

T.C.  
ONDOKUZ MAYIS ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ



AKILLI ŞEBEKELERDE KULLANILAN SCADA SİSTEMLERİNİN YAZILIM VE  
DONANIMSAL OLARAK UYGULAMASI

ABDULKADİR GÖZÜOĞLU

YÜKSEK LİSANS TEZİ

T.C.  
ONDOKUZ MAYIS ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ

YÜKSEK LİSANS TEZİ

AKILLI ŞEBEKELERDE KULLANILAN SCADA SİSTEMLERİNİN YAZILIM VE  
DONANIMSAL OLARAK UYGULAMASI

ABDULKADİR GÖZÜOĞLU

ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI

SAMSUN  
2018

Her Hakkı Saklıdır.

## TEZ ONAYI

Abdulkadir Gözüođlu tarafından hazırlanan “Akıllı Şebekelerde Kullanılan SCADA Sistemlerinin Yazılımsal ve Donanımsal Uygulaması” adlı tez çalışması 18/12/2018 tarihinde aŐađıdaki jüri tarafından Ondokuz Mayıs Üniversitesi Fen Bilimleri Enstitüsü Elektrik Elektronik Mühendisliđi Anabilim Dalı’nda **Yüksek Lisans Tezi** olarak kabul edilmiŐtir.

**Danışman** Prof. Dr. Okan ÖZGÖNENEL  
Elektrik Elektronik Mühendisliđi Anabilim Dalı

### Jüri Üyeleri

**Başkan** Prof. Dr. Okan ÖZGÖNENEL  
Ondokuz Mayıs Üniversitesi  
Elektrik Elektronik Mühendisliđi Anabilim Dalı

**Üye** Doç. Dr. Mustafa AKTAŐ  
Ondokuz Mayıs Üniversitesi  
Elektrik Elektronik Mühendisliđi Anabilim Dalı

**Üye** Doç. Dr. Ünal Kurt  
Amasya Üniversitesi  
Elektrik Elektronik Mühendisliđi Anabilim Dalı

**Yukarıdaki sonucu onaylarım. .../.../2018**

**Prof. Dr. Bahtiyar ÖZTÜRK**  
**Enstitü Müdürü**

## ETİK BEYAN

Ondokuz Mayıs Üniversitesi Fen Bilimleri Enstitüsü tez yazım kurallarına uygun olarak hazırladığım bu tez içindeki bütün bilgilerin doğru ve tam olduğunu, bilgilerin üretilmesi aşamasında bilimsel etiğe uygun davrandığımı, yararlandığım bütün kaynakları atıf yaparak belirttiğimi beyan ederim.

18/12/2018

Abdulkadir GÖZÜOĞLU

## ÖZET

Yüksek Lisans Tezi

### AKILLI ŞEBEKELERDE KULLANILAN SCADA SİSTEMLERİNİN YAZILIM VE DONANIMSAL OLARAK UYGULAMASI

Abdulkadir Gözüoğlu

Ondokuz Mayıs Üniversitesi

Fen Bilimleri Enstitüsü

Elektrik Elektronik Mühendisliği Anabilim Dalı

Danışman: Prof. Dr. Okan Özgönenel

Teknolojideki gelişmeler her alanda olduğu gibi elektrik şebekelerinin yönetilme şekillerinde de yenilikler getirmektedir. Gerçekleştirilen düzenleme ve iyileştirmelere bağlı olarak Akıllı Şebeke sistemlerine geçişler yapılmaya çalışılmaktadır. Elektrik şebekelerinin geniş bir çevreye yayılmış olması ve karmaşıklığından dolayı akıllı şebeke uygulamalarının geliştirilmesi ve test edilmesi için son yıllarda benzetim ortamları oluşturulmaktadır. Ancak bahsedilen laboratuvarların kurulması için ortaya çıkan yüksek maliyetler özellikle yeni başlayanlar ve öğrenciler için bir engel oluşturmaktadır. Bu çalışmada akıllı şebeke modellerinde kullanılabilecek çeşitli benzetimlerin uygulanabileceği düşük maliyetli eğitim ortamının oluşturulması hedeflenmiştir.

Aralık 2018, 87 sayfa

Anahtar Kelimeler: Akıllı şebekeler, Scada, RTU, Veri tabanı, Uzaktan denetim ve izleme, Güç sistemleri benzetimi, OG şebeke izleme ve denetim

## **ABSTRACT**

Master's Thesis

### **SOFTWARE AND HARDWARE APPLICATION FOR SCADA SYSTEM USED IN SMART GRIDS**

Abdulkadir Gozuoglu

Ondokuz Mayıs University  
Graduate School of Sciences

Department of Electrical and Electronics Engineering

Supervisor: Prof. Dr. Okan Ozgonenel

The improvements on technology brings innovation on managing electrical networks as it is in all areas. According to the regulations and restorations, most companies tries to implement Smart Grids on existing networks. Since the electrical networks spread into wide area and due to complexity of the networks, many Smart Grid laboratories have appeared over the years to provide simulation environments for testing and improving of the modern networks. However, high cost values for setting up such a laboratory becomes a barrier for new comers and students. In this study, we propose to establish a hardware and software model of Smart Grids that is set up with low cost equipment and allows testing many scenarios especially for training purpose.

December 2018, 87 pages

Keywords: Smart grids, Scada, RTU, Database, Remote control and monitoring, Power System Simulator, MV network monitor and control.

## ÖNSÖZ VE TEŞEKKÜR

Akademik eğitim sürecimin bir üst noktası olan yüksek lisans tez çalışmalarım boyunca yardım ve desteğini benden esirgemeyen danışman hocam Sayın Prof. Dr. Okan Özgönel'e teşekkürü bir borç bilirim.

Ayrıca eski danışman hocam Sayın Dr. Öğretim Üyesi Muammer Özdemir'e yardımlarından dolayı teşekkür eder Allah'tan rahmet dilerim.

Yüksek lisans eğitimim boyunca benden desteklerini esirgemeyen biricik eşim ve canım oğluma çok teşekkür ederim.

Beni bugünlere getirmek için hiçbir fedakârlıktan kaçınmayan babama, anneme ve kardeşlerime sonsuz teşekkür ederim.

Aralık 2018, Samsun

Abdulkadir Gözüoğlu

## İÇİNDEKİLER DİZİNİ

ÖZET.....	i
ABSTRACT.....	ii
ÖNSÖZ VE TEŞEKKÜR.....	iii
İÇİNDEKİLER DİZİNİ.....	iv
SİMGELER VE KISALTMALAR.....	vi
ŞEKİLLER DİZİNİ.....	vii
1. GİRİŞ.....	1
2. KURAMSAL TEMELLER VE KAYNAK ÖZETLERİ.....	3
3. MATERYAL VE YÖNTEM.....	5
3.1. Donanımsal Sistemler.....	6
3.1.1. Mikro denetleyici (Arduino UNO R3).....	6
3.1.2. I <sup>2</sup> C iletişim protokolü.....	8
3.1.3. Transformatör.....	10
3.1.4. AC akım algılayıcı.....	11
3.1.5. AC gerilim algılayıcı.....	13
3.1.6. Basmalı Buton.....	14
3.1.7. Gerçek Zaman Modülü – DS3231.....	14
3.1.8. 4x20 LCD Ekran.....	15
3.2. Yazılımsal Sistemler.....	17
3.2.1. Processing 3.....	18
3.2.2. MySQL veri tabanı.....	19
4. AC GÜÇ KURAMI.....	21
4.1. Direnç Yükler.....	21
4.2. Tepkin Yükler.....	22
4.3. Etkin Güç, Tepkin Güç ve Görünür Güç.....	23
4.4. Güç Akışının Yönünün Belirlenmesi.....	25
4.5. Güç Sistemlerinde Matematiksel Hesaplamalar.....	26
4.6. Güç Sistemindeki Büyüklüklerin Arduino ile Hesaplanması.....	27
4.6.1. Arduino ile etkin gücün hesaplanması.....	27
4.6.2. Arduino ile RMS gerilimin hesaplanması.....	28
4.6.3. Arduino ile RMS akımın hesaplanması.....	29
4.6.4. Arduino ile görünür gücün hesaplanması.....	29
4.6.5. Arduino ile güç faktörünün hesaplanması.....	29
4.6.6. Arduino “Emonlib” kütüphanesi.....	30
5. AKILLI ŞEBEKE SİMÜLASYON SENARYOSU.....	31
5.1. RTU – Ana Denetleyici.....	32
5.2. Röle Modülü – Uydu Denetleyici 1.....	37
5.3. Röle Modülü – Uydu Denetleyici 2.....	40
5.4. Röle Modülü – Uydu Denetleyici 3.....	43
5.5. Röle Modülü – Uydu Denetleyici 4.....	46
5.6. GSM Modülü – Uydu Denetleyici 5.....	49
5.7. Gerçek Zaman Modülü – DS3231 Çalışması ve Görevi.....	51
5.8. Güç İşaretinin Dağıtılması ve Ölçülmesi.....	52
5.9. Donanımsal ve Yazılımsal Sistemler Arasındaki İletişim.....	53
5.10. Scada Yazılımında Verilerin İşlenmesi.....	54
5.11. Ölçüm Verilerinin Arayüzde Görüntülenmesi ve Denetim İşaretleri.....	55
5.12. Eğitim Takımında Senaryoların İşletilmesi.....	59



5.12.1. Gerilim yükselmesi senaryosu .....	60
5.12.2. Gerilim düşüşü senaryosu .....	60
5.12.3. Kısa devre senaryosu .....	60
5.12.4. Elektrik kesintisi senaryosu .....	61
5.13. Alarm Üretilmesi .....	61
5.14. Verilerin Dönemsel Olarak Veri Tabanına Kaydedilmesi.....	62
6. TEZ ÇALIŞMASININ ELEKTRİK ŞEBEKELERİNE UYGULANMASI .....	64
7. BULGULAR VE TARTIŞMA .....	66
8. SONUÇ VE ÖNERİLER .....	69
KAYNAKLAR .....	72
ÖZGEÇMİŞ .....	75



## SİMGELER VE KISALTMALAR

### SİMGELER

∅ Faz çısı

### KISALTMALAR

COM	Seri İletişim Bilgisayar Bağlantı Noktası
DSP	Sayısal İşaret İşleme
FFT	Hızlı Fourier Dönüşümü
I <sup>2</sup> C	İki Kablolu İletişim Şekli
ICSP	Devre İçerisinde Seri Programlama
LOT	Nesnelerin İnterneti
MV	Orta Gerilim
OG	Orta Gerilim
PWM	Darbe Genişlik Modülasyonu
RMS	Örneklenen İşaretlerin Kareleri Toplamının Ortalamasının Karekökü
RTC	Gerçek Zaman Saati
RTU	Uzak Denetim Cihazı
RX	Seri İletişim Veri Alım Ucu
SCADA	Danışmalı Denetim ve Veri Toplama Sistemi
SCL	I <sup>2</sup> C İletişimi Zamanlama Hattı
SDA	I <sup>2</sup> C İletişimi Veri Hattı
SPI	Seri Çevresel Arayüzü
TTL	Mikro Denetleyicilerde 5V Gerilim Seviyesi
TX	Seri İletişim Veri Gönderim Ucu
UART	Seri İletişim
USB	Evrensel Seri Veri Yolu

## ŞEKİLLER DİZİNİ

Şekil 3.1.	Benzetim için genel çalışma algoritması .....	5
Şekil 3.2.	Donanımsal genel yapı.....	6
Şekil 3.3.	Arduino UNO R3 modeli.....	7
Şekil 3.4.	Örnek bir I <sup>2</sup> C bara bağlantısı .....	8
Şekil 3.5.	İki Arduino UNO I <sup>2</sup> C bağlantıları .....	9
Şekil 3.6.	Transformatör.....	10
Şekil 3.7.	Örnek olarak kullanılan dağıtım şebekesi .....	11
Şekil 3.8.	AC akım algılayıcı .....	12
Şekil 3.9.	AC akım algılayıcı bağlantı diyagramı.....	12
Şekil 3.10.	AC gerilim ölçüm devresi.....	13
Şekil 3.11.	Basmalı buton .....	14
Şekil 3.12.	DS3231 RTC modülü .....	15
Şekil 3.13.	4x20 LCD ekran.....	15
Şekil 3.14.	LCD ekran I <sup>2</sup> C Modülü.....	16
Şekil 3.15.	Arduino I <sup>2</sup> C Modülü ve LCD ekran bağlantısı.....	17
Şekil 3.16.	Genel Scada arayüzü.....	18
Şekil 3.17.	MySQL veri tabanı .....	20
Şekil 4.1.	Direnç yüklerde akım gerilim faz ilişkisi .....	21
Şekil 4.2.	Tepkin yüklerde akım gerilim ilişkisi.....	22
Şekil 4.3.	Doğrusal olmayan yükler için akım gerilim işareti .....	24
Şekil 4.4.	Elektrik enerjisinin üretilmesi durumunda akım gerilim ilişkisi.....	25
Şekil 4.5.	Arduino ile etkin gücün hesaplanması.....	28
Şekil 4.6.	Arduino ile rms gerilimin hesaplanması.....	28
Şekil 4.7.	Arduino ile rms akımın hesaplanması .....	29
Şekil 4.8.	Arduino ile görünür gücün hesaplanması .....	29
Şekil 4.9.	Arduino ile güç faktörünün hesaplanması .....	29
Şekil 5.1.	Donanımsal bölüm genel bağlantı çizelgesi .....	31
Şekil 5.2.	Donanımsal bölüm mikro denetleyici genel bağlantı çizelgesi.....	32
Şekil 5.3.	RTU ana-uydu bağlantı çizelgesi.....	33
Şekil 5.4.	Uydu denetleyiciden elde edilen veriler .....	34
Şekil 5.5.	RTU tarafından dönemsel olarak rölelerden talep edilen ölçüm verilerinin Arduino arayüzündeki yazılım biçimi .....	35
Şekil 5.6.	Seri iletişim yolu ile aktarılacak verinin tek harf dizini biçiminde düzenlenmesi.....	36
Şekil 5.7.	<i>Processing</i> ortamında gelen verinin split edilmesi.....	36
Şekil 5.8.	Anahtarlama yapısındaki butonlar ve Arduino bağlantısı.....	38
Şekil 5.9.	Anahtarlama yapısı gösterge LED'leri ve Arduino bağlantısı .....	39
Şekil 5.10.	Arduino röle modülleri bağlantısı.....	40
Şekil 5.11.	Anahtarlama yapısındaki butonlar ve Arduino bağlantısı .....	41
Şekil 5.12.	Anahtarlama yapısı gösterge LED'leri ve Arduino bağlantısı .....	42
Şekil 5.13.	Arduino röle modülleri bağlantısı.....	43
Şekil 5.14.	Anahtarlama yapısındaki butonlar ve Arduino bağlantısı .....	44
Şekil 5.15.	Anahtarlama yapısı gösterge LED'leri ve Arduino bağlantısı .....	45
Şekil 5.16.	Arduino röle modülleri bağlantısı.....	46
Şekil 5.17.	Anahtarlama yapısındaki butonlar ve Arduino bağlantısı .....	47
Şekil 5.18.	Anahtarlama yapısı gösterge LED'leri ve Arduino bağlantısı .....	48
Şekil 5.19.	Arduino röle modülleri bağlantısı.....	49

Şekil 5.20. Cinterion MC35i GSM modem .....	50
Şekil 5.21. GSM modem RS232 pinleri (Anonymous, 2008) .....	50
Şekil 5.22. RS232-TTL dönüştürücü .....	51
Şekil 5.23. DS3231- RTC Modülü bağlantı çizelgesi .....	51
Şekil 5.24. Trafo karakteristik tablosu .....	52
Şekil 5.25. Model şebekede kullanılan örnek yük ve dirençler .....	53
Şekil 5.26. <i>Processing</i> ve Arduino bağlantı çizelgesi .....	53
Şekil 5.27. Scada arayüz genel görünüm .....	55
Şekil 5.28. Scada arayüzü seri iletişim parametre seçim bölümü .....	56
Şekil 5.29. Scada arayüzü bağlantının gerçekleştiği durum .....	57
Şekil 5.30. Scada arayüzü durum çubuğu bilgileri .....	58
Şekil 5.31. Scada arayüzü veri tabanı işlemleri .....	59
Şekil 5.32. MySQL veri tabanı ve trafo tabloları .....	62
Şekil 5.33. Veri tabanı tablo içeriği .....	63
Şekil 6.1. Tez çalışmasının gerçek şebekelere uygulanması .....	64



## 1. GİRİŞ

Teknolojideki gelişmeler her alanda sürekli bir iyileşme ve hayatı kolaylaştırma olarak ortaya çıkmaktadır. Bu gelişmeler düşünüldüğü zaman elektriğin ilk sıralarda yer aldığı görülmektedir. Elektrik şebekelerinde gerçekleştirilen düzenleme ve iyileştirmeler sonucunda Akıllı Şebeke sistemlerine geçişler yapılmaya çalışılmaktadır. Böylece şebeke durumunun ve tüm saha donanımlarının uzaktan gözlem ve denetimi sağlanabilmektedir.

Geleneksel yöntemde Elektrik ölçümünü yapan cihazlar tüketilen enerjiyi hesaplamak için sürekli akım ve gerilim ölçümleri yaparak kilowatt-saat cinsinden elde ettikleri ölçümleri kendi ekranlarında görüntüleme işini yaparlar (Rahman vd, 2015). Bu ölçümleri kaydetme ve belli bir arayüzde görüntüleme gibi özellikler bulunmaz. Ayrıca anahtarlama yapılarının durumlarının değiştirilmeleri de el ile gerçekleştirilmektedir. Otomatik veya uzaktan manevra özelliği bulunmamaktadır.

Akıllı Şebeke tüm bağlı olan donanımların hareket ve davranışlarının akıllı bir şekilde bütünleşmesine olanak tanıyan, elektriğin sürekli, ekonomik, güvenli ve verimli olarak iletilmesini sağlayan elektrik şebeke formu olarak ifade edilebilir (Chren vd, 2016; Clastres, 2011).

Modern Akıllı Şebekelerin çeşitli amaçlarının olmasının yanı sıra çoğu zorluklarla da karşılaşma durumları bulunabilmektedir, bunlar şebekenin çoklu-disiplinler doğası, çeşitli güç donanımlarından verilerin toplanarak veri analizlerinin yapılması ve güç şebekesinin akıllılık seviyesini arttırmak şeklinde sıralanabilir (Schvarcbacher ve Rossi, 2017). Ayrıca şebekede yer alan farklı unsurlar arasında iletişim kurulması temel aşamalar arasında yer almaktadır, bu durumda şebekenin farklı kısımları için çalışanların eğitilmesi en önemli adımlardan biri olacaktır (Strasser vd, 2014). Bu nedenle son zamanlardaki çalışmalar, gerçek ve büyük ölçekli şebekelerin yazılım ve donanımlarının test edilebileceği ve denetimli ortamlarda öğrencilerin de eğitilebileceği Akıllı Şebeke laboratuvarlarının ortaya çıkması şeklindedir (Andreadou vd, 2016).

Ancak ortalama bir Akıllı Şebeke laboratuvarının kurulma maliyeti 2M€ seviyelerinde olabilirken daha gelişmiş laboratuvarların kurulması 30M€ tutarlarını

bulabilmektedir (Andreadou vd, 2016). Bu seviyedeki maliyetler eğitim amaçlı olarak kurulması planlanan Akıllı Şebeke laboratuvarları için ciddi bir engel oluşturmaktadır (Schvarcbacher ve Rossi, 2017). Bu nedenle bu çalışmada öğrencilerin test amaçlı kullanabilecekleri ve çeşitli senaryoları uygulayabilecekleri, düşük maliyetli ve sanallaştırılmış bir modelin oluşturulması önerilmektedir (Schvarcbacher ve Rossi, 2017).

Oluşturulan benzetim ortamı yazılımsal ve donanımsal olarak iki aşamadan oluşmaktadır. Donanımsal çözüm olarak Arduino ortamı kullanılmıştır. Arduino çok yaygın bir kullanım alanına sahip olması, çeşitli kütüphanelerinin bulunması ve bilgisayar ile seri olarak haberleşebildiğinden dolayı tercih edilmiştir. Yazılım ortamı olarak ise *Processing 3* tercih edilmiştir. *Processing* uygulaması esnek bir yazılıma sahip olduğundan ve altyapısının Arduino ile benzerlik ve uyum gösterdiğinden dolayı çeşitli senaryoların uygulanmasına olanak sağlamaktadır. Çalışmamızda oluşturduğumuz ilk aşama, donanımsal olarak modellenen şebekeye ait AC analog ölçümlerin ve sayısal bilgilerin I<sup>2</sup>C (Inter-integrated Circuit) iletişim protokolü ile RTU (Remote Terminal Unit)'ya taşınması şeklindedir. İkinci aşama ise bu verilerin seri iletişim yolu ile bilgisayara ve böylece Scada (Supervisory Control and Data Acquisition) yazılımına aktarılması, arayüzde anlık olarak gözlemlenmesi, denetlenmesi şeklindedir. Bu iki aşamanın elde edilmesinden sonra çeşitli senaryoların çalıştırılabileceği bir ortam elde edilmiş olur.

Çalışmamızda Bölüm 2 Akıllı Şebeke'nin test edilebilmesi için kullanılan düşük maliyetli laboratuvarın oluşturulması için daha önce yapılmış çalışmaları içermektedir. Bölüm 3'te kullanılan yazılım ve donanımsal yapıları ve genel mimari hakkında bilgi vermektedir. Bölüm 4 AC güç kuramını açıklamaktadır. Bölüm 5'te Akıllı Şebeke benzetimi için oluşturulmuş olan çeşitli senaryolar ve uygulamalar hakkında bilgi verilmiştir. Bölüm 6 ve 7 elde ettiğimiz bulgular ve sonuçlar üzerindeki değerlendirmelerimizi göstermektedir.

## 2. KURAMSAL TEMELLER VE KAYNAK ÖZETLERİ

Akıllı Şebeke sistemlerinin yaygınlaşmasına bağlı olarak bu sistemler için gerekli olan çalışan eğitimi ihtiyacı da aynı oranda artmaktadır. Çalışanların yeni oluşumlara uyarlanabilmeleri için de ilgili konularda eğitilmeleri önem arz etmektedir. Bu durum düşük maliyetli akıllı şebeke laboratuvarlarına olan ilgiyi de arttırmaktadır.

Akıllı Şebeke modellemelerinde donanımsal olarak *Raspberry Pi* ve *Arduino* geniş kullanım alanı bulmaktadır, bunlar çoğunlukla Akıllı Sayaç ve Müşteri Güç Tüketimi için uzaktan ölçüm yapan cihazları içermektedir (Bonganay vd, 2014). Ayrıca çok etkenli yöntemlerde şebekenin kendi kendini iyileştirmesi durumlarını test etmek (Eriksson vd, 2015) veya ikincil şebekelerdeki düzenin yeniden sağlanması testlerinin yapılması için kullanılmaktadır (Rodríguez vd, 2016).

Aurilio vd (2014) *Raspberry Pi* tabanlı bir veri yoğunlaştırıcı oluşturarak güç hattı dağıtım noktası üzerinden verilerin taşınmasıyla güç şebekesinin uzaktan yönetilmesi ve bağlı olan yüklerin izlenmesi için düşük maliyetli bir çözüm elde etmişlerdir (Aurilio vd, 2014).

Bu çalışmada sahadan (dağıtım şebekesi) gerçek veriler alınmamaktadır. Onun yerine donanımsal bir ortam oluşturup saha verilerinin de benzetimi sağlanmıştır. Bu sayede çeşitli senaryoların bilgisayar benzetimleri yapılırken sahaya bağımlı kalmadan ama sahadaki verilere çok benzer işaretlerin alınması hedeflenmiştir.

Armendariz vd (2014) *Raspberry Pi* ve *Arduino* kullanarak gerçek zamanlı güç sistemleri benzetimi (*Opal-RT*) ve iletişim şebekesi benzetim (*OPNET*) ortamlarını kullanarak otomatik ölçüm yapabilmek için düşük maliyetli bir ortam geliştirmiştir.

Nannen vd (2015) *Mosaik* sistem ile bütünleşmeyi sağlayabilecek *Raspberry Pi* tabanlı bir çözüm önerisinde bulunmuşlardır. Burada *Raspberry Pi* uzak bir noktada yer alan fotovoltaik dönüştürücüsünü denetlemek için bir geçit olarak kullanılmıştır (Nannen vd, 2015).

Bizim çalışmamız belirtilen daha önceki çalışmalardan çok daha farklı bir yapıdadır. Temelde herhangi bir laboratuvar cihazına ve sahadaki gerçek verilere bağımlı olmayan ve pek çok senaryo için esnek olabilecek bir ortam oluşturmaktır.

Mikro denetleyici olarak kullandığımız *Arduino* çok esnek bir yazılıma ve çok sayıda uyarlanmış genişletme donanımına sahiptir. Yazılım ortamı olarak

kullandığımız *Processing* de çoęu uygulama için kolaylık sağlamaktadır. *Windows* ortamında hazırlanan formlar sayesinde verilerin anlık olarak gözlemlenmesi, matematiksel işlemler ve veri tabanına kaydedilmesi gibi çoęu işlemin gerçekleştirilebilmesine olanak vermektedir.

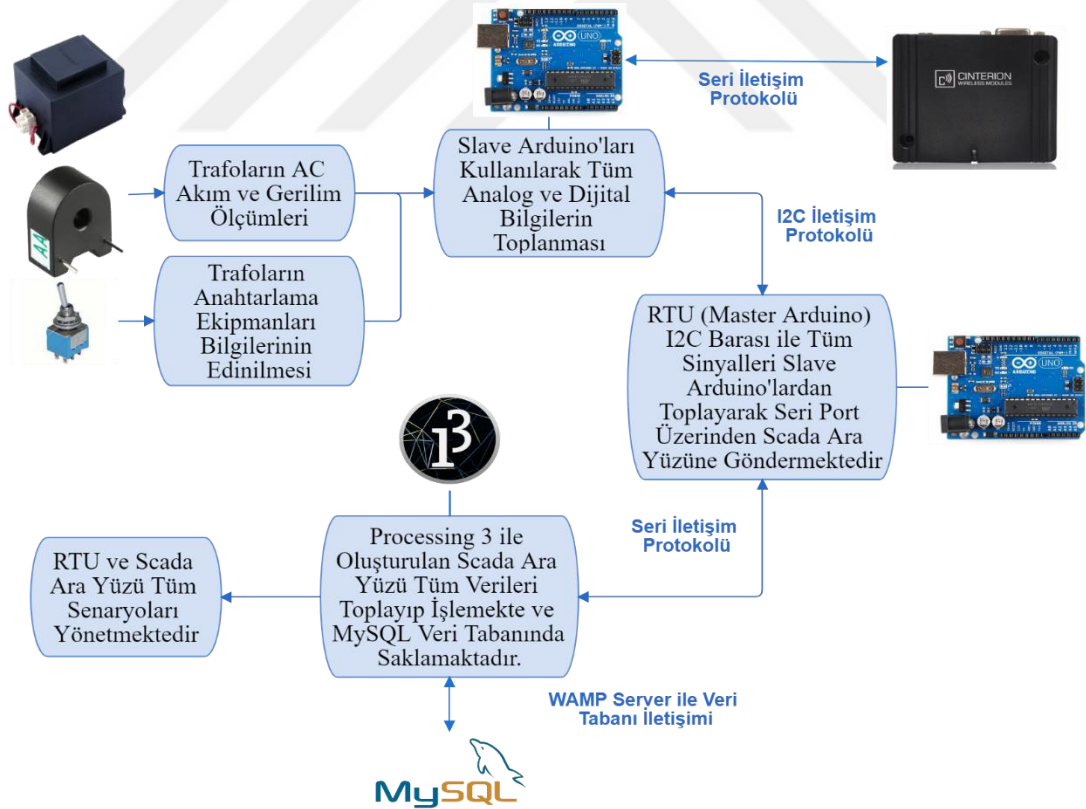




### 3. MATERYAL VE YÖNTEM

Bu çalışmada benzetim ortamı yazılımsal sistemler ve donanımsal sistemler olmak üzere iki kısımdan oluşmaktadır. Donanımsal bölüm *Arduino* tabanlı olarak tasarlanmıştır. Temel olarak sahadan gelmesi gereken verilerin oluşturulması için kullanılmaktadır. Böylece sahadan gerçek verilerin alınmasına ihtiyaç olmadan yazılım ortamının ihtiyaç duyduğu veriler temin edilmiş olacaktır. Ayrıca uygulanacak senaryoların yönetimi yine *Arduino* tarafından gerçekleştirilmektedir.

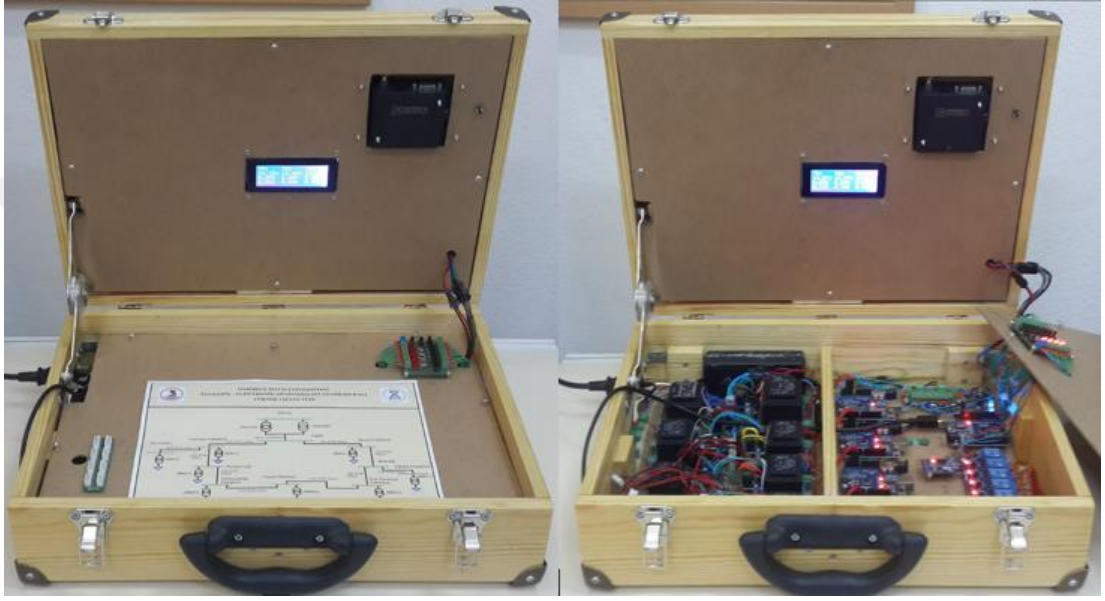
Yazılım bölümü ise donanım bölümünden alınan verilerin uygun bir arayüzde denetlenmesini ve gözlemlenmesini sağlamaktadır. Scada arayüzü *Processing* uygulaması ile oluşturulmuştur. Arayüzde donanım bölümünden gelen veriler anlık olarak görüntülenmekte ve denetim işaretleri gönderilerek donanım bölümünde yer alan yüklerin devreye alınıp devreden çıkarılma işlemleri yerine getirilmektedir. Akıllı Şebeke benzetimi için genel çalışma algoritması görsel olarak Şekil 3.1'de gösterilmiştir.



Şekil 3.1. Benzetim için genel çalışma algoritması

### 3.1. Donanımsal Sistemler

Çalışmanın donanımsal bölümünde mikro denetleyici olarak *Arduino UNO R3* modeli kullanılmıştır. Uygulamada *Arduino* seçilmesinin temel nedenleri çok yaygın olarak kullanılması, çeşitli kütüphanelerinin oluşturulmuş olması, çok sayıda algılayıcı ve genişletme yapısı ile adaptasyonunun sağlanmış olması ve kullandığımız *Processing* yazılımı ile seri olarak iletişim kurması şeklinde sıralanabilir. Donanımsal sistemin genel yapısı Şekil 3.2’de gösterilmiştir.

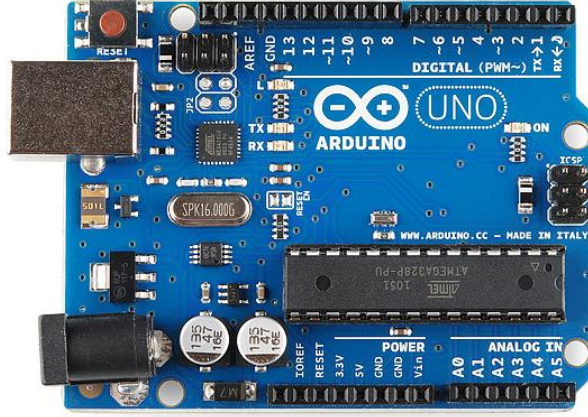


Şekil 3.2. Donanımsal genel yapı

#### 3.1.1. Mikro denetleyici (Arduino UNO R3)

*Arduino UNO R3* modeli ATmega328P mikro denetleyici tabanlı, 14 sayısal giriş/çıkış, 6 analog giriş pinlerine sahip, 16 MHz frekansı ile çalışan, USB portu, güç soketi, ICSP bağlantı noktası ve yenileme butonu bulunan bir denetleyicidir (Anonymous, 2018). *Arduino UNO R3* modeli Şekil 3.3’te gösterilmiştir.

Kullandığımız bu mikro denetleyici kart ile her türlü analog veya sayısal algılayıcı bilgisini okuma, sayısal çıkış verme ve PWM pinleri ile analog çıkış verebilme gibi birçok görev yerine getirilebilmektedir. *Arduino UNO* UART TTL seri iletişim, I<sup>2</sup>C ve SPI gibi iletişim protokollerini desteklemektedir (Anonymous, 2018).



Şekil 3.3. Arduino UNO R3 modeli

Çalışmamızda bir *Arduino* RTU olarak görev yapmaktadır. Bunun dışında şebeke koruma ve denetim rölelerini modelleme için dört Arduino daha kullanılmıştır. Donanım bölümünde yer alan Arduino'lar arasındaki veri transferini sağlamak için I<sup>2</sup>C iletişim protokolü seçilmiştir. Arduino UART TTL seri iletişim, I<sup>2</sup>C ve SPI gibi iletişim protokollerini desteklemektedir. Ayrıca UART TTL–RS485 dönüştürücüsü kullanılarak Arduino'lar arasında Modbus iletişim protokolü de kullanılabilir. Ancak harici bir yapıya ihtiyaç duymadan sadece iki kablo ile iletişimi gerçekleştirdiği ve standart olarak 25m mesafeye kadar sorunsuz çalıştığı için I<sup>2</sup>C iletişim protokolü seçilmiştir.

RTU ana denetleyici olarak görev yapmaktadır. Diğer röleler ise uydu biçiminde çalışmaktadır. Bu yapı I<sup>2</sup>C iletişim protokolünü kullanarak iki yönlü olarak çalışmaktadır. Bu durumda ana denetleyici olan RTU Scada arayüzünden gelen komutları ilgili uydu rölelerine göndermekle sorumlu iken aynı zamanda rölelerin elde ettiği ölçüm sonuçlarını da toplayarak Scada arayüze göndermektedir. Böylece iki yönlü iletişimi gerçekleştirmiş olur.

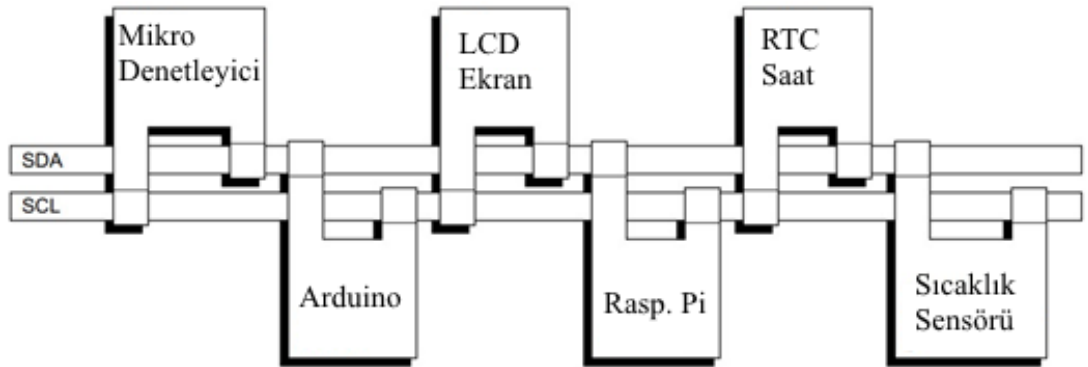
Belirtilen tüm bu işaretler bir araya getirildiği zaman her bir trafo için AC akım, AC gerilim, kesici durum bilgisi olmak üzere toplam üç adet veri I<sup>2</sup>C iletişim ortamı üzerinden RTU'ya taşınmaktadır. Ayrıca I<sup>2</sup>C dağıtım noktasına bağlanan DS3231 gerçek zaman modülü sayesinde RTU verileri Scada ortamına aktarılırken saat etiketi ile aktarılmış olacaktır. Scada yazılımında veriler değerlendirilirken saat bilgisi bilgisayardan da alınabilmektedir. Ancak bazı senaryolarda tarih-saat bilgisinin donanım bölümünden alınması gerektiğinde bu modül kullanılarak ilgili işlemler gerçekleştirilmiş olacaktır.

### 3.1.2. I<sup>2</sup>C iletişim protokolü

I<sup>2</sup>C protokolü birden fazla cihaz arasında SCL (Serial Clock Line) ve SDA (Serial Data Line) şeklinde ifade edilen sadece iki kablo kullanarak iletişim kurulabilmesine olanak sağlayan bir ortamdır, I<sup>2</sup>C protokolü tek ana çoklu uydu, çoklu ana çoklu uydu ve çoklu ana tek uydu durumlarında çalışabilmektedir (Megalingam vd, 2016). *Arduino* 'da dâhili olarak bulunan *Wire* kütüphanesi I<sup>2</sup>C protokolünün *Arduino* 'lar arasında iletişim sağlamasını kolaylaştırmıştır (Patel ve Soni, 2013).

I<sup>2</sup>C protokolü birden fazla cihazın birbirine bağlanmalarını ve iletişim kurmalarını sağlayan çok kullanışlı bir ortamdır. Bağlanabilecek cihaz sayısı 128'e kadar çıkabilmektedir. Tüm cihazlar sadece iki kablo ile birbiri ile iletişim kurabilmektedir. Tüm cihazlara ayrı adresler verilebilmesi doğru iletişim ve bilgi güvenliğini ortaya koymaktadır (Gammon, 2011).

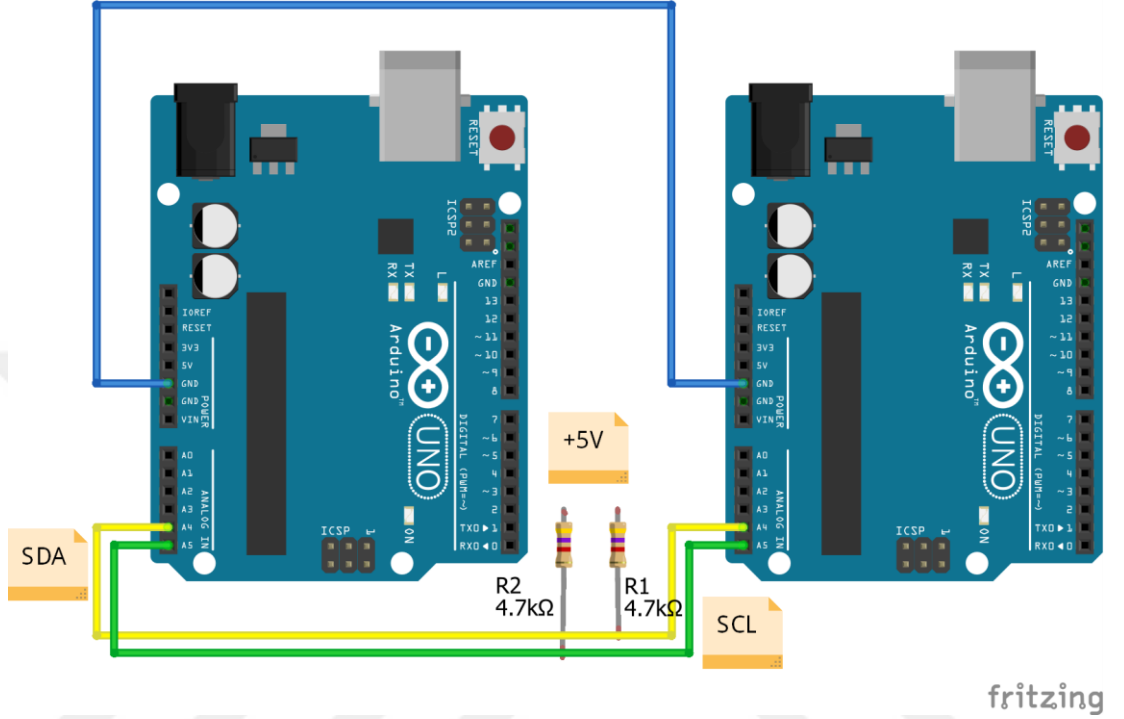
I<sup>2</sup>C protokolü ile seri olarak 8-bit tabanlı iki yönlü veri transferini Standart-biçim ile 100kbit/s, Hızlı-biçim olarak 400kbit/s, Hızlı-biçim artı olarak 1Mbit/s, Yüksek-Hızlı biçim olarak ise 3,4Mbit/s seviyelerinde gerçekleştirebilmektedir (Anonymous, 2014). Ayrıca I<sup>2</sup>C protokolünde birden çok ana biçimi şeklinde çalışılma durumlarında veri çarpışmalarını önleme sistemi de veri kaybını önleyecek şekilde bulunmaktadır (Anonymous, 2014). Şekil 3.4'te iki mikro denetleyicinin örnek bir I<sup>2</sup>C bara bağlantısı görülmektedir.



Şekil 3.4. Örnek bir I<sup>2</sup>C bara bağlantısı

Mikro denetleyici olarak kullanılan *Arduino UNO* I<sup>2</sup>C pinleri aşağıda listelenmiş ve Şekil 3.5'te gösterilmiştir.

- Analog port 4 (A4) = **SDA** (Serial Data Line)
- Analog port 5 (A5) = **SCL** (Serial Clock Line)



Şekil 3.5. İki Arduino UNO I<sup>2</sup>C bağlantıları

I<sup>2</sup>C protokolü kullanılarak cihazların iletişim kurabilmeleri için GND bağlantılarının ortak olmaları gerekmektedir. Bu nedenle donanımsal sistemde kullanılan tüm yapıların ve güç kaynaklarının nötr (GND) bağlantıları kısa devre edilmiştir.

Şekil 3.5'te görüldüğü gibi SCL ve SDA dağıtım noktalarına birer tane gerilimi yüksek tutma direnci bağlanmıştır. Normal durumda I<sup>2</sup>C dağıtım noktası 5V seviyesinde tutulması gerekmektedir bu da dirençler ile sağlanmaktadır (Gammon, 2011). Ancak gerilimi yüksek tutma direnç değeri seri iletişimde oluşan kare dalgaları etkilemeyecek şekilde kullanılmalıdır. Bu değer dağıtım noktasına bağlı cihaz sayısına bağımlı olmakla beraber 2k $\Omega$  ile 10k $\Omega$  arasında tutulmalıdır (Gammon, 2011). Çalışmamızda SCL ve SDA dağıtım noktaları ayrı ayrı iki tane 4,7k $\Omega$  dirençler bağlanmıştır.

*Arduino UNO* ve diğler cihazlar arasında I<sup>2</sup>C bağlantısı kullanılırken *Wire* kütüphanesi kullanılmıştır. Bu kütüphane sayesinde iki yönlü I<sup>2</sup>C veri alışverişi başarılı bir şekilde sağlanmıştır. Uydu adresler ana denetleyici durumunda bulunan RTU *Arduino*'suna ölçüm yaptıkları tüm bilgileri gönderirken RTU Scada arayüzünden aldığı komutları uydu cihazlarına göndermektedir.

### 3.1.3. Transformatör

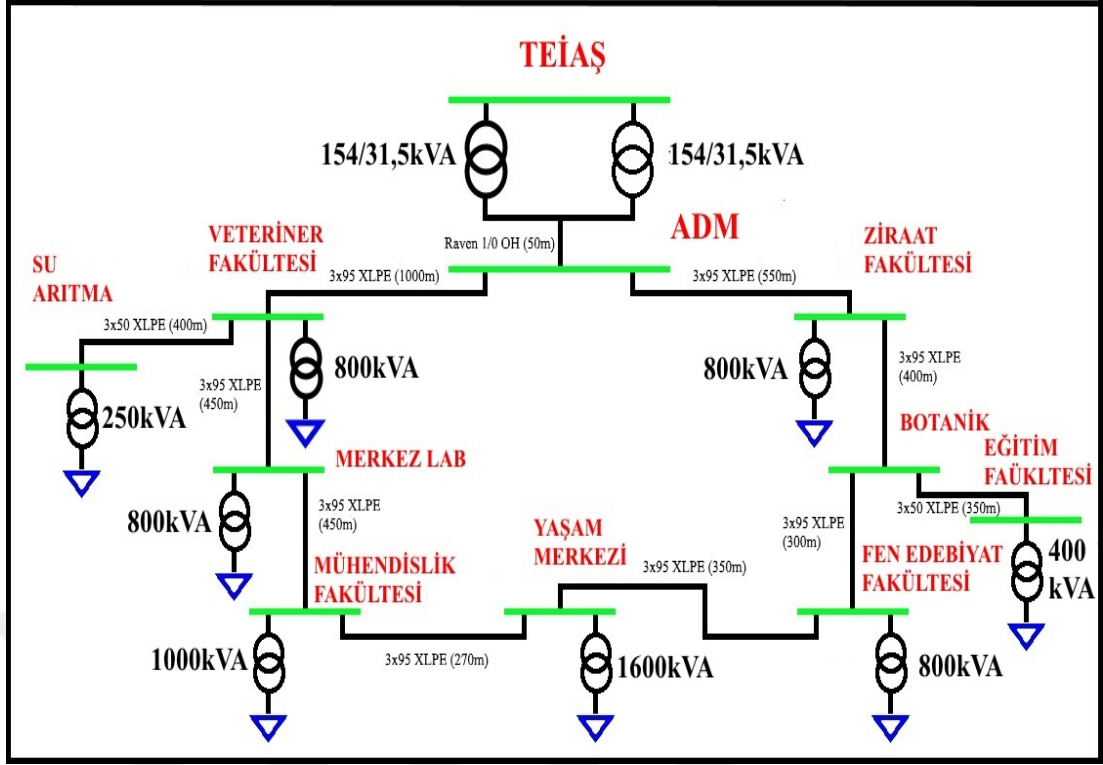
Oluşturulan çalışmada Aslan marka kapalı tipte düşük güçteki trafolar kullanılmıştır. Örnek kapalı tip trafo Şekil 3.6'da görülmektedir. Kullanılan trafoların birincil gerilimi 230V sekonder gerilimi ise 12V şeklindedir. Trafoların güçleri ise model şebekeye bağlı olarak aynı oranları elde edecek şekilde seçilmiştir.



Şekil 3.6. Transformatör

Eğitim takımında model şebeke olarak üniversitemizdeki dağıtım şebekesinin bir bölümü referans alınmıştır. Şebekede kullanılan trafolar 1/100000 oranında küçültülerek deney ortamı oluşturulmuştur. Model olarak kullanılan dağıtım şebekesi Şekil 3.7'de görülmektedir. Üniversite şebekesinde yer alan trafoların güç değerleri sırasıyla 1600kVA, 1000kVA, 800kVA, 800kVA, 800kVA, 800kVA, 400kVA ve 250kVA şeklinde iken eğitim takımında kullandığımız trafo güçleri sırasıyla 16VA, 10VA, 8VA, 8VA, 8VA, 8VA, 3.2VA ve 2.6VA şeklindedir.





Şekil 3.7. Örnek olarak kullanılan dağıtım şebekesi

### 3.1.4. AC akım algılayıcı

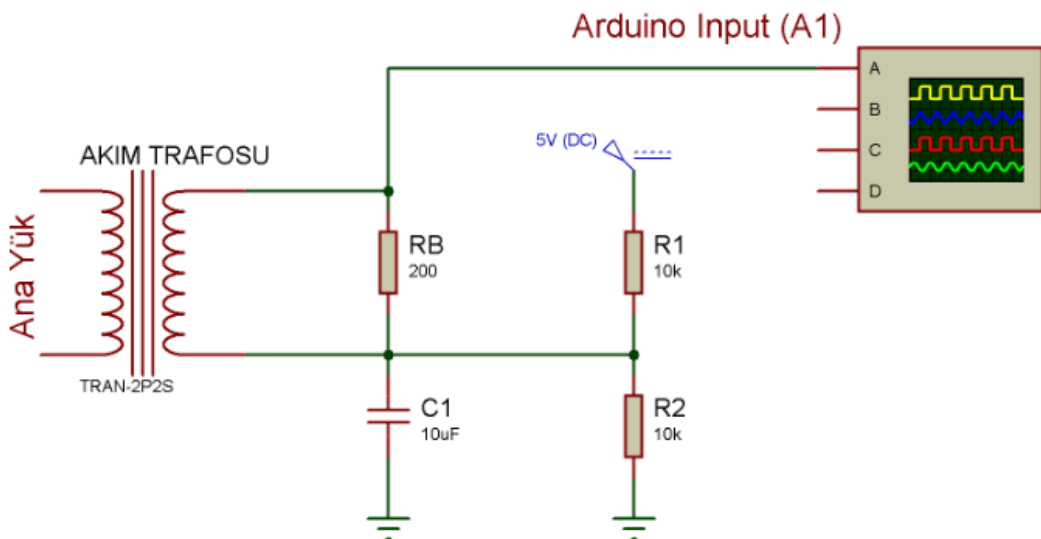
Her bir trafo için akım ölçümü toroid tipte AC akım algılayıcı kullanılarak gerçekleştirilmiştir. AC akım algılayıcı Şekil 3.8'de görülmektedir. Akım algılayıcı bir trafonun ikincil bölümü gibi davranmaktadır. Burada birincil sargı yüke bağlanmış olan ve üzerinden akım geçen iletkenin kendisidir. Akım ölçümünün yapılabilmesi için toroidin ortasından kablunun geçirilmesi yeterlidir. Daha hassas ölçümler alabilmek için birincil kabloyu toroidin içerisinden birkaç defa geçirmek gerekir. Bizim çalışmamızda mili amper seviyesindeki akımları da ölçebilmek için akım taşıyan kablo üç defa geçirilmiştir.



Şekil 3.8. AC akım algılayıcı

Kullanılan akım algılayıcı Zeming marka ZMCT103C model en fazla 5A rms ölçüm yapabilmektedir. Algılayıcı çalışma şekli 5A/5mA şeklinde olup çıkış pinlerine Şekil 3.9'da görüldüğü gibi  $R_B=200\Omega$  örnekleme direnci bağlanmıştır. Bu sayede hem akım trafosu açık uçlu kalmamış oluyor hem de algılayıcı çıkış işareti ölçümü değerlendirecek olan *Arduino* analog giriş yapısına uygun olarak gerilim biçimine dönüştürülmüş oluyor. *Arduino* analog giriş yapısı akımı ölçecek şekilde değil de gerilimi ölçecek şekilde düzenlenmiştir. Kullanılan algılayıcıda örnekleme direnci bulunmasaydı harici olarak eklenmesi gerekirdi. Şekil 3.9'da belirtilen  $R_1$  ve  $R_2$  dirençleri gerilim bölücü olarak çalışmaktadır ve değerleri eşit olarak  $10k\Omega$  olarak seçilmiştir.  $C_1$  kondansatörün değeri ise  $10\mu F$  seçilmiştir.

### Akım Ölçümü



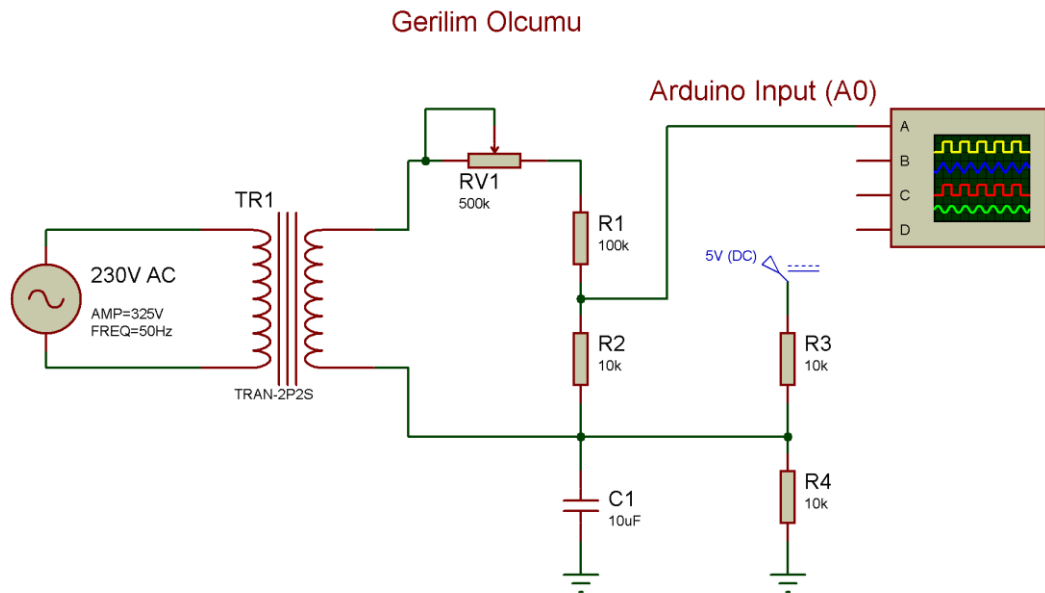
Şekil 3.9. AC akım algılayıcı bağlantı diyagramı



AC akım algılayıcı çıkış işareti 1V genliğe sahip sinüs biçimli şeklindedir. Çıkış işaretinin bu şekilde *Arduino* analog giriş pinine bağlanması eksi gerilimden dolayı zarar verecektir. *Arduino* analog giriş pinleri 0-5V aralığında çalışmaktadır. Bunun önlenmesi için çıkış işaretine 2,5V DC gerilim eklenerek AC akım algılayıcı çıkış işaretinin 0-5V arasında salınımı sağlanmıştır (Stusek vd, 2017; Tamkittikhun vd, 2015b). Şekil 3.9’da belirtilen  $R_1$  ve  $R_2$  dirençleri gerilim bölücü olarak çalışmaktadır ve 2,5V DC elde etmek için kullanılmıştır.

### 3.1.5. AC gerilim algılayıcı

*Arduino* kullanarak AC gerilim ölçümünün sağlanması için çeşitli algılayıcı bulunmaktadır. Bu algılayıcı 250V’a kadar AC gerilimi trafo ve küçük bir elektronik devre yapısıyla mikro denetleyicilere uygun hale getirmektedir. Bu çalışmada gerilim algılayıcısı yerine örnek şebekedeki trafolar kullanılarak ölçüm işlemi gerçekleştirilmiştir. Trafoların sekonder gerilimleri 12V olduğu için bu bölüme eklenen küçük bir devre yapısıyla *Arduino* analog girişine uygun işaretler elde ederek AC gerilim ölçme işlemi gerçekleştirilmiştir. AC gerilim ölçüm devresi Şekil 3.10’da görülmektedir.



Şekil 3.10. AC gerilim ölçüm devresi

AC geriliminin ölçülmesi sürecinde 500k $\Omega$  değerinde bir potansiyometre kullanılmıştır. Bu potansiyometrenin kullanımı isteğe bağlıdır. Ancak bu direnç değerinin değiştirilmesi algılayıcı çıkış işaretini kalibre edebildiği için önemlidir.

Algılayıcı ait çıkış işareti *Arduino* analog giriş pinine bağlanmadan önce bir Osiloskop bağlanarak genliğinin değeri görülmelidir. *Arduino* 0-5V aralığında çalıştığından salınan sinüs biçimli işaretin bu değerleri aşmayacak şekilde ayarlanması gerekmektedir. Bu da 500k $\Omega$  değerinde potansiyometre ile sağlanır. Devrede kullanılan 2 adet 10k $\Omega$  değerindeki dirençler ise gerilim bölücü olarak kullanılmıştır. AC gerilim algılayıcısından elde ettiğimiz işaret artı ve eksi bölgelerde salınmaktadır. *Arduino*'ya eksi işaret verilemeyeceğinden AC çıkış işaretine 2,5V DC seviye yükseltme değeri eklenerek çıkış işaretinin 0-5V aralığının tam ortasında salınması sağlanmaktadır.

### 3.1.6. Basmalı Buton

Elektrik şebekelerinde kullanılan Kesici anahtarlamalarının benzetimi için donanım bölümünde basmalı buton kullanılmıştır. Her anahtarlama yapısının benzetimi gerçekleştirecek şekilde her bir trafo için bir adet buton kullanılmıştır. Bu butonların tetiklenmesi elektrik şebekelerindeki Kesicilerin manevra durumlarının benzetimine olanak sağlamaktadır. Çalışmada kullanılan basmalı butonlar Şekil 3.11'da görülmektedir.



Şekil 3.11. Basmalı buton

### 3.1.7. Gerçek Zaman Modülü – DS3231

DS3231 düşük maliyetli, oldukça doğru I<sup>2</sup>C tabanlı gerçek zaman modülüdür. Devre kartı üzerinde bulunan düğme pil sayesinde kartın enerjisiz bulunduğu durumlarda dahi saat bilgisini hafızasında tutabilmekte ve böylece tekrar ayar gerekmemektedir. İçerisinde bulunan sıcaklık algılayıcısı ile  $\pm 3^{\circ}\text{C}$  doğruluk oranı ile sıcaklık bilgisini de verebilmektedir (Anonymous, 2015).

DS3231 modülünün sayıcısı Saniye, Dakika, Saat, Gün, Ay, Yıl, Hafta Günü ve Artık Yıl bilgisini 2100 tarihine kadar verebilmektedir (Anonymous, 2015). DS3231 RTC modülü Şekil 3.12’de görülmektedir.



Şekil 3.12. DS3231 RTC modülü

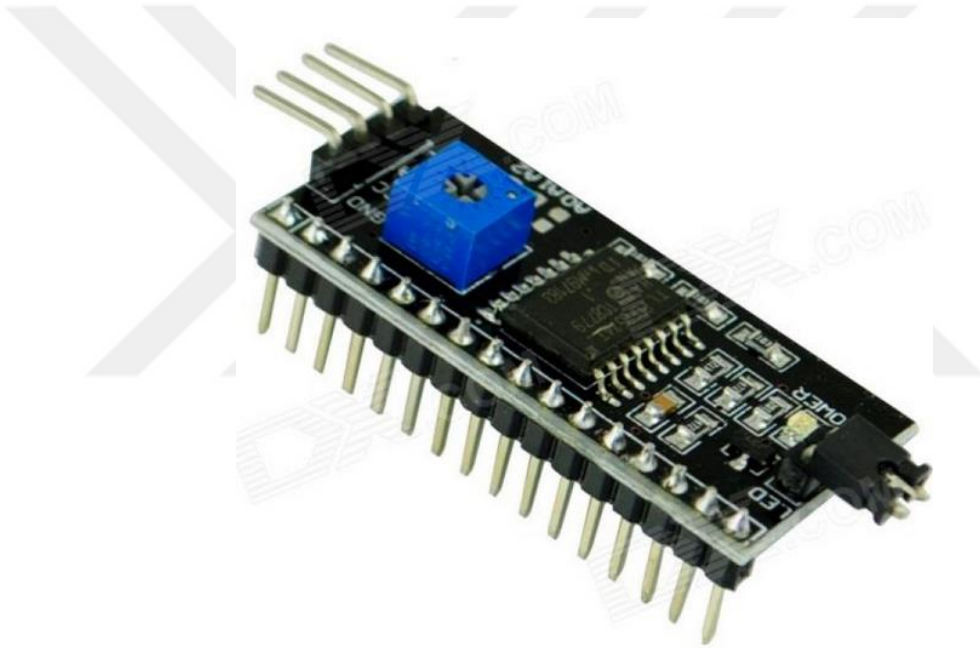
### 3.1.8.4x20 LCD Ekran

Çalışmamızda ölçüm değerlerinin arayüz üzerinde görüntülenmesinin yanı sıra donanımsal sistem üzerine yerleştirilmiş olan ekran üzerinde de görüntüleme yapılabilmektedir. Ekran olarak 4x20 LCD ekranları seçilmiştir. Bu ekranların seçilmesinin nedeni *Arduino* ile çok kolay iletişim kurabilmesi ve Likit Kristal numerik değerlerin çok rahat gösterilmesidir. 4x20 LCD ekranı Şekil 3.13’te görülmektedir.



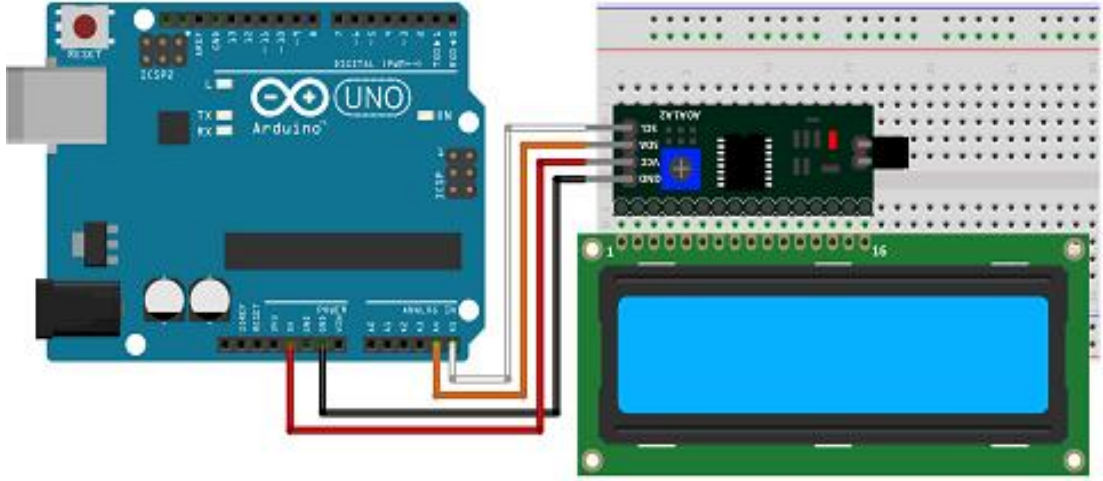
Şekil 3.13. 4x20 LCD ekran

4x20 LCD ekran mikro denetleyiciler ile çalıştırılabilmesi için 16 pin kullanılmaktadır. Yazılımsal bazı kolaylıklar ile bu bağlantı 12 pine düşürülebilmektedir. Ancak bu sayı da çok fazla olarak değerlendirilmektedir. Hem bağlantı karışıklığını ortadan kaldırmak hem de *Arduino*'da çok fazla sayısal pin kullanmamak için bu ekranlara I<sup>2</sup>C modülü bağlanmaktadır. Görsel olarak Şekil 3.14'te görülmektedir. LCD I<sup>2</sup>C modülü ekranda bulunan 16 pini kullanarak iletişim şeklini doğrudan bağlantı yerine I<sup>2</sup>C bağlantısı biçimine dönüştürmektedir. Böylece bu ekranı çalıştırmak için çok sayıda kablo yerine sadece 2 adet kablo yeterli olmaktadır. Çalışmamızda röle olarak kullanılan *Arduino* denetleyiciler arasındaki iletişimi I<sup>2</sup>C ile sağladığımız için ekranı çalıştırmak için modül çıkış kablolarını I<sup>2</sup>C dağıtım noktasına bağlamak donanımsal olarak yeterli olur.



Şekil 3.14. LCD ekran I<sup>2</sup>C Modülü

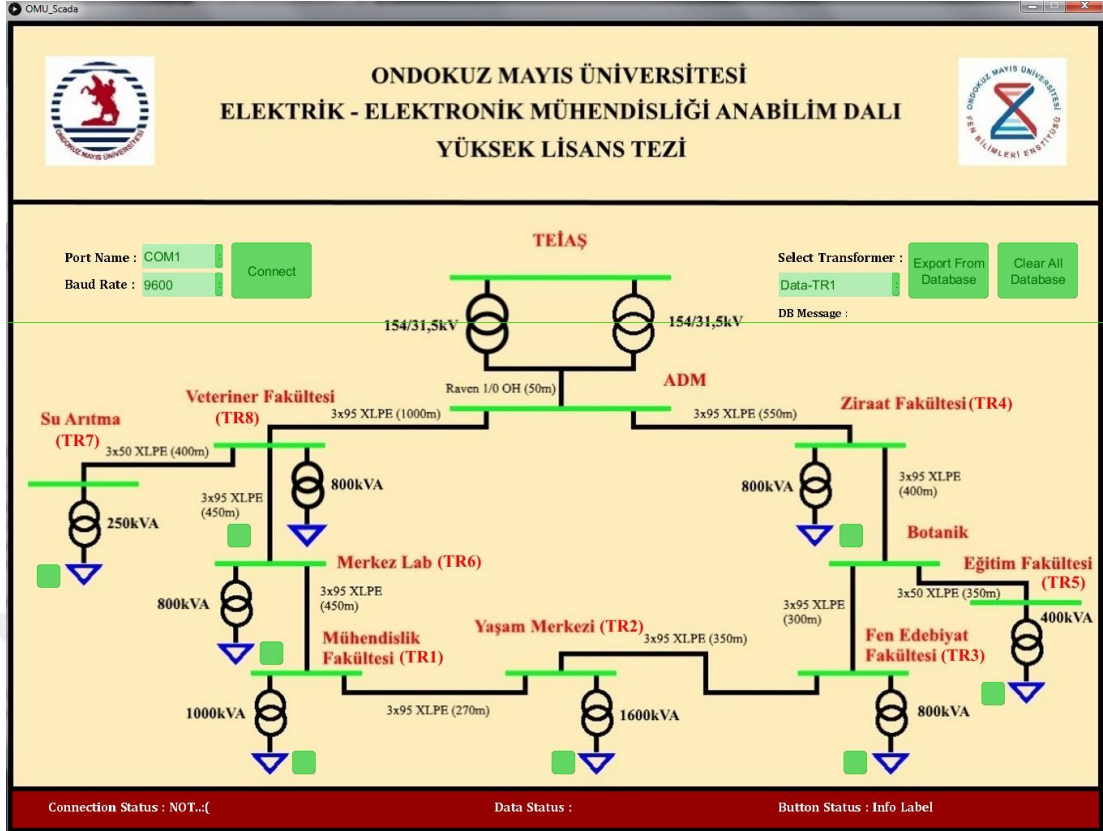
LCD ekranın *Arduino* ile bağlantısını gerçekleştirmek için I<sup>2</sup>C modülünün pinlerini I<sup>2</sup>C dağıtım noktasına bağlamak yeterli olur. *Arduino* I<sup>2</sup>C Modülü ve LCD ekran bağlantısı Şekil 3.15'de görülmektedir.



Şekil 3.15. Arduino I<sup>2</sup>C Modülü ve LCD ekran bağlantısı

### 3.2. Yazılımsal Sistemler

Yazılım ortamı olarak *Processing 3* kullanılmıştır. *Processing*, sayısal sanat ve görsel tasarım ile ilgilenenlere programlamanın temellerini görsel bir yaklaşımla öğretmek için kullanılan veya esnek elektronik bir taslak defteri görevi de görebilen açık kaynaklı bir programlama dili ve tümleşik geliştirme ortamıdır (Anonymous, 2012). 2001 yılında *Massachusetts Institute of Technology*'de ortaya çıkan, Casey Reas ve Benjamin Fry (Anonymous, 2012) tarafından geliştirilen proje Java üzerine kurulmuş olup daha basit bir çizelge programlama modeli ve sözdizimine sahiptir. Scada arayüzünün genel görüntüsü Şekil 3.16'da görülmektedir.



Şekil 3.16. Genel Scada arayüzü

### 3.2.1. Processing 3

*Processing* ilk ortaya çıktığı zamanlarda çoğunlukla sayısal sanat ve görsel tasarım gibi projelerde kullanılmaktaydı. Ancak *Arduino* yazılım ortamı ile olan benzerliği ve kullanışlılığı *Arduino* tabanlı projelerde kullanımını yaygın hale getirmektedir. Oluşturulan projelerin bilgisayar ortamlarında görselleştirilmesi *Processing* kullanılarak çok kolay ve esnek bir şekilde gerçekleştirilebilmektedir.

Ayrıca *Processing* ve *Arduino* arasındaki seri iletişim çok sağlıklı ve hızlı bir şekilde çalışmaktadır. Herhangi bir veri kaybı olmaması da büyük faydalarından birisidir. Gerçekleştirdiğimiz deneysel çalışmalarda Veri kaybı yaşamadan ve basit şifrelemeler oluşturularak veri güvenliği konusunu da başarılı bir şekilde gerçekleştirilmiş oldu. Donanım bölümünden gelen verilerin doğruluğunu kontrol etmek için bilginin baş bölümü “#Data” ile başlamalı ve her bir ölçümün arasında virgül “,” bulunmalıdır

Bu çalışmada *Processing* ortamında hazırlanan Scada arayüzü RTU’dan seri iletişim yolu ile gelen verilerin ayrıştırılıp uygun şekilde yönlendirilmesi ve

görüntülenmesini sağlamaktadır. Bu çalışmada tüm veriler her bir saniyede bir güncellenecek şekilde ayarlanmıştır.

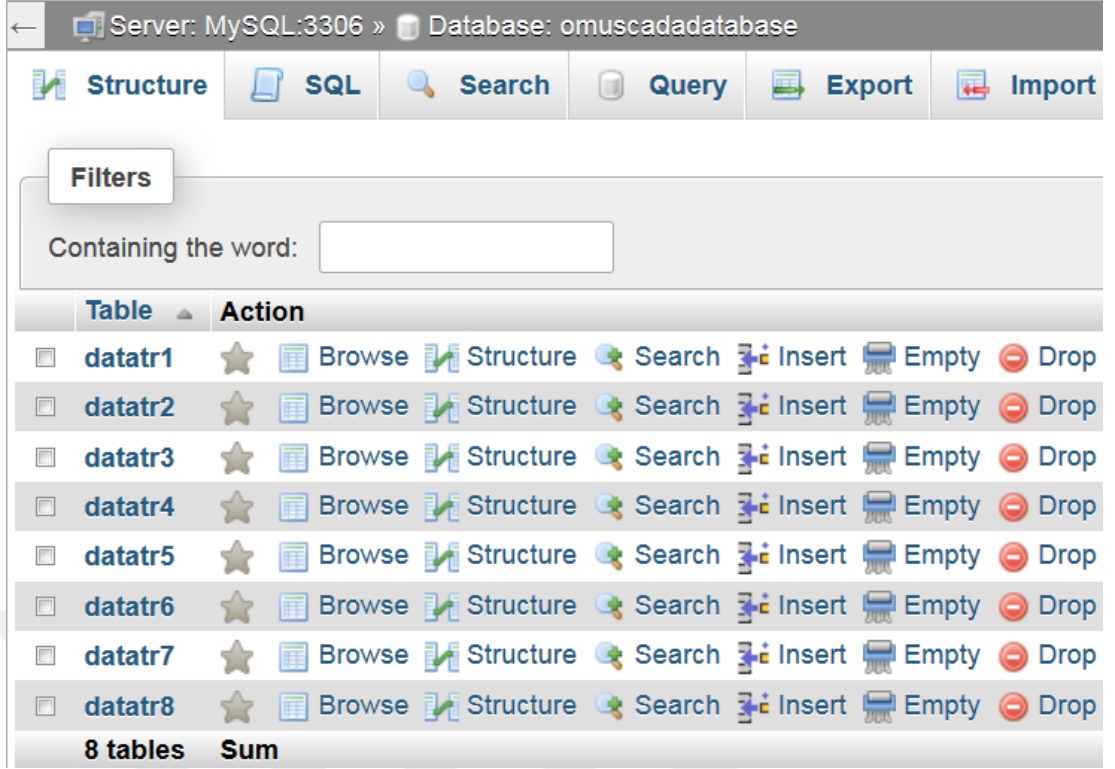
Scada arayüzü RTU'dan her bir trafo için akım, gerilim, durum bilgilerini ve Gerçek Zamanlı Saat Modülünden elde ettiği zaman bilgisini işleyebilmektedir. *Processing* esnek bir yazılım olmasından dolayı çoğu senaryonun uygulanmasına olanak sağlamaktadır.

### **3.2.2. MySQL veri tabanı**

Akıllı Şebeke uygulamalarında enerji sisteminden gelen verilerin gözlem ve denetiminin yanı sıra bir veri tabanında kaydedilmesi ve gerektiğinde raporlanabilmesi büyük önem arz etmektedir. Bu amaçla verilerin kaydedilmesi ve raporlanmasının benzetimi için Scada arayüzüne uyarlanmış *MySQL* veri tabanı Şekil 3.17'de görüldüğü gibi oluşturulmuştur (Suwansit vd, 2014). Enerji sisteminden gelen veriler belirli aralıklarla (saniyede bir) veri tabanındaki tablolara kaydedilmektedir.

*MySQL* veri tabanının oluşturmak için *Wamp Server* kullanılmıştır. *Wamp Server* *Windows* için hazırlanmış bir kurulum paketidir. Bir yerel çalışma programı olarak çalışmaktadır. Diğer programlardan farkı ise içerisinde *Mysql*, *Apache*, *Php* kurulumları bulunmaktadır.





Şekil 3.17. MySQL veri tabanı

*Processing* ortamında hazırlanmış olan Scada arayüzü *Wamp Server* kullanılarak *MySQL* veri tabanı ile iletişim kurabilmekte ve verileri kaydedebilmektedir. Scada arayüzü kullanılarak istenilen trafo için kaydedilmiş veriler *CSV* dosyası şeklinde dışa aktarılıp çeşitli çizelgeler elde edilebilmekte ve geçmişe yönelik analizler yapılabilmektedir.

*MySQL* ortamında önceden hazırlanacak çeşitli raporlama şablonları ile otomatik raporlar elde edilebilmektedir. Bu durum Akıllı Şebekeler’de yaygın olarak kullanılan yönetimsel veya analiz amaçlı raporlama süreçlerinin benzetiminde bir çözüm olarak sunulmuştur.



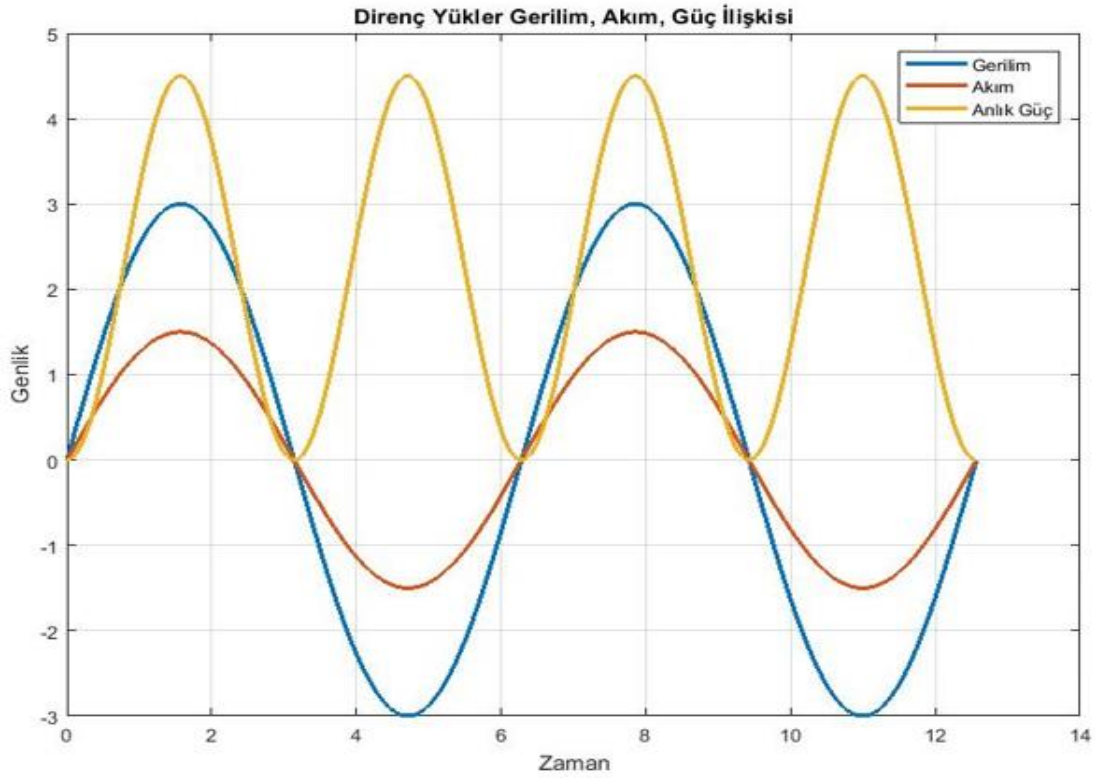
## 4. AC GÜÇ KURAMI

Enerji sistemleri ölçüm uygulamaları şebekeye bağlı olan tüm yüklerin ölçümlerini görüntüleme ve denetim etmeye dayalıdır. Ölçme ve görüntüleme uygulamalarının nasıl çalıştığını anlamak için yüklerin elektrik şebekesinde nasıl davrandıklarını bilmek faydalı olacaktır.

Farklı tipteki yüklerin şebekedeki davranışları temel olarak aynı prensiplere bağlıdır. Yüklerin davranışlarını daha kolay anlamak için ilk olarak direnç yükler hakkında bilgi verilip kullandıkları güçlerin hesabı hakkında bilgi verilecek daha sonrasında tepkin yükler ve birazda doğrusal olmayan yükler hakkında bilgi verilecektir.

### 4.1. Direnç Yükler

Ampuller, elektrikli su ısıtıcıları, ütüler, elektrikli fırınlar direnç yüklerine örnek verilebilir. Bu yükler kendilerine verilen tüm enerjiyi tüketirler. Direnç yüklerinin üzerinden geçen akım gerilimin dirence oranı şeklinde hesaplanabilir (*Ohm kanunu*). Saf bir direnç yük Şekil 4.1’de görülen şekilde akım gerilim ve güç işareti oluşturur.

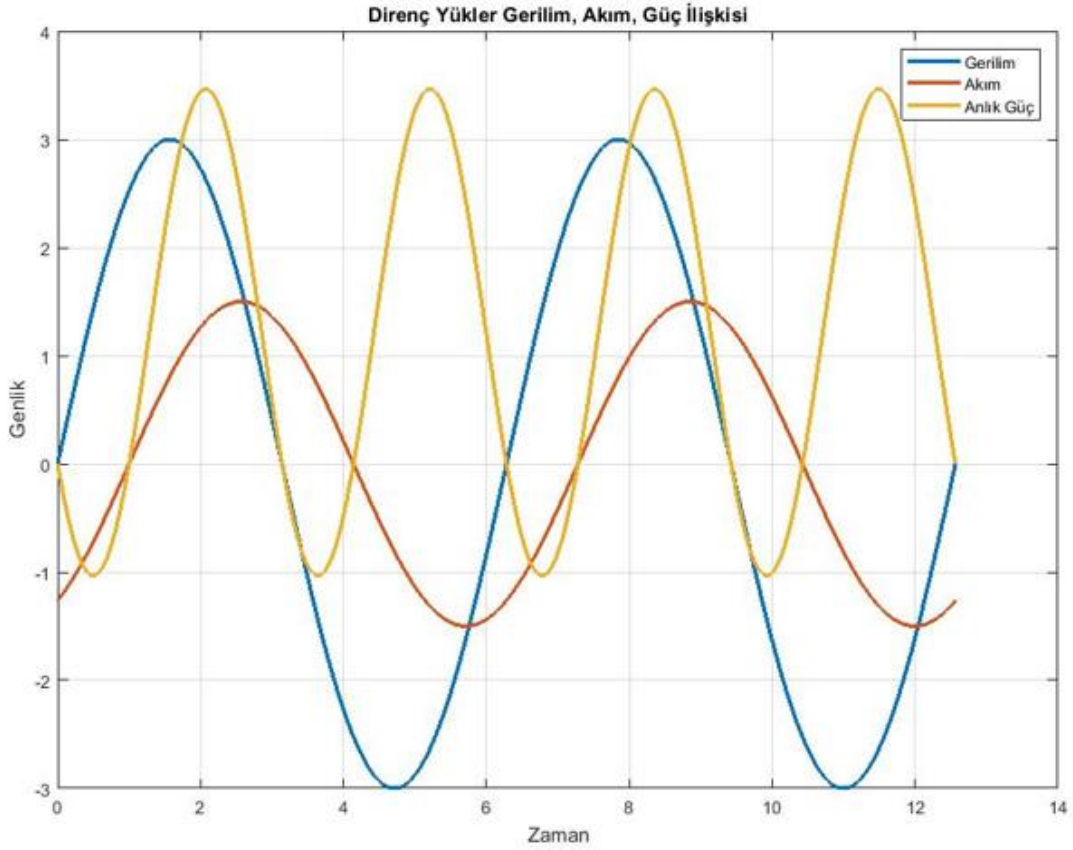


Şekil 4.1. Direnç yüklerde akım gerilim faz ilişkisi

Şekil 4.1’de sarı renk ile belirtilen işaret herhangi bir andaki güç tüketimini göstermektedir ve **Anlık Güç** olarak ifade edilmektedir. Belirtilen güç değeri herhangi bir zamandaki akım ve gerilimin çarpımına eşittir. Şekil 4.1’deki güç işareti daima artı değerdedir. Bu durum enerjinin kaynaktan yüke doğru olduğunu göstermektedir.

#### 4.2. Tepkin Yükler

Buzdolabı, çamaşır makinesi, matkaplar, kaynak makineleri verilen tüm enerjiyi tüketmezler. Bu yükler verilen enerjinin belli bir kısmını tüketirler bir kısmını ise kaynağa geri gönderirler. Bu tip yükler kendi içerisinde direnç bölümün yanı sıra Endüktif ve Kapasitif gibi bölümler de içerirler. İçerisinde kısmi olarak Endüktif bölüm bulunduran bir yükün dalga yapısı Şekil 4.2’de görülmektedir.



Şekil 4.2. Tepkin yüklerde akım gerilim ilişkisi

Şekil 4.2’de görüldüğü gibi sarı renkli işaret (Anlık Güç) bazı zamanlarda eksi değer almaktadır. İşaretin artı bölümü yük aktarılan enerjiyi ifade ederken eksi parçalar ise yük tarafından enerji kaynağına gönderilen enerjiyi ifade etmektedir. Şekil 4.2’de incelenmesi gereken başka bir ayrıntı ise akım ve gerilim işaretlerinin direnç yüklerde

olduğu gibi aynı hızda olmamalarıdır. Akım ve gerilim arasındaki bu hıza farklılığı **Faz Kayması** şeklinde ifade edilmektedir. Şekil 4.2’de belirtilen işaret Endüktif yük olduğu için akım gerilime göre daha ilerdedir. Nacak Kapasitif yüklerde gerilim akımın önünde yer almaktadır.

### 4.3. Etkin Güç, Tepkin Güç ve Görünür Güç

Şebeke gerilim ve akım işaretlerinin anlık değer olarak ifade edilmesi büyük zorluk oluşturmaktadır. Çünkü bu işaretlerin şebeke frekansında saniyede 50/60 kere salınmalarından dolayı işaretlerdeki değişimi görüp değerlendirmek güç bir durum oluşturmaktadır. Bu nedenle şebekede güç ifadesi için daha kolay bir çözüm elde edilmeye çalışılmıştır. Bu güç değeri ise anlık güç değerinin ortalaması şeklinde olan ve **Etkin Güç** veya **Gerçek Güç** şeklinde ifade edilen büyüklüktür Birimi Watt’tır.

Etkin güç genellikle yük tarafından kullanılan ve faydalı iş yapılmasını sağlayan güçtür. Şekil 4.2’de sarı renk ile belirtilen işaretin artı bölümü kaynaktan yüke doğru giden güç olup eksi parça ise yükten kaynağa gönderilen güçtür. Bu işarete etkin güç yük tarafından tamamen kullanılan güç olup işaretin artı parçası ile eksi parçasının farkına eşittir.

**Tepkin** veya **Sanal Güç** ise yük ve kaynak arasında sürekli gidip gelme hareketi yapan ve faydalı bir iş sağlamayan güçtür ve birimi “Var” olarak ifade edilir.

AC sistemlerde yaygın olarak kullanılan diğer güç tanımı ise **Görünür Güç**’tür. Görünür güç akım ve gerilim işaretlerinin rms değerlerinin çarpımına eşittir. Say direnç yüklerde etkin güç değeri görünür güç değerine eşittir. Bunun dışındaki tüm yük tiplerinde etkin güç görünür güçten küçüktür. Görünür güç etkin ve tepkin güç ölçümlerini tanımlamaktadır. Ancak bu iki gücün toplamına eşit değildir. Etkin ve tepkin gücün sayısal olarak toplanması faz farkının işleme alınmadan bir güç değeri sonucu verir ve bu da yanlış bir değer olarak ortaya çıkar (Anonymous, 2010).

İdeal bir sinüs biçimli işaret için etkin güç, tepkin güç ve görünür güç arasındaki ilişki aşağıdaki şekildedir:

$$Etkin Güç = Görünür Güç \times \cos\theta \quad (4.1)$$

$$Tepkin Güç = Görünür Güç \times \sin\theta \quad (4.2)$$

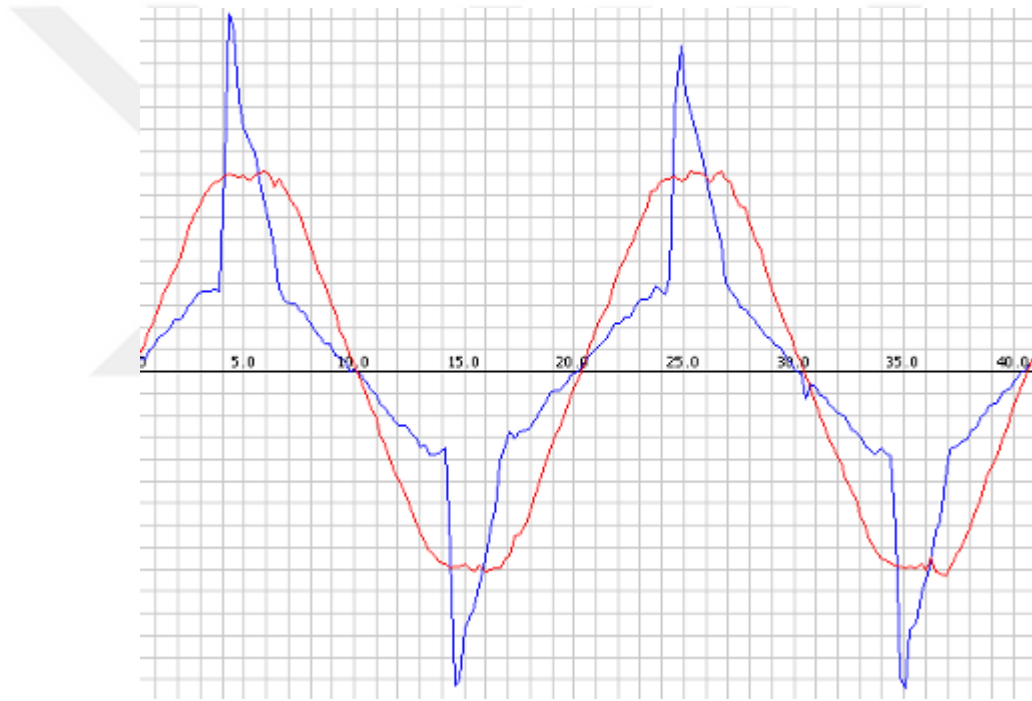
$$\text{Güç Faktörü} = \text{Etkin Güç} \div \text{Görünür Güç} \quad (4.3)$$

$\cos\theta$  **Güç Faktörü** şeklinde ifade edilmektedir.

Belirtilen güç faktörü ilişkisi sadece sinüs biçimli yükler için geçerlidir. Çoğu güç kaynağı Laptop bilgisayarları gibi doğrusal olmayan DC yapılar için enerji iletmemektedir. Bu tip yüklerin akım gerilim şekilleri Şekil 4.3'teki gibidir. Bu durumda güç faktörünü aşağıdaki eşitliğe göre yine de hesaplayabiliriz.

Ancak aşağıdaki eşitlik sadece saf sinüs biçimli işaretler için geçerlidir.

$$(\text{Görünür Güç})^2 = (\text{Etkin Güç})^2 + (\text{Tepkin Güç})^2 \quad (4.4)$$



Şekil 4.3. Doğrusal olmayan yükler için akım gerilim işareti

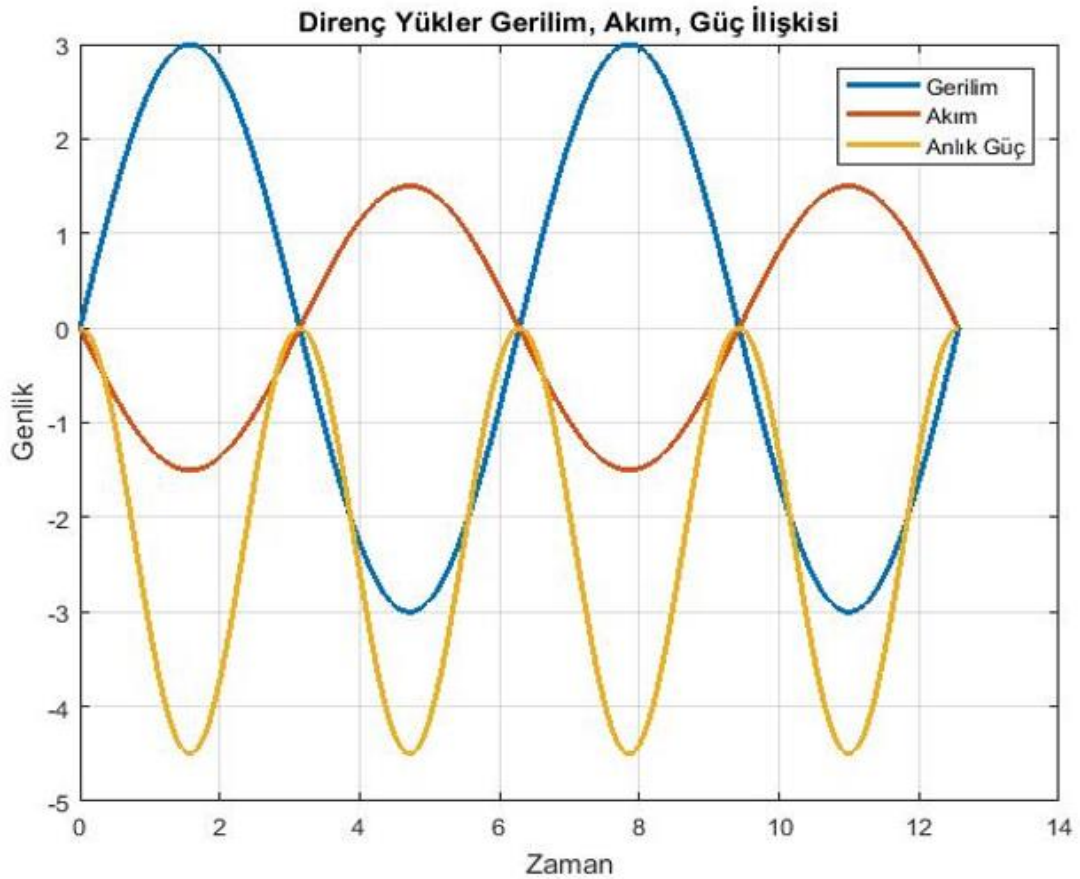
Görünür güç, etkin güç ve tepkin güç arasındaki ilişkiyi gösteren yukarıdaki eşitlik doğrusal olmayan yükler için geçerliliğini kaybeder. Aynı zamanda *güç faktörü* =  $\cos\theta$  eşitliği de geçersiz olur. Bu tür işaretler için gerilim ve akım işaretlerindeki yüksek seviye işaret bozukluklarının değerlendirilmeye alınması gerekmektedir (Anonymous, 2010).

*Güç Faktörü* değeri güç kaynağı verimliliğinin faz kayması  $\theta$  ve işaret bozukluğu bileşenleri tarafından ne kadar etkilendiğinin bir ölçümü şeklinde ifade edilebilir (Anonymous, 2010).

#### 4.4. Güç Akışının Yönünün Belirlenmesi

Güç sistemlerinde elektriğin yüke doğru mu olduğu yoksa üretilip kaynağa doğru mu yöneldiği önem arz etmektedir. Şekil 4.1’de belirtilen işaretlerde akım ve gerilim işaretleri her zaman aynı işareti almaktadır. Bunların çarpımı durumunda olan anlık güç değeri de sürekli olarak artı değerde hesaplanmaktadır. Bu durumda yük güç tüketiyor anlamına gelmektedir.

Eğer yük güç tüketmeyip Şekil 4.4’te gösterildiği gibi işaret yapısına sahipse bu durumda akım ve gerilim işaretleri ters işaretli olmaktadır. Gerilimin artı olduğu durumlarda akım işareti eksi olmaktadır.



Şekil 4.4. Elektrik enerjisinin üretilmesi durumunda akım gerilim ilişkisi

Anlık güç değeri akım ve gerilimin çarpımına eşit olduğundan her zaman eksi değer alacaktır. Elektrik işareti saniyede 50/60 kere salındığından dolayı elektriğin üretildiğini veya tüketildiğini anlık değerlerin incelenmesi zorluk oluşturmaktadır. Etkin güç değerinin işareti gücün tüketilip üretildiğini belirten çok kullanılan bir referans olmaktadır (Anonymous, 2010).

#### 4.5. Güç Sistemlerinde Matematiksel Hesaplamalar

Bu bölümde etkin güç, görünür güç, güç faktörü büyüklüklerinin hesaplanması rms gerilim ve rms akım değerlerinin anlık gerilim ve akım ölçümlerinden nasıl elde edildiği gibi konular işlenmektedir. Bu hesaplamaların ayrık zaman eşitliklerine de yer verilmiştir. Çünkü *Arduino* ile yapılan tüm ölçümler ve hesaplamalar ayrık zaman kullanılarak gerçekleştirilmiştir.

Etkin güç yük tarafından faydalı bir iş yapmak için harcanan güçtür. Matematiksel olarak Denklem 4.5'te görüldüğü gibi  $u(t)$  gerilimi ve  $i(t)$  akımının çarpımlarının integraline eşittir.

$$P = \frac{1}{T} \int u(t) \cdot i(t) dt \equiv U \cdot I \cdot \cos\theta \quad (4.5)$$

Etkin güç denklem açıklaması:

U: RMS gerilim

I: RMS akım

cos $\theta$ : Güç faktörü

Etkin güç eşitliğini ayrık zaman olarak ifade edilmiş şekli:

$$P = \frac{1}{T} \sum_{n=0}^{N-1} u(n) \cdot i(n) \quad (4.6)$$

Etkin güç değeri basitçe N sayıdaki akım gerilim çarpımlarının ortalaması şeklinde hesaplanmaktadır. Bu yöntemin sinüs biçimli veya bozulmuş işaretler için de geçerliliği bulunmaktadır (Anonymous, 2010). Etkin güç ayrık zaman denklem açıklaması:

u(n): u(t) geriliminin örneklenen birimi

i(n): i(t) akımının örneklenen birimi

N: örnekleme sayısı

Dönemsel olarak değişen işaretlerin rms değeri, tam bir dönem içinde alınmış anlık verilerin kareleri toplamının ortalamasının alınıp elde edilen sonucun

karekökünün hesaplanmasıyla elde edilir (Anonymous, 2010). Gerilimin rms değerinin ayırık zamanda hesaplanması Denklem 4.7 ile ifade edilmektedir.

$$U_{rms} = \sqrt{\frac{\sum_{n=0}^{N-1} u^2(n)}{N}} \quad (4.7)$$

Akımın rms değerinin ayırık zamanda hesaplanması Denklem 4.8 ile ifade edilmektedir.

$$I_{rms} = \sqrt{\frac{\sum_{n=0}^{N-1} i^2(n)}{N}} \quad (4.8)$$

Çalışmamızda görünür güç ve güç faktörü sırasıyla Denklem 4.9 ve Denklem 4.10'da gösterilmiştir.

$$\text{Görünür Güç} = \text{RMS Gerilim} \cdot \text{RMS Akım} \quad (4.9)$$

$$\text{Güç Faktörü} = \text{Aktif Güç} \div \text{Görünür Güç} \quad (4.10)$$

#### 4.6. Güç Sistemindeki Büyüklüklerin Arduino ile Hesaplanması

*Arduino* ile güç sistemlerindeki büyüklüklerin hesaplanmasında ayırık zaman eşitlikleri kullanılmıştır. Ölçümlerin elde edilmesinde ilk aşama örnekleme alınmasıdır. *Arduino* kullanarak her 20ms içerisinde 50 ile 100 arasında örnek alınabilmektedir. Eğer sadece akım bilgisi alınacaksa 100 örnek hem akım hem de gerilim bilgisi alınacaksa 50 örnek alınmaktadır. Buradaki kısıtlama *Arduino* "Analog Read" komutu ve hesaplama hızından kaynaklanmaktadır. Örnekleme sonucu elde edilen her bir değer anlık akım ve gerilim okumasıdır.

##### 4.6.1. Arduino ile etkin gücün hesaplanması

Etkin güç anlık güçlerin ortalaması alınarak hesaplanır. Etkin güç anlık gerilim ölçümünün anlık akım ölçümüyle çarpılması sonucu elde edilir. Belli sayıda örnekleme sonucu elde edilen anlık güç ölçüm değerleri toplanıp örnekleme sayısına bölünür. Etkin gücün hesaplanması için *Arduino*'da yer alan yazılım bölümü Şekil 4.5'de gösterilmektedir.

```
for (n = 0; n //Saf ADC girişinden gelen inst_voltage
        //ve inst_current değerleri buraya gelir)
{
    inst_power = inst_voltage * inst_current;
    sum_inst_power += inst_power;
}
real_power = sum_inst_power / number_of_samples;
```

Şekil 4.5. Arduino ile etkin gücün hesaplanması

#### 4.6.2. Arduino ile RMS gerilimin hesaplanması

Gerilimin rms değeri örnekleme sonucu elde edilen değerlerin karesi alınır. Örneklenen ve karesi alınan tüm değerler toplanır ve ortalaması alınır. Son aşama olarak ise elde edilen değerlerin karekökü alınarak sonuç elde edilmiş olur. RMS gerilimin hesaplanması için Arduino’da yer alan yazılım bölümü Şekil 4.6’de gösterilmektedir.

```
for (n = 0; n //Saf ADC girişinden gelen
        //inst_voltage değerleri buraya gelir)
{
    squared_voltage = inst_voltage * inst_voltage;
    sum_squared_voltage += squared_voltage;
}
mean_square_voltage = sum_squared_voltage / number_of_samples;
root_mean_square_voltage = sqrt(mean_square_voltage);
```

Şekil 4.6. Arduino ile rms gerilimin hesaplanması

*Arduino* tarafından gerçekleştirilen örneklemeyle ilgili olarak ilgili analog değerler elde edilmiş olur. RMS gerilim ölçümü analog giriş pinlerinden elde edilen analog değerlerin Şekil 4.6’da belirtilen eşitlikler kullanılarak dönüştürülmesi ve donanım bölümünde eklenmiş olan +2,5V DC işaret yükseltme değeri de göz önüne alınarak sonuçlandırılmış olur (Santis vd, 2016).



#### 4.6.3. Arduino ile RMS akımın hesaplanması

Akımın rms değeri örnekleme sonucu elde edilen değerlerin karesi alınır. Örneklenen ve karesi alınan tüm değerler toplanır ve ortalaması alınır. Son aşama olarak ise elde edilen değerlerin karekökü alınarak sonuç elde edilmiş olur. RMS akımın hesaplanması için *Arduino*'da yer alan yazılım bölümü Şekil 4.7'de gösterilmektedir.

```
for (n = 0; n //Saf ADC girişinden gelen
      //inst_current değerleri buraya gelir)
{
  squared_current = inst_current * inst_current;
  sum_squared_current += squared_current;
}
mean_square_current = sum_squared_current / number_of_samples;
root_mean_square_current = sqrt(mean_square_current);
```

Şekil 4.7. Arduino ile rms akımın hesaplanması

#### 4.6.4. Arduino ile görünür gücün hesaplanması

Görünür güç rms akım değeri ile rms gerilim değerinin çarpımı ile elde edilir. Görünür gücün hesaplanması için *Arduino*'da yer alan yazılım bölümü Şekil 4.8'de gösterilmektedir.

```
apparent_power = root_mean_square_voltage * root_mean_square_current;
```

Şekil 4.8. Arduino ile görünür gücün hesaplanması

#### 4.6.5. Arduino ile güç faktörünün hesaplanması

Güç faktörü etkin güç değeri ile görünür güç değerinin oranlanması ile elde edilir. Güç faktörünün hesaplanması için *Arduino*'da yer alan yazılım bölümü Şekil 4.9'de gösterilmektedir.

```
power_factor = real_power / apparent_power;
```

Şekil 4.9. Arduino ile güç faktörünün hesaplanması

#### **4.6.6. Arduino “Emonlib” kütüphanesi**

*Arduino* kullanılarak güç sistemlerine ait büyüklüklerin hesaplanabilmesi *Emonlib* adında bir kütüphane oluşturulmuştur. *Emonlib* kütüphanesi *Arduino* en üst sınırlarda örneklemelerin alınmasını sağlayan ve bu verileri kullanarak ölçüm sonuçlarının elde edilmesini sağlayan kullanışlı bir yazılımdır.



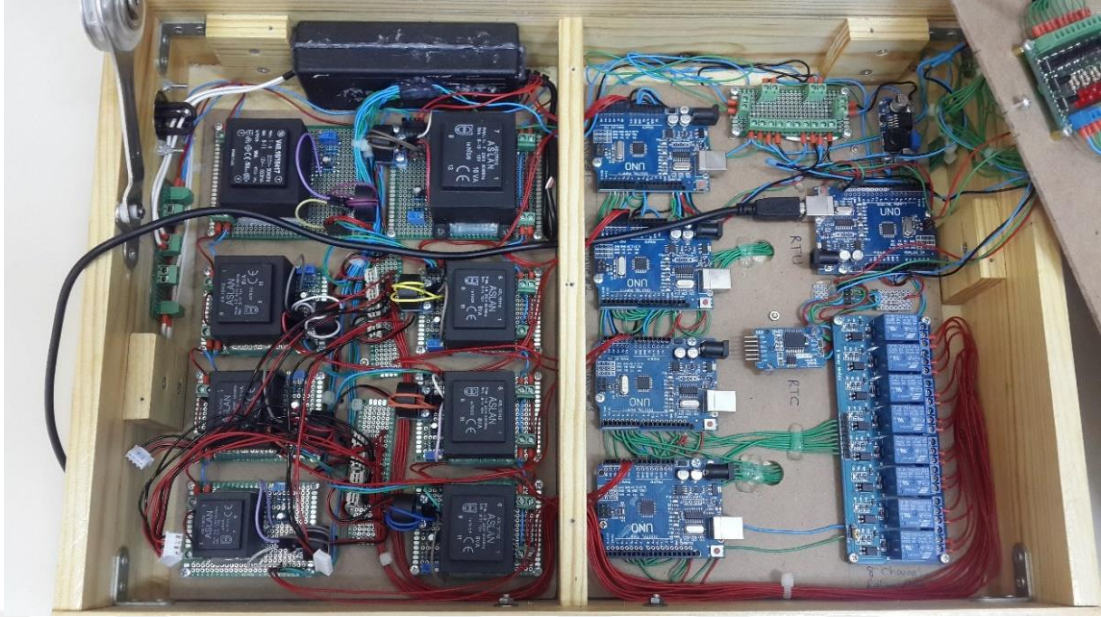
## 5. AKILLI ŐEBEKE SİMÜLASYON SENARYOSU

Çalışmamızda yazılım ortamının temeli olarak *Processing* kullanılmıştır. Donanımsal sistemin merkezinde ise *Arduino* yer almaktadır. Akıllı Őebekelerin gerçek uygulamalarında oluşturulan otomasyon iki temel bölümden oluşmaktadır. Birinci bölüm sistemin ana çekirdeđi olan Scada modülü ikinci bölüm ise Scada modülüne eklenen ve uygulamaları kolaylařtıran yardımcı modüller olmaktadır. Bu yardımcı modüller kesinti yönetim sistemi, yük akışı yönetim sistemi vb. gibi çeşitli yapılardan oluşabilmektedir. Yardımcı modüller Scada yazılımında kullanılan Őebeke ihtiyaçlarına göre ve sistemi kuran firmaya göre deđişiklikler göstermektedir. Belirtilen tüm bu sebeplerden dolayı Scada benzetim için çok çeşitli uygulamalar ve senaryolar ortaya çıkabilmektedir.

Çalışmamızda Ondokuz Mayıs Üniversitesine ait bir dağıtım Őebekesi örnek alınmıştır. Model Őebekede 8 adet trafo bulunmaktadır. Her iki trafoya ait bilgiler bir adet uydu Arduino tarafından deđerlendirilmektedir. Őekil 5.1 ve Őekil 5.2’de donanımsal bölümün ölçüm rölelerinin iletişimini sađlayan genel bađlantı çizelgesi görölmektedir.



Őekil 5.1. Donanımsal bölüm genel bađlantı çizelgesi

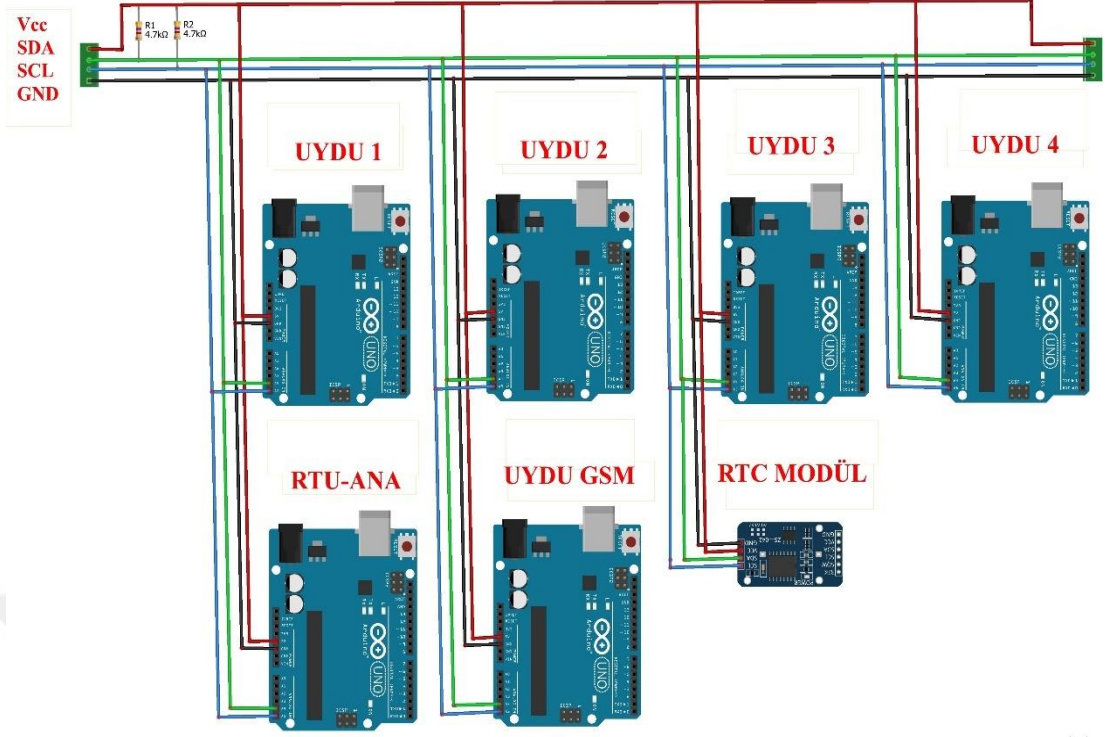


Şekil 5.2. Donanımsal bölüm mikro denetleyici genel bağlantı çizelgesi

Bu çalışmada akıllı şebekelerin yapı taşı olan Scada modülünün benzetimi temel alınmış ve Güç Kalitesi parametrelerine bağlı olarak çeşitli senaryolar oluşturulmuştur. Kullanılan yazılım ve donanım yapılarının geliştirmeye açık olmalarından dolayı istenen senaryoların oluşturulmalarını sağlamak için yeniden düzenlemelere gitmeye olanak sağlamaktadır. Oluşturulan bu deney ortamı ile donanımsal yapılar arasındaki iletişimi denetleme, donanımsal bölümde iki yönlü veri transferi gerçekleştirilebilmektedir. Ölçüm değerlerinin Scada arayüzüne aktarılması, Scada arayüzü veya mikro denetleyicilerde aritmetik işlemlerin gerçekleştirilmesi, gerçeğe yönelik senaryoların işletilmesi, verilerin dönemsel olarak veri tabanında saklanması ve raporlama yapılabilmektedir. Ayrıca ölçüm verilerinin Scada arayüzünde ve donanımsal bölümde gerçek ekranlarda anlık olarak görüntülenmesi ve arayüzden donanımsal sisteme komutların gönderilmesi gibi çok çeşitli uygulamaların gerçekleştirilmesi ve analizlerinin yapılmasına olanak sağlamaktadır.

### 5.1. RTU – Ana Denetleyici

Çalışmamızda RTU – Ana Denetleyici bölümü donanım bölümünün merkezi yönetim kısmı olarak görev yapmaktadır. Scada arayüzü ile donanım yapıları arasında köprü gibi çalışmaktadır. RTU olarak temelde bir adet Arduino UNO kullanılmaktadır. Diğer uydu yapıları, RTC modülü ve LCD ekran ile I<sup>2</sup>C arayüzü ile bağlantı sağlamaktadır. RTU genel bağlantı çizelgesi Şekil 5.3'te görülmektedir.



Şekil 5.3. RTU ana-uydu bağlantı çizelgesi

Bu çalışmada asıl amaç gömülü sistem kullanarak akıllı görüntüleme ve denetim ortamı oluşturmaktır (Barai vd, 2015; Zheng vd, 2013). Bu sebeple çalışmanın temelinde *Arduino UNO* kullanılmıştır. *Arduino* özellikle veri işleme çalışmalarında kullanılabilir çok ucuz ve esnek bir mikro denetleyicidir (Esposito vd, 2015). Bu mikro denetleyicinin önceki sürümleri de aynı uygulamalar için oluşturulmuştu. Bu çalışmada *Arduino* tarafından izleme uygulamalarının gerçekleştirilmesinin yanı sıra otomatik müdahaleler ve işlemlerde ön plana çıkmıştır (Fransiska vd, 2013; Rahman vd, 2015).

RTU uydu rölelerinde elde edilen ölçüm işaretlerini I<sup>2</sup>C protokolü ile taşımaktadır. I<sup>2</sup>C protokolü ile bir anda sadece 1 byte değerinde veri taşınabilmektedir. Ancak uydu denetleyicilerde elde edilen akım gerilim ölçümleri örnekleme sonucu elde edilmektedir. Ölçülen akım gerilim gibi değerler ondalık sayı biçiminden oldukları için *Arduino*’da ondalık sayı tipinde tanımlanmaktadır. Ondalık sayı tipi 4 byte büyüklüğündedir. Bu da ölçümlerin I<sup>2</sup>C üzerinden tek parça halinde taşınmasını engellemektedir. Bu sorunu gidermek için röle denetleyicilerde elde edilen ölçümler bir tampon oluşturularak taşınmaları sağlanmıştır. *Arduino* standart kurulumunda *Wire* kütüphanesinde tampon büyüklüğü 32 byte olarak ayarlanmıştır. Çalışmamızda her bir



uydu adresinden 52 byte büyüklüğünde veri transfer edildiği için kütüphanedeki tampon büyüklük değeri 64 byte olarak ayarlanmıştır.

Her uydu elde ettiği ölçüm ve durum bilgisi değerlerini ve adres bilgisini bir dizi haline dönüştürmekte ve bu şekilde RTU'ya göndermektedir. Uydu'da oluşturulan dizilerin Arduino arayüzünde oluşturulmuş hali Şekil 5.4'te görülmektedir. Her dizi on üç elemandan oluşmaktadır.

```
nodePayload[0] = NODE_ADDRESS;  
nodePayload[1] = supplyVoltage1;  
nodePayload[2] = Irms1;  
nodePayload[3] = realPower1;  
nodePayload[4] = apparentPower1;  
nodePayload[5] = TR1_fault;  
nodePayload[6] = supplyVoltage2;  
nodePayload[7] = Irms2;  
nodePayload[8] = realPower2;  
nodePayload[9] = apparentPower2;  
nodePayload[10] = TR2_fault;  
nodePayload[11] = RL1_durum;  
nodePayload[12] = RL2_durum;
```

Şekil 5.4. Uydu denetleyiciden elde edilen veriler

Rölelerden ölçüm işaretlerinin taşınması ve RTU denetim işaretlerinin rölelere gönderilmesi I<sup>2</sup>C protokolü kullanılarak ve RTU denetiminde gerçekleşmektedir. Veri aktarma işlemi iki yönlü olarak bir saniyelik dönemlerle gerçekleşmektedir. RTU her saniye rölelere ait ölçüm dizilerini talep etmekte ve Scada arayüzünden gelen komutları rölelere göndermektedir. Bu yapıda karar veren yapı RTU'dur. RTU önceden belirlenebilen bir aralıkla verileri talep etmekte ve komutları rölelere aktarabilmektedir. İki yönlü olarak çalışan bu iletişim yapısının hiçbir veri kaybı yaşamadan çalıştığı yapılan testlerde gözlemlenmiştir. RTU tarafından dönemsel olarak rölelerden talep edilen ölçüm verilerinin Arduino arayüzündeki yazılım biçimi Şekil 5.5'te görülmektedir.

RTU tarafından rölelerden talep edilen veri dizilerinde her trafo için 6 adet bilgi bulunmaktadır. Bunlar gerilim algılayıcı bilgisi, akım algılayıcı bilgisi, etkin güç, görünür güç, arıza durumu ve kesici durum bilgisi şeklindedir. Tüm bu değerler 4 byte boyutunda ve işlenmiş şekildedir. Rölelerde hesaplanmış ve düzenlenmiş olan tüm bu bilgiler RTU da toplanarak çeşitli dizilerde saklanmaktadır. Bu dizilerde yer alan

veriler harf dizini yazı tip biçimine dönüştürülerek seri iletişim yolu ile Scada arayüzüne gönderilmek üzere hazır hale getirilmiştir.

```
for (int nodeAddress = START_NODE; nodeAddress <= NODE_MAX; nodeAddress++)
{ // we are starting from Node address 1
Wire.requestFrom(nodeAddress, PAYLOAD_SIZE); // request data from node#
if (Wire.available() == PAYLOAD_SIZE)
{ // if data size is available from nodes
for (int i = 0; i < PAYLOAD_SIZE / 4; i++)
I2C_readAnything(nodePayload[i]); // get nodes data
if (k < NODE_MAX)
{
Vrms1[k] = nodePayload[1]; //Node 1,3,5,7
Irms1[k] = nodePayload[2]; //Node 1,3,5,7
realPower1[k] = nodePayload[3]; //Node 1,3,5,7
apparentPower1[k] = nodePayload[4]; //Node 1,3,5,7
faultState1[k] = nodePayload[5]; //Node 1,3,5,7
cbState1[k] = nodePayload[11]; //Node 1,3,5,7
Vrms2[k] = nodePayload[6]; //Node 2,4,6,8
Irms2[k] = nodePayload[7]; //Node 2,4,6,8
realPower2[k] = nodePayload[8]; //Node 2,4,6,8
apparentPower2[k] = nodePayload[9]; //Node 2,4,6,8
faultState2[k] = nodePayload[10]; //Node 2,4,6,8
cbState2[k] = nodePayload[12]; //Node 2,4,6,8
k++;
}
}
}
```

Şekil 5.5. RTU tarafından dönemsel olarak rölelerden talep edilen ölçüm verilerinin Arduino arayüzündeki yazılım biçimi

RTU röle temelli olarak diziler şeklinde aldığı verileri yeni bir düzenleme ile Şekil 5.5'te görüldüğü gibi farklı dizilere çevirmektedir. Bu diziler röle temelli olarak değil de Gerilim, Akım, Kesici Durum Bilgisi dizilerine çevirmektedir. Dizilerin bir biçiminden başka bir biçime dönüştürülmesinin sebebi Scada arayüzüne Seri İletişim yolu ile taşınmasının daha kolay olmasından kaynaklanmaktadır.

RTU ve diğer röleler arasındaki iletişim I<sup>2</sup>C ile sağlandığı daha önce belirtilmişti. I<sup>2</sup>C dağıtım noktasından gerçek zaman bilgisinin alınabilmesi için DS3231 RTC modülü bağlanmıştır. DS3231 modülü Arduino ile iletişimi I<sup>2</sup>C ile gerçekleştirmektedir (Anonymous, 2014). Bu modül sayesinde donanım bölümünden alınan verilerin saat etiketi ile birlikte alınabilmesi sağlanmaktadır.

RTU birimi uydu rölelerinden elde ettiği diziler şeklindeki verileri Şekil 5.5'te gösterildiği şekilde tekrardan düzenlemektedir. RTU Seri iletişim yolu ile verileri Scada arayüzüne aktarırken tekrar bir düzenleme yapmaktadır. Bu düzenleme tüm

verileri sıralayarak harf dizini biçimine dönüştürme olarak gerçekleştirilmektedir. Verilerin başına DS3231 RTC modülünden elde edilen tarih ve saat bilgisi eklenmektedir. Bu verinin yanına tüm analog ve sayısal veriler aralarında “,” karakteri olacak şekilde yan yana dizilmektedir. Şekil 5.6’da gösterildiği gibi elde edilen bu harf dizini seri iletişim yolu ile arayüze aktarılmaktadır.

```
tobeSendString = String(Vrms1[0]) + "," + String(Irms1[0]) + "," + String(cbState1[0]) + ","  
+ String(Vrms2[0]) + "," + String(Irms2[0]) + "," + String(cbState2[0]) + ","  
+ String(Vrms1[1]) + "," + String(Irms1[1]) + "," + String(cbState1[1]) + ","  
+ String(Vrms2[1]) + "," + String(Irms2[1]) + "," + String(cbState2[1]) + ","  
+ String(Vrms1[2]) + "," + String(Irms1[2]) + "," + String(cbState1[2]) + ","  
+ String(Vrms2[2]) + "," + String(Irms2[2]) + "," + String(cbState2[2]) + ","  
+ String(Vrms1[3]) + "," + String(Irms1[3]) + "," + String(cbState1[3]) + ","  
+ String(Vrms2[3]) + "," + String(Irms2[3]) + "," + String(cbState2[3]);
```

Şekil 5.6. Seri iletişim yolu ile aktarılabacak verinin tek harf dizini biçiminde düzenlenmesi

Tarih saat ve ölçüm verilerinin yan yana dizilmesi doğru ve kayıpsız olarak arayüze aktarılması için oluşturulmuş. Aralarına eklenen “,” karakteri ise *Processing* ortamında Şekil 5.7’de görüldüğü gibi “*Split*” fonksiyonu ile ayrıştırılıp uygun adreslere gönderilmelerini sağlamaktadır. Tarih ve saat bilgisinin gönderilmesi ise zaman etiketinin Scada arayüzünden değil donanım bölümünden alınmasını sağlamaktadır. Bazı senaryolarda zaman bilgisinin donanım bölümünden gelmesi istendiğinde bu özellik büyük kolaylık sağlamaktadır.

```
// trim crap  
data = trim(data);  
port.clear();  
dataCheck = data.substring(0, 5);  
String rowData = data.substring(5, data.length());  
String splitData[] = split(rowData, ',');
```

Şekil 5.7. *Processing* ortamında gelen verinin split edilmesi



## 5.2. Röle Modülü – Uydu Denetleyici 1

Çalışmamızda Akıllı Şebeke saha modelinin benzetimi için bir model şebeke sistemi oluşturulmuştur. Model şebekede 8 adet trafo bulunmaktadır. Her bir trafo ile halinin 1/100000 oranında küçültülmüş hali seçilerek kullanılmıştır. Her iki trafo analog ve sayısal bilgileri bir Arduino tarafından işlenmekte ve düzenlenmektedir.

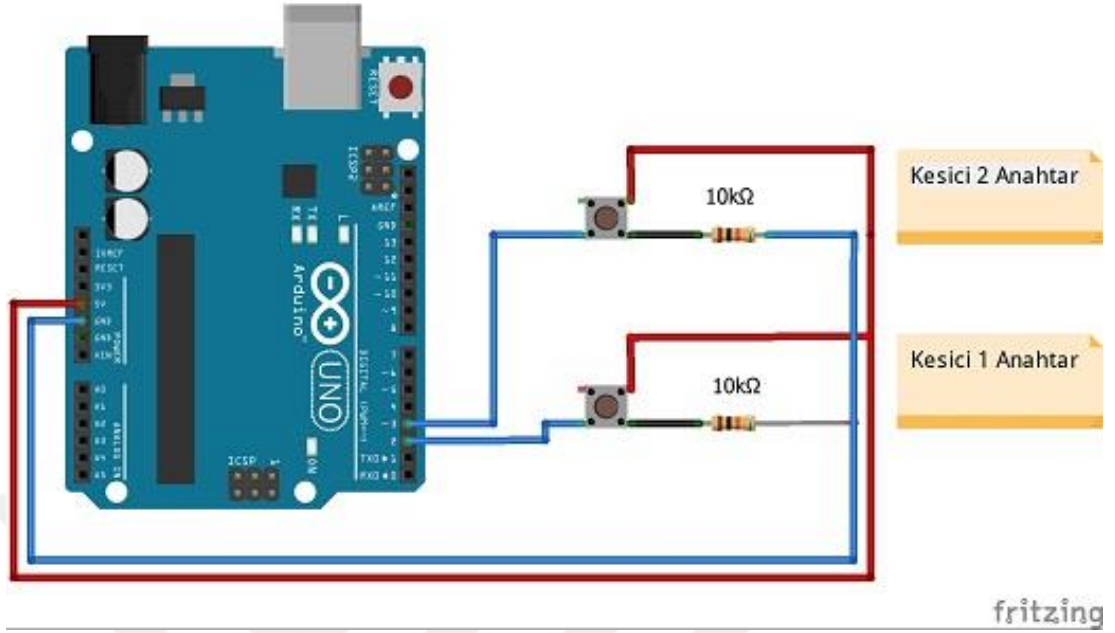
Birinci röle durumunda bulunan *Arduino* adres numarası 1 olan uydu denetleyicisi olarak görev yapmaktadır. Bu mikro denetleyici iki adet trafo için gerekli olan ölçüm ve durum bilgilerini alarak I<sup>2</sup>C dağıtım noktası üzerinden RTU'ya göndermektedir.

Birinci uydu olarak kullanılan röle modülü birinci ve ikinci trafoya ait bilgileri işlemektedir. Bu röle modülüne ölçüm değerine dönüştürülmesi gereken AC akım ve gerilim algılayıcı bilgileri ulaşmaktadır. Bu algılayıcı bilgileri 0-5V arasında olmaktadır. *Emonlib* kütüphanesi ile gerekli örneklemeler alınarak iki trafo için akım ve gerilim ölçümleri elde edilmektedir. Birinci trafo analog girişleri *Arduino* A0 ve A1 pinlerine, ikinci trafo analog girişleri ise *Arduino* A2 ve A3 pinlerine bağlanmıştır (Tamkittikhun vd, 2015a). *Arduino UNO*'da toplam 6 adet analog giriş bulunmaktadır. Geriye kalan A4 ve A5 pinleri ise I<sup>2</sup>C pinleri olarak kullanılmaktadır.

Rölelerde akım ve gerilim ölçümü AC algılayıcı ile gerçekleştirilmektedir. Ölçüm değeri *Arduino* 10 bit ADC analog girişi ile gerçekleştirildiğinden dolayı 0-1023 arasında değer vermektedir. *Arduino* analog işaretler için 20ms'de 50-100 arasında örnekleme alarak ve *Emonlib* kütüphanesini kullanarak AC akım ve gerilimi ölçer ve I<sup>2</sup>C dağıtım noktası ile ölçüm değerlerini RTU'ya ulaştırır.

Kesici Durum Bilgisi benzetimlerinin gerçekleştirilmesi için her rölede 2 adet buton anahtar kullanılmıştır. Bu anahtarlar Şekil 5.8'de görüldüğü gibi Akıllı Şebekelerde yer alan anahtarlama yapılarını temsil etmektedir ve sayısal giriş işareti vermektedir.

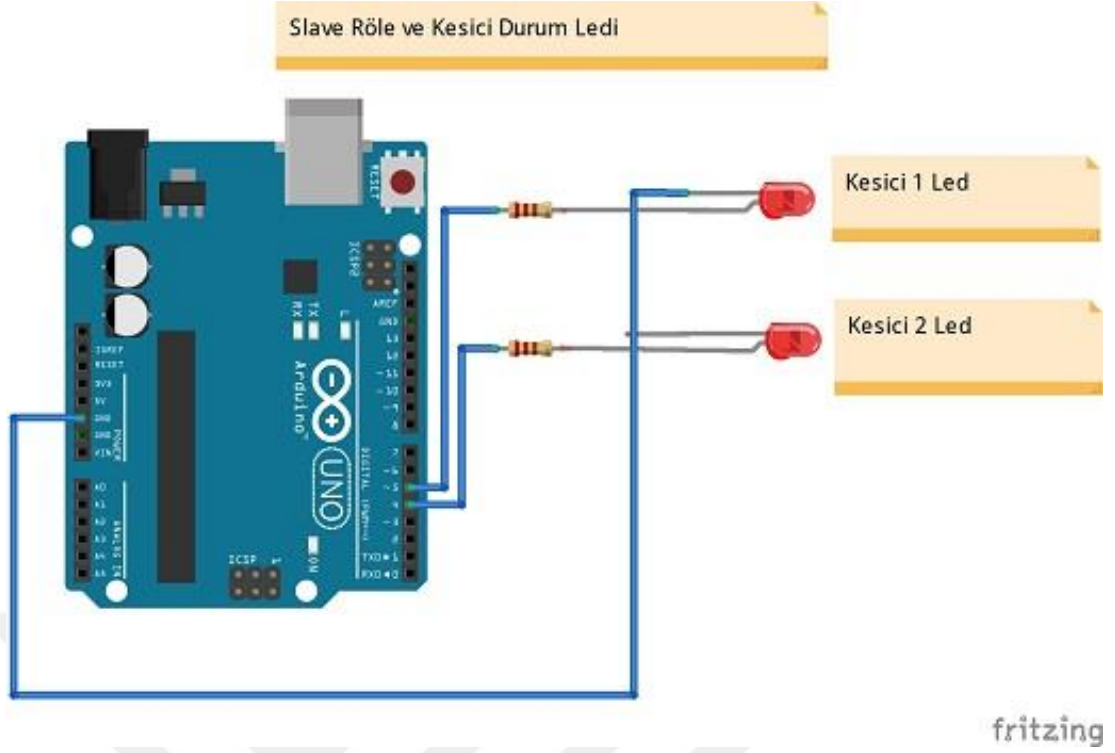
### Slave Röle ve Buton Bağlantı Şeması



Şekil 5.8. Anahtarlama yapısındaki butonlar ve Arduino bağlantısı

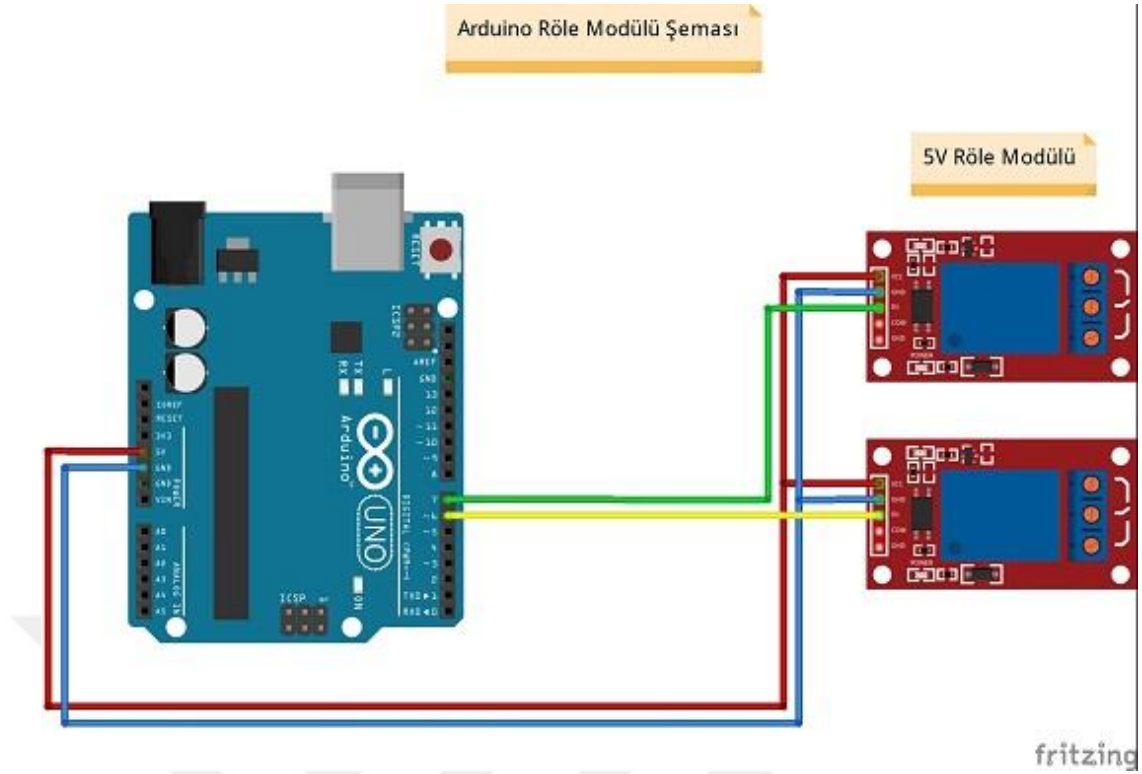
Anahtarların işaret uçları 10 k $\Omega$  değerinde dirençlerle toprağa bağlanmıştır. Bu dirençler sayısal girişlerin net “0” ve “1” şeklinde davranmalarını ve istikrarlı olmalarını sağlamaktadır. Butonlar Arduino 2. ve 3. sayısal pinlerine bağlanmıştır.

Kesici anahtarlama yapılarını sağlayan röle modülü Scada arayüzünden denetlenebildiği gibi belirtilen butonlar ile de denetlenebilmektedir. Ayrıca 4. ve 5. sayısal pinlere bağlı olan 2 adet LED anahtarlama yapılarının etkin olup olmadıklarını görsel olarak ifade etmek için kullanılmaktadır.



Şekil 5.9. Anahtarlama yapısı gösterge LED'leri ve Arduino bağlantısı

*Arduino*'ya bağlı olan röle modülü Scada arayüzünden gönderilen kesici komutlarının benzetimi için kullanılmaktadır. Trafo kesicisinin uzaktan veya el ile açılıp kapanması senaryolarını gerçekleştirmek için yerleştirilmiştir. Şekil 5.10'da görüldüğü gibi *Arduino* 6. ve 7. sayısal pinlerine bağlı olan iki adet röle modülü 10A'ye kadar dayanabilmektedir. Bu röle modülleri trafoların yüklere bağlanıp ayrılmasını elektronik denetimi şekilde gerçekleştirmektedir.



Şekil 5.10. Arduino röle modülleri bağlantısı

### 5.3. Röle Modülü – Uydu Denetleyici 2

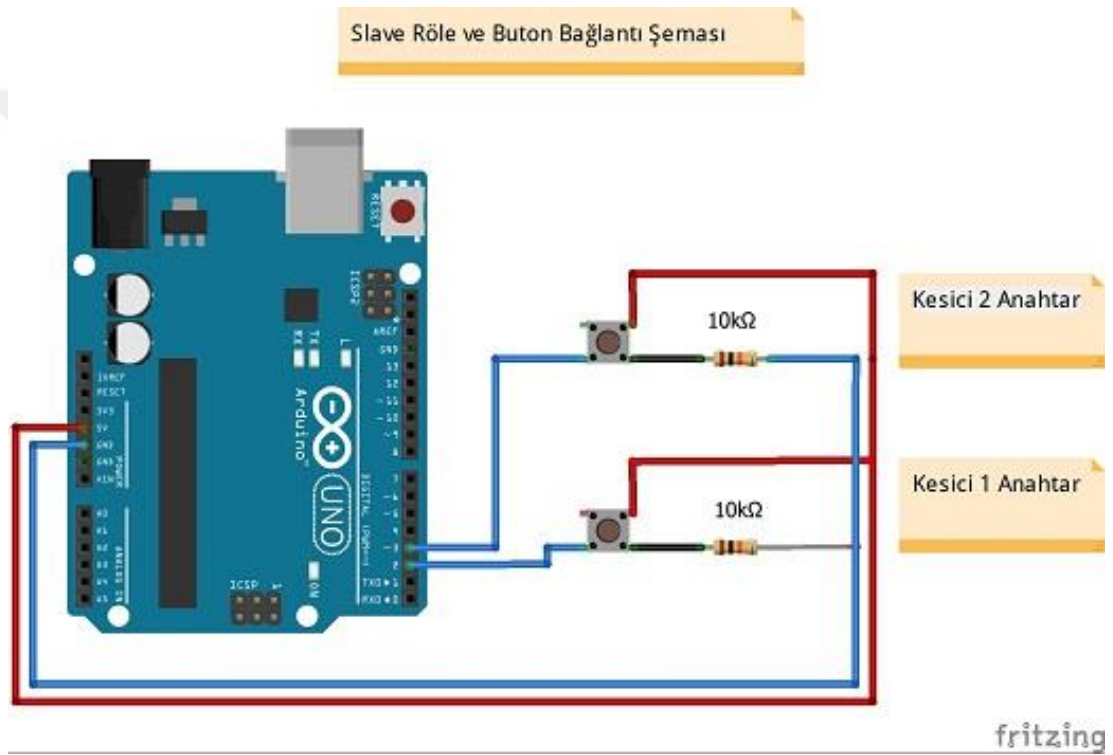
Çalışmamızda Akıllı Şebeke saha modelinin benzetimi için bir model şebeke sistemi oluşturulmuştur. Model şebekede 8 adet trafo bulunmaktadır. Her bir trafo ilk halinin 1/100000 oranında küçültülmüş hali seçilerek kullanılmıştır. Her iki trafo analog ve sayısal bilgileri bir Arduino tarafından işlenmekte ve düzenlenmektedir.

İkinci röle durumunda bulunan *Arduino* adres numarası 2 olan uydu denetleyicisi olarak görev yapmaktadır. Bu mikro denetleyici iki adet trafo için gerekli olan ölçüm ve durum bilgilerini alarak I<sup>2</sup>C dağıtım noktası üzerinden RTU'ya göndermektedir.

İkinci uydu olarak kullanılan röle modülü üçüncü ve dördüncü trafoya ait bilgileri işlemektedir. Bu röle modülüne ölçüm değerine dönüştürülmesi gereken AC akım ve gerilim algılayıcı bilgileri ulaşmaktadır. Bu algılayıcı bilgileri 0-5V arasında olmaktadır. *Emonlib* kütüphanesi ile gerekli örneklemeler alınarak iki trafo için akım ve gerilim ölçümleri elde edilmektedir. Birinci trafo analog girişleri *Arduino* A0 ve A1 pinlerine, ikinci trafo analog girişleri ise *Arduino* A2 ve A3 pinlerine bağlanmıştır. *Arduino UNO*'da toplam 6 adet analog giriş bulunmaktadır. Geriye kalan A4 ve A5 pinleri ise I<sup>2</sup>C pinleri olarak kullanılmaktadır.

Rölelerde akım ve gerilim ölçümü AC algılayıcı ile gerçekleştirilmektedir. Ölçüm değeri *Arduino* 10 bit ADC analog girişi ile gerçekleştirildiğinden dolayı 0-1023 arasında değer vermektedir. *Arduino* analog işaretler için 20ms'de 50-100 arasında örnekleme olarak ve *Emonlib* kütüphanesini kullanarak AC akım ve gerilimi ölçer ve I<sup>2</sup>C dağıtım noktası ile ölçüm değerlerini RTU'ya ulaştırır.

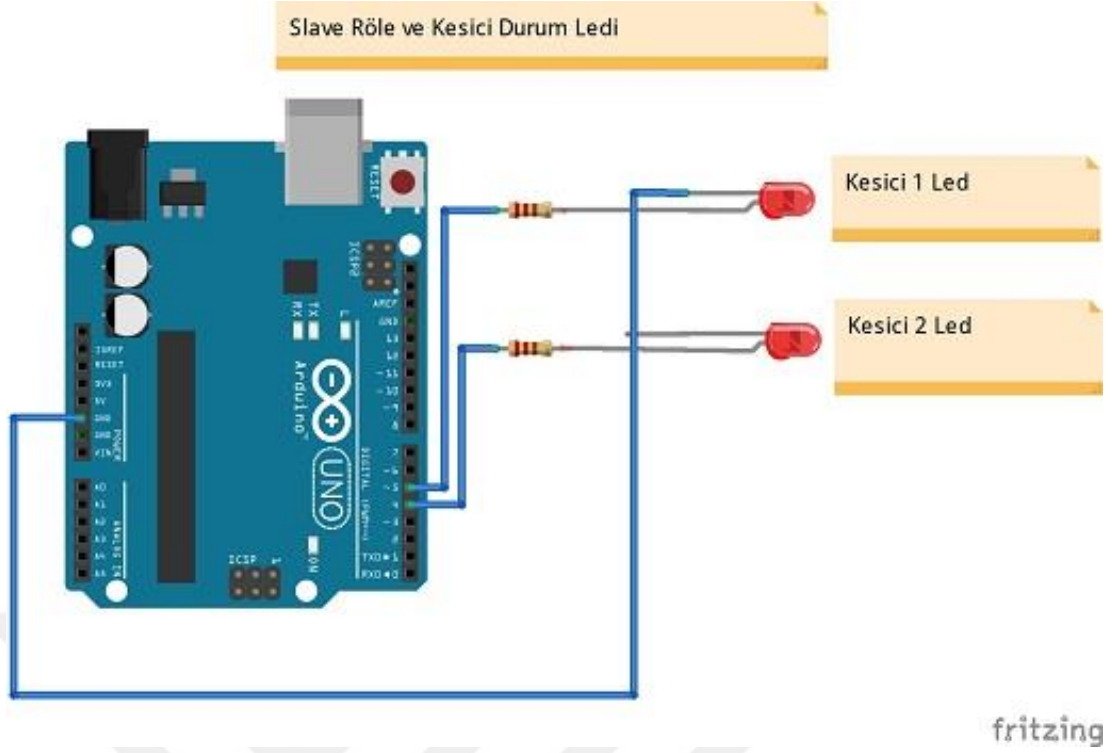
Kesici Durum Bilgisi benzetimlerinin gerçekleştirilmesi için her rölede 2 adet buton anahtar kullanılmıştır. Bu anahtarlar Şekil 5.11'de görüldüğü gibi Akıllı Şebekelerde yer alan anahtarlama yapılarını temsil etmektedir ve sayısal giriş işareti vermektedir.



Şekil 5.11. Anahtarlama yapısındaki butonlar ve Arduino bağlantısı

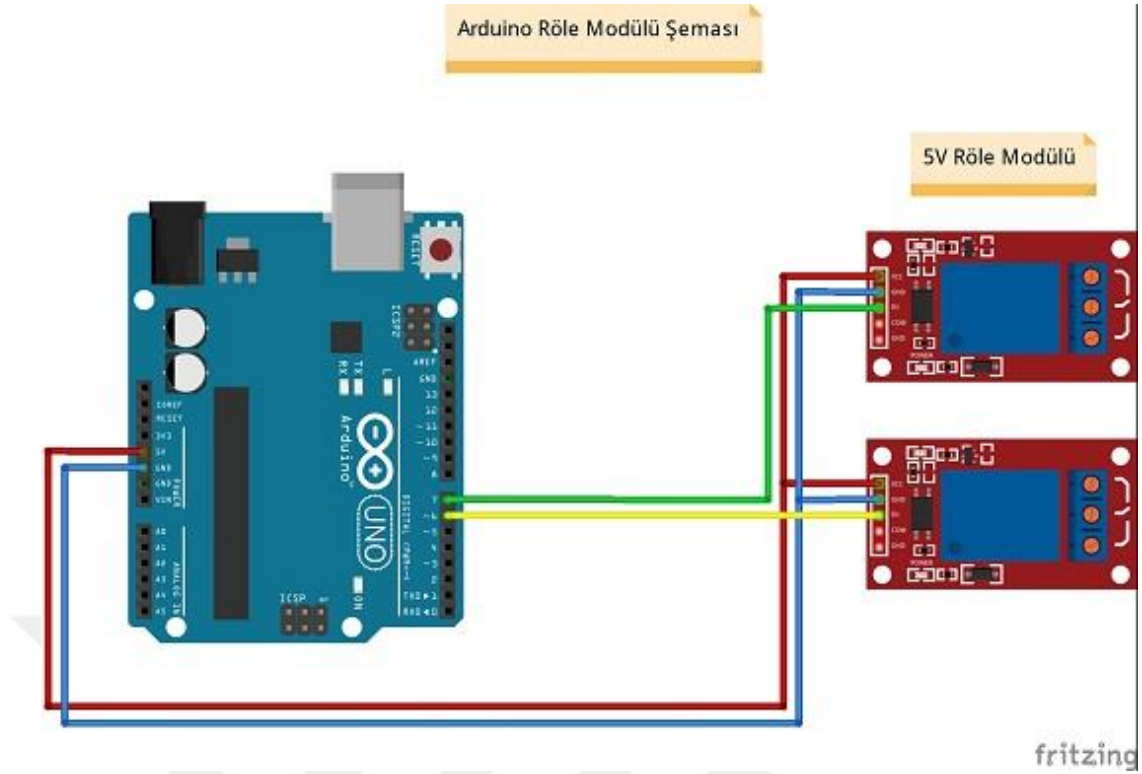
Anahtarların işaret uçları 10 k $\Omega$ 'değerinde dirençlerle toprağa bağlanmıştır. Bu dirençler sayısal girişlerin net "0" ve "1" şeklinde davranmalarını ve istikrarlı olmalarını sağlamaktadır. Butonlar *Arduino* 2. ve 3. sayısal pinlerine bağlanmıştır.

Kesici anahtarlama yapılarını sağlayan röle modülü Scada arayüzünden denetlendiği gibi belirtilen butonlar ile de denetlenebilmektedir. Ayrıca 4. ve 5. sayısal pinlere bağlı olan 2 adet LED anahtarlama yapılarının etkin olup olmadıklarını görsel olarak ifade etmek için kullanılmaktadır.



Şekil 5.12. Anahtarlama yapısı gösterge LED'leri ve Arduino bağlantısı

*Arduino*'ya bağlı olan röle modülü Scada arayüzünden gönderilen kesici komutlarının benzetimi için kullanılmaktadır. Trafo kesicisinin uzaktan veya el ile açılıp kapanması senaryolarını gerçekleştirmek için yerleştirilmiştir. Şekil 5.13'te görüldüğü gibi *Arduino* 6. ve 7. sayısal pinlerine bağlı olan iki adet röle modülü 10A'ye kadar dayanabilmektedir. Bu röle modülleri trafoların yüklere bağlanıp ayrılmasını elektronik denetimi şekilde gerçekleştirmektedir.



Şekil 5.13. Arduino röle modülleri bağlantısı

#### 5.4. Röle Modülü – Uydu Denetleyici 3

Çalışmamızda Akıllı Şebeke saha modelinin benzetimi için bir model şebeke sistemi oluşturulmuştur. Model şebekede 8 adet trafo bulunmaktadır. Her bir trafo ilk halinin 1/100000 oranında küçültülmüş hali seçilerek kullanılmıştır. Her iki trafo analog ve sayısal bilgileri bir Arduino tarafından işlenmekte ve düzenlenmektedir.

Üçüncü röle durumunda bulunan *Arduino* adres numarası 3 olan uydu denetleyicisi olarak görev yapmaktadır. Bu mikro denetleyici iki adet trafo için gerekli olan ölçüm ve durum bilgilerini alarak I<sup>2</sup>C dağıtım noktası üzerinden RTU'ya göndermektedir.

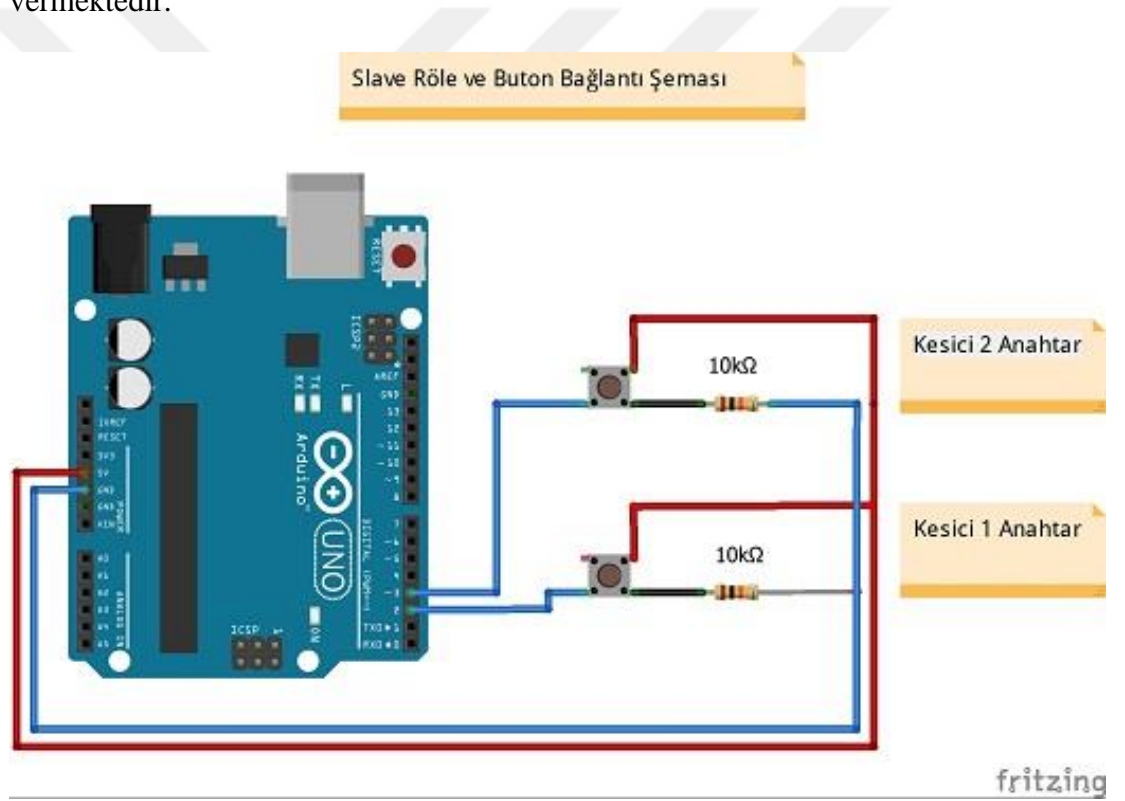
Üçüncü uydu olarak kullanılan röle modülü beşinci ve altıncı trafoya ait bilgileri işlemektedir. Bu röle modülüne ölçüm değerine dönüştürülmesi gereken AC akım ve gerilim algılayıcı bilgileri ulaşmaktadır. Bu algılayıcı bilgileri 0-5V arasında olmaktadır. *Emonlib* kütüphanesi ile gerekli örneklemeler alınarak iki trafo için akım ve gerilim ölçümleri elde edilmektedir. Birinci trafo analog girişleri Arduino A0 ve A1 pinlerine, ikinci trafo analog girişleri ise *Arduino* A2 ve A3 pinlerine bağlanmıştır.



*Arduino UNO*'da toplam 6 adet analog giriş bulunmaktadır. Geriye kalan A4 ve A5 pinleri ise I<sup>2</sup>C pinleri olarak kullanılmaktadır.

Rölelerde akım ve gerilim ölçümü AC algılayıcı ile gerçekleştirilmektedir. Ölçüm değeri *Arduino* 10 bit ADC analog girişi ile gerçekleştirildiğinden dolayı 0-1023 arasında değer vermektedir. *Arduino* analog işaretler için 20ms'de 50-100 arasında örnekleme alarak ve *Emonlib* kütüphanesini kullanarak AC akım ve gerilimi ölçer ve I<sup>2</sup>C dağıtım noktası ile ölçüm değerlerini RTU'ya ulaştırır.

Kesici Durum Bilgisi benzetimlerinin gerçekleştirilmesi için her rölede 2 adet buton anahtar kullanılmıştır. Bu anahtarlar Şekil 5.14'te görüldüğü gibi Akıllı Şebekelerde yer alan anahtarlama yapılarını temsil etmektedir ve sayısal giriş işareti vermektedir.



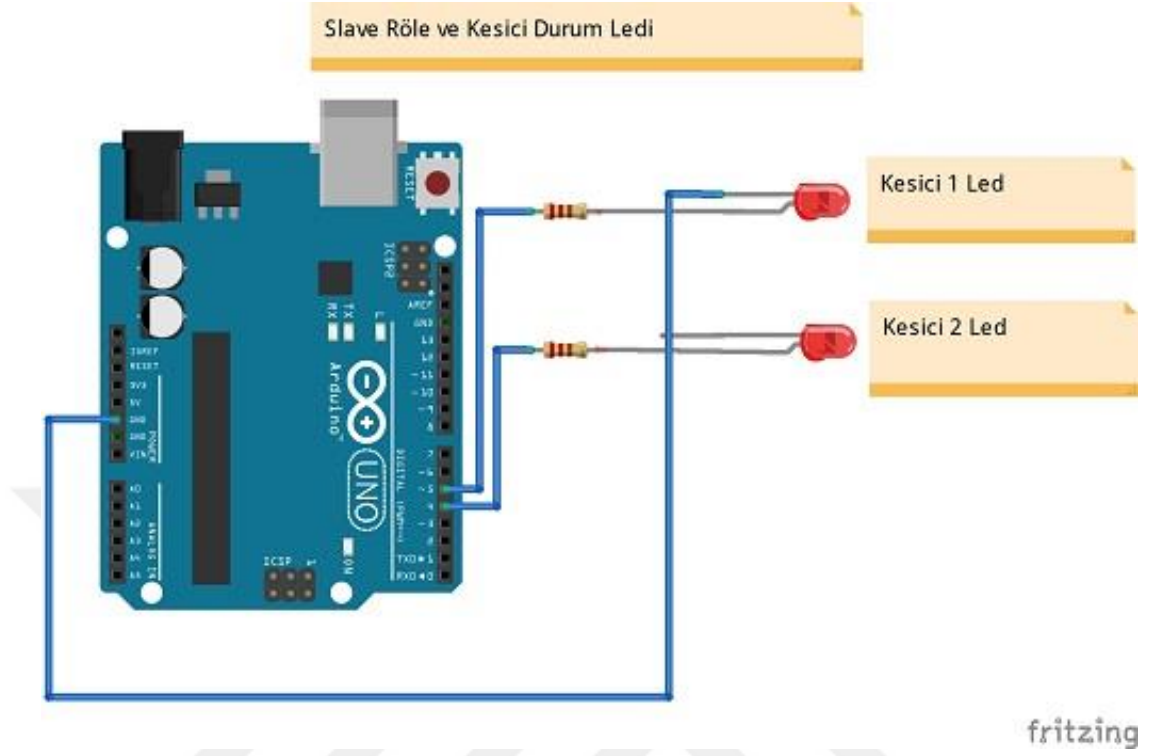
Şekil 5.14. Anahtarlama yapısındaki butonlar ve *Arduino* bağlantısı

Anahtarların işaret uçları 10 k $\Omega$ 'değerinde dirençlerle toprağa bağlanmıştır. Bu dirençler sayısal girişlerin net "0" ve "1" şeklinde davranmalarını ve istikrarlı olmalarını sağlamaktadır. Butonlar *Arduino* 2. ve 3. sayısal pinlerine bağlanmıştır.

Kesici anahtarlama yapılarını sağlayan röle modülü Scada arayüzünden denetlenebildiği gibi belirtilen butonlar ile de denetlenebilmektedir. Ayrıca 4. ve 5.

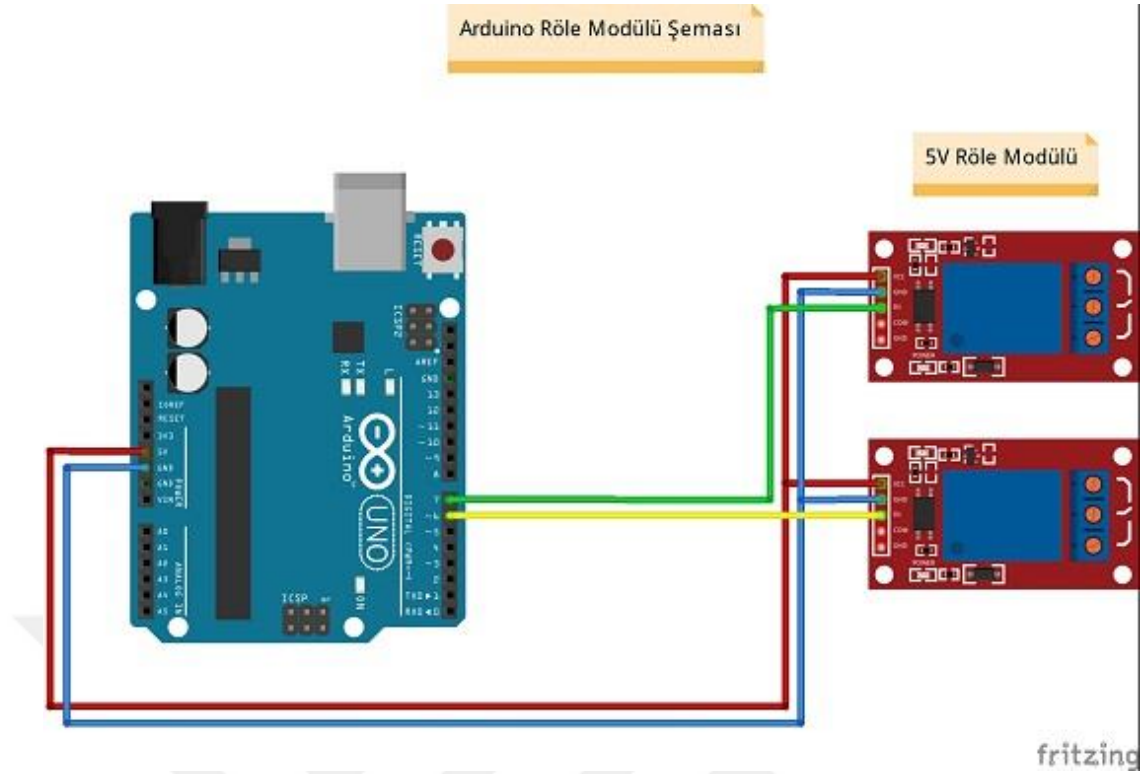


sayısal pinlere bağı olan 2 adet LED anahtarlama yapılarının etkin olup olmadıklarını görsel olarak ifade etmek için kullanılmaktadır.



Şekil 5.15. Anahtarlama yapısı gösterge LED'leri ve Arduino bağlantısı

*Arduino*'ya bağı olan röle modülü Scada arayüzünden gönderilen kesici komutlarının benzetimi için kullanılmaktadır. Trafo kesicisinin uzaktan veya el ile açılıp kapanması senaryolarını gerçekleştirmek için yerleştirilmiştir. Şekil 5.16'da görüldüğü gibi *Arduino* 6. ve 7. sayısal pinlerine bağı olan iki adet röle modülü 10A'e kadar dayanabilmektedir. Bu röle modülleri trafoların yüklere bağlanıp ayrılmasını elektronik denetimi şekilde gerçekleştirmektedir.



Şekil 5.16. Arduino röle modülleri bağlantısı

#### 5.5. Röle Modülü – Uydu Denetleyici 4

Çalışmamızda Akıllı Şebeke saha modelinin benzetimi için bir model şebeke sistemi oluşturulmuştur. Model şebekede 8 adet trafo bulunmaktadır. Her bir trafo ilk halinin 1/100000 oranında küçültülmüş hali seçilerek kullanılmıştır. Her iki trafo analog ve sayısal bilgileri bir Arduino tarafından işlenmekte ve düzenlenmektedir.

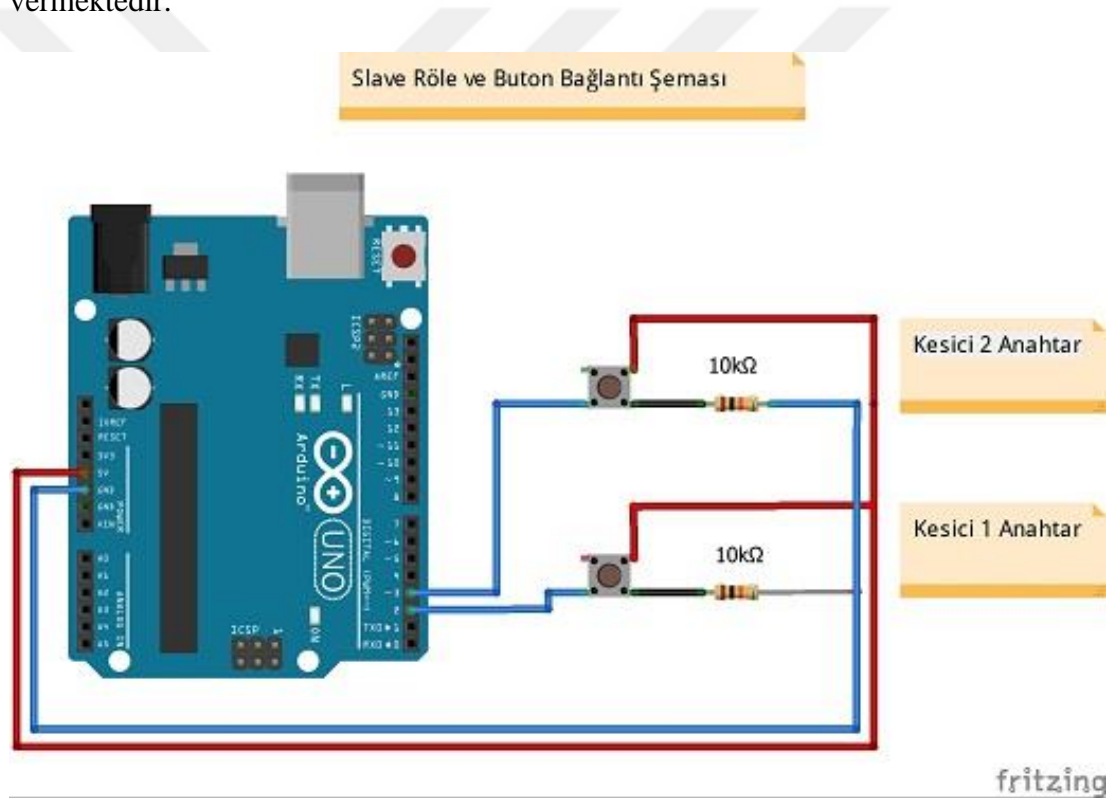
Dördüncü röle durumunda bulunan *Arduino* adres numarası 4 olan uydu denetleyicisi olarak görev yapmaktadır. Bu mikro denetleyici iki adet trafo için gerekli olan ölçüm ve durum bilgilerini alarak I<sup>2</sup>C dağıtım noktası üzerinden RTU'ya göndermektedir.

Dördüncü uydu olarak kullanılan röle modülü yedinci ve sekizinci trafoya ait bilgileri işlemektedir. Bu röle modülüne ölçüm değerine dönüştürülmesi gereken AC akım ve gerilim algılayıcı bilgileri ulaşmaktadır. Bu algılayıcı bilgileri 0-5V arasında olmaktadır. *Emonlib* kütüphanesi ile gerekli örneklemeler alınarak iki trafo için akım ve gerilim ölçümleri elde edilmektedir. Birinci trafo analog girişleri *Arduino* A0 ve A1 pinlerine, ikinci trafo analog girişleri ise *Arduino* A2 ve A3 pinlerine bağlanmıştır.

*Arduino UNO*'da toplam 6 adet analog giriş bulunmaktadır. Geriye kalan A4 ve A5 pinleri ise I<sup>2</sup>C pinleri olarak kullanılmaktadır.

Rölelerde akım ve gerilim ölçümü AC algılayıcı ile gerçekleştirilmektedir. Ölçüm değeri *Arduino* 10 bit ADC analog girişi ile gerçekleştirildiğinden dolayı 0-1023 arasında değer vermektedir. *Arduino* analog işaretler için 20ms'de 50-100 arasında örnekleme alarak ve *Emonlib* kütüphanesini kullanarak AC akım ve gerilimi ölçer ve I<sup>2</sup>C dağıtım noktası ile ölçüm değerlerini RTU'ya ulaştırır.

Kesici Durum Bilgisi benzetimlerinin gerçekleştirilmesi için her rölede 2 adet buton anahtar kullanılmıştır. Bu anahtarlar Şekil 5.17'de görüldüğü gibi Akıllı Şebekelerde yer alan anahtarlama yapılarını temsil etmektedir ve sayısal giriş işareti vermektedir.

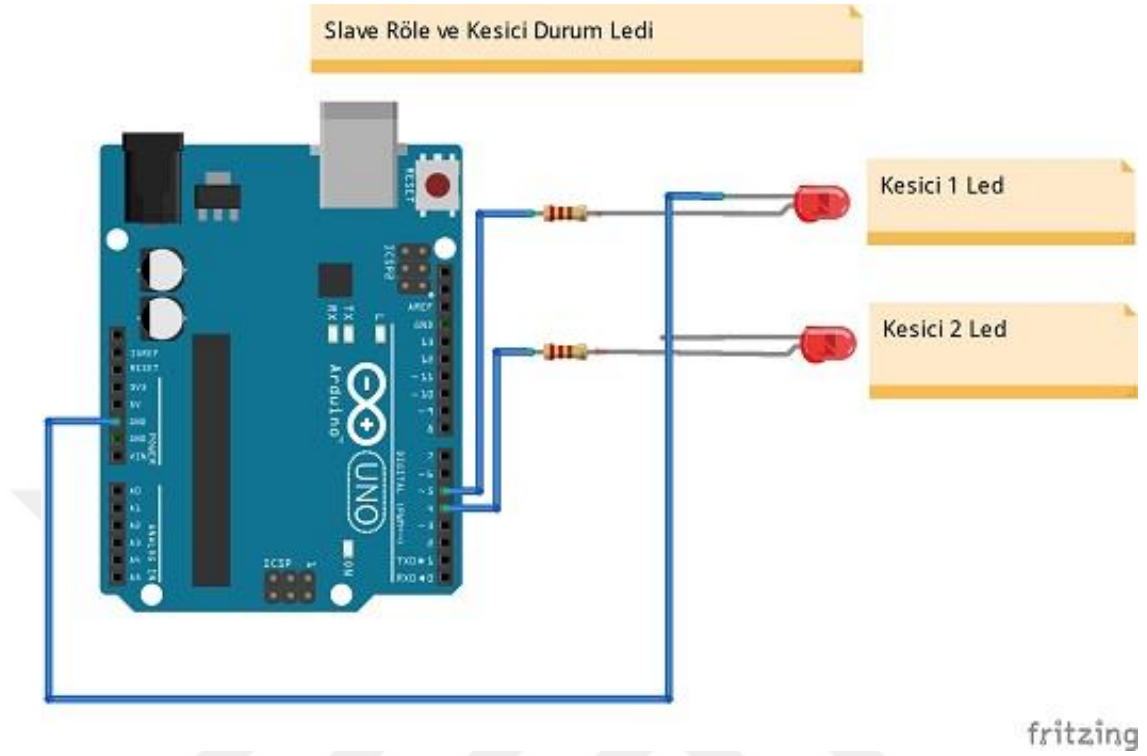


Şekil 5.17. Anahtarlama yapısındaki butonlar ve *Arduino* bağlantısı

Anahtarların işaret uçları 10 k $\Omega$  değerinde dirençlerle toprağa bağlanmıştır. Bu dirençler sayısal girişlerin net “0” ve “1” şeklinde davranmalarını ve istikrarlı olmalarını sağlamaktadır. Butonlar *Arduino* 2. ve 3. Sayısal pinlerine bağlanmıştır.

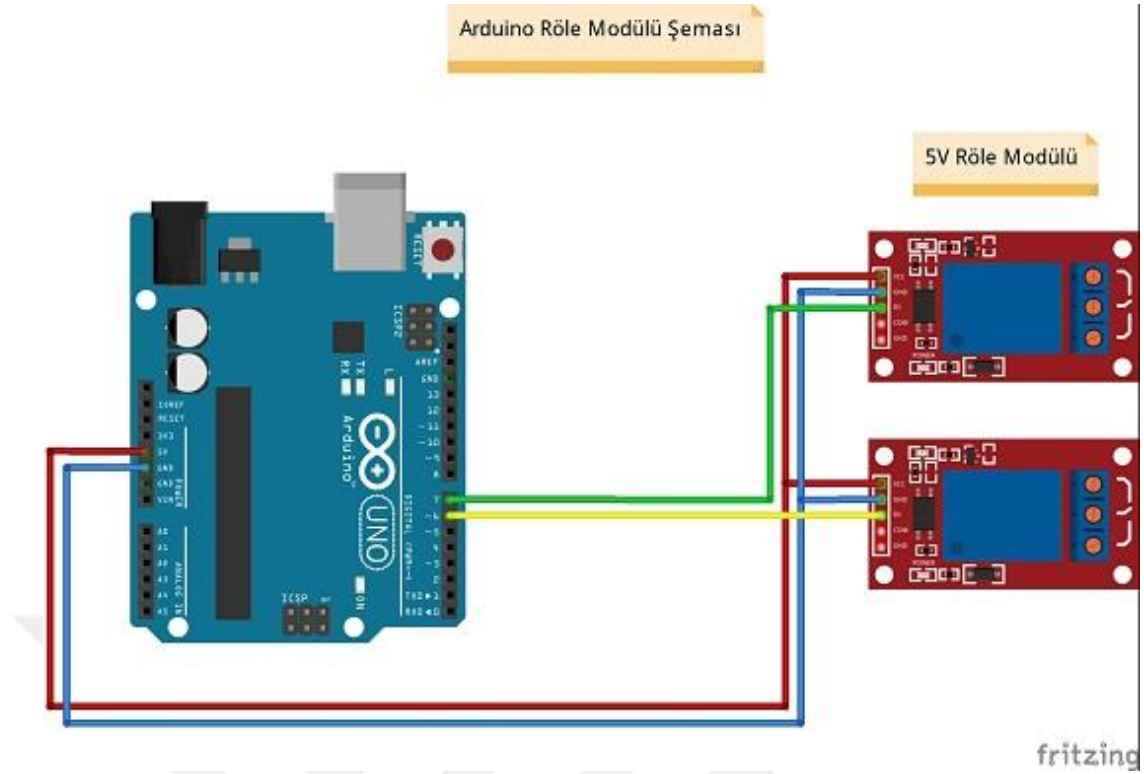
Kesici anahtarlama yapılarını sağlayan röle modülü Scada arayüzünden denetlenebildiği gibi belirtilen butonlar ile de denetlenebilmektedir. Ayrıca 4. ve 5.

sayısal pinlere bağı olan 2 adet LED anahtarlama yapılarının etkin olup olmadıklarını görsel olarak ifade etmek için kullanılmaktadır.



Şekil 5.18. Anahtarlama yapısı gösterge LED'leri ve Arduino bağlantısı

Arduino'ya bağı olan röle modülü Scada arayüzünden gönderilen kesici komutlarının benzetimi için kullanılmaktadır. Trafo kesicisinin uzaktan veya el ile açılıp kapanması senaryolarını gerçekleştirmek için yerleştirilmiştir. Şekil 5.19'da görüldüğü gibi Arduino 6. ve 7. sayısal pinlerine bağı olan iki adet röle modülü 10A'e kadar dayanabilmektedir. Bu röle modülleri trafoların yüklere bağlanıp ayrılmasını elektronik denetimi şekilde gerçekleştirmektedir.



Şekil 5.19. Arduino röle modülleri bağlantısı

## 5.6. GSM Modülü – Uydu Denetleyici 5

Elektrik şebekelerinin daha teknolojik oluşumlarla desteklenmesi birçok alanda olduğu gibi arızalara müdahale etmeyi de kolaylaştırmaktadır. Özellikle saha çalışanlarının ve uygulama mühendislerinin oluşan arızalardan ve arızanın oluşum noktasından hızlı bir şekilde haberdar olması kesinti sürelerini büyük ölçüde kısaltmaktadır. Gerçek hayatta çok önemli yer tutan bu özelliği deney ortamımızda benzetimi için RTU olarak kullanılan Arduino'ya bir GSM Modem bağlantısı gerçekleştirdik (Visalatchi ve Sandeep, 2017). GSM Modem önceden belirlenmiş olan telefon numaralarına Elektrik Güç Kalitesi (Chattopadhyay vd, 2011) verilerine göre oluşturduğumuz Senaryolara bağlı bilgilendirme mesajları gönderebilmektedir.

Çalışmada Şekil 5.20'de görüldüğü gibi *Cinterion MC35i* tipinde GSM modem kullanılmıştır. GSM modem Arduino arasındaki bağlantı seri iletişim ile sağlanmaktadır. GSM modem üzerinde yer alan RS232 portundan *RX* ve *TX* pinleri kullanılarak bağlantı gerçekleştirilmiştir (Anonymous, 2018). Arduino UNO kartı üzerinde sayısal 0. ve 1. Pinleri, seri iletişim için kullanılan TTL (0-5V) seviyesinde çalışan sırasıyla *RX* ve *TX* pinleridir.



Şekil 5.20. Cinterion MC35i GSM modem

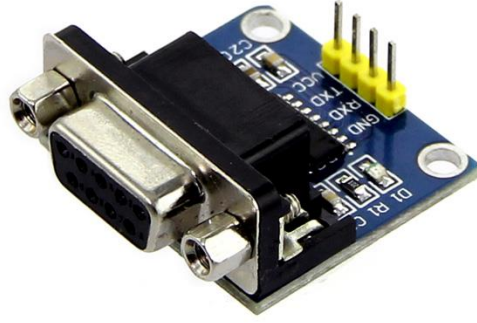
GSM modem üzerinde yer alan RS232 portu kullanılarak oluşturulan seri iletişim bağlantısında port pinlerinden Şekil 5.21’de görüldüğü gibi 2, 3 ve 5. pinler kullanılmıştır. 2 numaralı pin RX, 3 numaralı pin TX ve 5 numaralı pin GND pinleridir (Anonymous, 2008).



Şekil 5.21. GSM modem RS232 pinleri (Anonymous, 2008)

GSM modem RS232 seri bağlantı işaret seviyesi  $-10V - 10V$  arasında olduğundan (Anonymous, 2008) bu gerilim seviyesi Arduino’nun kullandığı  $0-5V$  aralığına dönüştürülmesi gerekmektedir. Gerilim seviye dönüştürme işlemi Şekil 5.22’de görüldüğü gibi RS232-TTL dönüştürücüsü ile sağlanmıştır. Kullanılan ara kart sayesinde Arduino ve GSM modem sağlıklı bir şekilde iletişim kurabilip gerekli mesajları gönderebilmektedir.

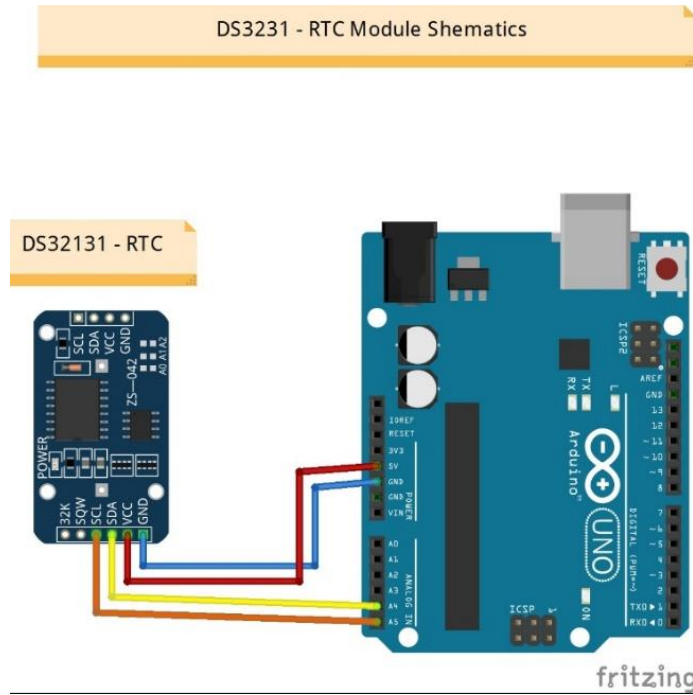




Şekil 5.22. RS232-TTL dönüştürücü

### 5.7. Gerçek Zaman Modülü – DS3231 Çalışması ve Görevi

Verilerin Scada arayüzünde gözlemlenirken doğruluğunun teyit edilmesinde ve veri tabanına kaydedilmesi esnasında zaman bilgisine ihtiyaç duyulmaktadır. Scada arayüzü bilgisayar ortamında oluşturulup kullanıldığından dolayı zaman bilgisi bilgisayar üzerinden de alınabilmektedir. Ancak Akıllı Şebeke uygulamalarında zaman bilgisi genellikle RTU'dan alınmaktadır. Bu durumun benzetimi için donanım bölümünden bulunan I<sup>2</sup>C dağıtım noktasına DS3231 RTC modülü bağlanmıştır. DS3231 Arduino bağlantısı Şekil 5.23'te görülmektedir.



Şekil 5.23. DS3231- RTC Modülü bağlantı çizelgesi

DS3231 düşük maliyetli, doğruluk derecesi yüksek, I<sup>2</sup>C tabanlı gerçek zaman modülüdür. Devre kartı üzerinde bulunan düğme pil sayesinde kartın enerjisiz bulunduğu durumlarda dahi saat bilgisini hafızasında tutabilmekte ve böylece tekrar ayar gerekmemektedir. Bu modül sayesinde RTU donanım bölümünden topladığı verileri zaman etiketi ile etiketleyerek arayüze gönderebilmektedir.

RTU tarafından arayüze gönderilen zaman etiketi ilk olarak ana Scada penceresinde görüntülenmektedir. Bu durum aradaki iletişim doğruluğu hakkında bilgi verirken gelen verilerin eski veri olmadığı yeni güncel bilgi olduğunu göstermektedir.

Zaman etiketi veri tabanına aktarılan verilerin bir düzen zaman bilgisi ile kaydedilmesini de sağlamaktadır. Zaman bilgisi verilerin hem belli bir düzene göre kaydedilmesini sağlarken geçmişe yönelik rapor ve analizleri için de büyük kolaylıklar sağlamaktadır.

## 5.8. Güç İşaretinin Dağıtılması ve Ölçülmesi

Donanım bölümünde mikro denetleyici ve algılayıcılar için gerekli olan DC akım ve gerilim 12V 5A güç kaynağı tarafından sağlanmaktadır. 12V güç kaynağına bağlı 5V 3A değerinde bir DC güç çıkışı bulunmaktadır. Bu çıkış 5V gerilim ihtiyacını karşılamaktadır. Yükleri beslemek için ise trafolar kullanılmıştır. Trafoların güç, gerilim ve akım değerleri Şekil 5.24'te görülmektedir. Tabloda her bir trafoya bağlanması gereken yük değerlerinin hesapları da yer almaktadır.

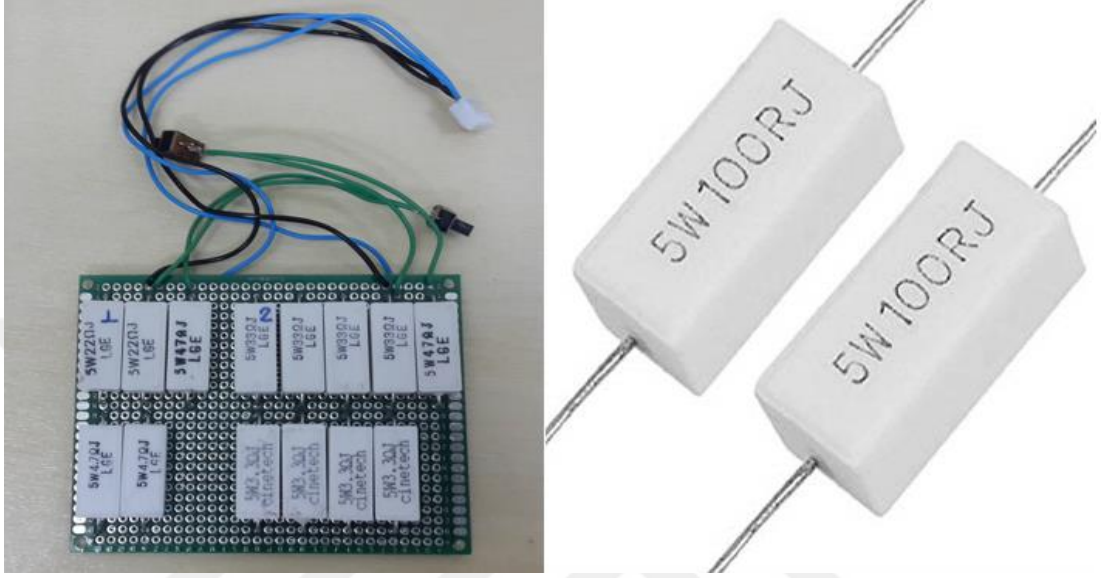
VERİLER	TR1	TR2	TR3	TR4	TR5	TR6	TR7	TR8
Güç (VA) Nominal	10,0	16,0	8,0	8,0	3,2	8,0	2,6	8,0
Gerilim (V) Nominal	12,0	12,0	12,0	12,0	12,0	12,0	12,0	12,0
Tam Yük Akımı(A)	0,833	1,333	0,667	0,667	0,267	0,667	0,217	0,667
Yük Direnç Sayısı	2	4	2	2	1	2	1	2
Yük Direnci	14,4	9,0	18,0	18,0	45,0	18,0	55,4	18,0
Gerçek En Büyük Yük Direnci	26,7	36,3	33,0	33,0	47,0	33,0	57,0	33,0
Gerçek Yük Direnci	13,4	9,1	16,5	16,5	47,0	16,5	57,0	16,5
Gerçek Tam Yük Akımı(A)	0,899	1,322	0,727	0,727	0,255	0,727	0,211	0,727
Gerçek Yük Direnci(W)	10,8	15,9	8,7	8,7	3,1	8,7	2,5	8,7
Güç Farkı	0,8	- 0,1	0,7	0,7	- 0,1	0,7	- 0,1	0,7
Gerekli Direnç	22+4,7	33+3,3	33,0	33,0	47,0	33,0	47+10	33,0
Fazladan Eklenen Direnç	47	47	100	100	220	100	220	100

Şekil 5.24. Trafo karakteristik tablosu

Yük olarak 5W gücünde ve çeşitli değerlerde seramik dirençler kullanılmıştır. Şekil 5.24'te görüldüğü gibi yük seçimi tüm trafoları tam olarak yükleyebilecek şekilde seçilmiştir. Trafolara bağlanan direnç yükleri TR1:26,7Ω+47Ω,



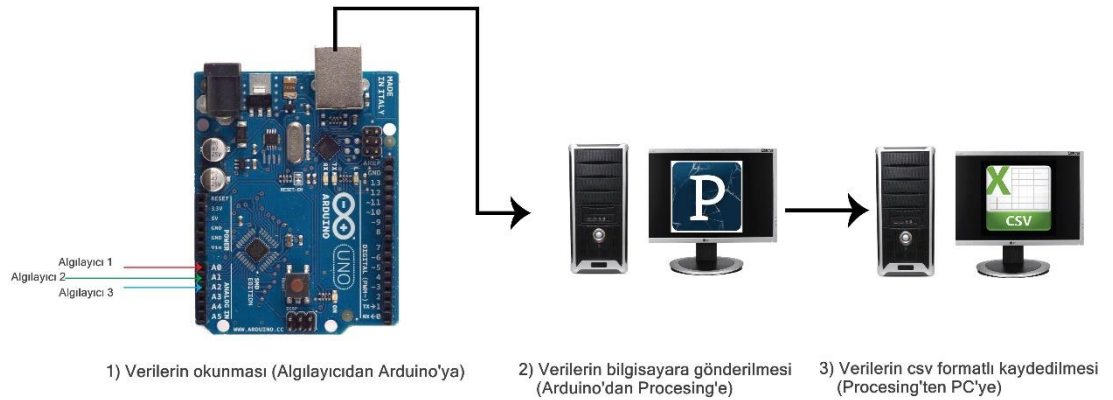
TR2:36,3Ω+47Ω, TR3:33Ω+100Ω, TR4:33Ω+100Ω, TR5:47Ω+220Ω, TR6:33Ω+100Ω, TR7:57Ω+220Ω, TR8:33Ω+100Ω şeklindedir. Bu durumu gerçekleştirmek için çeşitli değerlerde dirençler seri veya paralel bağlanmıştır. 1 ve 2 numaralı trafolar için düzenlenmiş 26,7Ω ve 36,3Ω değerinde direnç yükler Şekil 5.25'te görülmektedir.



Şekil 5.25. Model şebekede kullanılan örnek yük ve dirençler

## 5.9. Donanımsal ve Yazılımsal Sistemler Arasındaki İletişim

Donanımsal bölüm ile Scada arayüzü arasındaki iletişimi RTU sağlamaktadır. RTU bölümünde yer alan Arduino üzerindeki USB portu sayesinde seri olarak iletişim sağlanmaktadır. RTU ve Scada arayüzü arasındaki iletişim Şekil 5.26'da görülmektedir.



Şekil 5.26. Processing ve Arduino bağlantı çizelgesi

Arduino üzerinde bulunan ATmega328 mikro denetleyicisi UART TTL (5V) seri iletişimi desteklemektedir. Bu bağlantı Arduino kartının 0 (RX) ve 1 (TX) sayısal pinleri ile sağlanmaktadır. Kart üzerinde yer alan ATmega16U2 mikro denetleyicisi bu iletişimi USB biçimine dönüştürerek cihazın bilgisayarda sanal com port olarak gözükmelerini sağlamaktadır (Anonymous, 2018). Böylece Arduino’da elde ettiğimiz veriler USB portu üzerinden seri iletişim yolu ile arayüze iletilmektedir. Çalışmamızda Arduino ile Scada arayüzü arasındaki iletişim için kullandığımız seri iletişim parametreleri aşağıdaki şekildedir:

- ✓ Port Name: **COM4** (Seçilen seri bağlantı noktası)
- ✓ Baud Rate: **9600** (Seçilen seri bağlantı hızı)
- ✓ Parity: **None**
- ✓ Data Bits: **8**
- ✓ Stop Bits: **One**

RTU DS3231 RTC modülünden ve I<sup>2</sup>C iletişim protokolü ile uydu adreslerinden elde ettiği verileri harf dizini halinde yan yana sıralamaktadır. Verilerin aralarına “,” karakteri eklenerek arayüzde ayrılmalarına olanak sağlamaktadır. RTU bir saniyelik dönemlerle tüm harf dizini verisini arayüze göndermektedir.

### 5.10. Scada Yazılımında Verilerin İşlenmesi

Scada arayüzünde verilerin işleme başlamasından önce arayüzün veri tabanı ve USB portu üzerinden RTU ile iletişim kurması gerekmektedir. USB portu ile iletişim sağlanabilmesi için *Processing* uygulamasındaki “*SerialPort*” fonksiyonu kullanılmaktadır. Bu fonksiyon USB üzerinden seri iletişim sağlanması için gerekli tüm parametrelerin girilmesine olanak sağlayarak güvenli bir iletişim ortamı oluşturmaktadır. *MySQL* veri tabanı ile iletişim ise “*Wamp Server*” kullanılmaktadır. *Wamp Server* Windows için hazırlanmış bir kurulum paketidir. Bir yerel çalışma programı olarak çalışmaktadır. Diğer programlardan farkı ise içerisinde *Mysql*, *Apache*, *Php* kurulumları bulunmaktadır.

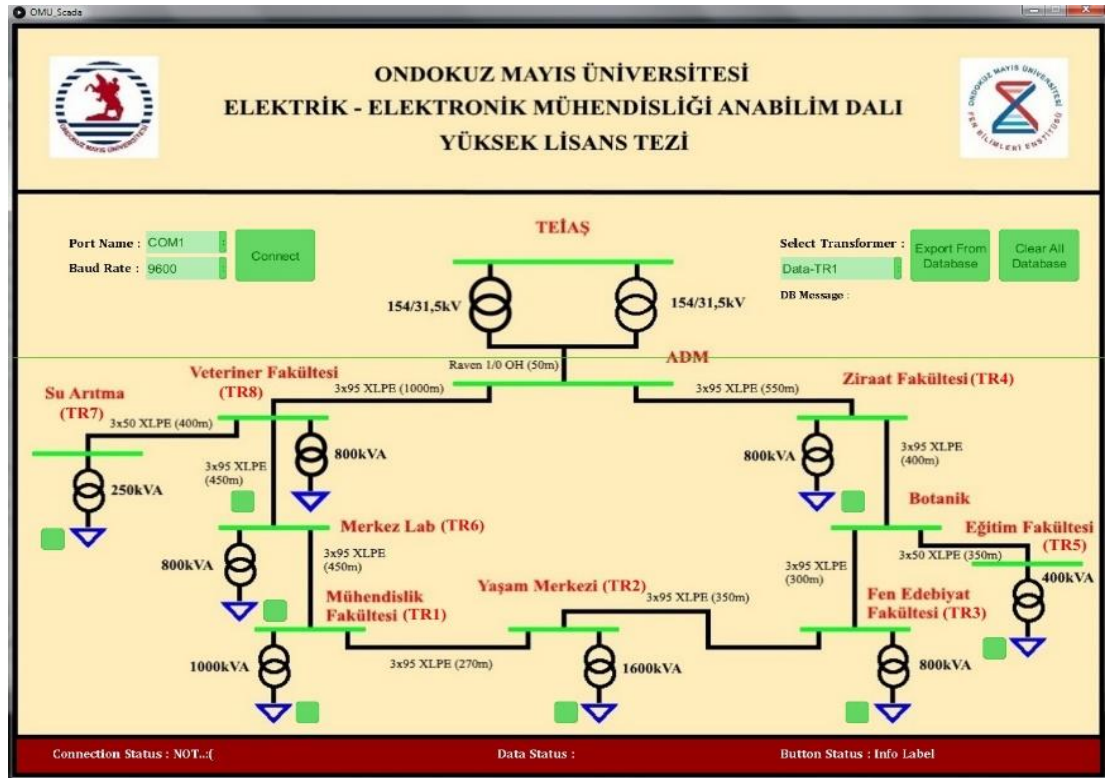
Scada arayüzü RTU ve veri tabanı ile olan iletişimi denetler. Bu iletişimler uygun bir şekilde bulunuyorsa verileri almaya başlar. RTU’dan harf dizini şeklinde gelen veriler arayüzde “*Split*” fonksiyonu ile “,” karakterine göre ayrılmaktadır.

Ayrılan bu veriler arayüzde uygun bir şekilde görüntülenebilmeleri için yönlendirilmektedir. Ayırma komutu ile birbirinden ayrılan veriler yeni bir diziyi oluşturacak şekilde düzenlenir. Bu dizinin elemanlarının arayüzde uygun yerlerde görüntülenmesi sağlanarak donanım bölümündeki ölçüm ve durum bilgilerinin arayüzde anlık olarak görüntülenmesi sağlanmış olur.

### 5.11. Ölçüm Verilerinin Arayüzde Görüntülenmesi ve Denetim İşaretleri

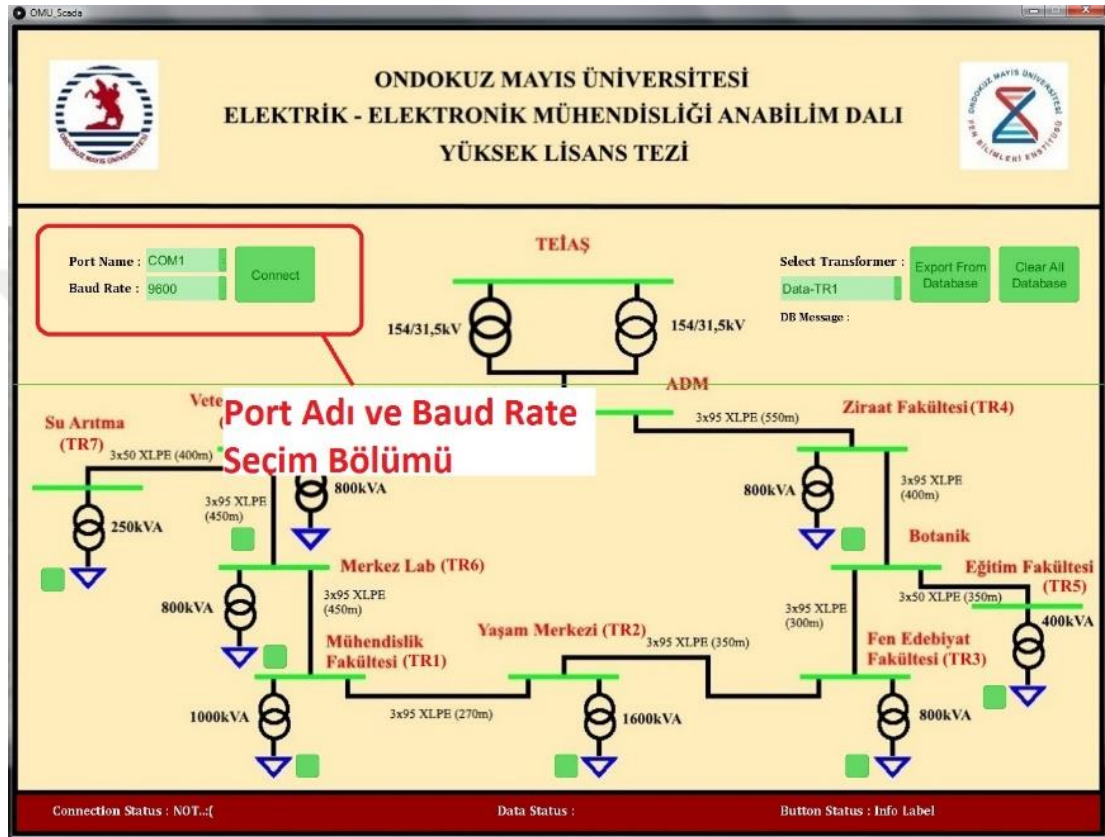
Scada uygulaması için oluşturulan arayüz ilk açılışta RTU'ya seri iletişim yolu ile bağlanmamış durumdadır. Scada arayüzü genel görüntüsü Şekil 5.27'de görülmektedir. Arayüz şebekenin genel görüntüsünü gösterecek şekilde tasarlanmıştır. Şebeke diyagramında trafoların gerçek isim ve güçleri belirtilmiş haldedir. Arayüzün sol üst köşesinde seri iletişim yolu ile bağlantı seçenekleri yer almaktadır. Sağ üst köşede ise veri tabanı ile ilgili işlemler yer almaktadır. En alt bölüm ise durum çubuğu gibi RTU bağlantı durumu ve gelen verinin güvenilirliği hakkında bilgi vermektedir.

Scada arayüzü durum çubuğu RTU'ya bağlı olmadığı durumda kırmızı renk olacak şekilde düzenlenmiştir. Böylece kullanıcının bağlantı durumunu takip etmesi kolaylaşmaktadır.



Şekil 5.27. Scada arayüz genel görünüm

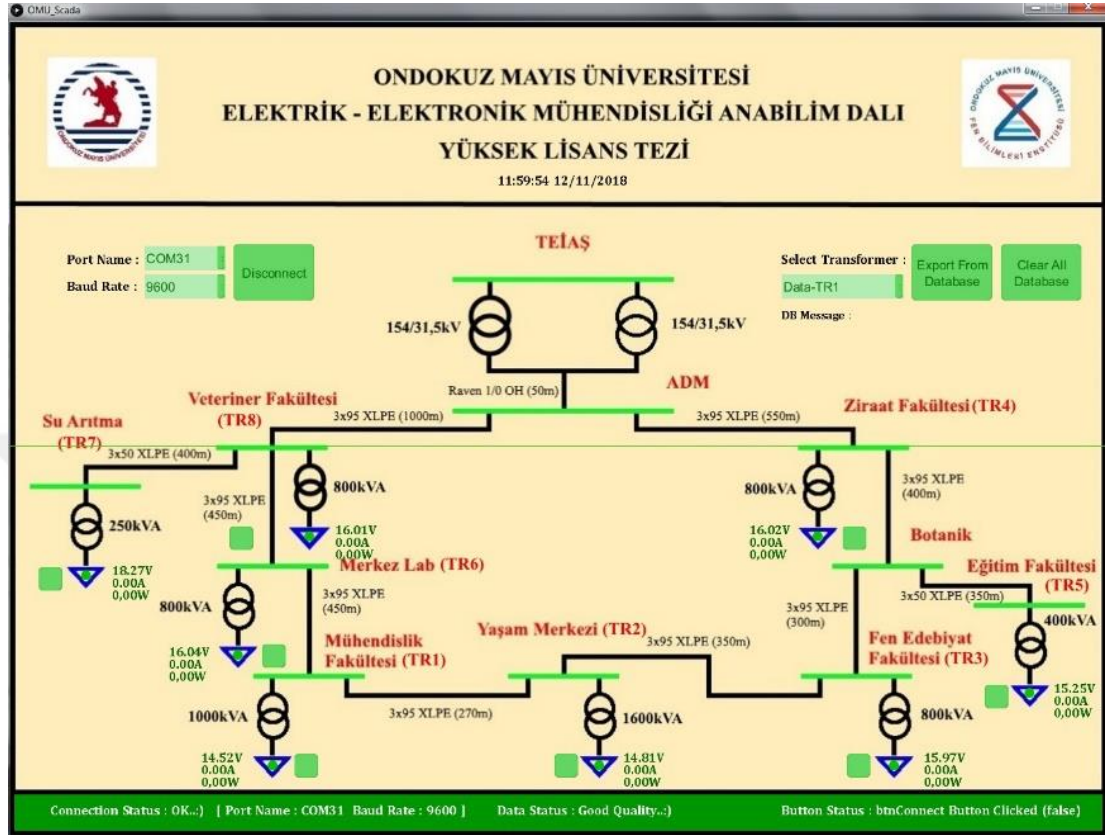
Scada arayüzünün RTU ile bağlantı sağlayabilmesi için port adı ve baud rate gibi değerlerin Şekil 5.28’teki gibi seçilmesi gerekmektedir. RTU’nun hangi COM bağlantı noktasına bağlı olduğu bilinmiyorsa bilgisayarda aygıt yöneticisi bölümünden öğrenilebilir. Çalışmamızda tüm mikro denetleyiciler 9600 iletişim hızı değerinden çalışacak şekilde düzenlenmiştir. Üst seviyelerde iletişim hızı seçilebileceği gibi bu değer bizim çalışmamız için yeterli olmaktadır.



Şekil 5.28. Scada arayüzü seri iletişim parametre seçim bölümü

Arayüzde seri iletişim parametrelerinin seçilmesinden sonra “Connect” butonuna basıldığında durum çubuğunun Şekil 5.29’da olduğu gibi yeşil renk olarak değişmesi gerekmektedir. Bu durum bağlantının sağlıklı bir şekilde gerçekleştiğini göstermektedir. Durum çubuğunu renklendirmek kullanıcıya bağlantı hakkında hızlı yorum yapma olanak sağlamaktadır. Bağlantı gerçekleştikten sonra arayüzün üst bölgelerinde donanım bölümünde yer alan DS3231 RTC modülünden gelen gerçek zaman bilgisi görüntülenmektedir. Her trafonun yanında donanım bölümünden gelen gerilim akım ve güç bilgileri yazdırılmaktadır. Ayrıca her trafonun diğer yanında konumlanmış olan butonlar kesici modellerinin manevralarını gerçekleştirmektedir. Trafoların içerisinde yer alan kırmızı ve yeşil yuvarlaklar ise donanım bölümündeki

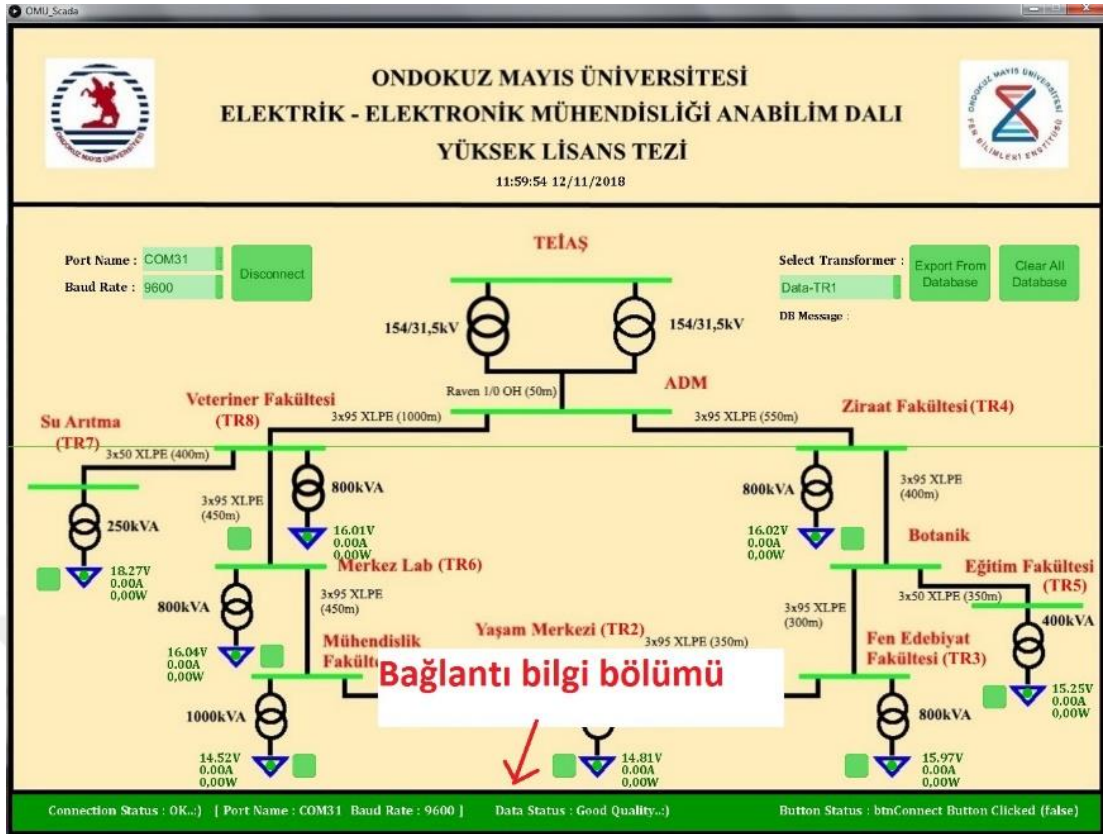
kesicilerin durum bilgilerini göstermektedir. Yapılan manevralara göre kapalı anahtar için *Kırmızı*, açık anahtar için *Yeşil* renk almaktadır. *Processing* ortamında buton yapıları için “*G4P*” kütüphanesi kullanılmıştır.



Şekil 5.29. Scada arayüzü bağlantının gerçekleştiği durum

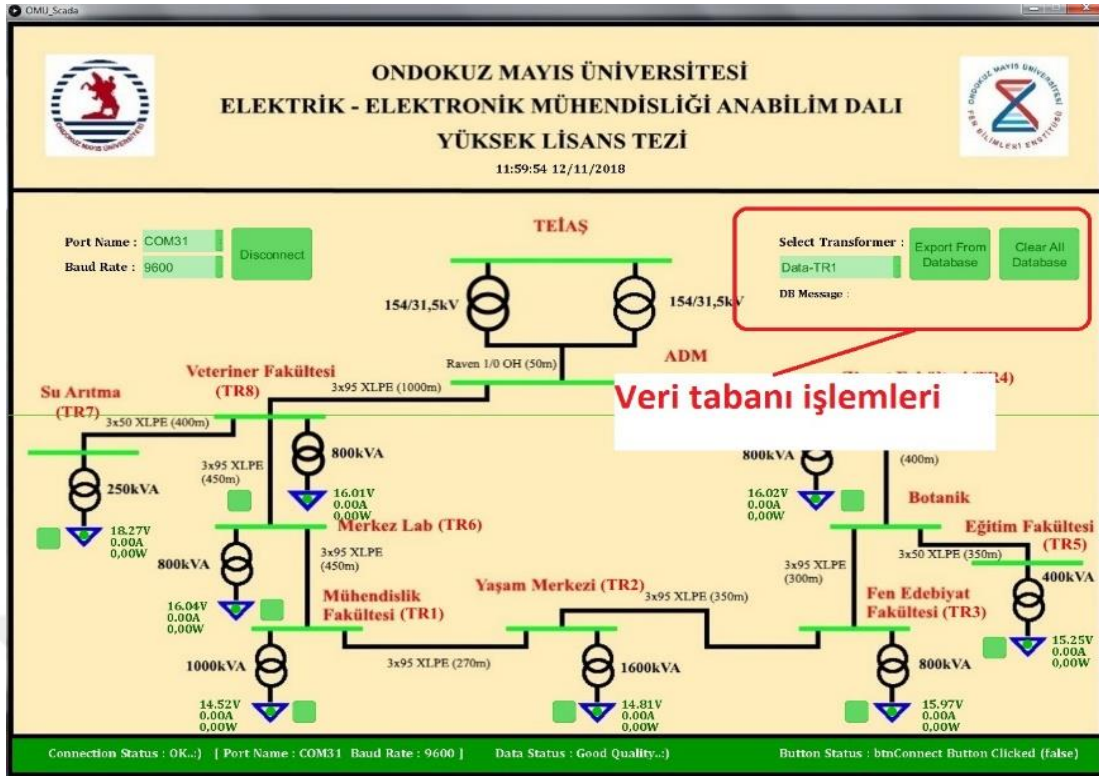
Durum çubuğu bağlantının gerçekleştiği dışında Şekil 5.30’da görüldüğü gibi faydalı bilgiler de sağlamaktadır. Durum çubuğu bağlantı gerçekleşmesi durumunda “*Connection Status*” bölümünde bağlantının gerçekleştiğini yazı ile belirtir. Ayrıca hangi COM port kullanıldığını ve Baud Rate değerini de arka plandan otomatik olarak elde edip gösterir. Durum çubuğu orta bölümünde gelen verinin belli parametrelere uygun olup olmadığı denetlenabilir ve yazı ile kullanıcıya bilgi verilir. Gelen veri “*Data#*” ile başlamalıdır. Bu kurala uymayan veri analizden geçirilmez ve kötü etiketi olarak arayüzde görüntülenmez. Durum çubuğu sol bölümde ise analiz amaçlı hangi butona basıldığı ve durum bilgisi yazdırılır.





Şekil 5.30. Scada arayüzü durum çubuğu bilgileri

Scada arayüzü sol üst bölümünde Şekil 5.31’de görüldüğü gibi veri tabanı ile ilgili uygulamalar bulunmaktadır. Burada herhangi bir trafo seçilerek trafoya ait verilerin veri tabanından “Export” edilmesi sağlanabilmektedir. CSV biçiminde elde edilen dosya çeşitli süzgeçler ve çizelgelerle analiz edilebilir hale getirilebilmektedir. Ayrıca temizleme butonu ile tüm veri tabanı arayüzden silinebilmektedir.



Şekil 5.31. Scada arayüzü veri tabanı işlemleri

## 5.12. Eğitim Takımında Senaryoların İşletilmesi

Elde edilen tüm veriler ile yazılım ve donanım sistemleri arasında kurulan anlık iletişim çeşitli senaryoların çalıştırılmasına olanak sağlamaktadır. Çalışmamızda oluşturduğumuz senaryo Elektrik Güç Kalitesi (Chattopadhyay vd, 2011) verilerine göre akım gerilim limit değerleri kullanılarak oluşturulmuştur. Senaryonun çalışma şekli ilgili trafo için limit değerlerine ulaşıldığında ya çeşitli şekillerde uyarı verme şeklinde ya da gerekli yüklerin otomatik olarak şebekeden çıkarılıp arızanın veya aşırı yüklenmenin giderilmesinin ardından Scada üzerinden veya el ile trafonun devreye alınması şeklinde belirtilebilir.

Elektrik Güç Kalitesi'nde (Chattopadhyay vd, 2011) belirtilen tanımlamalara göre oluşturduğumuz senaryolar: Gerilim Yükselmesi, Gerilim Düşüşü, Kısa Devre ve Elektrik Kesintisi şeklinde sıralanabilir.

Elektrik Güç Kalitesi'ndeki (Chattopadhyay vd, 2011) tanımlama ve limit değerlerine göre oluşturduğumuz senaryolar Scada arayüzünde ve donanım bölümünde yer alan ekranda alarm oluşturduğu gibi GSM Modem kullanılarak önceden belirlenmiş telefon numaralarına da mesaj olarak iletilebilmektedir.

### **5.12.1. Gerilim yükselmesi senaryosu**

Elektrik Güç Kalitesi'inde (Chattopadhyay vd, 2011) gerilim yükselmesi bozukluğu gerilim değerinin üstüne çıkması şeklinde tanımlanmaktadır. Çalışmamızda kullanılan trafoların sekonder nominal gerilimleri 12V AC'dir.

Gerilim yükselmesi senaryosunu uygulayabilmek için trafonun yükü alınarak trafonun boшта çalışması sağlanmıştır. Boş yükte çalışan trafo gerilimi nominal değerinin %30-50 oranında yükseğine çıkmaktadır. Uyguladığımız gerilim limit değeri ise %10'tir. Gerilim değeri nominal değerinin %10 fazlasını aşması durumunda (Chattopadhyay vd, 2011) donanım bölümünde yer alan ekranda alarm oluşturmakta ve önceden belirlenmiş numaralara GSM modem kullanılarak SMS ile haber verilebilmektedir.

### **5.12.2. Gerilim düşüşü senaryosu**

Elektrik Güç Kalitesi'inde (Chattopadhyay vd, 2011) gerilim düşüşü bozukluğu gerilim değerinin aşağısına inmesi şeklinde tanımlanmaktadır. Çalışmamızda kullanılan trafoların sekonder nominal gerilimleri 12V AC'dir.

Gerilim düşüşü senaryosunu uygulayabilmek için trafonun yüküne paralel dirençler ekleyerek eşdeğer direncin düşmesi, çekilen akımın normalden fazla olması ve böylece gerilimin düşmesi adımları gerçekleştirilmiştir. Uyguladığımız gerilim limit ise %10'tir. Gerilim değeri nominal değerinin %10 fazla azalması durumunda (Chattopadhyay vd, 2011) donanım bölümünde yer alan ekranda alarm oluşturmakta ve önceden belirlenmiş numaralara GSM modem kullanılarak SMS ile haber verilebilmektedir.

### **5.12.3. Kısa devre senaryosu**

Elektrik Güç Kalitesi'inde (Chattopadhyay vd, 2011) kısa devre bozukluğu gerilim değerinin %10'undan daha düşük seviyelere düşmesi şeklinde tanımlanmaktadır. Çalışmamızda kullanılan trafoların sekonder nominal gerilimleri 12V AC'dir.

Kısa devre senaryosunu uygulayabilmek için trafonun yüküne gerilimi nominal değerinin %10'unun aşağısına indirecek seviyede dirençler eklenmesi, çekilen akımın normalden çok fazla olması ve böylece gerilimin düşmesi adımları gerçekleştirilmiştir.



Uyguladığımız gerilim limiti trafolarla zarar vermemek için ise 2V şeklinde düzenlenmiştir. Gerilim değeri 2V'un aşağısına düşmesi durumunda donanım bölümünde yer alan ekranda alarm oluşturmakta ve önceden belirlenmiş numaralara GSM modem kullanılarak SMS ile haber verilebilmektedir.

#### **5.12.4. Elektrik kesintisi senaryosu**

Elektrik Güç Kalitesi'nde (Chattopadhyay vd, 2011) elektrik kesintisi herhangi bir arıza veya kesici manevrası sonucunda elektriğin olmaması şeklinde tanımlanmaktadır. Çalışmamızda kullanılan trafoların sekonder nominal gerilimleri 12V AC'dir.

Elektrik kesintisi senaryosunu uygulayabilmek için trafo kesicilerinin donanım bölümünden el ile tetikleme, Scada arayüzünden denetim işaretleri gönderme veya arıza sonucu otomatik açma gibi aşamalar gerçekleştirilebilir. Arıza tipindeki tetikleme tiplerinde donanım bölümünde yer alan ekranda alarm oluşturmakta ve önceden belirlenmiş numaralara GSM modem kullanılarak SMS ile haber verilebilmektedir.

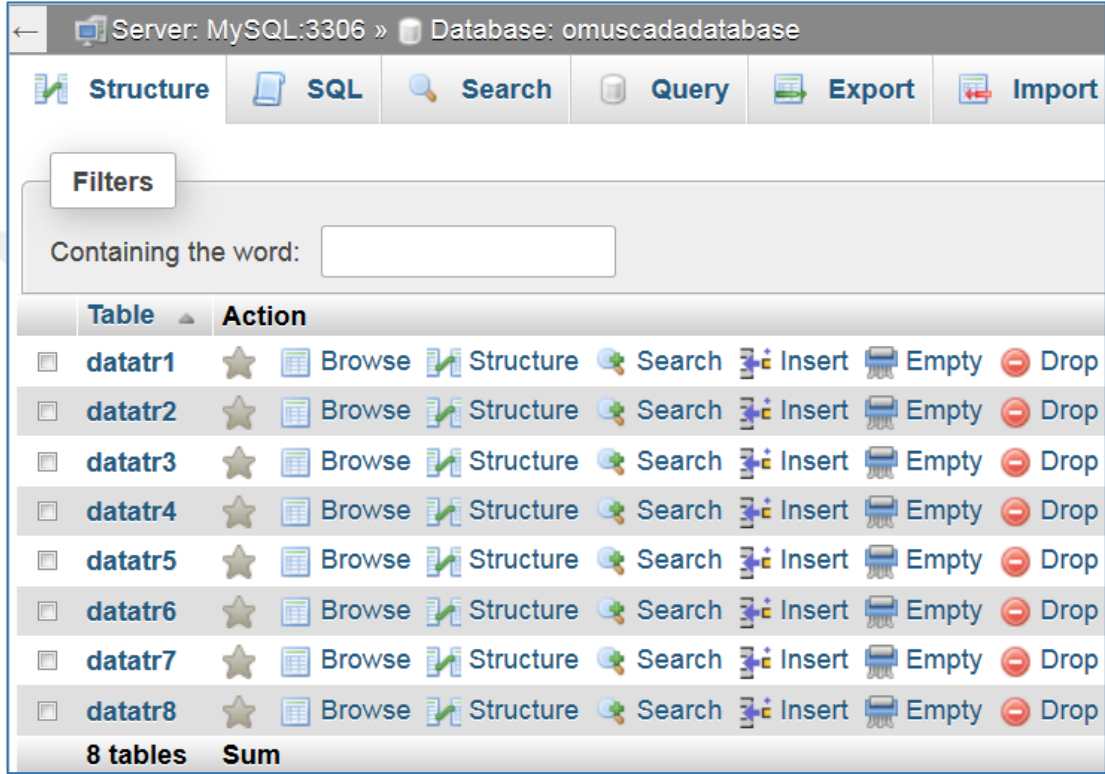
#### **5.13. Alarm Üretilmesi**

Elektrik Güç Kalitesi'indeki tanımlama ve limit değerlerine göre oluşturduğumuz senaryolar Scada arayüzünde ve donanım bölümünde yer alan ekranda alarm oluşturduğu gibi GSM Modem kullanılarak önceden belirlenmiş telefon numaralarına da mesaj olarak iletilebilmektedir. Arıza benzetimleri sonucu limit aşımına bağlı oluşturulan örnek alarmlar aşağıda belirtilen listedeki gibidir:

- ✓ Gerilim Yükselmesi: **TR7 – VH Arıza**
- ✓ Gerilim Düşüşü: **TR7 – VL Arıza**
- ✓ Kısa Devre: **TR7 – SC Arıza**
- ✓ Elektrik Kesintisi: **TR7 – Elektrik Kesinti**

#### 5.14. Verilerin Dönemsel Olarak Veri Tabanına Kaydedilmesi

Akıllı Şebeke uygulamalarında sahadan gelen verilerin gözlem ve denetiminin yanı sıra bir veri tabanında kaydedilmesi ve gerektiğinde raporlanabilmesi büyük önem arz etmektedir. Bu amaçla verilerin kaydedilmesi ve raporlanmasının benzetimi için Scada arayüzüne uyarlanmış *MySQL* veri tabanı Şekil 5.32'deki gibi oluşturulmuştur. Sahadan gelen veriler belirli aralıklarla veri tabanındaki tablolara kaydedilmektedir.



Şekil 5.32. MySQL veri tabanı ve trafo tabloları

*MySQL* veri tabanının oluşturma için *Wamp Server* kullanılmıştır. *Wamp Server* Windows için hazırlanmış bir kurulum paketidir. Bir yerel çalışma programı olarak görev yapmaktadır. Diğer programlardan farkı ise içerisinde *Mysql*, *Apache*, *Php* kurulumları bulunmaktadır. Veri tabanı adı "*omuscadadatabase*" olarak belirlenmiştir. Veri tabanında her trafo için ayrı ayrı tablolar oluşturulmuştur.

Veri tabanındaki her tabloda kayıt altına alınan veri topluluğu birbirinden benzersiz numaralarla kayıt altına alınmaktadır. Bu numaralar "*ComponentID*" sütunlarında görülmektedir. Her tabloda Şekil 5.33'te görüldüğü gibi RTU'dan gelen tarih bilgisi, saat bilgisi, trafo adı, gerilim, akım ve güç ölçümleri ve anahtarlama yapılarının durum bilgileri dönemsel olarak kayıt edilmektedir.

Kayıt altına alınan tüm bu veriler ilgili sistem için çok önemli analiz bilgilerini içermektedir. Operatör, uygulama mühendisleri veya yöneticiler bu kayıtlar sayesinde ilgili trafo için geçmişe yönelik analizler yapabilir. Ayrıca ihtiyaca bağlı çeşitli raporlama formları oluşturularak otomatik veya el ile raporlama ortamları oluşturulabilir. Bu da sistemin analizinde kolaylık getiren bir uygulamadır.

componentID	timeData	trName	voltageData	currentData	powerData	cbStatus
1977	11:02:05 07/12/2018	TR7	17.98V	0.02A	0,36W	Opened
1978	11:02:06 07/12/2018	TR7	12.19V	0.21A	2,56W	Closed
1979	11:02:07 07/12/2018	TR7	12.24V	0.21A	2,57W	Closed
1980	11:02:09 07/12/2018	TR7	12.21V	0.21A	2,56W	Closed
1981	11:02:10 07/12/2018	TR7	12.22V	0.21A	2,57W	Closed
1982	11:02:11 07/12/2018	TR7	12.18V	0.21A	2,56W	Closed
1983	11:02:12 07/12/2018	TR7	12.19V	0.21A	2,56W	Closed
1984	11:02:13 07/12/2018	TR7	12.22V	0.21A	2,57W	Closed
1985	11:02:14 07/12/2018	TR7	12.19V	0.21A	2,56W	Closed
1986	11:02:16 07/12/2018	TR7	12.20V	0.21A	2,56W	Closed
1987	11:02:17 07/12/2018	TR7	12.19V	0.21A	2,56W	Closed
1988	11:02:18 07/12/2018	TR7	0.00V	0.00A	0,00W	Closed
1989	11:02:19 07/12/2018	TR7	18.01V	0.00A	0,00W	Opened
1990	11:02:20 07/12/2018	TR7	17.99V	0.00A	0,00W	Opened
1991	11:02:21 07/12/2018	TR7	17.97V	0.00A	0,00W	Opened
1992	11:02:22 07/12/2018	TR7	17.96V	0.00A	0,00W	Opened
1993	11:02:24 07/12/2018	TR7	17.96V	0.00A	0,00W	Opened

Şekil 5.33. Veri tabanı tablo içeriği

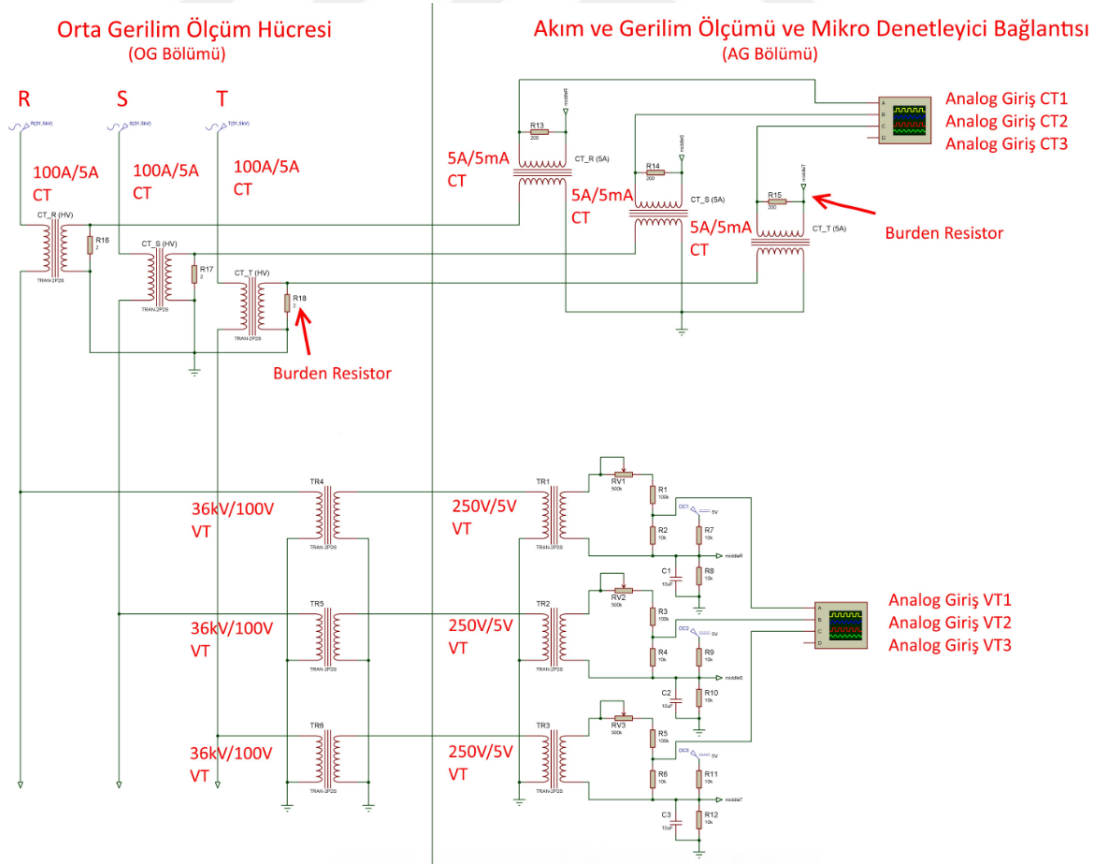
Scada arayüzü sol üst bölümünde Şekil 5.31’de görüldüğü gibi veri tabanı ile ilgili uygulamalar bulunmaktadır. Burada herhangi bir trafo seçilerek trafoya ait verilerin ver tabanından dışa aktarılması sağlanabilmektedir. CSV biçiminde elde edilen dosya çeşitli süzgeçler ve çizelgelerle analiz edilebilir hale getirilebilmektedir. Ayrıca temizleme butonu ile tüm veri tabanı arayüzden silinebilmektedir.

Veri tabanı üzerindeki uygulamalar Scada arayüzünden yapılabileceği gibi *Wamp Server*’de bulunan *phpMyServer* arayüzünden de gerçekleştirilebilir. Burada veri tabanı için özelleştirilmiş çok sayıda uygulama kullanıcılar için bir kolaylık sağlamaktadır.

## 6. TEZ ÇALIŞMASININ ELEKTRİK ŞEBEKELERİNE UYGULANMASI

Bu çalışmada model şebeke olarak OMÜ elektrik dağıtım şebekesinin bir bölümü kullanılmıştır. Trafoların belli oranda küçük halleri kullanılarak senaryo ve analizler gerçekleştirilmiştir. Elde edilen sonuçlar bu eğitim takımındaki yazılım ve donanım oluşumların gerçek ve akıllı olmayan bir elektrik şebekesine uygulanabileceğini göstermiştir. Eğitim takımında kullanılan şebeke bir fazlıdır. Bu nedenle her bir trafo için bir akım ve bir gerilim algılayıcısı yeterli olmaktadır. Kullanılan akım algılayıcı 5A AC, gerilim algılayıcı ise 250V AC seviyesindedir.

Standart elektrik şebekelerine ait ana dağıtım merkezlerinde her dağıtım noktasına ait akım-gerilim ölçü hücreleri yer almaktadır. Ölçü hücrelerinde yer alan akım trafolarının çıkışı genellikle 5A'dır. Gerilim trafolarının çıkışı ise 100V'tur. Bu akım ve gerilim seviyeleri eğitim takımında kullanılan algılayıcıların çalışma aralığı içindedir. Şekil 6.1'de belirtilen bir devre şeması ile gerçek şebekede bir fiderin akım ve gerilim ölçümü gerçekleştirilebilir.



Şekil 6.1. Tez çalışmasının gerçek şebekelere uygulanması

Tez çalışmasının gerçek bir şebekeye uygulanması için akım-gerilim ölçü hücrelerinden akım ve gerilim işaretlerini alması gerekir. Her bir fider için bir mikro denetleyici, üç akım algılayıcı ve üç gerilim algılayıcı yeterli olmaktadır. Mikro denetleyici olarak Arduino DUE veya DSP tabanlı bir devre kartı seçilebilir. Böylece ayrıntılı arıza analizleri, harmonikler ve FFT çalışmaları gerçekleştirilebilir. Kesicilerin ve diğer anahtarlama araçlarının durum bilgisinin alınabilmesi için hücrelerdeki küçük anahtarlar kullanılabilir.

Ana dağıtım merkezinde her fider ayrı bir mikro denetleyiciye sahip olmalı ve Şekil 6.1'deki devre uygulanmalıdır. Tüm fiderlerde elde edilen ölçümler bir iletişim protokolü ile her ana dağıtım merkezinde bulunması gereken Ana RTU'ya gönderilmesi gerekmektedir. I<sup>2</sup>C iletişim protokolü 25m'den fazla mesafelerde özel yapılara ihtiyaç duyduğundan ve işaret kaybı yaşadığından dolayı daha gelişmiş bir iletişim protokolü kullanılmalıdır. Bu iletişim şekli RS-485 (Modbus) veya güç hattı iletişimi olabilir. Ana RTU'ya ulaşan tüm ölçümler IOT uygulamaları, kablosuz haberleşme veya seri iletişim yolu ile Scada arayüzüne ulaştırılmalıdır. Böylece eğitim takımı şeklinde oluşturulan bu çalışma gerçek elektrik şebekelerine de uygulanabilir hale gelmektedir.

## 7. BULGULAR VE TARTIŞMA

Çalışmamızda benzetim süreci algılayıcı bilgilerinin okunması ve durum bilgilerinin alınmasıyla başlamaktadır. Akım ve gerilim ölçümü AC algılayıcı kullanılarak sağlanmaktadır. Akım ve gerilim algılayıcı Arduino analog giriş pinlerine bağlanarak ölçümler yapılmaktadır. Anahtarlama yapılarının durum bilgilerinin benzetimi için buton anahtar kullanılmaktadır. Bu sayede her bir trafo için akım, gerilim ölçümleri ve anahtarlama yapıları durum bilgileri temin edilmektedir.

RTU ile uydu röleler arasındaki iletişim I<sup>2</sup>C protokolü ile gerçekleştirilmektedir. Bu nedenle ana ve uydu yapılar birbirinden farklı olarak ifade edilmektedir. Bu adresleme verilerin uygun bir şekilde taşınmasını ve ilgili yapının doğru bir şekilde tetiklenmesini sağlamaktadır. Donanım bölümünde I<sup>2</sup>C protokolü iki yönlü iletişim sağlayacak şekilde düzenlenmiştir. Bu sayede Scada arayüzünden gelen verilerin RTU üzerinden ilgili uydu adresine taşınması sağlanırken uydu yapılarında ölçülen verilerin RTU'ya taşınması sağlanmış oluyor. RTU'ya ulaşan tüm veriler çeşitli dizilerde saklanarak seri iletişim yolu ile Scada arayüzüne ulaşması sağlanmaktadır.

*Processing* ortamında hazırlanan Scada arayüzü bir saniyelik aralıkla RTU'da toplanan tüm veriyi seri iletişim protokolü ile almaktadır. RTU'dan alınan ve bütün şeklinde bulunan veri uygun ayrıştırma ve adresleme yöntemleri ile çözümlenmekte ve arayüzde bulunan ilgili yerlerde anlık olarak görüntülenmeleri sağlanmaktadır. I<sup>2</sup>C protokolünde olduğu gibi RTU ve arayüz arasında kullanılan seri iletişim de iki yönlü çalışacak şekilde düzenlenmiştir. RTU'da toplanan verilerin arayüze taşınması birinci yön olarak düşünülürse arayüzdeki butonlarla donanım bölümüne gönderilen denetim komutları ikinci yön olarak değerlendirilebilir.

Elde edilen tüm veriler ile yazılım ve donanım sistemleri arasında kurulan anlık iletişim çeşitli senaryoların çalıştırılmasına olanak sağlamaktadır. Çalışmamızda oluşturduğumuz senaryo Elektrik Güç Kalitesi (Chattopadhyay vd, 2011) verilerine göre akım gerilim limit değerleri kullanılarak oluşturulmuştur. Senaryonun çalışma şekli ilgili trafo için limit değerlerine ulaşıldığında gerekli yüklerin otomatik olarak şebekeden çıkarılıp arızanın veya aşırı yüklenmenin giderilmesinin ardından Scada üzerinden veya el ile trafonun devreye alınması şeklinde belirtilebilir.

Elektrik Güç Kalitesi'nde (Chattopadhyay vd, 2011) belirtilen tanımlamalara göre oluşturduğumuz senaryolar: Yüksek Gerilim, Alçak Gerilim, Kısa Devre ve

Elektrik Kesintisi şeklinde sıralanabilir. Elektrik Güç Kalitesi'indeki (Chattopadhyay vd, 2011) tanımlama ve limit değerlerine göre oluşturduğumuz senaryolar Scada arayüzünde ve donanım bölümünde yer alan ekranda alarm oluşturduğu gibi GSM Modem kullanılarak önceden belirlenmiş telefon numaralarına da mesaj olarak iletilebilmektedir.

Scada arayüzüne bütünleştirilmiş ve *MySQL* ortamında oluşturulmuş olan veri tabanı anlık olarak gelen verilerin dönemsel olarak saklanmasını sağlamaktadır. Veri tabanında veriler her bir trafo için ayrı ayrı olacak şekilde saklanmaktadır. Elde edilen bu veri yığını geçmişe yönelik analizlerde ve otomatik – el ile raporlamalarda kullanılabilir. Arayüz üzerinde yer alan açıklama ve işaretler veri tabanına ve donanım bölümüne olan bağlantıyı takip etmekte kullanıcıyı bilgilendirmektedir. Bu da herhangi bir bağlantı kopması durumlarında uygulanan senaryoların doğruluğunu teyit etme açısından önem arz etmektedir.

Elektrik şebekeleri yazılım ve donanım uygulamaları dünya genelinde çok sayıda firma tarafından gerçekleştirilmektedir. Ürünlerin farklı olmalarından dolayı çok çeşitli yazılım, donanım ortamları ve iletişim protokolleri kullanılmaktadır. Tüm bu farklılıklar Akıllı Şebekelerin uygulanmasında zorluk oluşturmaktadır. Çeşitli ürünlerin birbiriyle uyum sağlamamaları ve bu sorunların üstesinden gelmek yaygın olarak karşılaşılan sorunlar arasındadır. Bu karışıklıkları gidermek için yazılım ve donanım ortamlarında dünya genelinde standart yapı oluşturulmaya çalışılmaktadır. Donanım bölümündeki standart yapı IEC 61850 protokolü olurken yazılım bölümünde OPC UA yaygınlaşmaya başlamıştır.

IEC 61850 protokolü elektrik dağıtım şebekeleri otomasyon sistemlerinde kullanılan bir standarttır. Elektrik şebekeleri arasındaki iletişimi ana dağıtım merkezi içerisindeki standartları tanımlar (Srinivasan vd, 2013). Elektrik şebekelerinde çeşitli akıllı röleler kullanılmaktadır. Her firma üretmiş olduğu röle için farklı iletişim protokolü kullanmaktadır. Kullanılan iletişim protokollerinin farklı olması tüm şebekenin tek bir sistem olarak düzenlenmesini ve uyumunu zorlaştırmaktadır. Bu farklılıkların üstesinden gelmek için IEC 61850 Standardı, 57. *IEC Technical Committee (TC)* tarafından otomasyonla denetlenen şebekelerdeki haberleşme ağ ve sistemleri için uluslararası bir standart oluşturmak amacıyla hazırlanmıştır. Bu standart ile ilgili sistem parçaları arasındaki haberleşmenin sağlanması hedeflenmiştir (Gezer

vd, 2008). Çalışmamızda kullandığımız iletişim protokolü özellikle eğitim amaçlı uygulamalarda IEC 61850 protokolünün anlaşılması için bir örnek teşkil etmektedir.

*Open Process Control (OPC)* son zamanlarda endüstride arayüz olarak kullanılmaktadır. Yeni sürüm olarak ise *OPC Unified Architecture (OPC UA)* ortaya çıkmaktadır. Bu yeni sürüm eski sürümlerde bulunan çoğu eksikliği gidermiş durumdadır (Srinivasan vd, 2013). IEC 61850 iletişim protokolünde olduğu gibi OPC UA Scada yazılım ortamlarının standart hale gelmesi için oluşturulmuş bir uygulamadır. Son yıllarda yapılan çalışmalar özellikle IEC 61850 protokolünün ve OPC UA uygulamalarının birlikte kullanılması üzerine odaklanmaktadır (Srinivasan vd, 2013). Böylece sahada bulunan çeşitli teknolojilere sahip cihazları ortak bir iletişim protokolü ve Scada arayüzü ile uyuşturmak mümkün olmaktadır. Oluşturduğumuz deney ortamı farklı sistemlerden veri alınması ve bu verilerin Scada arayüzünde gözlemlenmesi süreçlerini değerlendirme açısından önem taşımaktadır.



## 8. SONUÇ VE ÖNERİLER

Teknolojik gelişmeler her alanda hayatı kolaylaştırma olarak ortaya çıkmaktadır. Teknolojinin gelişme aşamaları düşünüldüğü zaman bu gelişmelerin kalbinde elektriğin olduğu görülmektedir. Elektriğin sürekli ve kesintisiz olarak temini büyük önem arz etmektedir. Elektrik şebekelerinin klasik olarak yönetilmesinde ortaya çıkan en önemli sorunlar; eski yapıların kullanılıyor olması, anahtarlama yapılarına sadece yerinde ve el ile olarak müdahale edilebilmesi, saha ölçümlerinin bir merkezden görüntülenememesi, anahtarlama yapılarında değişiklik olması ve arıza durumlarında herhangi bir bilgi veren bir sistemin bulunmaması şeklinde sıralanabilir. Tüm bu sorunlar şebeke durumunun izlenmesi, kumanda ve denetiminin yapılması ve arızalara kısa süre içinde müdahale edilmesini engelleyen durumlar arasındadır.

Teknolojinin gelişmesi sürecinde elektriğin sürekliliği en önemli unsurlardan biridir. Bu nedenle elektrik şebekelerinde daha teknolojik bir şekilde yönetilme eğilimine gidilmektedir. Bu da Akıllı Şebeke sistemlerine geçiş yapılarak sağlanmaktadır. Akıllı şebekelerin yaygınlaşması sürecinde bu sistemleri kullanan kişilerin eğitilmesi geliştirilmiş olan şebekenin daha verimli ve doğru kullanılmasını sağlayacaktır.

Son yıllarda Akıllı Şebeke Laboratuvarları kurularak yeni başlayanlar ve öğrenciler için eğitim ortamları oluşturulma sürecine girilmiştir. Akıllı Şebeke laboratuvarlarının kurulması genellikle çok yüksek maliyetler gerektirmektedir. Bu durum özellikle öğrenciler için bir engel olarak ortaya çıkmaktadır. Tüm bu sebeplerden dolayı son yıllardaki çalışmalar Akıllı Şebeke laboratuvarlarının düşük maliyetli oluşturulması şeklinde gelişim göstermektedir.

Geçmiş yıllarda yapılan çalışmalarda donanım olarak *Raspberry Pi* ve *Arduino* kullanıldığı görülmektedir. Ancak oluşturulmuş olan çoğu sistemde laboratuvar malzemeleri veya bazı hazır Scada yazılımlarından faydalanılmıştır. Bizim çalışmamızda ise maliyeti yüksek olan herhangi bir yazılım veya cihaz kullanılmaması ve çoğu senaryo için uygun bir ortam olması önceki çalışmalardan farkını ortaya çıkarmaktadır. Oluşturulan bu deney setinin esnek yazılımları sayesinde çoğu senaryonun oluşturulması için kullanışlı bir ortam olduğu görülmektedir.

Eğitim takımı Arduino mikro denetleyici ve *Processing* yazılımı tabanlı olduğundan dolayı uygulaması gerçekleştirilmiş olan senaryoların dışında birçok

senaryonun oluşturulmasına olanak sağlamaktadır. *Arduino* donanımsal çalışmalarda dünya genelinde yaygın olarak kullanıldığı için düşünülebilecek çoğu uygulamaya örnek çalışma sağlamaktadır. Bu özelliği sayesinde analog veya sayısal giriş bilgilerini alabilen matematiksel hesaplamaları gerçekleştiren ve sayısal veya PWM çıkış işaretleri verebilen esnek ve kullanışlı yapısını ortaya çıkarmaktadır.

Oluşturulan bu eğitim takımı ile ilgili ileriki zamanlarda yapılabilecek ek uygulamalardan biri kullanımı yaygınlaşan LOT uygulamaları ve internet bağlantı. *Arduino* kartlarına bağlanabilen ve mikro denetleyiciyi ucuz ve esnek bir şekilde internete bağlayan modüller bulunmaktadır. Bu uygulamalar eğitim takımının ve *Scada* arayüzünün internet üzerinden denetim ve gözlemlenmesine olanak sağlamaktadır.

*Arduino* kullanılarak gerçekleştirilebilen çoğu uygulamanın yanı sıra bazı sınırlamalar da bulunmaktadır. Bu sınırlamalar 16Mhz çalışma frekansı, örnekleme hızı, sınırlı sayıdaki giriş ve çıkış pinleri şeklinde sıralanabilir. Özellikle ölçülen işaretteki bozukluklara ait frekansların belirlenmesi, FFT uygulamaları gibi çalışmalarda *Arduino* çok esnek olamayabilir.

Elektrik Güç Kalitesi'inde (Chattopadhyay vd, 2011) belirtilen tanımlamalardan biri olan işaret bozukluklarını belirlenmesi ileriki çalışmalarda eklenebilecek bir senaryo olabilir. Çevredeki manyetik alanlar sistem akım gerilim işaretlerine etki edebilmektedir. Bozulmuş olan işaretteki çeşitli frekansların tespit edilmesi ve değerlendirilmesi Akıllı Şebekeler'de kullanıcılar için önemli bir analiz yöntemi olabilmektedir.

Eğitim takımında mikro denetleyici olarak *Arduino UNO R3* modeli kullanılmıştır. *Arduino UNO* seri iletişim tampon büyüklüğü ve giriş-çıkış pin sayıları sınırlı sayıda olduğu için tüm trafoları ait analog ve sayısal bilginin işlenmesi için çok sayıda uydu *Arduino* kullanılmıştır. Ayrıca *Arduino* analog giriş birimi 0-5V aralığında çalıştığı için analog girişine bağlanacak olan AC ölçüm işaretlerine +2,5V DC işaret eklenmek zorunda kalmıştır. AC işaret doğrudan bağlanamamıştır. Daha sonraki çalışmalarda bu sorunların üstesinden gelebilmek için çok sayıda giriş çıkış pinlerine sahip olan, analog girişi artı ve eksi değerleri kabul eden yüksek frekanslarda mikro denetleyici veya DSP'ye sahip olan kartlar kullanılabilir. Böylece eğitim takımına ait tüm işaretler tek bir kart üzerinden denetlenebileceği gibi yüksek hızlarda

örnekleme ve FFT senaryoları gerçekleştirilebilecek ve böylece arıza analizleri de çok daha ayrıntılı yapılabilecektir.

Oluşturulan eğitim takımının ileriki çalışmalarda mikro denetleyici yapısının DSP yapısına dönüştürülmesine bağlı olarak Yük Akışı veya Kesinti Yönetim Sistemleri gibi senaryolar da çalıştırılabilir.

Yük Akışı senaryosu için oluşturulabilecek DSP donanımsal yapısına mühendislik uygulamalarında sıklıkla kullanılan *Matlab (MATrix LABoratory)* yazılımı bütünleştirilebilirse “*Gauss Elimination*” veya “*Newton-Raphson*” (Glover vd, 2010) gibi yöntemler kullanılarak yakınsamalar yapılabilir. Böylece geniş şebeke modellerinde tüm trafolardan ölçüm almak yerine önemli noktalardan ölçümler alınarak ve belirli yöntemlerle elde edilen yakınsama sonuçlarına bağlı olarak tüm şebekenin yaklaşık yüklenme değerleri elde edilebilir.

Kesinti yönetim sistemi senaryosunda ise trafolara bağlı olan müşteri bilgisinin eşleştirilmesi ve veri tabanlarına kaydedilmesine bağlı olarak herhangi bir kesinti durumunda etkilenen müşteri sayısı, kesinti süresi gibi veriler elde edilebilir. Bu bilgiler kullanılarak müşterilere elektrik sağlayıcılar tarafından herhangi bir ceza ödemesi konusunda yönlendirme yapılabilmektedir.

## KAYNAKLAR

- Andreadou, N., Guardiola, M. O., Papaioannou, I. ve Prettico, G. (2016). *Smart Grid Laboratories Inventory 2016*. Retrieved from JRC Science for Policy Report: [https://ses.jrc.ec.europa.eu/sites/ses.jrc.ec.europa.eu/files/u24/2017/report\\_sg\\_1\\_abs/jrc104803\\_jrc104803\\_sgli\\_report\\_2016\\_pdf\\_version.pdf](https://ses.jrc.ec.europa.eu/sites/ses.jrc.ec.europa.eu/files/u24/2017/report_sg_1_abs/jrc104803_jrc104803_sgli_report_2016_pdf_version.pdf) (Eriřim tarihi: 01.11.2018)
- Anonymous, 2008. Cinterion MC35i. Retrieved from [https://primteplo.ru/d/283643/d/mc35i\\_hid.pdf](https://primteplo.ru/d/283643/d/mc35i_hid.pdf) (Eriřim tarihi: 01.11.2018)
- Anonymous, 2010. Open Energy Monitor. Retrieved from <https://learn.openenergymonitor.org/> (Eriřim tarihi: 01.11.2018)
- Anonymous, 2012. Overview. A short introduction to the Processing software and projects from the community. Retrieved from <https://processing.org/overview/> (Eriřim tarihi: 01.11.2018)
- Anonymous, 2014. I2C-bus specification and user manual. *UM10204*. Retrieved from <https://www.nxp.com/docs/en/user-guide/UM10204.pdf?fsrch=1&sr=8&pageNum=1> (Eriřim tarihi: 01.11.2018)
- Anonymous, 2015. Extremely Accurate I2C-Integrated RTC/TCXO/Crystal. *DS3231*. Retrieved from <https://datasheets.maximintegrated.com/en/ds/DS3231.pdf> (Eriřim tarihi: 01.11.2018)
- Anonymous, 2018. Arduino. *Arduino UNO R3*. Retrieved from <https://store.arduino.cc/usa/arduino-uno-rev3> (Eriřim tarihi: 01.11.2018)
- Armendariz, M., Chenine, M., Nordström, L. ve Al-Hammouri, A. (2014). A co-simulation platform for medium/low voltage monitoring and control applications. ISGT 2014, 19-22 Feb. 2014, 1-5.
- Aurilio, G., Gallo, D., Landi, C., Luiso, M. ve Graditi, G. (2014). A low cost smart meter network for a smart utility. 2014 IEEE International Instrumentation and Measurement Technology Conference (I2MTC) Proceedings, 12-15 May 2014, 380-385.
- Barai, G. R., Krishnan, S. ve Venkatesh, B. (2015). Smart metering and functionalities of smart meters in smart grid - a review. 2015 IEEE Electrical Power and Energy Conference (EPEC), 26-28 Oct. 2015, 138-145.
- Bonganay, A. C. D., Magno, J. C., Marcellana, A. G., Morante, J. M. E. ve Perez, N. G. (2014). Automated electric meter reading and monitoring system using ZigBee-integrated raspberry Pi single board computer via Modbus. Electrical, Electronics and Computer Science (SCEECS), 2014 IEEE Students' Conference on, 1-2 March 2014, 1-6.
- Chattopadhyay, S., Mitra, M. ve Sengupta, S. 2011. *Electric Power Quality*. Springer Science+Business Media, 5-12,

- Chren, S., Rossi, B. ve Pitner, T. (2016). Smart grids deployments within EU projects: The role of smart meters. 2016 Smart Cities Symposium Prague (SCSP), 26-27 May 2016, 1-5.
- Clastres, C. 2011. Smart grids: Another step towards competition, energy security and climate change objectives. *Energy Policy*, 39:9, 5399-5408.
- Eriksson, M., Armendariz, M., Vasilenko, O. O., Saleem, A. ve Nordström, L. 2015. Multiagent-Based Distribution Automation Solution for Self-Healing Grids. *IEEE Transactions on Industrial Electronics*, 62:4, 2620-2628.
- Esposito, W. J., Mujica, F. A., Garcia, D. G. ve Kovacs, G. T. A. (2015). The Lab-In-A-Box project: An Arduino compatible signals and electronics teaching system. 2015 IEEE Signal Processing and Signal Processing Education Workshop (SP/SPE), 9-12 Aug. 2015, 301-306.
- Fransiska, R. W., Septia, E. M. P., Vessabhu, W. K., Frans, W., Abednego, W. ve Hendro (2013). Electrical power measurement using Arduino Uno microcontroller and LabVIEW. 2013 3rd International Conference on Instrumentation, Communications, Information Technology and Biomedical Engineering (ICICI-BME), 7-8 Nov. 2013, 226-229.
- Gammon, N., 2011. I2C - Two-Wire Peripheral Interface - for Arduino. Retrieved from <http://www.gammon.com.au/i2c> (Erişim tarihi: 16.02.2018)
- Gezer, D., Nadar, A. ve Özay, N. (2008). IEC 61850 Standardı ve Hidroelektrik Santrallerin Otomasyon Sistemlerine Uygulanması. Elektrik - Elektronik ve Bilgisayar Mühendisliği Sempozyumu, 26-30 Kasım 2008, Bursa.
- Glover, J. D., Sarma, M. S. ve Overbye, T. J. 2010. *Power System Analysis and Design* (Fifth Edition), 294-379,
- Megalingam, R. K., Varghese, J. M. ve Anil, S. A. (2016). Distance estimation and direction finding using I2C protocol for an auto-navigation platform. 2016 International Conference on VLSI Systems, Architectures, Technology and Applications (VLSI-SATA), 10-12 Jan. 2016, 1-4.
- Nannen, O., Piech, K., Lehnhoff, S., Rohjans, S., Schlögl, F., Velasquez, J., Andren, F. ve Strasser, T. (2015). Low-cost integration of hardware components into co-simulation for future power and energy systems. IECON 2015 - 41st Annual Conference of the IEEE Industrial Electronics Society, 9-12 Nov. 2015, 005304-005309.
- Patel, J. J. ve Soni, B. H. 2013. Design And Implementation of I2C Bus Controller Using Verilog. *Journal of Information, Knowledge and Research in Electronics and Communication Engineering*, 02:02, 520-522.
- Rahman, M. M., Noor, E. J., Islam, M. O. ve Salakin, M. S. (2015). Arduino and GSM based smart energy meter for advanced metering and billing system. 2015 International Conference on Electrical Engineering and Information Communication Technology (ICEEICT), 21-23 May 2015, 1-6.

- Rodríguez, F. J., Fernandez, S., Sanz, I., Moranchel, M. ve Bueno, E. J. 2016. Distributed Approach for SmartGrids Reconfiguration Based on the OSPF Routing Protocol. *IEEE Transactions on Industrial Informatics*, 12:2, 864-871.
- Santis, D. D., Giampetruzzi, D. A., Abbatantuono, G. ve Scala, M. L. (2016). Smart metering for low voltage electrical distribution system using Arduino Due. 2016 IEEE Workshop on Environmental, Energy, and Structural Monitoring Systems (EESMS), 13-14 June 2016, 1-6.
- Schvarcbacher, M. ve Rossi, B. (2017). Smart Grids Co-Simulations with Low-Cost Hardware. 2017 43rd Euromicro Conference on Software Engineering and Advanced Applications (SEAA), Aug. 30 2017-Sept. 1 2017, 252-255.
- Srinivasan, S., Kumar, R. ve Vain, J. (2013). Integration of IEC 61850 and OPC UA for Smart Grid automation. 2013 IEEE Innovative Smart Grid Technologies-Asia (ISGT Asia), 10-13 Nov. 2013, 1-5.
- Strasser, T., Stifter, M., Andrén, F. ve Palensky, P. 2014. Co-Simulation Training Platform for Smart Grids. *IEEE Transactions on Power Systems*, 29:4, 1989-1997.
- Stusek, M., Pokorny, J., Masek, P., Hajny, J. ve Hosek, J. (2017). A non-invasive electricity measurement within the smart grid landscape: Arduino-based visualization platform for IoT. 2017 9th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT), 6-8 Nov. 2017, 423-429.
- Suwansit, K., Konsombut, B., Hankongkaew, P. ve Tantidham, T. (2014). PMA: Power Monitoring Application for Android. 2014 Third ICT International Student Project Conference (ICT-ISPC), 26-27 March 2014, 69-72.
- Tamkittikhun, N., Tantidham, T. ve Intakot, P. (2015a). AC power meter design based on Arduino: Multichannel single-phase approach. 2015 International Computer Science and Engineering Conference (ICSEC), 23-26 Nov. 2015, 1-5.
- Tamkittikhun, N., Tantidham, T. ve Intakot, P. (2015b). AC power meter design for home electrical appliances. 2015 12th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), 24-27 June 2015, 1-6.
- Visalatchi, S. ve Sandeep, K. K. (2017). Smart energy metering and power theft control using arduino & GSM. 2017 2nd International Conference for Convergence in Technology (I2CT), 7-9 April 2017, 858-961.
- Zheng, J., Gao, D. W. ve Lin, L. (2013). Smart Meters in Smart Grid: An Overview. 2013 IEEE Green Technologies Conference (GreenTech), 4-5 April 2013, 57-64.

## ÖZGEÇMİŞ

Adı ve Soyadı : Abdulkadir Gözüoğlu  
Doğum Yeri : Mardin  
Doğum Tarihi : 21.04.1988  
Yabancı Dili : İngilizce

### Eğitim Durumu

Lise : Mardin (YDA) Lisesi (Eylül 2002 – Haziran 2006)  
Lisans : Gaziantep Üniversitesi, Mühendislik fakültesi, Elektrik ve Elektronik Mühendisliği Bölümü (Eylül 2006 – Haziran 2011)  
Yüksek Lisans : Ondokuzmayıs Üniversitesi, Fen Bilimleri Enstitüsü, Elektrik Elektronik Mühendisliği Anabilim Dalı (Eylül 2015 – Aralık 2018)

### Çalıştığı Kurum/Kurumlar ve Yıl

Gülsan Holding (2011-2012)  
Vezirköprü Orman Ürünleri ve Kâğıt sanayi A.Ş. (2012-2013)  
Çalık YEDAŞ (2013-2015)  
Gaziosmanpaşa Üniversitesi Tokat Teknik Bilimler MYO Elektrik Pr. (2015 – Devam Ediyor)

### Yayımlar

Gozuoglu A & Ozgonenel O (2018). Akıllı Şebeke ve Scada Benzeşimi için Eğitim Takımı Tasarımı. Eleco2018, 30 Kasım - 1 Aralık 2018, Bursa.