler, J. Olorunfemi Ojo, A.H. Ertas, Analysis of Wavelet Controller for Robustness in Electronic Differential of Electric Vehicles: An Investigation and Numerical Developments, *Electr. Power Components Syst.* 44 (2016) 763–773. <https://doi.org/10.1080/15325008.2015.1131771>.
 - [24] C. Zhang, Y. Yuan, N. Wang, C. Li, Study on Electronic Differential Control System Used for In-wheel Motor Driven EV, in: *Int. Forum Energy, Environ. Sustain. Dev.*, Atlantis Press, 2016: pp. 809–815. <https://doi.org/10.2991/ifeesd-16.2016.148>.
 - [25] L. Clavero-Ordóñez, J. Fernández-Ramos, A. Gago-Calderón, Electronic differential system for Light Electric Vehicles with two inwheel motors, *Renew. Energy Power Qual. J.* 1 (2018) 325–329. <https://doi.org/10.24084/repqj16.300>.
 - [26] H. Tian, W. Zhu, S. Wang, Adaptive electronic differential control of vehicle by torque balance, *Mob. Networks Appl.* 25 (2020) 1604–1610. <https://doi.org/10.1007/s11036-019-01365-w>.
 - [27] H.T. Al-Fiky, M.S. Asfoor, M.I. Yacoub, A.M. Sharaf, Electronic Differential Optimization for Electric Vehicle Full Model for In-Wheel Permanent Magnet Brushless DC Motors, in: *2019 IEEE 7th Int. Conf. Control. Mechatronics Autom. ICCMA 2019*, Institute of Electrical and Electronics Engineers Inc., 2019: pp. 15–20. <https://doi.org/10.1109/ICCMA46720.2019.8988695>.
 - [28] C. Lee, J.H. Moon, Robust lane detection and tracking for real-time applications, *IEEE Trans. Intell. Transp. Syst.* 19 (2018) 4043–4048. <https://doi.org/10.1109/TITS.2018.2791572>.
 - [29] D. Rojdestvenskiy, M. Cvetković, P. Bouchner, Real-time driver advisory system for improving energy economy based on advance driver assistant systems interface, in: *2018 Smart Cities Symp. Prague, SCSP 2018*, Institute of Electrical and Electronics Engineers Inc., 2018: pp. 1–6. <https://doi.org/10.1109/SCSP.2018.8402655>.
 - [30] A. Gupta, A. Choudhary, A Framework for Camera-Based Real-Time Lane and Road Surface Marking Detection and Recognition, *IEEE Trans. Intell. Veh.* 3 (2018) 476–485. <https://doi.org/10.1109/TIV.2018.2873902>.
 - [31] V.D. Nguyen, D.T. Tran, J.Y. Byun, J.W. Jeon, Real-Time Vehicle Detection Using an Effective Region Proposal-Based Depth and 3-Channel Pattern, *IEEE Trans. Intell. Transp. Syst.* 20 (2019) 3634–3646. <https://doi.org/10.1109/TITS.2018.2877200>.
 - [32] C. Bao, C. Chen, H. Kui, X. Wang, Safe driving at traffic lights: An image recognition based approach, in: *Proc. - IEEE Int. Conf. Mob. Data Manag.*, Institute of Electrical and Electronics Engineers Inc., 2019: pp. 112–117. <https://doi.org/10.1109/MDM.2019.00-67>.
 - [33] A. Nair, S. Mansoori, R. Moghe, P. Shah, K. Talele, Driver assistant system using haar cascade and Convolution Neural Networks(CNN), in: *Proc. Int. Conf. Trends Electron. Informatics, ICOEI 2019*, Institute of Electrical and Electronics Engineers Inc., 2019: pp. 1261–1263. <https://doi.org/10.1109/ICOEI.2019.8862779>.
 - [34] A. Zaarane, I. Slimani, W. Al Okaishi, I. Atouf, A. Hamdoun, An automated night-time vehicle detection system for driving assistance based on cross-correlation, in: *Proc. - 2019 4th Int. Conf. Syst. Collab. Big Data, Internet Things Secur. SysCoBioTS 2019*, Institute of Electrical and Electronics Engineers Inc., 2019. <https://doi.org/10.1109/SysCoBioTS48768.2019.9028038>.
 - [35] S. Coicheci, I. Filip, Self-driving vehicles: Current status of development and technical challenges to overcome, in: *SACI 2020 - IEEE 14th Int. Symp. Appl. Comput. Intell. Informatics, Proc.*, Institute of Electrical and Electronics Engineers Inc., 2020: pp. 255–260. <https://doi.org/10.1109/SACI49304.2020.9118809>.

- [36] E. Bertolazzi, F. Biral, M. Da Lio, Future advanced driver assistance systems based on optimal control: The influence of “risk functions” on overall system behavior and on prediction of dangerous situations, in: *IEEE Intell. Veh. Symp. Proc.*, 2004: pp. 386–391. <https://doi.org/10.1109/ivs.2004.1336414>.
- [37] H.H. Chiang, Y.L. Chen, B.F. Wu, T.T. Lee, Embedded driver-assistance system using multiple sensors for safe overtaking maneuver, *IEEE Syst. J.* 8 (2014) 681–698. <https://doi.org/10.1109/JSYST.2012.2212636>.
- [38] J.S. Hu, X.C. Lin, F.R. Hu, Direct yaw-moment control for in-wheel motor electric vehicles, in: *2014 IEEE/SICE Int. Symp. Syst. Integr. SII 2014*, Institute of Electrical and Electronics Engineers Inc., 2014: pp. 475–479. <https://doi.org/10.1109/SII.2014.7028085>.
- [39] M.W. Chong, Y.P. Yang, Vehicle stability control with electric propulsion failure for an electric vehicle driven by multiple motors, in: *ICNSC 2015 - 2015 IEEE 12th Int. Conf. Networking, Sens. Control*, Institute of Electrical and Electronics Engineers Inc., 2015: pp. 515–520. <https://doi.org/10.1109/ICNSC.2015.7116090>.
- [40] S. Zhao, T. Zhang, N. Wu, H. Ogai, S. Tateno, Vehicle to vehicle communication and platooning for EV with wireless sensor network, in: *2015 54th Annu. Conf. Soc. Instrum. Control Eng. Japan, SICE 2015*, Institute of Electrical and Electronics Engineers Inc., 2015: pp. 1435–1440. <https://doi.org/10.1109/SICE.2015.7285493>.
- [41] E. Kural, S. Jones, A.F. Parrilla, A. Grauers, Traffic light assistant system for optimized energy consumption in an electric vehicle, in: *2014 Int. Conf. Connect. Veh. Expo, ICCVE 2014 - Proc.*, Institute of Electrical and Electronics Engineers Inc., 2014: pp. 604–611. <https://doi.org/10.1109/ICCVE.2014.7297619>.
- [42] B. Soner, A. Içke, A.C. Mert, U. Başaran, S.T. Impram, I. Şahin, Development of an electronic control unit for PMSM drives in automotive applications, in: *ELECO 2015 - 9th Int. Conf. Electr. Electron. Eng.*, Institute of Electrical and Electronics Engineers Inc., 2016: pp. 680–684. <https://doi.org/10.1109/ELECO.2015.7394493>.
- [43] S.K. Kommuri, M. Defoort, H.R. Karimi, K.C. Veluvolu, A Robust Observer-Based Sensor Fault-Tolerant Control for PMSM in Electric Vehicles, *IEEE Trans. Ind. Electron.* 63 (2016) 7671–7681. <https://doi.org/10.1109/TIE.2016.2590993>.
- [44] C. Bian, G. Yin, M. Hu, N. Zhang, N. Chen, Fault Tolerant Control Design for All Wheels Independent Drive and Steering Electric Vehicle with Singular Effective Drive Wheel, in: *2016 IEEE Veh. Power Propuls. Conf. VPPC 2016 - Proc.*, Institute of Electrical and Electronics Engineers Inc., 2016. <https://doi.org/10.1109/VPPC.2016.7791688>.
- [45] M.O. Shin, S.W. Seo, Vehicle recognition using common appearance captured by 3D LIDAR and monocular camera, in: *IEEE Intell. Veh. Symp. Proc.*, Institute of Electrical and Electronics Engineers Inc., 2017: pp. 399–404. <https://doi.org/10.1109/IVS.2017.7995751>.
- [46] H.A. Attia, S. Ismail, H.Y. Ali, Design of enhanced vehicle safety system based on the evaluation of driving errors, in: *2017 Int. Conf. Electr. Comput. Technol. Appl. ICECTA 2017*, Institute of Electrical and Electronics Engineers Inc., 2017: pp. 1–4.
- [47] J. Han, A. Sciarretta, L.L. Ojeda, G. De Nunzio, L. Thibault, Safe-and Eco-Driving Control for Connected and Automated Electric Vehicles Using Analytical State-Constrained Optimal Solution, *IEEE Trans. Intell. Veh.* 3 (2018) 163–172. <https://doi.org/10.1109/TIV.2018.2804162>.
- [48] S. Wei, Y. Zou, X. Zhang, T. Zhang, X. Li, An Integrated Longitudinal and Lateral Vehicle Following Control System with Radar and Vehicle-to-Vehicle Communication, *IEEE Trans. Veh. Technol.* 68 (2019) 1116–1127. <https://doi.org/10.1109/TVT.2018.2890418>.
- [49] A.K. Madhusudhanan, X. Na, Effect of a traffic speed based cruise control on an electric vehicles performance and an energy consumption model of an electric vehicle, *IEEE/CAA J. Autom. Sin.* 7 (2020) 386–394. <https://doi.org/10.1109/JAS.2020.1003030>.
- [50] M.S. Mökükcü, Electric vehicle powertrain design and implementation - Tez Arşivi, Yüksek Lisans Tezi, İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, 2014.
- [51] C.C. Chan, K.T. Chau, An overview of power electronics in electric vehicles, *IEEE Trans. Ind. Electron.* 44 (1997) 3–13. <https://doi.org/10.1109/41.557493>.
- [52] K. Tural, Elektronik Diferansiyelli Elektrikli Araçların Modellenmesi Ve Kontrolü, Fen Bilimleri Enstitüsü, 2014. <https://polen.itu.edu.tr/handle/11527/2939> (accessed June 28, 2021).
- [53] M. Ehsani, Y. Gao, S. Longo, K. Ebrahimi, Modern Electric, Hybrid Electric, and Fuel Cell

- Vehicles - 3rd Edition, CRC Press, 2018. <https://www.routledge.com/Modern-Electric-Hybrid-Electric-and-Fuel-Cell-Vehicles/Ehsani-Gao-Longo-Ebrahimi/p/book/9781498761772>.
- [54] www.wikipedia.org.tr Elektrikli Otomobil,(n.d.). https://tr.wikipedia.org/wiki/Elektrikli_otomobil.
- [55] G. Varol, Elektrikli araçlarda enerji depolama ve geri kazanımlı frenleme sistemlerinin incelenmesi, Yüksek lisans tezi, Kocaeli Üniversitesi, Fen Bilimleri Enstitüsü, 2019.
- [56] EV A to Z Encyclopedia - 1: Understanding EV Components, (n.d.). <https://news.hyundaimotorgroup.com/Article/Understanding-EV-Components>.
- [57] M. Rigan, Elektrikli araçlarda batarya yönetim sistemi tasarımı, Yüksek lisans tezi, Kahramanmaraş Sütçü İmam Üniversitesi, Fen Bilimleri Enstitüsü, 2020.
- [58] K. Yoldaş, An improved speed controller for PMSM drive with unbalanced load using load torque observer, Yüksek lisans tezi, Özyeğin Üniversitesi, Fen Bilimleri Enstitüsü, 2019.
- [59] P. Pillay, R. Krishnan, Modeling, Simulation, and Analysis of Permanent-Magnet Motor Drives, Part I: The Permanent-Magnet Synchronous Motor Drive, IEEE Trans. Ind. Appl. 25 (1989) 265–273. <https://doi.org/10.1109/28.25541>.
- [60] O. Aydogmus, E. Deniz, Design and implementation of two-phase permanent magnet synchronous motor fed by a matrix converter, IET Power Electron. 10 (2017) 1054–1060.
- [61] A. Yıldız, Sürekli mıknatıslı senkron motorun FPGA ile konum tahmin yöntemi ile alan yönlendirmeli kontrollü sürücü tasarımı ve gerçekleştirilmesi, Yüksek lisans tezi, İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, 2019.
- [62] G. Altıntaş, Sürekli Mıknatıslı Senkron Motorda Algılayıcısız Kontrol Yöntemini Geliştirmeye Katkıları, Yüksek lisans tezi, İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, 2017.
- [63] E. Çelik, Sürekli mıknatıslı senkron motor hız denetiminin genetik tabanlı bulanık mantık denetleyici ile arayüz destekli simülasyonu, Yüksek lisans tezi, Gazi Üniversitesi, Fen Bilimleri Enstitüsü, 2012.
- [64] B. Gördük, Design and implementation of sensorless vector controlled drive for PMSMs, Yüksek lisans tezi, İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, 2019.
- [65] İ. Aışkan, A. Sel, Sürekli Mıknatıslı Senkron Motorların Hız Denetiminde Doğal Frekans ve Bulanık Mantık, Karaelmas Fen ve Müh. Derg. 6 (2016) 423–431. <http://fd.beun.edu.tr> (accessed January 8, 2022).
- [66] H. Aydoğan, M. Acaroğlu, Günümüz İçten Yanmalı Motorlarında Kullanılan Kontrol ve Veri İletim Sistemleri ve Gelişmeler, Taşıt Teknol. Elektron. Derg. 1 (2009) 51–62.
- [67] Milli Eğitim Bakanlığı, Motorlu Araçlar Teknolojisi, Araç Yönetim Sistemleri, 2012.
- [68] R. Bayır, E. Soylu, Endüstriyel İletişim Sistemleri Ders Notu, Karabük, 2014.
- [69] N. Çayan, 3. Ulusal Hidrolik Pnömatik Kongresi ve Sergisi, in: Pnömatikte Alansal Veriyolu Uygulamaları, n.d.: pp. 453–475.
- [70] T.. Deepika, P. Bhagya, Dual Redundancy CAN-bus Controller based on FPGA, Int. J. Sci. Res. Publ. 4 (2014) 1–4.
- [71] A.R.. Khan, Thakare A.P., FPGA Based Design & Implementation of Serial Data Transmission Controller, Int. J. Eng. Sci. Technol. 2 (2010) 5526–5533.
- [72] K. Linek, A. Hernandez, A. Wyglinski, Encoding and Physical Study of the CANBus Sensor Network, 2017.
- [73] C. Fil, İçten yanmalı motorların j1939 canbus’tan alınan bilgilerinin ve hata mesajlarının işlenmesi ve mobil uygulamada görüntülenmesi, Yüksek lisans tezi, İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, 2019.
- [74] Local Interconnect Network - Wikipedia, (n.d.). https://en.wikipedia.org/wiki/Local_Interconnect_Network (accessed June 27, 2021).
- [75] LIN, Bus Explained - A Simple Intro (2021), (n.d.). <https://www.csselectronics.com/screen/page/lin-bus-protocol-intro-basics/language/en> (accessed June 27, 2021).
- [76] H. Kohler, MOST150-The Next Generation Automotive Infotainment Backbone, n.d.
- [77] R. Özalp, Çip üzerinde sistem mimarili fpga kullanarak gerçek zamanlı görüntü işleme algoritmalarının gerçekleştirilmesi, Yüksek lisans tezi, Fırat Üniversitesi, Fen Bilimleri Enstitüsü, 2018.
- [78] YUV - Vikipedi, (n.d.). <https://tr.wikipedia.org/wiki/YUV> (accessed June 27, 2021).

- [79] OpenCv Görüntü İşleme ve Renk Uzayı, (n.d.). <https://medium.com/@hafizegungor/opencv-görüntü-işleme-ve-renk-uzayı-1a76562ff715> (accessed June 27, 2021).
- [80] Özçelik- Mehmet Fatih, Temel Görüntü İşleme Algoritmalarının Gerçek Zamanlı Olarak FPGA ile Gerçeklenmesi, Yüksek lisans tezi, Kocaeli Üniversitesi, Fen Bilimleri Enstitüsü, 2015.
- [81] R. Kızılkaya, Implementation of image processing algorithms on FPGA demonstration board, Fen Bilimleri Enstitüsü, 2012.
- [82] Konvolüsyon - Vikipedi, (n.d.). <https://tr.wikipedia.org/wiki/Konvolüsyon> (accessed June 27, 2021).
- [83] S. Fırat, HSA'ların FPGA Üzerinde Gerçeklenmesi ve Görüntü İşleme Uygulamaları, Yüksek lisans tezi, Yıldız Teknik Üniversitesi, Fen Bilimleri Enstitüsü, 2011.
- [84] Online Kurslar - Programınıza Uygun Şekilde Her Şeyi Öğrenin | Udemy, (n.d.). <https://www.udemy.com/tr/> (accessed July 5, 2021).
- [85] M.M. Çil, Temel Görüntü İşleme Algoritmalarının FPGA Üzerinde Gerçeklenmesi, Yüksek lisans tezi, İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, 2015.
- [86] M.F. Özçelik, Görüntü İşleme Algoritmalarının FPGA Üzerinde Gerçeklenmesi, Yüksek lisans tezi, Gazi Üniversitesi, Bilişim Enstitüsü, 2012.
- [87] E. Onat, Sobel, Robert, Prewitt ve Laplace Filtreleri Kullanarak Gerçek Zamanlı Video Sinyallerinin İşlenmesinin FPGAda Gerçeklenmesi, in: 2017 25th Signal Process. Commun. Appl. Conf. SIU 2017, Institute of Electrical and Electronics Engineers Inc., 2017. <https://doi.org/10.1109/SIU.2017.7960586>.
- [88] W. Rong, Z. Li, W. Zhang, L. Sun, An improved Canny edge detection algorithm, 2014 IEEE Int. Conf. Mechatronics Autom. IEEE ICMA 2014. (2014) 577–582.
- [89] TTControl HYDAC International, n.d. <https://www.hydac.com.au/ourcompanies/ttcontrol/> (accessed July 5, 2021).
- [90] HY-TTC 50_60_90 User Manual V2.2 , n.d.
- [91] HYDAC International Universal Mobile Controller HY-TTC 60 Data Sheet, n.d. <https://www.hydac.com/de-en/products/control-technology-connectivity/mobile-controller-displays-connectivity/controller/hy-ttc-60.html> (accessed July 5, 2021).
- [92] PCAN-USB User Manual V2.1.3, (n.d.). <https://www.google.com/search?q=PCAN-USB+User+Manual+V2.1.3&oq=PCAN-USB+User+Manual+V2.1.3&aqs=chrome..69j0j9&sourceid=chrome&ie=UTF-8>.
- [93] User Manual for PLC Programming with CoDeSys 2.3, n.d.
- [94] AMK RACING KIT 4 wheel drive “Formula Student Electric,” 2017. www.amk-group.com (accessed July 5, 2021).
- [95] GPIO - Raspberry Pi Documentation, (n.d.). <https://www.raspberrypi.org/documentation/usage/gpio/> (accessed June 28, 2021).
- [96] Basys3™ FPGA Board Reference Manual, n.d. www.digilentinc.com (accessed June 28, 2021).
- [97] Ö. Karaduman, Ortam etkileşimli güvenli sürüş destek sistemlerinin modellenmesi, Doktora tezi, Fırat Üniversitesi, Fen Bilimleri Enstitüsü, 2015.
- [98] Degree of Curvature, (2005). <http://du.edu/~jcalvert/railway/degcurv.htm> (accessed December 28, 2021).
- [99] GitHub - windowsub0406/Advanced-lane-finding: Advanced lane finding, (n.d.). <https://github.com/windowsub0406/Advanced-lane-finding> (accessed December 28, 2021).
- [100] C. Yuan, H. Chen, J. Liu, D. Zhu, Y. Xu, Robust lane detection for complicated road environment based on normal map, IEEE Access. 6 (2018) 49679–49689.
- [101] İmage segmentasyon (Görüntü Bölütleme), (n.d.). <https://docplayer.biz.tr/12381442-Image-segmentasyon-goruntu-bolutleme.html>.
- [102] H. Mobahi, S.R. Rao, A.Y. Yang, S.S. Sastry, Y. Ma, Segmentation of Natural Images by Texture and Boundary Compression, Int. J. Comput. Vis. 95 (2011) 86–98. <https://doi.org/10.1007/s11263-011-0444-0>.

ÖZGEÇMİŞ

Merve YILDIRIM

[illegible]

AKADEMİK FAALİYETLER

Makaleler:

1. Yildirim, M., Catalbas, M. C., Gulten, A., & Kurum, H. (2016). Computation of the speed of four in-wheel motors of an electric vehicle using a radial basis neural network, *Engineering, Technology & Applied Science Research*, 6(6), 1288-1293.
2. Catalbas, M. C., Yildirim, M., Issever, D., Gulten, A., & Kurum, H. (2016). An Approach to Adaptive Structuring Element Size Selection for Feature Extraction, *International Journal of Computer Science and Information Security*, 14(5), 187.
3. Catalbas, M. C., Yildirim, M., Gulten, A., Kurum, H., & Dobrisek, S. (2016). Estimation of trajectory and location for mobile sound source, *International Journal of Advanced Computer Science and Applications*, 7(9), 237-241.
4. Yildirim, M. & Kurum, H. (2017). Influence of Stator Embrace on Torque of In-wheel Switched Reluctance Motor, *European Journal of Technique (EJT)*, 7(2), 78-84.

5. Yildirim, M., Kurum, H., Miljavec, D., & Corovic, S. (2018). Influence of Material and Geometrical Properties of Permanent Magnets on Cogging Torque of BLDC, *Engineering, Technology & Applied Science Research*, 8(2), 2656-2662.
6. Omaç, Z., Polat, M., Öksüztepe, E., Yıldırım, M., Yakut, O., Eren, H., ... & Kürüm, H. (2018). Design, analysis, and control of in-wheel switched reluctance motor for electric vehicles, *Springer Electrical Engineering*, 100(2), 865-876.

Kitap:

1. Omaç, Z., Polat, M., Kaya, M., Öksüztepe, E., Eren, H., Yıldırım, M., & Kürüm, H. (2018). Outer Rotor SRM Design for Electric Vehicle without Reducer via Speed-Up Evolutionary Algorithm, *IntechOpen New Trends in Electrical Vehicle Powertrains*.

Bildiriler:

1. Polat, M., Öksüztepe, E., Omaç, Z., Yıldırım, M., & Kürüm, H. (2013, November). Examination of radial force with Finite Element Method in Switched Reluctance Motor, *IEEE 8th International Conference on Electrical and Electronics Engineering (ELECO)*, 576-580.
2. Yıldırım, M., Polat, M., & Kürüm, H. (2014, September). A survey on comparison of electric motor types and drives used for electric vehicles, *IEEE 16th International Power Electronics and Motion Control Conference and Exposition*, 218-223.
3. Yıldırım, M., Polat, M., Öksüztepe, E., Omaç, Z., Yakut, O., Eren, H., ... & Kürüm, H. (2014, September). Designing in-wheel switched reluctance motor for electric vehicles. *IEEE 16th International Power Electronics and Motion Control Conference and Exposition*, 793-798.
4. Yıldırım, M., Öksüztepe, E., Tanyeri, B., & Kürüm, H. (2015, November). Electronic differential system for an electric vehicle with in-wheel motor, *IEEE 9th International Conference on Electrical and Electronics Engineering (ELECO)*, 1048-1052.
5. Yıldırım, M., Öksüztepe, E., Tanyeri, B., & Kürüm, H. (2016, February). Design of electronic differential system for an electric vehicle with in-wheel motor, *IEEE Power and Energy Conference at Illinois (PECI)*, 1-5.
6. Yıldırım, M., Catalbas, M. C., Gulten, A., & Kurum, H. (2016, June). Modelling and estimation parameters of electronic differential system for an electric vehicle using radial basis neural network, *IEEE 16th International Conference on Environment and Electrical Engineering (EEEIC)*, 1-5.
7. Catalbas, M. C., Yıldırım, M., Gulten, A., & Kurum, H. (2016, September). Optimal component selection for image segmentation via Parallel Analysis, *IEEE 8th International Conference on Intelligent Systems (IS)*, 499-502.
8. Catalbas, M. C., Yıldırım, M., Gulten, A., & Kurum, H. (2017, June). Estimation of optimal locations for electric vehicle charging stations, *IEEE International Conference on Environment and Electrical Engineering and IEEE Industrial and Commercial Power Systems Europe (EEEIC/I&CPS Europe)*, 1-4.
9. Yıldırım, M., & Kurum, H. (2018, August). Influence of Poles Embrace on In-Wheel Switched Reluctance Motor Design, *IEEE 18th International Power Electronics and Motion Control Conference (PEMC)*, 562-567.
10. Yaman, S., Yıldırım, M., Kamışlıoğlu, B., Erol, Y., & Kürüm, H. (2019, June). Image and Video Processing Applications Using Xilinx System Generator, *IEEE 7th International Symposium on Digital Forensics and Security (ISDFS)*, 1-5.

11. Dal, Ö., Yıldırım, M., & Kürüm, H. (2019, September). Optimization of permanent magnet synchronous motor design by using PSO, *IEEE 4th International Conference on Power Electronics and their Applications (ICPEA)*, 1-6.
12. Yıldırım, M., & Kurum, H. (2020, May). Electronic Differential System for an Electric Vehicle with Four In-wheel PMSM, *IEEE 91st Vehicular Technology Conference (VTC2020-Spring)*, 1-5.

Projeler:

1. Fırat Racing Formula Student Elektrikli Araç Yarış Projesi

Ödüller:

1. Mühendislik Fakültesi Elektrik-Elektronik Mühendisliği Bölümü 2020 yılı Araştırma Görevlileri Akademik Performans Bölüm Birincisi Ödülü

