



**BİLECİK
ŞEYH EDEBALI ÜNİVERSİTESİ**

**Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Ana Bilim Dalı**

**RAYLI SİSTEM TAM VE EŞ ZAMANLI
BENZETİM ORTAMININ YAZILIMSAL UYARLANMASI**

**Osman YİĞİT
Yüksek Lisans**

**Tez Danışmanı
Prof. Dr. Cihan KARAKUZU**

**BİLECİK, 2019
Ref. No: 10289885**



BİLECİK ŞEYH EDEBALI
ÜNİVERSİTESİ

**BİLECİK
ŞEYH EDEBALI ÜNİVERSİTESİ**

**Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Ana Bilim Dalı**

**RAYLI SİSTEM TAM VE EŞ ZAMANLI
BENZETİM ORTAMININ YAZILIMSAL UYARLANMASI**

**Osman YİĞİT
Yüksek Lisans**

**Tez Danışmanı
Prof. Dr. Cihan KARAKUZU**

BİLECİK, 2019



**BİLECİK
ŞEYH EDEBALI UNIVERSITY**

**Graduate School of Sciences
Department of Computer Engineering**

**COMPLETE AND SIMULTANEOUS SIMULATION
SOFTWARE ADAPTATION IN RAIL SYSTEM**

**Osman YİĞİT
Master's**

**Thesis Advisor
Prof. Dr. Cihan KARAKUZU**

BİLECİK, 2019



BİLECİK ŞEYH EDEBALI ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

YÜKSEK LİSANS
JÜRİ ONAY FORMU

Bilecik Şeyh Edebali Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulunun 24/07/2019 tarih ve 40-11 sayılı kararıyla oluşturulan jüri tarafından 21/08/2019 tarihinde tez savunma sınavı yapılan Osman YİĞİT'in "Raylı Sistem Tam ve Eş Zamanlı Benzetim Ortamının Yazılımsal Uyarlanması" başlıklı tez çalışması Bilgisayar Mühendisliği Ana Bilim Dalında YÜKSEK LİSANS tezi olarak oy birliği ile kabul edilmiştir.

JÜRİ

Prof. Dr. Cihan KARAKUZU
(TEZ DANIŞMANI) :

Prof. Dr. Nejat YUMUŞAK :
(JÜRİ BAŞKANI)

Doç. Dr. Uğur YÜZGEÇ:

ONAY

Bilecik Şeyh Edebali Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulunun
.../.../..... tarih ve/..... sayılı kararı.

İMZA/ MÜHÜR

TEŐEKKÖR

Tez alıřmamamın her ařamasında deęerli katkı ve eleřtiriyle yol gōsteren, sonsuz sabırla beni her zaman alıřmaya teřvik eden ve gŸven veren danıřmanım Sayın Prof. Dr. Cihan Karakuzu'ya en iten teřekkŸrlerimi sunarım. 1501- TŸBİTAK Sanayi AR-GE Projeleri Destekleme Programı kapsamında “3180351” numaralı proje lideri olarak gōrev aldıęım “EndŸstri 4.0 Kapsamında Raylı Sistemler Tam ve Eř Zamanlı SimŸlasyon ve Koordinasyon Merkezi Projesi” yŸrŸtŸcŸsŸ Durmazlar Makine AŐ'ye, proje yŸrŸtŸcŸsŸ Ertan Bahtiyar'a, proje ekibinde yer alan Salih İbrahim Akın, Gōkhan Kurt ve İbrahim Yōnel'e destekleri iin ok teřekkŸr ederim. Ayrıca bu tez alıřmasında her zaman yanımda olan ve her tŸrlŸ desteęi saęlayan sevgili eřim, hayat arkadařım, Nurseli Meral Yięit'e ve bana her daim yol gōsteren, sonsuz sabırla beni alıřmaya teřvik eden ve her tŸrlŸ desteęi saęlayan, babam Prof. Dr. Abdulvahap Yięit'e ok teřekkŸr ederim.

BEYANNAME

Bilecik Şeyh Edebali Üniversitesi Fen Bilimleri Enstitüsü Tez Yazım Kılavuzu'na uygun olarak hazırladığım bu tez çalışmada, tez içindeki tüm verileri akademik kurallar çerçevesinde elde ettiğimi, görsel ve yazılı tüm bilgi ve sonuçların akademik ve etik kurallara uygun olarak sunulduğunu, kullanılan verilerde herhangi bir tahrifat yapılmadığını, başkalarının eserlerinden yararlanılması durumunda ilgili eserlere bilimsel normlara uygun olarak atıfta bulunulduğunu, tezde yer alan verilerin bu Üniversite veya başka bir üniversitede herhangi bir tez çalışmada kullanılmadığını beyan ederim.

.../.../2019

Osman YİĞİT

RAYLI SİSTEM TAM VE EŞ ZAMANLI BENZETİM ORTAMININ YAZILIMSAL UYARLANMASI

ÖZET

Son yıllarda şehirlerde yaşayan nüfusun artmasıyla birlikte raylı sistem yatırımları bütün dünyada ve ülkemizde çok hızlı bir şekilde artmaktadır. Artan nüfusa bağlı olarak artan ihtiyaçların karşılanması için raylı sistem araç üretimlerindeki hız çok önemli bir faktör haline gelmiştir. Bu tez çalışmasında; raylı sistem üreticileri ve işletmeleri için, araç üretilmeden önce, aracın bire bir aynı kontrol paneli, araç kontrol ünitesi, giriş-çıkış modülleri, insan makine arayüzleri ve yazılımların benzetimlerinin yapılacağı ayrıca tramvay araçlarında kullanılan ekipmanların testlerinin de yapılabilceğı DurSim (Durma Simölätör) benzetim ortamı üzerinde çalışılmıştır. Bu benzetim ortamı ile yapılan testler sonucunda ekipmanların yeterliliğı ve kullanılabilirliğı belirlenebilecek olup, mevcut araç yazılımlarının testleri birebir araç üzerinde yapılmış gibi DurSim yardımıyla dinamik olarak test edilebilecektir. Bu tez çalışması sonucunda geliştirilen DurSim ile aracın üretiminden sonra ortaya çıkması muhtemel yazılımsal veya donanımsal eksiklikler ve hataların tespit edilmesi, tespit edilen hatalar ve eksikliklerin raylı sistem araç üretiminden önce giderilmesi ve bu sayede sıfır hatalı raylı sistem araç üretimi hedeflenmektedir. Ayrıca DurSim ile raylı sistem araç sürücülerinin eğitimlerinin verilmesi hedeflenmektedir, bu sayede sürücülerin kontrol paneli ve insan makine arayüzünün kullanımını öğrenmeleri sağlanacaktır. Böylece hiçbir risk faktörü olmadan sürücü eğitimlerinin güvenli, yetkin ve düşük maliyetli bir şekilde gerçekleştirilmesi sağlanmış olacaktır.

Anahtar Kelimeler: DurSim; Tam ve Eş Zamanlı Benzetim; Raylı Sistem Araç Simölätörü; Raylı Sistem Araç Yazılımı;

COMPLETE AND SIMULTANEOUS SIMULATION SOFTWARE

ADAPTATION IN RAIL SYSTEM

ABSTRACT

With the increase in the population living in cities in recent years, rail system investments have been increasing rapidly all over the world and in our country, and the speed of rail vehicle production has become a very important factor in order to meet the increasing needs due to the increasing population. In this thesis; DurSim (Durma Simulator) simulation environment that exactly the same control panel as that of the vehicle, vehicle control unit, input-output modules, human machine interfaces and software simulations of the vehicle will be simulated and tested before the vehicle is produced for the rail system manufacturers and enterprises, has been studied. As a result of the tests carried out with this simulation environment, the adequacy and usability of the equipment can be determined and the existing vehicle software can be tested dynamically with the help of DurSim. With the help of DurSim developed in this thesis, it is aimed to detect software and hardware deficiencies and errors that may come up after the production of the vehicle and to eliminate these errors before production and thus zero-defect production of rail system vehicles. In addition, it is aimed to provide trainings of the rail system vehicle drivers with DurSim so that the drivers will learn to use the control panel and human machine interface. This will ensure that driver trainings are conducted safely, competently and cost-effectively without any risk factors.

Key Words: DurSim; Complete and Simultaneous Simulation; Rail System Vehicle Simulator; Rail System Vehicle Software

İÇİNDEKİLER

	Sayfa No
TEŞEKKÜR	
BEYANNAME	
ÖZET.....	I
ABSTRACT	II
İÇİNDEKİLER	III
ŞEKİLLER DİZİNİ	IV
ÇİZELGELER DİZİNİ	V
SİMGELER ve KISALTMALAR DİZİNİ	VI
1. GİRİŞ	1
2. DURSİM TRAMVAYSİMÜLATÖR SİSTEMİ	11
2.1. DurSim Kurulumu	12
2.2. DurSim Kontrol Paneli Elemanları	13
3. DURSİM YAZILIM BİLEŞENLERİ	23
3.1. RailDriver Dinamik Bağlantı Kitaplığı	25
4. SONUÇ ve ÖNERİLER.....	37
KAYNAKLAR	40
EKLER.....	42
ÖZ GEÇMİŞ.....	

ŞEKİLLER DİZİNİ

	Sayfa No
Şekil 1.1.	Bursa’da kullanılan İpekböceği Tramvayının fotoğrafı.2
Şekil 1.2.	Bursa’da kullanılan HRS aracının fotoğrafı.3
Şekil 1.3.	Üsküdar-Çekmeköy hattında kullanılan metro aracının fotoğrafı.4
Şekil 1.4.	Beiyuan Guo ve arkadaşlarının kurduğu benzetim ortamının görünümü9
Şekil 2.1.	DurSim benzetim sistemi genel görünümü 11
Şekil 2.2.	DurSim benzetim ortamı panosu genel görünümü..... 12
Şekil 2.3.	DurSim kontrol paneli sürüş kolu, zil ve anahtar 13
Şekil 2.4.	DurSim kontrol paneli araç canlandırma butonları 15
Şekil 2.5.	DurSim kontrol paneli orta sol butonlar 16
Şekil 2.6.	DurSim kontrol paneli orta sağ butonlar 17
Şekil 2.7.	DurSim kontrol paneli sağ butonlar 18
Şekil 2.8.	DurSim kontrol paneli sol göstergeler..... 20
Şekil 2.9.	DurSim kontrol paneli sağ göstergeler 20
Şekil 2.10.	Buton MMI ana ekran görüntüsü 22
Şekil 2.11.	Buton MMI GPS ekran görüntüsü 22
Şekil 3.1.	DurSim yazılımı akış şeması 24
Şekil 3.2.	DurSim HataLoglama() fonksiyonu kodları..... 25
Şekil 3.3.	DurSim RailDriver.cs sınıfı kodları 26
Şekil 3.4.	HMI ana ekran görüntüsü 35
Şekil 4.1.	Çoklu ekran benzetim ortamı örneği 38
Şekil 4.2.	Gerçek kabinli benzetim ortamı örneği 39

ÇİZELGELER DİZİNİ

Sayfa No

Çizelge 3.1. GetControllerList() ID ve min-max değerleri.....	26
Çizelge 3.2. SetControllerValue() için kullanılan anahtarlar.....	32
Çizelge 3.3. GetControllerValue() için kullanılan geribildirimler.....	34



SİMGELER ve KISALTMALAR DİZİNİ**Kısaltmalar**

ADNKS	: Adrese Dayalı Nüfus Kayıt Sistemi
API	: Application Programming Interface
BCU	: Brake Control Unit
CAN Bus	: Controller Area Network Bus
DCU	: Door Control Unit
Dll	: Dynamic Link Library
DurSim	: Durma Simülatör
GPS	: Global Positioning System
HMI	: Human Machine Interface
HRS	: Hafif Raylı Sistemler
MMI	: Machine Machine Interface
TCU	: Traction Control Unit
UPA	: Uygulama Programlama Arayüzü
VCU	: Vehicle Control Unit

1. GİRİŞ

Gelişmiş ülkelerin büyük kentlerinde, XIX. yüzyılın sonlarından başlayan raylı sistem ağlarının genişletilme çabaları, günümüzde de sürmektedir. Bu çabalar, 1973-1974 yıllarındaki enerji bunalımı ve 1990 yıllarında sonra çevreye olan duyarlılığın artması nedeniyle hızlanmıştır. 1970'li yıllarda, gelişmiş ülkelerde, nüfusu 300.000'in altında hatta 100.000 dolayında olan kentlerde bile (Almanya'da Ulm, Würzburg, Mainz) raylı sistem uygulamaları yapılmıştır. Ancak bu uygulamalar genel eğilimi yansıtmamaktadır. Dünyada 1970'li yıllarda hızlanan kentsel raylı sistem geliştirme çabaları son yıllarda ülkemize yansımıştır. Aksaray - Esenler raylı sistemi, 1989 yılında işletmeye açılarak yeni bir gelişmenin başlangıcını oluşturmuştur (Öğüt ve Evren, 2006).

Raylı sistemler, ulaşımında sürekli artan yolculuk taleplerini verimli, hızlı, konforlu ve güvenli bir şekilde karşılamasının yanında diğer ulaşım seçeneklerine göre kullandığı elektrik enerjisi nedeniyle de çevreye en az zarar veren ulaşım türüdür. Raylı sistemler denildiğinde ilk akla gelenler, kullanımındaki yaygınlığına bağlı olarak tramvaylar, metrolar, hafif raylı sistemler ve banliyö trenleridir. Yukarıda bahsedilen sistemler ülkemizin farklı kentlerinde etkin olarak kullanılmaktadır (Torun, 2017).

Artan şehir nüfusu ile raylı sistemler arasında doğru bir orantı vardır. 2018 yılı ADNKS verilerine göre İstanbul 15.067.724, Ankara 5.503.985, İzmir 4.320.519, Bursa 2.994.521, Antalya 2.426.356, Adana 2.220.125, Konya 2.205.609, Şanlıurfa 2.035.809, Gaziantep 2.028.563, Kocaeli 1.906.391 nüfusa sahiptir (URL-3 TUIK, 2018). Bu illerin hepsinde raylı sistemler mevcuttur.

Raylı Sistemlerin tercih edilmesindeki faktörleri kısaca sıralarsak,

- Kitle taşımacılığı yapabilmesi nedeniyle, nüfus yoğunluğunun fazla olduğu büyük kentler için ideal bir toplu ulaşım sistemidir.
- Raylı sistemler, mevcut teknolojisi ve özellikleri nedeniyle diğer ulaşım seçeneklerine oranla daha hızlı, konforlu, güvenilir bir ulaşım sağlar.
- Kendine ait bir hat üzerinde ilerler, trafik yoğunluğu gibi durumlardan etkilenmez.
- Kullandığı enerji nedeniyle çevre kirliliğine etkisi minimum düzeydedir.
- Mevcut güzergâhlarındaki arazi kullanım oranları karayoluna göre çok daha düşüktür.

- Günümüzün en büyük sorunlarından biri olan karayollarındaki trafik kazalarının neden olduğu maddi ve manevi zararlara raylı sistemlerde çok fazla rastlanmaz (Torun, 2017).

Tramvaylar; karayolu ulaşım araçları ile aynı alanı kullanan, yol ve trafik durumuna göre bir sürücü tarafından kumanda edilen, elektrik enerjisini katenerden alan, günümüzde daha çok bir adım atılarak binilebilen alçak zeminli araçların kullanıldığı en düşük yolcu kapasiteli raylı toplu taşım sistemleridir. Karayoluna aynı seviyede döşenen raylar üzerinde hareket ettiğinden mevcut karayolu trafik düzenine uymak zorunda olup bu araçlara geçit ve kavşaklarda karayolu araçlarına göre geçiş üstünlüğü sağlanmaktadır. Ülkemizde birçok şehirde bu sistemler mevcuttur (Bayram, 2018). Şekil 1.1’de Bursa’da üretilen ve hizmet veren Türkiye’nin ilk yerli tramvayı ipekböceğinin resmi bulunmaktadır.



Şekil 1.1. Bursa’da kullanılan İpekböceği Tramvayının fotoğrafı.

Hafif raylı sistemler (HRS), gerektiğinde diğer trafikle birlikte kent caddelerini paylaşan, koşullara göre değişik biçimlerde ayrılmış yollardan veya tünellerden yararlanan, talebe göre kapasitenin arttırılmasını, dolayısıyla yatırımın aşamalı biçimde

gerçekleştirilmesini sağlayan, bu özellikleriyle de kentsel raylı sistemlere geniş olanaklar kazandıran sistemlerdir. Tramvay ile metro arasındaki bu sistemler, metro aşamasıyla sonlanma özelliğine de sahip bulunmaktadır. HRS, şehir içi raylı toplu taşımacılık sistemleri arasında önemli bir yere sahiptir. Tramvay sistemlerine oranla daha yüksek yolculuk kapasitesine sahip sistemlerdir. Saatteki maksimum yolculuk kapasiteleri 35000 yolcu/yön şeklindedir. Bu sistemler yolculuk taleplerinin yüksek olduğu ulaşım koridorlarında, ana ulaşım sistemleri olarak tercih edilmekle birlikte çok kalabalık şehirlerde daha yüksek kapasiteli sistemlerle entegre çalışan tali ulaşım sistemleri olarak da inşa edilebilmektedir. Hafif metro hatları tam tecritli güvenli sistemlerdir. Sistemin tramvaydan farkı, tecritli yolda gittiğinden daha hızlı ve güvenli ulaşım sağlamasıdır; çalışma prensibi olarak aralarında fark yoktur (Bayram, 2018). Şekil 1.2’de Bursa’da hizmet veren HRS araçlarının resmi bulunmaktadır.



Şekil 1.2. Bursa’da kullanılan HRS aracının fotoğrafı.

Metrolar yüksek yolcu kapasitesine sahip sistemlerdir. Maksimum saatteki yolcu kapasiteleri 100.000 yolcu/yön dür. İkili, üçlü, dörtlü ve daha fazla olacak şekilde yolcu kapasitesi ve hesaplanan peron uzunluğuna göre set olarak çalıştırılabilmektedir. Büyük şehirlerde en yüksek yolculuk taleplerinin tespit edildiği hatlarda metro sistemleri tercih edilmektedir. Tam tecritli raylı ulaşım sistemleri olan metrolar, genellikle yüzeydeki trafik yüklerini hafifletmek amacıyla derin tünel yöntemleri ile yeraltında inşa edilirler. Arazinin yapısına bağlı olarak aç kapa tünel veya delme (tünel) olarak inşa edilebilen metro hatları bazen yüzeyde hemzemin şeklinde veya viyadük üzerinde de inşa edilebilmektedirler (Bayram, 2018). Şekil 1.3’de İstanbul’da Üsküdar Çekmeköy metrolarında kullanılan araçların resmi bulunmaktadır. Bu metro araçları tam otomatik sürücüsüz metro araçlarına bir örnektir.



Şekil 1.3. Üsküdar-Çekmeköy hattında kullanılan metro aracının fotoğrafı.

Raylı sistem araçları bir çok alt ekipman ve bu ekipmanların kontrol ünitelerinden oluşmaktadır. Bu ekipmanları ve kontrol ünitelerinin birçoğu raylı sistem araçları için hayati önem taşımaktadır. Bu sistemlerin araç üretilmeden birbirleriyle uyumu ve haberleşmesinin test edilmesi ve bir bütün haline getirilmesi çok önemlidir, test edilmeyen üniteler veya yazılımlar araç yolculu işletime verildiğinde herhangi bir hata ile karşılaşılması durumunda çok daha büyük zararlara yol açabilmektedir.

Raylı sistem araçlarının ekipmanlarından bazıları;

- Araç Kontrol Ünitesi VCU (Vehicle Control Unit),
- Tahrik Sistemi Kontrol Ünitesi TCU (Traction Control Unit),
- Fren Sistemi Kontrol Ünitesi BCU (Brake Control Unit),
- Kapı Sistemi Kontrol Ünitesi DCU (Door Control Unit),

• Pantograf, elektrikli taşıtlarda elektrik enerjisini katener telinden taşıta aktarmaya yarayan devre elemanıdır. Aracın çatısında bulunur. Şaseden izolatörlerle izole edilmiştir. Hattın genişliğine göre büyük veya küçük pantograf kullanılır. İki adet karbon grafit kömür arşe üzerine monte edilmiştir. Bu iki kömürün pantografin her yüksekliğinde aynı yatay doğrultuda olması gerekir. Aksi hâlde kömürlerde dengesiz aşınma meydana gelir. Pantograf katener telinden alınan elektrik enerjisini mafsal bağlantı noktalarında kamçı diye adlandırılan örgülü iletkenlerle kesiciye aktarır. Pantograf mekanik gövdesi de elektrik enerjisini aktarmada iletken olarak kullanılmıştır (URL-1 MEB, 2013).

• Sürüş Kolu, raylı sistem araçlarında çekiş ve frenleme sisteminin komut bilgilerinin kontrolünün sağlandığı ekipmandır. Sürüş hazır olan araçta, sürüş kolu ileri itilerek, motorla güç verilir, araç ileri hareket eder. Kol orta konumuna getirildiğinde araca güç veya fren uygulanmaz, araç eylemsizdir. Kol geri çekilerek hareket halindeki araca fren uygulanır, araç yavaşlar.

• Olay Kaydedici, raylı sistem araçlarının kara kutusudur. Raylı sistem araçlarındaki analog veya dijital olarak neredeyse tüm mevcut sinyalleri kaydetmektedir ayrıca tren hızı, tren konumu veya fren sıcaklığı gibi güvenlikle ilgili kritik bilgileri kaydetmektedir. Olay Kaydedici ayrıca yerleşik dijital video kaydedicilerden (DVR) veya doğrudan kameralardan video akışlarını kaydedebilir.

• Kumlama Sistemi, raylı sistem araçlarında, tekerlek ve ray arasındaki kuru sürtünmeyi arttırmak için kullanılır. Çalıştırma işlemi ya otomatik olarak gerçekleşir ya da sürücü tarafından araçların ilk hareketi ve frenlenmesi esnasında hedef gözeterek uygulanır. Bir aracın acil frenlenmesi durumunda kumlama işlemi otomatik olarak devreye girer. Kum, bir kum sandığı içinde bulunur ve ihtiyaç olması durumunda basınçlı hava ile tekerlek ve ray arasına püskürtülür.

• İklimlendirme Sistemi, yolcu konforu için raylı sistem araçlarının iç ısısının 18 ile 22 derece sıcaklıklar arasında olması gereklidir. Ancak dış hava sıcaklığının

mevsimlere bağılı olarak deęişken olması iç sıcaklığı uygun deęerlerde muhafaza etmek için iklimlendirme araçlarına gerek göstermektedir. Bunun için iklimlendirme sistemleri kullanılmaktadır (URL-2 MEB, 2013).

- Yaęlama Sistemi, Raylı sistem araçlarının makaslarda ve kulplarda sürtünmeyi azaltması için kullanılan sistemdir, otomatik yada sürücü tarafından kulplara yada makaslara gelmeden elle yapılabilir.

- Yolcu Bilgilendirme Sistemi; Raylı sistem araçlarının içindeki ve duraklardaki insanlara LED paneller yada LCD ekranlar vasıtasıyla araç hakkında bilgi akışı sağlamaktadır ayrıca sesli anonslar, reklam ekranları gibi bütün ekipmanları kapsamaktadır.

- Makas Kontrol Ünitesi, araç üstü ve hat boyu olarak ikiye ayrılabilir. Araç üzerindeki verici sinyali, yol üzerinde iki ray arasına yerleştirilmiş olan alıcıya gönderilir. Alıcı da gelen bu sinyale göre makas ayarlamasını yapar. Makas talepleri makinist tarafından buton ile makas bölgesine gelmeden belirlenmiş talep noktasında verilir, yada önceden tanımlanan hatlar sayesinde otomatik olarak yapılabilir.

Batı dillerinde benzetim karşılığı olarak şu terimler kullanılır: simulation (Almanca, Fransızca, İngilizce), simulazione (İtalyanca), simulación (İspanyolca), simulação (Portekizce) ve simulatie (Felemenkçe). Buterimler, “benzer” anlamına gelen “similis” kökünden gelen, bir şeyin benzerini (taklidini) yapmak demek olan ve 14. yüzyıldan beri Latince kullanılan “simulare” sözcüğünden türetilmiş olup, teknik olmayan anlamda, bir şeyin benzeri veya sahtesi anlamında kullanılır. Bu terimler ancak 20. yüzyılda teknik bir anlam kazanmıştır. Günümüzde, Batı dillerinde benzetim terimi teknik olan ve olmayan anlamları ile kullanılmakta ve yerine göre hangi anlama geldiği anlaşılmaktadır (Ören, 2006).

Türkçede, teknik olmayan anlamda “simulation” karşılığı olarak “yalancı” ,“sahte” sözcükleri kullanılır ve teknik anlamda benzetim terimi uzun zamandan beri bilinir (Ören vd., 1985).

Benzetimin tarihi 5000 yıl öncesinde WEICH olarak bilinen ilk simülasyonlar Çin savaş oyunlarından gelmektedir. Bu oyunlar daha sonra ordu ve donanma stratejilerinin gelişimini sağlamak amacıyla da kullanılmıştır. Prussian’ların bu oyunları ordularındaki trenlerde kullanmalarından beri, tüm askeri stratejilerin geliştirilmesinde,

havacılık, nükleer enerji, endüstri gibi alanlarda kullanılmaya başlanmıştır (Shah vd., 2007).

Benzetim tarihindeki ikinci önemli adım 1929 yılında Edward Link tarafından geliştirilen ilk uçak simülatörü ile atılmıştır. 1949'da ücretli eğlence sürüşleri için tasarlanan Link'in simülatörü ordu ve ticari havacılık alanında eğitim ve değerlendirmelerde yaygın olarak kullanılmaya başlanmıştır. Günümüzde inşaattan moleküler biyolojiye, havacılık ve otomobil sektöründen tıp uygulamalarına kadar hayatın her alanında benzetim uygulamalarını görmek mümkündür. Kovboyların yarışma yaptığı hareketli taklit bizon makineleri, bir şehrin trafik akışını planlayan simülatörler, askeri amaçlı simülasyonlar, insanları taklit eden robotlar gibi pek çok benzer simülatör sayılabilir (Patrick, 2002).

Üniversitelerde ve işletmelerde benzetim tekniğinin kullanımı ile ilgili çeşitli araştırmalar yapılmıştır. Bu araştırmalardan Bazıları;

- (1978) Case Western Reserve Üniversitesinde- Yöneylem Araştırması Bölümünde yüksek lisans öğrencileri arasında yapılan bir araştırma sonucunda; benzetim 15 teknik arasında 5. sırada yer almıştır. Aynı çalışmanın doktora öğrencileri ile ilgili bölümünde ise İstatiksel metotlar birinci sırada olmak üzere doğrusal programlama ile benzetim ikinci sırayı paylaşmaktadır.

- Thomas ve Do Costa (1979), 137 firma arasında yapılan bir ankette benzetimin bu firmaların %84 ü tarafından kullanıldığını belirlemiştir. İstatiksel Analiz ise %93 kullanım oranı ile 1. sıradadır.

- Shanon, Long ve Buckles (1980), A.B.D. De Endüstri Mühendisleri Topluluğunun YA(OR) Bölümündeki üyeleri arasında bir araştırma yapmıştır. Bu araştırma sonuçlarına göre, benzetim 12 metot arasında, doğrusal programlamadan sonra 2. sırada yer almıştır.

- Forgionne (1983) ve Harpell, Lane ve Monsour (1989), büyük şirketler arasında yaptığı bir araştırmada, sekiz farklı metot arasında benzetimin 2.sırada yer aldığını göstermiştir.

Benzetim, karmaşık sistemlerin tasarımı ve analizinde kullanılan en güçlü analiz araçlarından birisidir. Genel anlamda benzetim, zaman içinde sistemin işleyişinin taklididir. Benzetim, çeşitli koşullar altında sistemin tavrının gözlemlenebilmesi için, bu sistemin modellenmesi olarak da tanımlanabilir. Zaman içinde değişiklik gösteren bir

sistemin tavrı, geliştirilen bir benzetim modeli ile incelenir. Bu model, sistemin çalışması ile ilgili kabuller setinden oluşur. Bu kabuller, sistemin ilgilenilen nesneleri (varlıkları) arasındaki matematiksel, mantıksal ve sembolik ilişkiler ile ifade edilir (Dengiz, 2012).

Benzetim, dinamik bir sistemin özelliklerini ve davranışlarını bilgisayar aracılığıyla değerlendiren bir tekniktir. Bir simülasyon modeli, temel olarak “ne-eğer” (what-if) analizlerinin yapılmasını sağlayan bir araç olarak ele alınmalıdır.

Deneyim kazanmanın ya da deneyin gerçek sistem yerine bir modeli ile yapılması gerektiği durumlarda benzetim gerekli olur. Bu durumlar şöyle özetlenebilir:

- Gerçek sistemin olmaması,
- Gerçek sisteme erişimin kolay olmaması,
- Gerçek sistemle deneyin tehlikeli olması,
- Gerçek sistemin maliyetli olması,
- Gerçek sistemle deneyin rahatsız edici olması,
- Analitik çözüm tekniklerinin olmaması veya zor olması,
- Sistemin çok yavaş veya çok hızlı olması (Güven ve Ören, 2005).

Bir benzetim çalışmasının temel amaçları şöyle sıralanabilir:

- Bir gerçek hayat sistemini girdi ve çıktılarıyla matematiksel olarak ifade etmek,
- Gerçek sistemi, kurulan model üzerinden tanıyıp araştırmak, değişik kararları ve seçenekleri gerçek sistemde hiçbir değişiklik yapmadan deneyebilmek,
- Elde edilen bilgiler ışığında, sistemle ilgili ön görüşlerde bulunabilmek ve uygulamaya esas olan kararları belirlemek (URL-4).

Literatüre baktığımızda, raylı sistem araçları için bu tarz fiziksel benzetim ortamına pek rastlanmamaktadır, daha çok otomobil, uçak, gemi, tank araçları için benzetim ortamı üzerine çalışmalar yapılmıştır. Raylı sistem araçları için yapılan çalışmalardan örnek vermek gerekirse; Guo ve arkadaşları yüksek hızlı tren benzetim ortamı kurmuşlardır ve tren hızının ve arka plan görüntüsü karmaşıklığının yüksek hızlı tren sürücüsünün görsel davranışı üzerindeki etkilerini anlamak için bir deneylerini yapmışlardır. 20 adet katılımcı ile yapılan bu deneyde her katılımcı doğrudan simülasyon sisteminin önündeki sürücü koltuğuna oturmuş ve pasif olarak treni sürmüşler, yani treni önceden ayarlanmış bir hızla sürdüğünü izlediler, ancak bu hızı

kontrol etmek için bir sürücü denetleyicisi kullanmadılar. Her deneyin başında, bir hız belirlenmiş ve simülatör yalnızca ayarlanan hızda çalışmış ve kullanıcıların görsel davranış analizleri yapılmıştır (Gua vd., 2015).Şekil 1.4’de Guo ve arkadaşlarının kurmuş olduğu benzetim ortamının genel görünümü bulunmaktadır.



Şekil 1.4.Guo ve arkadaşlarının kurduğu benzetim ortamının görünümü.

Bu tez çalışmasında ise raylı sistem üreticileri ve işletmeleri için, araç üretilmeden önce, aracın bire bir aynı kontrol paneli, araç kontrol ünitesi, giriş-çıkış modülleri, insan makine arayüzleri ve yazılımların benzetimlerinin yapılacağı ayrıca tramvay araçlarında kullanılan ekipmanların testlerinin de yapılabileceği DurSim tam ve eş zamanlı benzetim ortamı üzerinde çalışılmıştır. Bu benzetim ortamı ile yapılan testler sonucunda ekipmanların yeterliliği ve kullanılabilirliği belirlenebilecek olup, mevcut araç yazılımların testleri birebir araç üzerinde yapılmış gibi DurSim yardımıyla dinamik

olarak test edilebilecektir. Bu tez çalışması çerçevesinde geliştirilen DurSim sayesinde aracın üretiminden sonra ortaya çıkması muhtemel yazılımsal veya donanımsal eksiklikler ve hataların tespit edilmesi ve üretimden önce bu hataların giderilip sıfır hatalı raylı sistem araç üretimi hedeflenmektedir. Ayrıca benzetim ortamında kurulan kontrol paneli sayesinde sürücülerin tam ve eş zamanlı sürüş eğitimlerinin verilmesi hedeflenmektedir, sürücü eğitimleri sayesinde sürücülerin kontrol paneli ekipmanlarını tanımaları ve işlevlerini öğrenmeleri hedeflenmektedir. Ayrıca sürücülerin araçlarda kullanılacak insan makine arayüzünün kullanımını öğrenmelerinin sağlanması ve sürücülerin araç üretilmeden önce sürüş tecrübesi kazanması hedeflenmektedir. Benzetim ortamı üzerinde yapılacak bu eğitimler sayesinde hiçbir risk faktörü olmadan sürücü eğitimlerinin güvenli, yetkin ve düşük maliyetli bir şekilde tam ve eş zamanlı olarak gerçekleştirilmesi sağlanmış olacaktır.

2. DURSİM TRAMVAY SİMÜLATÖR SİSTEMİ

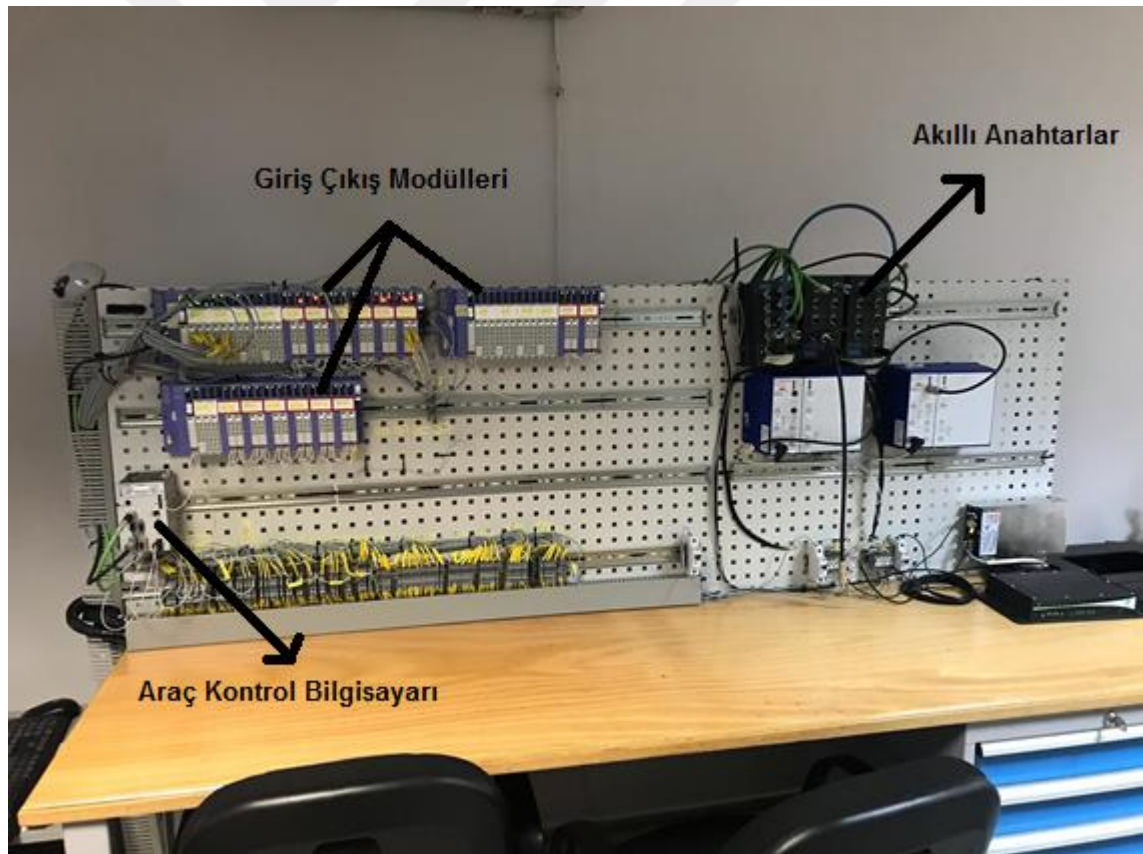
DurSim (Durma Simülatör) Durmazlar Makine A.Ş Raylı Sistemler Ar-Ge yazılım ekibi tarafından 1501- TÜBİTAK Sanayi AR-GE Projeleri Destekleme Programı kapsamında “3180351” numaralı “ENDÜSTRİ 4.0 KAPSAMINDA RAYLI SİSTEMLER TAM VE EŞ ZAMANLI SİMÜLASYON VE KOORDİNASYON MERKEZİ PROJESİ” başlıklı projenin çıktısıdır. Bu tez çalışması bu projenin lideri olarak görev yapan yazar tarafından yapılmıştır. Bu proje sayesinde araç üretilmeden önce, aracın birebir aynı kontrol paneli, araç kontrol ünitesi, giriş çıkış modülleri, insan makine arayüzleri ve yazılımların benzetimleri yapılmıştır ve bu ekipmanların testleri yapılmaktadır. Yapılan testler sonucunda ekipmanların yeterliliği, kullanılabilirliği belirlenmektedir, yazılımların testleri birebir araç üzerinde yapılmış gibi dinamik olarak test edilmektedir. Bu sayede araç üretiminden sonra ortaya çıkabilecek yazılımsal veya donanımsal eksiklikler ve hataların azaltılması hedeflenmektedir. Şekil 2.1’de kurulan DurSim benzetim ortamının genel görünümü bulunmaktadır.



Şekil 2.1. DurSim benzetim sistemi genel görünümü.

2.1. DurSim Kurulumu

DurSim benzetim ortamının kurulması için öncelikle İstanbul Eminönü-Alibeyköy hattında kullanılacak olan tramvay araçları için tasarlanan kontrol panelinin aynısı kurulmuştur. Kontrol paneli üzerinde bulunan butonlar, sürüş kolu, saclar, insan makine arayüzü ekranları araçta kullanılanların aynısı olacak şekilde tasarlanmıştır. Aynı zamanda araçta kullanılan araç kontrol ünitesi, giriş çıkış modüllerinin birebir aynılarının kurulumları araçların elektrik projesine bağlı kalınarak yapılmıştır ve ekipmanların birbiriyle bağlantısı için araçlarda kullanılan akıllı anahtarlar kullanılmıştır. İstanbul Eminönü-Alibeyköy tramvay hattında kullanılacak olan tramvay araçları için yazılan yazılımlar araç kontrol ünitesine ve insan makine arayüzleri ekranlarına yüklenmiştir. Şekil 2.2’de araç kontrol ünitesi, giriş çıkış modülleri ve anahtarların bulunduğu panonun genel görünümü bulunmaktadır.



Şekil 2.2 DurSim benzetim ortamı panosu genel görünümü.

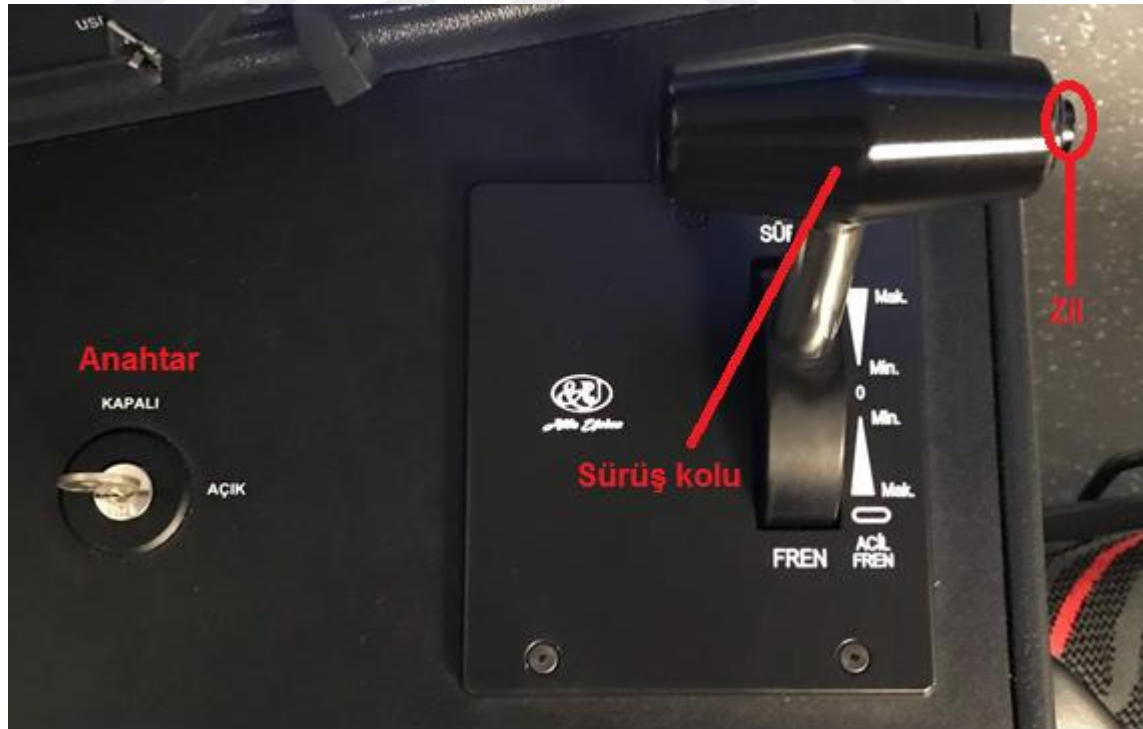
2.2. DurSim Kontrol Paneli Elemanları

Kontak Anahtarı: Kabin aktivasyon durumunu seçmek için iki konumlu (Açık/Kapalı) bir sürücü anahtarı yer almaktadır. Kabin aktif olmadan diğer butonlar kullanılamaz.

Sürüş Kolu: Sürüş kolu, aracın çekiş ve frenleme sisteminin komut bilgilerinin kontrolünün sağlandığı ekipmandır. Sürüşe hazır olan araçta, sürüş kolu ileri itilerek, araca cer verilir, araç ileri hareket eder. Kol orta konumuna getirildiğinde araca cer veya fren uygulanmaz, araç eylemsizdir. Kol geri çekilerek hareket halindeki araca fren uygulanır, araç yavaşlar. Kol en geriye çekildiğinde acil fren uygulanır. Bu durumda dinamik fren, mekanik fren, kumlama, ray freni, kayma/kızaklama kontrol ve yük ayarı devrede olacaktır. Sürüş kolu fren ya da sürüş pozisyonunda iken el çekildiğinde, kol bırakıldığı pozisyonda kalmaktadır.

Zil: Sürüş kolunda sağ tarafta bulunur, basınca zil aktif olur.

Şekil 2.3’de kontak anahtarı, sürüş kolu ve zilin kontrol panelinde bulunduğu bölümün resmi verilmiştir.



Şekil 2.3. DurSim kontrol paneli sürüş kolu, zil ve anahtar.

24V Akü Anahtarı: Aracın aküsünü açma ve kapatma için kullanılır. Devre dışı, 0 ve devrede pozisyonları vardır. Yay geri dönüşlü olarak çalışmaktadır. Anahtar açık konuma getirildikten sonra 24V akü butonu sağa çevrilerek akü açılır. Akü anahtarı sola çevrilerek kapatılır.

Ana Devre Kesici Anahtarı: Aracın yüksek hızlı devre kesicisini açma ve kapatma için kullanılır. Devre dışı, 0 ve devrede pozisyonları vardır. Yay geri dönüşlü olarak çalışmaktadır. Akü açıldıktan sonra pantograf kaldırılır ve ana devre kesici Devre dışı / Devrede anahtarı sağa çevrilerek ana devre kesici devreye alınır. Ana devre kesici anahtarı sola çevirerek, ana devre kesici devre dışı bırakılır.

Acil Aydınlatma Butonu: Aracın acil aydınlatmasını açma ve kapatma için kullanılır. Aç ve kapat pozisyonları vardır. Kalıcı olarak çalışmaktadır. Acil aydınlatma butonu çevrildiğinde (1) acil aydınlatmalar devreye alınır ve tavan-kapı acil aydınlatmaları yanar.

Temizlik Modu Butonu: Aracın iç yıkama işlemlerini açma ve kapatma için kullanılır. 0 ve 1 pozisyonları vardır. Yay geri dönüşlü olarak çalışmaktadır. Temizlik Modu butonu sağa (1) çevrilerek Temizlik Modu aktif olur ve tüm kapılar serbest konumundadır.

Şekil 2.4’de araç canlandırma butonlarının kontrol panelinde bulunduğu bölümün resmi verilmiştir.



Şekil 2.4. DurSim kontrol paneli araç canlandırma butonları.

Yön Seçimi butonu: İleri yön, geri yön ve 0 pozisyonları vardır. Kalıcı olarak çalışmaktadır. Aracın hareket yönünü seçmek için kullanılır. Sağa çevrildiğinde ileri yön sola çevrildiğinde geri yön seçilmektedir.

Sol Kapılar Aç butonu: Aracın sol tarafında bulunan kapıların açılması için kullanılmaktadır. Basmalı butondur, Bas-Bırak olarak çalışmaktadır. Durumunu gösteren mavi ikaz uyarı ışığı vardır.

Sol Kapılar Serbest butonu: Aracın sol tarafında yer alan kapıların serbest bırakılması için kullanılmaktadır. Basmalı butondur, Bas-Bırak olarak çalışmaktadır. Durumunu gösteren sarı ikaz uyarı ışığı vardır.

Şekil 2.5’de yön seçimi, sol kapılar aç ve sol kapılar serbest butonunun kontrol panelinde bulunduğu bölümün resmi verilmiştir.



Şekil 2.5. DurSim kontrol paneli orta sol butonlar.

Dış Aydınlatma Butonu: Aracın kısa ve uzun farlarını açma ve kapatma için kullanılır. Park, kısa far ve uzun far pozisyonları vardır. 3 kademeli kalıcı olarak çalışmaktadır. Dış aydınlatma seçici butonu kısa farlar seçili iken kısa far göstergesi yanar. Dış aydınlatma seçici butonu uzun farlar seçili iken uzun far göstergesi yanar.

Sağ Kapılar Aç Butonu: Aracın sağ tarafında bulunan kapıların açılması için kullanılmaktadır. Basmalı butondur, Bas-Bırak olarak çalışmaktadır. Durumunu gösteren mavi ikaz uyarı ışığı vardır.

Sağ Kapılar Serbest Butonu: Aracın sağ tarafında yer alan kapıların serbest bırakılması için kullanılmaktadır. Basmalı butondur, Bas-Bırak olarak çalışmaktadır. Durumunu gösteren sarı ikaz uyarı ışığı vardır.

İmdat Freni Butonu: İmdat freni sürücü konsolu üzerinde bulunan bir itmeli tip butondur. İmdat freni butonu, bir kere uygulandığında her ne olursa olsun araç durana dek geri alınamaz şekilde uygulanır. İmdat freni uygulandığında mekanik fren, ray frenleri ve kumlama devreye girecektir. Basmalı ve kilitlemeli kırmızı mantar butondur.

Şekil 2.6’da dış aydınlatma, sağ kapılar aç, sağ kapılar serbest ve imdat freni mantar butonunun kontrol panelinde bulunduğu bölümün resmi verilmiştir.



Şekil 2.6. DurSim kontrol paneli orta sağ butonlar.

İzolasyon Modu Butonu: Trenin yüksek gerilim kaynaklarının kesildiği moddur. Pantograf aşağıda pozisyonuna, tüm yerden besleme pabuçları yukarı pozisyonuna geçecektir. İlgili 750V batarya besleme kontağı açık pozisyonuna geçecektir. Mod geçişleri süresince yanıp sönecektir. Mod geçişi tamamlandığında durumunu gösteren uyarı ışığı sabit yanacaktır. Basmalı butondur, Bas-Bırak olarak çalışmaktadır. Durumunu gösteren yeşil ikaz uyarı ışığı vardır. Pantograf besleme ve yerden besleme mod geçişlerinde, pantograf besleme ve batarya besleme mod geçişlerinde önce izolasyon moduna geçilecek sonrasında ilgili besleme modu seçimi yapılacaktır.

Yerden Besleme Butonu: Aracın yerden besleme sistemini devreye almak için kullanılır. Yerden besleme butonuna basıldığında pabuç-1 ve pabuç-2, 750 V besleme rayına inecek, yerden besleme modu devreye girecektir. Mod geçişleri süresince yanıp sönecektir. Mod geçişi tamamlandığında durumunu gösteren uyarı ışığı sabit yanacaktır. Basmalı butondur, Bas-Bırak olarak çalışmaktadır. Durumunu gösteren kırmızı ikaz uyarı ışığı vardır.

Pantograf Butonu: Aracın pantograftan besleme sistemini devreye almak için kullanılır. Butona basıldığında pantograf yukarı yönlü hareket edecek, pantograftan besleme modu devreye girecektir. Mod geçişi tamamlandığında durumunu gösteren uyarı ışığı sabit yanacaktır. Basmalı butondur, Bas-Bırak olarak çalışmaktadır. Durumunu gösteren kırmızı ikaz uyarı ışığı vardır.

750V Batarya Butonu: Aracın bataryadan besleme sistemini devreye almak için kullanılır. Araç besleme gerilimi 750V bataryadan beslemeye geçecektir. Mod geçişi tamamlandığında durumunu gösteren uyarı ışığı sabit yanacaktır. Basmalı butondur, Bas-Bırak olarak çalışmaktadır. Durumunu gösteren kırmızı ikaz uyarı ışığı vardır.

Flaşör Butonu: Aracın sinyallerinin flaşör olarak çalışmasını açma ve kapatma için kullanılır. Basmalı butondur, kalıcı olarak çalışmaktadır. Durumunu gösteren ikaz uyarı ışığı vardır. Rengi Kırmızıdır.

Şekil 2.7’de izolasyon modu, yerden besleme, pantograf, 750V batarya ve flaşör butonunun kontrol panelinde bulunduğu bölümün resmi verilmiştir. Kontrol panelinde sağ tarafta yer almaktadır.



Şekil 2.7. DurSim kontrol paneli sağ butonlar.

Sol Sinyal Gösterge Grubu (İkaz Işığı): Aracın sola sinyal komutu seçili olduğunda lambaların yanıp söndüğünü gösteren yeşil ikaz ışığıdır.

Sağ Sinyal Gösterge Grubu (İkaz Işığı): Aracın sağa sinyal komutu seçili olduğunda lambaların yanıp söndüğünü gösteren yeşil ikaz ışığıdır.

Tüm Kapılar Kapalı Gösterge Grubu (İkaz Işığı)-Kapı Döngüsü: Tüm kapılar kapalı olunca yeşil renkte yanar. Şekil 2.8’de tüm kapılar kapalı ikaz ışığı yanmaktadır.

Sürüş Hazır Gösterge Grubu (İkaz Işığı): Aracın sürüş hazır olduğunu gösteren mavi ikaz ışığıdır. Şekil 2.8’de aracın sürüş hazır ikaz ışığı yanmaktadır.



Şekil 2.8. DurSim kontrol paneli sol göstergeler.

Fren Gösterge Grubu-Park Freni Gösterge Grubu (İkaz Işığı): Aracın fren uyguladığını gösteren kırmızı ikaz ışığıdır.

Arıza Gösterge Grubu (İkaz Işığı): Araçta sürüş engel bir arızanın olduğunu gösteren kırmızı ikaz ışığıdır.

Otomatik Servis Freni Gösterge Grubu (İkaz Işığı): Araçta otomatik servis freni uygulandığını gösteren kırmızı ikaz ışığıdır.

Yangın Alarmı Gösterge Grubu (İkaz Işığı): Duman sensörlerinden duman algılandığını gösteren kırmızı ikaz ışığıdır.

Şekil 2.9’da fren, arıza, otomatik servis freni, yangın alarmı göstergelerinin kontrol panelinde bulunduğu bölümün resmi verilmiştir.



Şekil 2.9. DurSim kontrol paneli sağ göstergeler.

Sürücü Aydınlatma: Aracın sürücü aydınlatmasını açma ve kapatma için kullanılır.

Sürücü Aydınlatma Parlaklık Ayarı: Aracın sürücü aydınlatmasının ışık şiddetini azaltmak ve artırmak için kullanılır.

Yolcu aydınlatma: Aracın yolcu aydınlatmasını açma ve kapatma için kullanılır.

Cam ısıtıcı: Aracın ön cam ısıtıcısını açma ve kapatma için kullanılır.

Silecek: Aracın silecek sisteminin hızlarının ayarlanması için kullanılmaktadır.

Sis Farı: Aracın sis farlarını açma ve kapatma için kullanılır.

Zorla Kapatma: Aracın açık olan kapılarını kapatmak için kullanılır.

Park Freni: Park frenini aktif etmek için kullanılır.

Sol Sinyal: Aracın sol sinyallerini açma ve kapatma için kullanılır.

Sağ Sinyal: Aracın sağ sinyallerini açma ve kapatma için kullanılır.

Kapı Elçek Reset: Yolcular tarafından kapı elçeği çekildiğinde sürücü tarafından sinyalin iptal edilmesi için kullanılacaktır.

Ölü Adam Reset: Ölü adam butonuna 10 saniyeden uzun süre basılmadığında ölü adam uygulaması aracı durduracaktır. Aracı tekrar hareket ettirmek için ölü adam serbest butonuna basarak freni resetlemek ve fonksiyonu aktif etmek için kullanılır.

Şekil 2.10'da yukarıdaki butonların bulunduğu buton MMI ana ekranının görüntüsü verilmiştir.



Şekil 2.10. Buton MMI ana ekran görüntüsü.

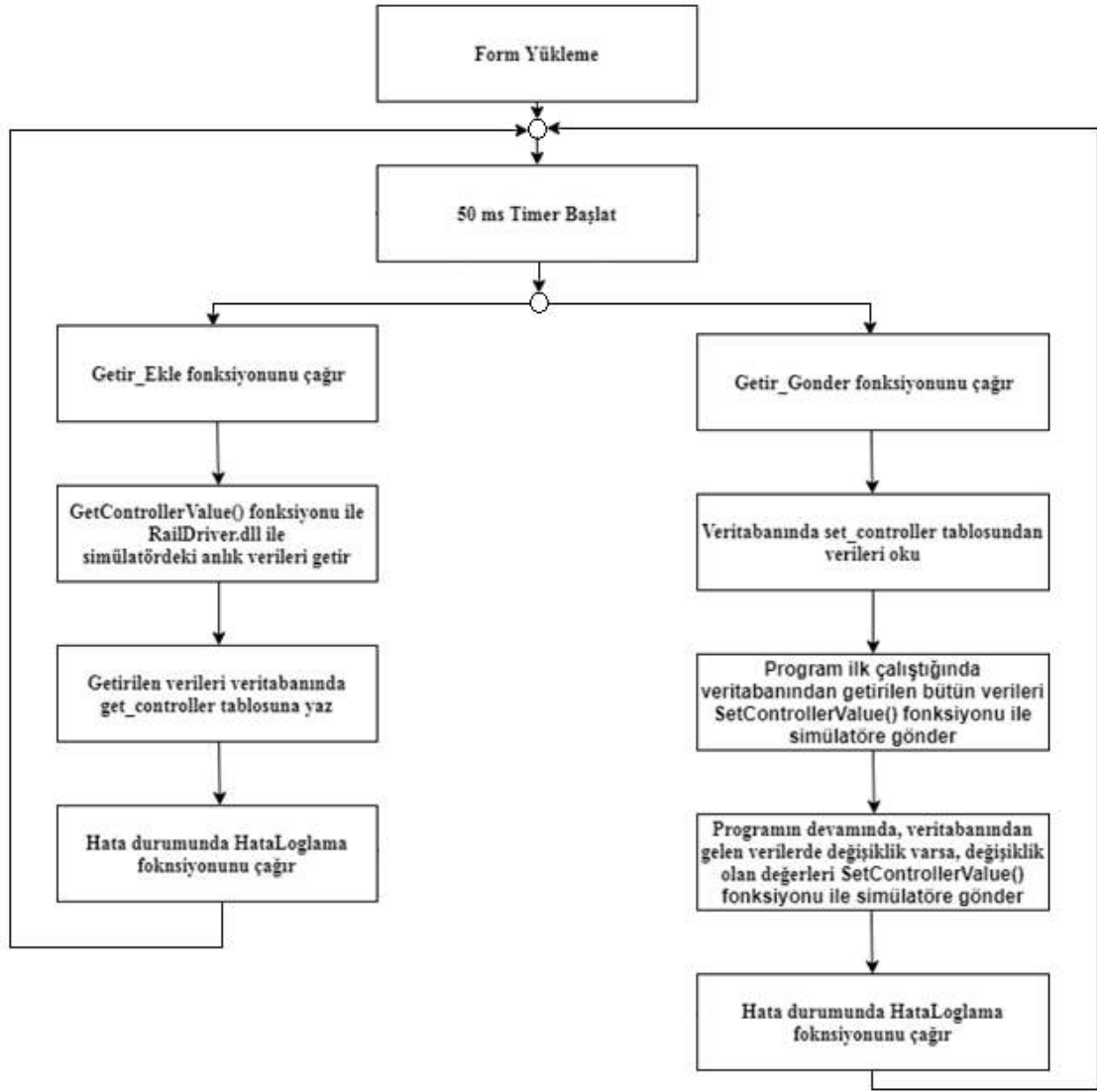
2. DURSİM YAZILIM BİLEŞENLERİ

DurSim yazılımı Visual Studio 2015 ortamında C# yazılım dili kullanılarak yazılmıştır, veritabanı olarak MySQL kullanılmıştır. Bu yazılımın amacı RailDriver.dll ile haberleşmeyi sağlayıp Train Simülör 2019'dan verilerin getirilmesi, getirilen verilerin veritabanına eklenmesi, kontrol panelinden gelen girişlerin Train Simülör 2019'a gönderilmesini sağlamaktır.

Yazılım içinde bulunan *Getir_Ekle* fonksiyonunda Train Simülör 2019'dan gelen bütün veriler *GetControllerValue()* fonksiyonu ile getirilir, getirilen bu veriler diziye aktarılır ardından dizinin bütün elemanları veritabanında *get_controller* tablosuna yazılır. Sonrasında HMI (İnsan Makine Arayüzü) yazılımı ile simülörden getirilen veriler örneğin hız, sürüş kolu pozisyonu, aracın konumu gibi değişkenler kontrol panelinde bulunan HMI ekranında gösterilir. DurSim yazılımındaki *Getir_Ekle* fonksiyonun kodları Ek-1'de verilmiştir.

Kurulan kontrol panelinden Train Simulator 2019'daki raylı sistem aracının kullanımı sağlanması için *Getir_Gonder()* fonksiyonu kullanılmaktadır. Her 50 ms'de bir bu fonksiyon çağırılmaktadır. Bu fonksiyonda; program ilk kez çalıştırıldığında *set_controller* veritabanındaki veriler çekilir, veritabanından çekilen bütün veriler geçici bir diziye aktarılır, ardından bütün veriler *SetControllerValue()* fonksiyonu ile Train Simulator 2019'a gönderilir. Sonrasında *set_controller* veritabanındaki veriler çekilir, veritabanından çekilen bütün veriler bir diziye aktarılır, bu dizi geçici dizi ile karşılaştırılır eğer bir değişiklik varsa değişiklik olan değerler geçici dizide güncellenmekte ve *SetControllerValue()* fonksiyonu ile Train Simulator 2019'a gönderilmektedir. DurSim yazılımındaki *Getir_Gonder* fonksiyonun kodları Ek-2'de verilmiştir.

Şekil 3.1'de DurSim yazılımının akış şeması verilmiştir.



Şekil 3.1. DurSim yazılımı akış şeması.

HataLoglama fonksiyonuyla program içerisindeki try-catch kontrol bloğuyla yapılan kontrollerde herhangi bir hata ile karşılaşılması durumunda err.log dosyasına hataları yazma işlemi yapılmaktadır. Bu sayede herhangi bir hata durumunda kayıtlar incelenerek teşhis edilebilmektedir. Şekil 3.2’de DurSim yazılımındaki *HataLoglama* fonksiyonunun kodları verilmiştir.

```
private void HataLoglama(string msgString)
{
    string date = DateTime.Now.Date.ToString("yyyyMMdd").TrimEnd(':');
    string filename = string.Format(@"err\" + date + "_err.log");
    DosyaOlustur = File.AppendText(filename);
    DosyaOlustur.WriteLine(DateTime.Now + " --- "
+ Application.ProductName + " --- " + msgString);
    DosyaOlustur.Close();
}
```

Şekil 3.2. DurSim *HataLoglama* fonksiyonu kodları.

3.1. RailDriver Dinamik Bağlantı Kitaplığı

Benzetim ortamı görsellik için “Steam” platformu üzerinden “Train Simulator 2019” satın alınmıştır ve bu simülatorün Uygulama Programlama Arayüzü (UPA) [İngilizce: Application Programming Interface – API] kullanılmıştır. Uygulama Programlama Arayüzü (UPA) işletim sisteminin, bir yazılımın başka bir yazılımda tanımlanmış işlevlerini kullanabilmesi için oluşturulmuş bir tanım bütünüdür (URL-5). Kullanılan bu UPA’da 26 adet fonksiyon yer almaktadır, bu çalışmada kullanılmakta olan fonksiyonlar aşağıdadır.

GetControllerList(): Araçtan getirilecek verilerin tamamını string değer olarak geri döndürmektedir. Tablo 1’ deki veriler bu fonksiyon ile elde edilmiş verilerdir.

GetLocoName(): Bu fonksiyon, kullanılan raylı sistem aracının Sağlayıcı, Ürün ve Model adını string değer olarak geri döndürmektedir. Bu form “PRODUCER.:PRODUCT.:MODEL” içindedir. Bu hangi raylı sistem aracını sürdüğünü doğru bir şekilde belirleme ve buna göre uyarılama olanağı sağlar. Örneğin; producer: UrbanTrainSim,product: Frankfurt U-Bahn Typ U5_V2,model: Frankfurt U5 Section 1

GetRailDriverConnected(): Bu fonksiyon, RailDriver.dll bağlantısını kontrol etmektedir ve sonucu bool değer olarak döndürmektedir.

GetControllerValue(int controllerId, Single value):Araçtan verilerin getirilmesi için kullanılan fonksiyondur, bu fonksiyon için getirilecek verinin ID’ si ve verinin hangi değeri getirilmek isteniyorsa ID’si girilir, aşağıda hangi değer için hangi değer girilmesi gerektiği gösterilmektedir;

- 0 - Current - Anlık değer
- 1 - Min - En küçük değer

- 2 - Max - En büyük değer

SetControllerValue(int controllerId, Single value): Verilerin araca gönderilmesi için kullanılan fonksiyondur, bu fonksiyon için gönderilecek verinin ID' si ve anlık değeri girilir. Anlık değer min-max aralığında olmalıdır.

Şekil 3.3'de DurSim yazılımındaki *RailDriver* sınıfının kodları verilmiştir.

```
class RailDriver
{
    protected const string uzanti = "D:/SteamLibrary/steamapps/common/RailWorks/plugins/RailDriver.dll";

    public enum RDid{...}

    public enum RDmod
    {
        Current,
        Min,
        Max
    }

    [DllImport(uzanti, CallingConvention = CallingConvention.Cdecl)]
    public static extern float GetControllerValue(int controllerId, int getType);

    [DllImport(uzanti, CallingConvention = CallingConvention.Cdecl)]
    public static extern float SetControllerValue(int controllerId, Single value);

    [DllImport(uzanti, CallingConvention = CallingConvention.Cdecl)]
    public static extern string GetLocoName();

    [DllImport(uzanti, CallingConvention = CallingConvention.Cdecl)]
    public static extern string GetControllerList();

    [DllImport(uzanti, CallingConvention = CallingConvention.Cdecl)]
    public static extern bool GetRailDriverConnected();
}
```

Şekil 3.3. DurSim RailDriver.cs sınıfı kodları.

GetControllerValue() ve *SetControllerValue()* fonksiyonunda kullanılan anahtarların ID listesi ve alabilecekleri en küçük ve en büyük değerler Çizelge 3.1'de gösterilmiştir; ID numarası 400 ile başlayanlar sanal kontrollerdir, *GetControllerList()* fonksiyonu ile bu değerler tespit edilemediğinden tabloda ilgili alanlar boş bırakılmıştır.

Çizelge 3.1. GetControllerList() ID ve min-max değerleri.

ID	Anahtar	MIN - MAX
0	AI	[0.0, 1.0]
1	isA	[0.0, 1.0]
2	IBIS_Play	[0.0, 1.0]
3	IBIS_Kursanzeige	[0.0, 99.0]

Çizelge 3.1. (Devam Ediyor) GetControllerList() ID ve min-max değerleri.

4	IBIS_IsSet	[0.0, 1.0]
5	IBIS_Next	[0.0, 1.0]
6	IBIS_On	[0.0, 1.0]
7	IBIS_Sound	[0.0, 1.0]
8	IBIS_Btn_1	[0.0, 1.0]
9	IBIS_Btn_2	[0.0, 1.0]
10	IBIS_Btn_3	[0.0, 1.0]
11	IBIS_Btn_4	[0.0, 1.0]
12	IBIS_RightBtn_1	[0.0, 1.0]
13	IBIS_RightBtn_2	[0.0, 1.0]
14	IBIS_RightBtn_3	[0.0, 1.0]
15	IBIS_RightBtn_4	[0.0, 1.0]
16	IBIS_Ziel	[0.0, 999.0]
17	IBIS_Linie	[0.0, 99.0]
18	IBIS_Curent_Station	[0.0, 999.0]
19	IBIS_Enter	[0.0, 1.0]
20	IBIS_Del	[0.0, 1.0]
21	IBIS_9	[0.0, 1.0]
22	IBIS_8	[0.0, 1.0]
23	IBIS_7	[0.0, 1.0]
24	IBIS_6	[0.0, 1.0]
25	IBIS_5	[0.0, 1.0]
26	IBIS_4	[0.0, 1.0]
27	IBIS_3	[0.0, 1.0]
28	IBIS_2	[0.0, 1.0]
29	IBIS_1	[0.0, 1.0]
30	IBIS_0	[0.0, 1.0]
31	Horn	[0.0, 1.0]
32	EmergencyBrake	[0.0, 1.0]

Çizelge 3.1. (Devam Ediyor) GetControllerList() ID ve min-max değerleri.

33	dver	[0.0, 1.0]
34	shtora	[0.0, 1.0]
35	kreslo	[0.0, 1.0]
36	fortl	[0.0, 1.0]
37	fortr	[0.0, 1.0]
38	DriverMirrors	[0.0, 0.7]
39	DriverKey	[0.0, 1.0]
40	DoorsClose	[0.0, 1.0]
41	Controlinverse	[0.0, 1.0]
42	TractiveEffort	[-1000.0, 1000.0]
43	Acceleration	[-1000.0, 1000.0]
44	HandBrake	[0.0, 1.0]
45	AbsoluteSpeed	[0.0, 100.0]
46	BrakeAmount	[0.0, 1.0]
47	DistanceTravelled	[0.0, 1.0]
48	Current	[0.0, 100000.0]
49	BrakeActive	[0.0, 1.0]
50	Warnblinker	[0.0, 1.0]
51	Luefter	[0.0, 1.0]
52	Stoermeldung	[0.0, 1.0]
53	Automat_ein	[0.0, 1.0]
54	Automat_aus	[0.0, 1.0]
55	Automat	[0.0, 1.0]
56	Abfahrt	[0.0, 1.0]
57	Sand	[0.0, 1.0]
58	Kupplung	[0.0, 1.0]
59	Number_sections	[0.0, 14.0]
60	SectionNum	[1.0, 9.0]
61	RollbandUP	[0.0, 1.0]
62	RollbandDOWN	[0.0, 1.0]

Çizelge 3.1.(Devam Ediyor) GetControllerList() ID ve min-max değerleri.

63	Weiche_alarm_l	[0.0, 1.0]
64	Weiche_alarm	[0.0, 1.0]
65	Weiche_r_Btn_l	[0.0, 1.0]
66	Weiche_l_Btn_l	[0.0, 1.0]
67	Weiche_l_Btn	[0.0, 1.0]
68	Weiche_l	[0.0, 1.0]
69	Weiche_r_Btn	[0.0, 1.0]
70	Weiche_r	[0.0, 1.0]
71	Cab_light_Btn	[0.0, 1.0]
72	Cab_light	[0.0, 1.0]
73	Pultlicht	[0.0, 1.0]
74	WipersON	[0.0, 1.0]
75	WipersONslow	[0.0, 1.0]
76	WipersState	[0.0, 2.0]
77	Wipers_interior	[0.0, 1.0]
78	Fwd_headlight_nah	[0.0, 1.0]
79	Fwd_headlight_nah_l	[0.0, 1.0]
80	Fwd_headlight_fern	[0.0, 1.0]
81	Fwd_headlight_fern_l	[0.0, 1.0]
82	Headlight	[0.0, 1.0]
83	Interior_light	[0.0, 2.0]
84	InteriorlightON	[0.0, 1.0]
85	InteriorlightOFF	[0.0, 1.0]
86	InteriorlightAUTO	[0.0, 1.0]
87	BataryBtnON	[0.0, 1.0]
88	BataryBtnOFF	[0.0, 1.0]
89	BatteryControl	[0.0, 1.0]
90	CloseDoors	[0.0, 1.0]
91	UnlockDoorsRight	[0.0, 1.0]
92	UnlockDoorsLeft	[0.0, 1.0]

Çizelge 3.1. (Devam Ediyor) GetControllerList() ID ve min-max değerleri.

93	OpenDoorsRight	[0.0, 1.0]
94	OpenDoorsLeft	[0.0, 1.0]
95	DoorsUnlockedRight	[0.0, 1.0]
96	DoorsUnlockedLeft	[0.0, 1.0]
97	DoorsOpenRight	[0.0, 1.0]
98	DoorsOpenLeft	[0.0, 1.0]
99	DoorsWait	[0.0, 1.0]
100	BlockDoors	[0.0, 1.0]
101	DoorControlBtn	[0.0, 1.0]
102	DoorControlBtn_l	[0.0, 1.0]
103	DoorControl	[0.0, 1.0]
104	Headlight1	[0.0, 1.0]
105	SpeedometerKPH	[0.0, 80.0]
106	Reverser	[-1.0, 1.0]
107	PantographControl	[0.0, 1.0]
108	PantographButton	[0.0, 1.0]
109	PantographUP	[0.0, 1.0]
110	PantographDOWN	[0.0, 1.0]
111	ThrottleAndBrake	[3.0, 89.0]
112	rest	[0.0, 1.0]
113	VirtualRegulator	[0.0, 1.0]
114	Regulator	[0.0, 1.0]
115	VirtualDynamicBrake	[0.0, 1.0]
116	DynamicBrake	[0.0, 1.0]
117	TrainBrakeControl	[0.0, 1.0]
118	brake_state	[0.0, 1.0]
119	DoorsOpenCloseLeft	[0.0, 1.0]
120	DoorsOpenCloseRight	[0.0, 1.0]
121	ReadyToGo	[0.0, 1.0]
122	control_emergency	[0.0, 1.0]

Çizelge 3.1. (Devam Ediyor) GetControllerList() ID ve min-max değerleri.

123	control_reset	[0.0, 1.0]
124	InLight	[0.0, 1.0]
125	door1_r_State	[0.0, 1.0]
126	door2_r_State	[0.0, 1.0]
127	door1_l_State	[0.0, 1.0]
128	door2_l_State	[0.0, 1.0]
129	DoorsInSections	[0.0, 14.0]
130	EngineSpeedRPM	[0.0, 1600.0]
131	EngineON	[0.0, 1.0]
132	DrivingSystemAMP	[0.0, 500.0]
133	DrivingSystemON	[0.0, 1.0]
134	Door1_R	[0.0, 1.0]
135	Door2_R	[0.0, 1.0]
136	Door1_L	[0.0, 1.0]
137	Door2_L	[0.0, 1.0]
138	Startup	[0.0, 1.0]
139	LanguageSelect	[0.0, 1.0]
140	C_Station	[0.0, 200.0]
141	Ansage_Linie	[0.0, 9.0]
142	Ansage_Richtung	[0.0, 4.0]
143	U6_JTP	[0.0, 1.0]
400	Latitude of Train	
401	Longitude of Train	
402	Fuel Level	
403	Is in a Tunnel?	
404	Gradient	
405	Heading	
406	Time of day hours	
407	Time of day minutes	
408	Time of day seconds	

Çizelge 3.2’de *SetControllerValue()* fonksiyonunda kullanılmakta olan anahtarlar, açıklamaları ve durumları yer almaktadır. Bu butonlardan bazıları kontrol paneli üzerinde, bazıları ise Buton MMI’da bulunmaktadır (Bkz. Şekil 2.10).

Çizelge 3.2. *SetControllerValue()* için kullanılan anahtarlar.

Anahtar	Açıklama	Durum
Horn	Zil butonu	0 harici bir rakam verilince zil sürekli çalınıyor.
EmergencyBrake	Acil Fren mantar butonu	0 harici bir rakam verilince acil fren aktif oluyor.
DriverKey	Kontak Anahtar	0 > Kabin pasif, 1 > Kabin aktif
Warnblinker	Flaşör butonu	0 harici bir rakam verilince Flaşör açıksa kapanıyor, kapalıysa açılıyor
Automat_ein	Ana Devre Kesici butonu	0 harici bir rakam verilince HSCB açılıyor
Automat_aus	Ana Devre Kesici butonu	0 harici bir rakam verilince HSCB kapanıyor
Weiche_l_Btn	Sol sinyal butonu (Buton MMI)	0 harici bir rakam verilince kapalıysa açılıyor, açıksa kapanıyor.
Weiche_r_Btn	Sağ sinyal butonu (Buton MMI)	0 harici bir rakam verilince kapalıysa açılıyor, açıksa kapanıyor.
Cab_light_Btn	Sürücü aydınlatma butonu (Buton MMI)	0 harici bir rakam verilince kapalıysa açılıyor, açıksa kapanıyor
WipersON	Silecek butonu (Buton MMI)	0 harici bir rakam verilince silecek normal açılıyor
WipersONslow	Silecek butonu (Buton MMI)	0 harici bir rakam verilince silecek yavaş mod açılıyor

Çizelge 3.2 (Devam Ediyor). *SetControllerValue()* için kullanılan anahtarlar.

Fwd_headlight_nah	Dış Aydınlatma butonu	0 harici bir rakam verilince park lambası açılır
Fwd_headlight_fern	Dış Aydınlatma butonu	0 harici bir rakam verilince farlar açılıyor
InteriorlightON	Yolcu Aydınlatma butonu (Buton MMI)	0 harici bir rakam verilince Yolcu Aydınlatma açılıyor
InteriorlightOFF	Yolcu Aydınlatma butonu (Buton MMI)	0 harici bir rakam verilince Yolcu Aydınlatma kapanıyor
InteriorlightAUTO	Yolcu Aydınlatma butonu (Buton MMI)	0 harici bir rakam verilince oto mod seçiliyor
BataryBtnON	24V Akü Anahtarı	0 harici bir rakam verilince batarya açılıyor
BataryBtnOFF	24V Akü Anahtarı	0 harici bir rakam verilince batarya kapanıyor
CloseDoors	Zorla kapatma butonu (Buton MMI)	0 harici bir rakam verilince kapılar kapanıyor
UnlockDoorsRight	Sağ Kapılar Serbest butonu	0 harici bir rakam verilince sağ kapılar serbest oluyor
UnlockDoorsLeft	Sol Kapılar Serbest butonu	0 harici bir rakam verilince sol kapılar serbest oluyor
OpenDoorsRight	sağ kapılar aç butonu	0 harici bir rakam verilince sağ kapılar açılıyor
OpenDoorsLeft	sol kapılar aç butonu	0 harici bir rakam verilince sağ kapılar açılıyor
Reverser	Yön seçimi butonu	-1 geri yön seçili 0 yön seçili değil 1 ileri yön seçili

Çizelge 3.2 (Devam Ediyor). *SetControllerValue()* için kullanılan anahtarlar.

PantographUP	Pantograf butonu	0 harici bir rakam verilince pantograf kalkıyor
PantographDOWN	İzolasyon Modu butonu	0 harici bir rakam verilince pantograf iniyor
ThrottleAndBrake	Sürüş Kolu	3 > %-100 , 89 > %100

Çizelge 3.3’de *GetControllerValue()* fonksiyonunda kullanılmakta olan geribildirimler bulunmaktadır, alınan bu değerler ile HMI ekranında sürücüler bilgilendirilmektedir.

Çizelge 3.3. *GetControllerValue()* için kullanılan geribildirimler.

Anahtar	Açıklama	Durum
DriverMirrors	Ayna	0 > kapalı, 0,7> açık
Automat	HSCB	0 > kapalı 1 > açık
Weiche_alarm	Flaşör	0 > kapalı 1 > açık
Weiche_r_Btn_1	Sağ Sinyal Lamba	Sağ sinyal lambaları yanınca 1 sönünce 0 oluyor. Sürekli 0-1 tekrarlıyor
Weiche_l_Btn_1	Sol Sinyal Lamba	Sol sinyal lambaları yanınca 1 sönünce 0 oluyor. Sürekli 0-1 tekrarlıyor
Weiche_l	Sol Sinyal	0 > kapalı 1 > açık
Weiche_r	Sağ Sinyal	0 > kapalı 1 > açık 0
Cab_light	Kabin aydınlatma	0 > kapalı 1 > açık
Pultlicht	Tepe lambası	0 > kapalı 1 > açık
WipersState	Silecek Durumu	0 > duruyor, 1 > Normal mod, 2 > Yavaş mod

Çizelge 3.3. (Devam Ediyor) *GetControllerValue()* için kullanılan geribildirimler.

Fwd_headlight_nah_1	Park lambası	0 > kapalı 1 > açık
Fwd_headlight_fern_1	Far	0 > kapalı 1 > açık
Interior_light	İç Aydınlatma	0 > kapalı 1 > açık
BatteryControl	Batarya	0 > kapalı 1 > açık
DoorsUnlockedRight	Sağ kapılar serbest	0 > serbest değil, 1 > serbest
DoorsUnlockedLeft	Sol kapılar serbest	0 > serbest değil, 1 > serbest
DoorsOpenRight	Sağ kapılar açık	0 > kapalı 1 > açık
DoorsOpenLeft	Sol kapılar açık	0 > kapalı 1 > açık
SpeedometerKPH	KM	Hız Bilgisi
PantographControl	Pantograf	Pantograf Yukarıda > 1, Pantograf aşağıda > 0
ReadyToGo	Sürüşe Hazır	1 > Sürüşe hazır, 0 > Sürüşe hazır değil
door1_r_State	Sağ kapı 1 açık	0 > kapalı 1 > açık
door2_r_State	Sağ kapı 2 açık	0 > kapalı 1 > açık
door1_l_State	Sol kapı 1 açık	0 > kapalı 1 > açık
door2_l_State	Sol kapı 2 açık	0 > kapalı 1 > açık
Latitude of Train	Enlem	Konum bilgisi
Longitude of Train	Boylam	Konum bilgisi
Is in a Tunnel?	Tünel	0 > Tünelde değil, 1 > Tünelde
Gradient	Eğim	kN değerleri için kullanılır.

Sürücülerin bilgilendirilmesi için HMI ekranı kullanılmaktadır, bu ekranda araçla alakalı bütün bilgiler ve hatalar sürücülere gösterilmektedir, ayrıca klima yönetimi de bu ekran üzerinden yapılmaktadır. Aşağıda şekil 3.4' de HMI ana ekranın görüntüsü yer almaktadır.



Şekil 3.4. HMI ana ekran görüntüsü.

4. SONUÇ ve ÖNERİLER

Yapılan araştırmalar sonucunda, dünya genelinde raylı sistem üreticisi ve araç üreticisi olmayan fakat ticari olarak bu tarz benzetim ortamı çalışması olan birçok firmaya rastlanmıştır. Türkiye’de raylı sistem alanında çalışan çok az sayıda firma mevcut olup, tam ve eş zamanlı raylı sistem simülatörü olan firma mevcut değildir. Bu eksikliğin giderilmesi için bu tez çalışması ile Türkiye’de ilk defa raylı sistem araçları üzerine tam ve eş zamanlı benzetim ortamı kurulmuştur. Kurulan bu benzetim ortamı ile araç üretilmeden sürücülerin eğitimleri verilerek kontrol paneli ekipmanlarının, insan makine ara yüzünün kullanımını öğrenmeleri sağlanacaktır. Bu tez çalışmasının ürünü olan DurSim kullanılarak yapılacak olan testler sonucunda ekipmanların yeterliliği, kullanılabilirliği, birbirleriyle olan uyumu belirlenebilecek olup yazılımların testleri birebir araç üzerinde yapılmış gibi dinamik olarak test edilebilecektir. Örneğin kurulan bu benzetim ortamı ile tramvay araçlarında kullanılacak olan hız sabitleme sistemi DurSim üzerinde test edilmektedir; Ayrıca sesli komut, sesli uyarı sistemleri de DurSim kullanılarak test edilmektedir. Bu sayede yazılım değişiklikleri araç üzerinde test ediliyormuş gibi DurSim üzerinde test edilmekte, yapılan testlerin sonucunda değişiklikler tramvay araçlarına uygulanmaktadır.

Ayrıca tez çalışmasının ürünü olan DurSim üzerinde sürücü eğitimleri verilerek zaman ve maliyet açısından getiri sağlanmaktadır. Bu sayede üretici firmalar tarafından üretilen raylı sistem araçlarının kabullerinin daha kısa sürede yapılacağı, dolayısıyla kısa sürede ürettikleri raylı sistem araçlarının ödemelerini daha kısa sürede alabileceği öngörülmektedir. İşletmeciler de satın almış oldukları raylı sistem araçlarından daha kısa sürede gelir elde etmeye başlayabileceklerdir. Ayrıca hizmet bekleyen insanların daha kısa sürede hizmet almaya başlamaları mümkün olabilecektir.

DurSim ile raylı sistem araçlarındaki CAN Bus haberleşme protokolü testleri yapılabilmektedir. Açılımı “Controller Area Network Bus” olan yani “Kontrol Alan Ağı Veri Yolu” olan CAN Bus başta otomobiller olmak üzere teknolojik araçlarda, maksimum verim ve güvenlik sağlanabilmesi için geliştirilen, otomotiv sektöründe kullanılan bir haberleşme protokolüdür. Tramvay araçlarında genellikle VCU, TCU, BCU, DCU gibi ekipmanlarda CAN Bus haberleşme protokolü kullanarak haberleşme sağlanmaktadır. CAN Bus haberleşme protokolündeki herhangi bir aksaklıkta araç acil sürüş moduna alınıp depoya çekilmektedir ve bundan dolayı işletmeler ve raylı sistem

yolcuları mağdur olmaktadır. Bu tez çalışmasının ürünü olan DurSim, bu mağduriyetlerin yaşanmaması için düşünülmüş ve bu çalışma kapsamında geliştirilmiştir.

DurSim ile raylı sistem araçlarındaki Ethernet haberleşme protokolü testleri yapılabilmektedir. Ethernet en bilinen ve en çok kullanılan ağ teknolojisidir. Tramvay araçlarında genellikle yolcu bilgilendirme sistemi, iklimlendirme sistemi Ethernet haberleşme protokolünü kullanmaktadır. Ayrıca araç kontrol ünitesinin insan makine arayüzü ile haberleşmesinde de Ethernet haberleşme protokolü kullanılmaktadır. Ethernet haberleşme protokolündeki herhangi bir aksama durumunda araç sürücüleri ve yolcular ciddi sıkıntılarla karşılaşmaktadır. Örneğin, klima sistemindeki bir aksaklık kışın araçların soğuk, yazın ise sıcak olmasına sebep olup yolcuları mağdur edebilir. Bu tez çalışmasının ürünü olan DurSim, bu gibi mağduriyetlerin yaşanmaması için düşünülmüş ve bu çalışma kapsamında geliştirilmiştir.

İlerleyen çalışmalarda tek ekranlı olan bu yapı çoklu ekran ile kurularak, benzetim ortamının görüntü kalitesi ve gerçekliği yükseltilebilir. Şekil 4.1’de çoklu ekranlı benzetim ortamına örnek bir çalışma gösterilmektedir.



Şekil 4.1. Çoklu ekran benzetim ortamı örneği (URL-6).

Ayrıca birebir araçta bulunan kabin ile birlikte kontrol paneli ve diğer bileşenleri ile bu sistem kurulursa benzetim ortamının gerçekçiliği yükseltilebilir. Şekil 4.2’de örnek bir çalışmadan bir görünüm verilmiştir.



Şekil 4.2. Gerçek kabinli benzetim ortamı örneği (URL-6).

Ayrıca birebir hat çizimleri yapılarak sürücülerin kendi kullanacakları hatta sürüş yapmaları sağlanabilir. Ayrıca sanal gerçeklik gözlüğü kullanılarak benzetimin gerçekliği ve hissiyatı yükseltilebilir. Ayrıca hareketli platforma benzetim ortamı kurulursa raylı sistem aracının sürüşünde; kalkışlarda, duruşlarda, virajlarda gerçeklik ve hissiyat artırılabilir. Ayrıca raylı sistem araçlarına bağlanarak yol kamera görüntüsü ekrana verilip çoklu ekran seçeneği ile ayna kameraların görüntüsü diğer ekranlara verilip, uzaktan raylı sistem araçları kullanılabilir, bu şekilde uzaktan araçların kullanılabilmesi öncelikli olarak yüksek hızlı ve kotasız sim kartlardan internet sağlanması gerekmektedir veya kullanılacak hatlarda WiFi mesh altyapısı kurulmalıdır. Diğer yandan veritabanına kaydedilen verilerden arşivleme yapıp, elde edilen büyük veriden meta sezgisel algoritmalar yardımı ya da farklı veri madenciliği teknikleri ile akıllı veriler elde edilebilir.

KAYNAKLAR

- Bayram, C S. (2018). *Toplu Taşımada Raylı Sistemlerin Önemi Ve Kentiçi Uygulamalarının Sağladığı Kazanımların Akçaray Üzerinden Değerlendirilmesi*. Yüksek Lisans Tezi, Kocaeli Üniversitesi, Fen Bilimleri Enstitüsü, Kocaeli.
- Dengiz, B., Benzetim ders notları, Başkent Üniversitesi, 2012
- Guo B., Mao Y., Hedge A. & Fang W. (2015) Effects of apparent image velocity and complexity on the dynamic visual field using a high-speed train driving simülator. *International Journal of Industrial Ergonomics*, 48, 99-109.
- Güven, İ. ve Ören T.İ. (2005). Fen Eğitiminde Benzetim. IETC 2005 – 5. Uluslararası Eğitim Teknolojileri Konferansı. Sakarya, Türkiye, Eylül 21-23.
- Öğüt, K. S., Evren, G. (2006). Türkiye’de Kentsel Raylı Sistemlerin Gerekliği Ve Uygulamada Dikkat Edilecek Konular. *Uluslararası Demiryolu Sempozyumu*. 13–16 Aralık.
- Ören, T.I. (2006). *Benzetim: Temel Kavramlar ve İlerlemeler*. Türkiye Bilişim Ansiklopedisi, Papatya Yayıncılık, İstanbul.
- Ören, T.I., Elçi, A., ve Köksal, A. (1985). *Benzetimin Temel Kavramları için Türkçe Terimler Önerisi*. Bilişim (Türkiye Bilişim Derneği Dergisi), sayı 21- 22 Eylül-Aralık 1985, s. 57-59.
- Patrik J. Simulation. In: Patric J, ed. Training: ResearchandPractice. London: AcademicPress, 2002: 487-508.
- Shah NH, Gor RV, Soni H. Simulations. In: Shah NH, Gor RV, Soni H eds. Operations Research. New Delphi: PrenticeHall of IndiaPrivate Limited, 2007: 486-488.
- Torun, M F. (2017). *İstanbulda Raylı Sistemler ve Kentiçi Raylı Sistemlerde Konfor Algısı*. Yüksek Lisans Tezi, Bahçeşehir Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul

KAYNAKLAR (Devam Ediyor)

URL-1 T.C. Millî Eğitim Bakanlığı. (2013). *Raylı Sistem Araçlarına Elektrik Enerjisinin Aktarılması*.

http://www.megep.meb.gov.tr/mte_program_modul/moduller_pdf/Rayl%C4%B1%20Sistem%20Ara%C3%A7lar%C4%B1na%20Elektrik%20Enerjisinin%20Aktar%C4%B1lmas%C4%B1.pdf. (Erişim Tarihi: 01.07.2019)

URL-2 T.C. Millî Eğitim Bakanlığı. (2013). *Raylı Sistem Araçları Aydınlatma ve İklimlendirme Sistemleri*.

http://www.megep.meb.gov.tr/mte_program_modul/moduller_pdf/Rayl%C4%B1%20Sistem%20Ara%C3%A7lar%C4%B1%20Ayd%C4%B1nlatma%20Ve%20%C4%B0klimlendirilme%20Sistemleri.pdf. (Erişim Tarihi: 01.07.2019)

URL-3 Türkiye İstatistik Kurumu (2018). *Adrese Dayalı Nüfus Kayıt Sistemi Sonuçları, 2018*. <http://www.tuik.gov.tr/PreHaberBultenleri.do?id=30709> (Erişim Tarihi: 01.07.2019)

URL-4 <http://www.ozelogretim.hacettepe.edu.tr/grup3/benzetim.php> (Erişim Tarihi: 01.07.2019)

URL-5 https://tr.wikipedia.org/wiki/Uygulama_programlama_arayüzü (Erişim Tarihi: 08.07.2019)

URL-6 <https://www.youtube.com/watch?v=Ogro9nLqFo0&t=23s> (Erişim Tarihi: 10.07.2019)

EKLER

EK-1

```

PrivatevoidGetir_Ekle()
{
    try
    {
        if(k4==true)
        {
            k4 = false;
            for (int i = 0; i < 500; i++)
            {
                gelenler[i] = RailDriver.GetControllerValue(i, 0);
            }
            gkmh= gelenler[104];
        }
        else
        {
            SingletramSpeed = RailDriver.GetControllerValue(104, 0);
            if (gkmh == tramSpeed)
            {
                kmsayac++;
                if(kmsayac == 21)
                {
                    for (int i = 0; i < 500; i++)
                    {
                        gelenler[i] = 0;
                    }
                    aktif = false;
                    kmsayac = 0;
                }
            }
        }
    }
}

```

```

else
{
    for (int i = 0; i < 500; i++)
    {
        gelenler[i] = RailDriver.GetControllerValue(i, 0);
    }
    gkmh = gelenler[104];
    aktif = true;
    kmsayac = 0;
}
}
con.Open();
string sorgu = "updateget_controller set AI = " + gelenler[0] +
", isA = " + gelenler[1] +
", IBIS_Play = " + gelenler[2] +
", IBIS_Kursanzeige = " + gelenler[3] +
", IBIS_IsSet = " + gelenler[4] +
", IBIS_Next = " + gelenler[5] +
", IBIS_On = " + gelenler[6] +
", IBIS_Sound = " + gelenler[7] +
", IBIS_Btn_1 = " + gelenler[8] +
", IBIS_Btn_2 = " + gelenler[9] +
", IBIS_Btn_3 = " + gelenler[10] +
", IBIS_Btn_4 = " + gelenler[11] +
", IBIS_RightBtn_1 = " + gelenler[12] +
", IBIS_RightBtn_2 = " + gelenler[13] +
", IBIS_RightBtn_3 = " + gelenler[14] +
", IBIS_RightBtn_4 = " + gelenler[15] +
", IBIS_Ziel = " + gelenler[16] +
", IBIS_Linie = " + gelenler[17] +
", IBIS_Curent_Station = " + gelenler[18] +
", IBIS_Enter = " + gelenler[19] +

```

```

", IBIS_Del = " + gelenler[20] +
", IBIS_9 = " + gelenler[21] +
", IBIS_8 = " + gelenler[22] +
", IBIS_7 = " + gelenler[23] +
", IBIS_6 = " + gelenler[24] +
", IBIS_5 = " + gelenler[25] +
", IBIS_4 = " + gelenler[26] +
", IBIS_3 = " + gelenler[27] +
", IBIS_2 = " + gelenler[28] +
", IBIS_1 = " + gelenler[29] +
", IBIS_0 = " + gelenler[30] +
", Horn = " + gelenler[31] +
", EmergencyBrake = " + gelenler[32] +
", dver = " + gelenler[33] +
", shtora = " + gelenler[34] +
", kreslo = " + gelenler[35] +
", fortl = " + gelenler[36] +
", fortr = " + gelenler[37] +
", DriverMirrors = " + gelenler[38] +
", DriverKey = " + gelenler[39] +
", DoorsClose = " + gelenler[40] +
", Controlinverse = " + gelenler[41] +
", TractiveEffort = " + gelenler[42] +
", Acceleration = " + gelenler[43] +
", HandBrake = " + gelenler[44] +
", AbsoluteSpeed = " + gelenler[45] +
", BrakeAmount = " + gelenler[46] +
", DistanceTravelled = " + gelenler[47] +
", Current = " + gelenler[48] +
", BrakeActive = " + gelenler[49] +
", Warnblinker = " + gelenler[50] +
", Luefter = " + gelenler[51] +

```

```
", Stoermeldung = " + gelenler[52] +  
", Automat_ein = " + gelenler[53] +  
", Automat_aus = " + gelenler[54] +  
", Automat = " + gelenler[55] +  
", Abfahrt = " + gelenler[56] +  
", Sand = " + gelenler[57] +  
", Kupplung = " + gelenler[58] +  
", Number_sections = " + gelenler[59] +  
", SectionNum = " + gelenler[60] +  
", RollbandUP = " + gelenler[61] +  
", RollbandDOWN = " + gelenler[62] +  
", Weiche_alarm_1 = " + gelenler[63] +  
", Weiche_alarm = " + gelenler[64] +  
", Weiche_r_Btn_1 = " + gelenler[65] +  
", Weiche_l_Btn_1 = " + gelenler[66] +  
", Weiche_l_Btn = " + gelenler[67] +  
", Weiche_l = " + gelenler[68] +  
", Weiche_r_Btn = " + gelenler[69] +  
", Weiche_r = " + gelenler[70] +  
", Cab_light_Btn = " + gelenler[71] +  
", Cab_light = " + gelenler[72] +  
", Pultlicht = " + gelenler[73] +  
", WipersON = " + gelenler[74] +  
", WipersONslow = " + gelenler[75] +  
", WipersState = " + gelenler[76] +  
", Wipers_interior = " + gelenler[77] +  
", Fwd_headlight_nah = " + gelenler[78] +  
", Fwd_headlight_nah_1 = " + gelenler[79] +  
", Fwd_headlight_fern = " + gelenler[80] +  
", Fwd_headlight_fern_1 = " + gelenler[81] +  
", Headlight = " + gelenler[82] +  
", Interior_light = " + gelenler[83] +
```

```

", InteriorlightON = " + gelenler[84] +
", InteriorlightOFF = " + gelenler[85] +
", InteriorlightAUTO = " + gelenler[86] +
", BataryBtnON = " + gelenler[87] +
", BataryBtnOFF = " + gelenler[88] +
", BatteryControl = " + gelenler[89] +
", CloseDoors = " + gelenler[90] +
", UnlockDoorsRight = " + gelenler[91] +
", UnlockDoorsLeft = " + gelenler[92] +
", OpenDoorsRight = " + gelenler[93] +
", OpenDoorsLeft = " + gelenler[94] +
", DoorsUnlockedRight = " + gelenler[95] +
", DoorsUnlockedLeft = " + gelenler[96] +
", DoorsOpenRight = " + gelenler[97] +
", DoorsOpenLeft = " + gelenler[98] +
", DoorsWait = " + gelenler[99] +
", BlockDoors = " + gelenler[100] +
", DoorControlBtn = " + gelenler[101] +
", DoorControlBtn_1 = " + gelenler[102] +
", DoorControl = " + gelenler[103] +
", SpeedometerKPH = " + gelenler[104] +
", Headlight1 = " + gelenler[105] +
", Reverser = " + gelenler[106] +
", PantographControl = " + gelenler[107] +
", PantographButton = " + gelenler[108] +
", PantographUP = " + gelenler[109] +
", PantographDOWN = " + gelenler[110] +
", ThrottleAndBrake = " + gelenler[111] +
", rest = " + gelenler[112] +
", VirtualRegulator = " + gelenler[113] +
", Regulator = " + gelenler[114] +
", VirtualDynamicBrake = " + gelenler[115] +

```

```

", DynamicBrake = " + gelenler[116] +
", TrainBrakeControl = " + gelenler[117] +
", brake_state = " + gelenler[118] +
", DoorsOpenCloseLeft = " + gelenler[119] +
", DoorsOpenCloseRight = " + gelenler[120] +
", ReadyToGo = " + gelenler[121] +
", control_emergency = " + gelenler[122] +
", control_reset = " + gelenler[123] +
", InLight = " + gelenler[124] +
", door1_r_State = " + gelenler[125] +
", door2_r_State = " + gelenler[126] +
", door1_l_State = " + gelenler[127] +
", door2_l_State = " + gelenler[128] +
", DoorsInSections = " + gelenler[129] +
", EngineSpeedRPM = " + gelenler[130] +
", EngineON = " + gelenler[131] +
", DrivingSystemAMP = " + gelenler[132] +
", DrivingSystemON = " + gelenler[133] +
", Door1_R = " + gelenler[134] +
", Door2_R = " + gelenler[135] +
", Door1_L = " + gelenler[136] +
", Door2_L = " + gelenler[137] +
", Startup = " + gelenler[139] +
", LanguageSelect = " + gelenler[139] +
", C_Station = " + gelenler[140] +
", Ansage_Linie = " + gelenler[141] +
", Ansage_Richtung = " + gelenler[142] +
", U6_JTP = " + gelenler[143] +
", LatitudeofTrain = " + gelenler[400] +
", LongitudeofTrain = " + gelenler[401] +
", FuelLevel = " + gelenler[402] +
", Tunnel = " + gelenler[403] +

```

```

        ", Gradient = " + gelenler[404] +
        ", Heading = " + gelenler[405] +
        ", Hours = " + gelenler[406] +
        ", Minutes = " + gelenler[407] +
        ", Seconds = " + gelenler[408] +
        " WHERE id_num = 1 ";
        MySqlCommandkmt = new MySqlCommand(sorgu, con);
        kmt.ExecuteNonQuery();
        k1 = true;
        label501.ForeColor = Color.Lime;
        label501.Text = "OK";
        label501.Update();
    }
    catch (Exceptionerr)
    {
        if (k1 == true)
        {
            label501.ForeColor = Color.Red;
            label501.Text = "NOK";
            label501.Update();
            HataLoglama(err.Message + " get_controller tablosuna veri
            ekleme yapılamadı..");
            k1 = false;
        }
    }
    con.Close();
}

```

EK-2

PrivatevoidGetir_Gonder()

```
{
    try{
        con.Open();
        string sql = "SELECT * FROM set_controller WHERE id_num=1 ";
        MySqlCommand cmd = new MySqlCommand(sql, con);
        MySqlDataReader rdr = cmd.ExecuteReader();
        if (kontrol == true)
        {
            Vargdt = new DataTable();
            gdt.Load(rdr);
            var gstringArr = gdt.Rows[0].ItemArray.Select(x => x).ToArray();
            for (int i = 0; i < 144; i++)
            {
                gecici[i] = Convert.ToSingle(gstringArr[i + 2]);
                RailDriver.SetControllerValue(i, gecici[i]);
            }
            kontrol = false;
        }
        else
        {
            var dt = new DataTable();
            dt.Load(rdr);
            var stringArr = dt.Rows[0].ItemArray.Select(x => x).ToArray();
            if (aktif == true)
            {
                for (int i = 0; i < 144; i++)
                {
                    array[i] = Convert.ToSingle(stringArr[i + 2]);
                    if (array[i] != gecici[i])
                    {
```

```

        RailDriver.SetControllerValue(i, array[i]);
        gecici[i] = array[i];
    }
}
else
{
    for (int k = 0; k < 144; k++)
    {
        gecici[k] = 0;
    }
}
}
k2 = true;
label501.ForeColor = Color.Lime;
label501.Text = "OK";
label501.Update();
}

catch (Exception err){
    if (k2 == true)
    {
        label501.ForeColor = Color.Red;
        label501.Text = "NOK";
        label501.Update();
        HataLoglama(err.Message + " set_controller tablosuna veri
ekleme yapılamadı..");
        k2 = false;
    }
}
con.Close();
}

```

ÖZ GEÇMİŞ

Kişisel Bilgiler

Adı Soyadı : Osman YİĞİT

Doğum Yeri ve Tarihi : Nilüfer / 1993



Eğitim Durumu

Lisans Öğrenimi : Sakarya Üniversitesi Bilgisayar Mühendisliği İÖ

Bildiği Yabancı Diller : İngilizce (İyi), Almanca(Orta)

İş Deneyimi

Stajlar : T.C. Bursa Uludağ Üniversitesi Sağlık Uygulama ve Araştırma Merkezi Hastanesi

Prestij Yazılım

Projeler : 1501-TÜBİTAK Endüstri 4.0 Kapsamında Raylı Sistemler

Tam ve Eş Zamanlı Simülasyon ve Koordinasyon Projesi, Proje Lideri

Çalıştığı Kurumlar : Durmazlar Makine Raylı Sistemler ARGE Yazılım Uzmanı
(2016 – Devam Ediyor)

İletişim

Adres : Demirci Mah. Gökhan Sok. No:4 D:5 Nilüfer/BURSA

E-Posta Adresi : osmnygt23@gmail.com

Tarih: 21/08/2019