

T.C.
BOLU ABANT İZZET BAYSAL ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ



MANUEL JENERATÖRÜN AKILLI JENERATÖRE
DÖNÜŞÜMÜ VE UZAKTAN İZLENMESİ

YÜKSEK LİSANS TEZİ

İBRAHİM ŞEREMET

BOLU, TEMMUZ - 2019

T.C.
BOLU ABANT İZZET BAYSAL ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
ELEKTRİK ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM
DALI



MANUEL JENERATÖRÜN AKILLI JENERATÖRE
DÖNÜŞÜMÜ VE UZAKTAN İZLENMESİ

YÜKSEK LİSANS TEZİ

İBRAHİM ŞEREMET

BOLU, TEMMUZ - 2019

KABUL VE ONAY SAYFASI

İbrahim ŞEREMET tarafından hazırlanan “**MANUEL JENERATÖRÜN AKILLI JENERATÖRE DÖNÜŞÜMÜ VE UZAKTAN İZLENMESİ**” adlı tez çalışması Elektrik Elektronik Mühendisliği Anabilim Dalı'nda **03/07/2019** tarihinde **Bolu Abant İzzet Baysal Üniversitesi Fen Bilimleri Enstitüsü** Yüksek Lisans Tezi olarak kabul edilmiştir.

Jüri Üyeleri

Danışman
Dr. Öğr. Üyesi Nihat DALDAL
Bolu Abant İzzet Baysal Üniversitesi

Üye
Doç. Dr. Kemal POLAT
Bolu Abant İzzet Baysal Üniversitesi

Üye
Dr. Öğr. Üyesi Fatih KORKMAZ
Çankırı Karatekin Üniversitesi

İmza


.....

.....

.....

Prof. Dr. Ömer ÖZYURT

Fen Bilimleri Enstitüsü Müdürü

✓



AiLEME,

ETİK BEYAN

Abant İzzet Baysal Üniversitesi, Fen Bilimleri Enstitüsü Tez Yazım Kurallarına uygun olarak hazırladığım bu tez çalışmada;

- Tez içinde sunduğum verileri, bilgileri ve dokümanları akademik ve etik kurallar çerçevesinde elde ettiğimi,
- Tüm bilgi, belge, değerlendirme ve sonuçları bilimsel etik ve ahlak kurallarına uygun olarak sunduğumu,
- Tez çalışmada yararlandığım eserlerin tümüne uygun atıfta bulunarak kaynak gösterdiğimi,
- Kullanılan verilerde herhangi bir değişiklik yapmadığımı,
- Bu tezde sunduğum çalışmanın özgün olduğunu,

bildirir, aksi bir durumda aleyhime doğabilecek tüm hak kayıplarını kabullendiğimi beyan ederim.

İBRAHİM ŞEREMET

vekil Burak Şeremet


ETİK BEYAN

Abant İzzet Baysal Üniversitesi, Fen Bilimleri Enstitüsü Tez Yazım Kurallarına uygun olarak hazırladığım bu tez çalışmada;

- Tez içinde sunduğum verileri, bilgileri ve dokümanları akademik ve etik kurallar çerçevesinde elde ettiğimi,
- Tüm bilgi, belge, değerlendirme ve sonuçları bilimsel etik ve ahlak kurallarına uygun olarak sunduğumu,
- Tez çalışmada yararlandığım eserlerin tümüne uygun atıfta bulunarak kaynak gösterdiğimi,
- Kullanılan verilerde herhangi bir değişiklik yapmadığımı,
- Bu tezde sunduğum çalışmanın özgün olduğunu,

bildirir, aksi bir durumda aleyhime doğabilecek tüm hak kayıplarını kabullendiğimi beyan ederim.

İBRAHİM ŞEREMET

ÖZET

**MANUEL JENERATÖRÜN AKILLI JENERATÖRE DÖNÜŞÜMÜ VE
UZAKTAN İZLENMESİ
YÜKSEK LİSANS TEZİ
İBRAHİM ŞEREMET
BOLU ABANT İZZET BAYSAL ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
ELEKTRİK ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI
(TEZ DANIŞMANI: DR. ÖĞR. ÜYESİ NİHAT DALDAL)**

BOLU, TEMMUZ - 2019

Bu çalışmada manuel olarak çalıştırılan düşük güçlü jeneratörlerin düşük maliyetle otomatik çalışır jeneratöre dönüştürülmesi üzerine bir çalışma gerçekleştirilmiş ve jeneratörler ve şebekedeki bilgilerin uzaktan izlenmesi gerçekleştirilmiştir. Sistemde öncelikle manuel çalışmayı otomatikleştirecek yapı anlatılmış ve bunun için mikrodenetleyici tabanlı kontrol sistemi geliştirilmiştir.

Çalışmanın diğer aşamasında ise şebeke ve jeneratör üzerindeki gerilim ve frekans değerlerini uzaktan izleyebilmek için ENC2860J ethernet modülü mikrodenetleyicili sistem kartı ile haberleştirilerek verilerin internet ortamına gönderimi sağlanmıştır. Uzaktan bu değerleri takip edebilmek için modem üzerinde alınacak olan IP adresi ile yerel ağda bilgilerin izlenmesi gerçekleştirilmiş daha sonra Mathswork'un entegrasyonu olan ThingSpeak adı verilen bir internet sitesinde kullanıcı hesabı oluşturularak jeneratör ve şebeke değerlerini uzaktan izleme gerçekleştirilmiş ve başarılı olunmuştur.

ANAHTAR KELİMELER: Manuel jeneratör, Otomatik jeneratör, Uzaktan izleme, Arduino Mega 2560, ThingSpeak

ABSTRACT

**MANUAL GENERATOR CONVERSION TO SMART GENERATOR AND
REMOTE MONITORING
MSC THESIS
IBRAHİM ŞEREMET
BOLU ABANT İZZET BAYSAL UNIVERSITY GRADUATE SCHOOL OF
NATURAL AND APPLIED SCIENCES
OF ELECTRICAL AND ELECTRONICS ENGINEERING DEPARTMENT
(SUPERVISOR: ASSIST. PROF. DR.NIHAT DALDAL)**

BOLU, JULY 2019

In this study, a study was carried out on the conversion of low power generators operated manually to low cost automatic generator and remote monitoring of the generators and network information was carried out. Firstly, the system to automate manual operation is explained and a microcontroller based control system has been developed.

In the other phase of the study, to monitor the voltage and frequency values on the network and generator remotely, ENC2860J ethernet module was communicated to the system board with microcontroller and the data was sent to the internet. In order to follow these values remotely, the IP address of the modem will be monitored on the local network and then a user account was created on a web site called ThingSpeak which was the integration of Mathswork and succeeded.

KEYWORDS: Manual Generator, Automatic Generator, Arduino Mega 2560, Thing Speak

İÇİNDEKİLER

Sayfa

ÖZET.....	v
ABSTRACT	vi
İÇİNDEKİLER	vii
ŞEKİL LİSTESİ.....	viii
ÇİZELGE LİSTESİ.....	ix
KISALTMA VE SEMBOLLER LİSTESİ	x
TEŞEKKÜR	xi
1. GİRİŞ.....	1
1.1 Jeneratörü Uzaktan İzleme ve Kontrol	1
2. JENERATÖR	4
2.1 Jeneratör Çalışma Prensibi	4
2.2 Jeneratör Yapısı.....	6
2.2.1 Stator	6
2.2.2 Rotor	6
2.3 Jeneratör Özellikleri	7
2.4 Manuel Jeneratör	8
2.5 Transfer Panosu	10
2.6 Manuel Jeneratör Çalışması.....	10
2.7 Manuel Jeneratör	11
3. ARDUİNO.....	13
3.1 Arduino Temel Bileşenleri	13
3.2 Arduino Kütüphane Ekleme ve Kurulması	15
4. TRANSFER PANOSUNDAKİ DEĞERLERİN İZLENMESİ.....	17
4.1 Uzaktan İzlemenin Avantajları.....	17
5. THINGSPEAK IOT KULLANIMI.....	19
5.1 ThingSpeak Kullanımı ve Yapısı.....	20
6. JENERATÖR ÇEVİRİMİ VE PROGRAMINAÇIKLAMASI.....	25
7. JENERATÖRÜ UZAKTAN İZLEME	34
7.1 Modemden Alınan IP adresi ve Yerel Ağ Bağlantısı Kullanılarak	34
7.2 ThingSpeaak IoT ile Uzaktan İzleme.....	36
8. SONUÇ	40
9. KAYNAKLAR.....	41
10.EKLER.....	46
EK A.....	46
11.ÖZGEÇMİŞ.....	62

ŞEKİL LİSTESİ

Sayfa

Şekil 2.1.	Manuel Jeneratör	9
Şekil 2.2.	Jeneratör Eş Değer Devresi	9
Şekil 2.3.	Jeneratör İç Yapısı	9
Şekil 2.4.	Jikle Teli	11
Şekil 2.5.	Temel Karbüratör ve Jikle Yapısı.....	11
Şekil 3.1.	Arduino IDE Gösterimi	14
Şekil 3.2.	Arduino Mega 2560 R3 Versiyon	15
Şekil 5.1.	ThingSpeak My Channel Oluşturma Ekranı	20
Şekil 5.2.	ThingSpeak New Channel Oluşturma Ekranı	21
Şekil 5.3.	ThingSpeak Kanal Oluşturulduktan Sonraki Görüntüsü	22
Şekil 5.4.	ThingSpeak Özel Görünüm Ekranı	23
Şekil 5.5.	ThingSpeak API Anahtar Kodu	24
Şekil 6.1.	Sistem Bloğu.....	25
Şekil 6.2.	Yazılımın Akış Diyagramı.....	26
Şekil 6.3.	Gerilim-Frekans Ölçüm Devresi	27
Şekil 6.4.	Gerilim-Frekans Ölçüm Sinyalleri	27
Şekil 6.5.	Kurmuş Olduğumuz Devrenin Bir Görüntüsü.....	28
Şekil 6.6.	Kurmuş Olduğumuz Devre Sisteme Bağlıyken Görüntüsü.....	28
Şekil 6.7.	Sistemin Kütüphane ve Değişkenlerinin Akış Diyagramı.....	29
Şekil 6.8.	Şebeke-Jeneratör Değerlerinin Okunmasının Akış Diyagramı	30
Şekil 6.9.	Sistemin Şebeke Üzerindeki Okumuş Olduğu Değeri Yazdırması	31
Şekil 6.10.	Jeneratörü Çalıştırmayı Denerken ve Çalıştığına Dair LCD Yazısı.....	32
Şekil 6.11.	Jeneratörün Çalıştırılmadığına Dair Akış Diyagramı	32
Şekil 6.12.	Jeneratörün Kapatılamadığına Dair Akış Diyagramı	33
Şekil 7.1.	Enc2860J Ethernet Modülü	34
Şekil 7.2.	Sistemin İnternet Ağı Akış Diyagramı	35
Şekil 7.3.	Ağ Bağlantısının Başlatıldığını Gösteren Lcd Yazısı.....	36
Şekil 7.4.	Modülün Modemden Aldığı IP Adresi ve Elde Edilen Veriler	36
Şekil 7.5.	Şebeke Gerilim Grafiği.....	37
Şekil 7.6.	Jenereatör Gerilim Grafiği.....	38
Şekil 7.7.	Şebeke Frekans Grafiği	38
Şekil 7.8.	Jeneratör Frekans Grafiği	39
Şekil 7.9.	Jeneratör Isı Grafiği	39

ÇİZELGE LİSTESİ

Sayfa

Çizelge 7.1. Enc2860J Modül Bağlantı Pin Numaraları.....	35
--	----



KISALTMA VE SEMBOLLER LİSTESİ

API	: Uygulama Programlama Arayüzü
AVR DUDE	: Arduino Üzerindeki Mikrodenetleyici Programlayan Yazılım
AVR-GCC	: Arduino Derleyicisi
EEPROM	: Electronically Erasable Programmable Read-Only Memory
EKG	: Elektrokardiyogram
EMK	: Elektro Motor Kuvveti
GPRS	: General Packet Radio Service
GUI	: Graphical User Interface
ID	: Kullanıcı Adı
IP	: Internet Protocol
IDE	: Arduino Geliştirme Programı
IoT	: Internet of Things
IP	: Internet Protocol
LAN	: Local Area Network
LCD	: Liquid Crystal Display
OS	: Operating System
PWM	: Darbe Genişlik Modülasyonu
RFID	: Radio Frekansı İle Tanımlama
SRAM	: Static Random Access Memory
WAN	: Wide Area Network
WiFi	: Wireless Fidelity
WSN	: Wireless Sensor Network

TEŞEKKÜR

Bu alıřmada, manuel jeneratörün otomatik jeneratöre dönüřtürölmesi ve řebeke ile jeneratör enerjisinin uzaktan takibi ele alınmıřtır. alıřma konusunun belirlenmesi ve hazırlanması sürecinde kıymetli tecrübeleri ile beni yönlendiren deęerli hocam, Dr. Öğr. Üyesi Nihat DALDAL'a en içten dileklerimle teşekkür ederim. Eğitim-öğretim hayatım boyunca deęerli bilgilerini benle paylaşan hocalarıma ve her zaman benden desteklerini esirgemeyen aileme teşekkür ederim.

1. GİRİŞ

Gelişen dünyada elektrik insanlar için vazgeçilmez bir unsur olmuştur. Elektrik dağıtımı ve kullanımı sırasında yaşanan problemler nedeniyle insanların elektriğe olan ihtiyacını devam ettirebilmek için jeneratörlere olan ihtiyacımız artmıştır. Konutlarda, iş yerlerinde, fabrikalarda enerji devamlılığını devam ettirebilmek için ihtiyacımız olan enerji değeri hesaplanarak farklı güç ve tipte otomatik jeneratörler seçilebilir. Seçilen bu jeneratörlerin kullanımı süreleri parametre değerleri, arıza detayları , bakım periyotlarını uzaktan izlemek kullanıcıya maliyeti arttırmaktadır. Bu yüzden daha pahalı olan otomatik jeneratörleri yerine manuel jeneratörleri otomatik jeneratörlere dönüştürerek maliyet düşürülebilir.

Bu çalışmamda manuel jeneratörlerin otomatik jeneratöre çevrilmesi ve üzerindeki gerilim ve frekans değerlerinin uzaktan takip edilmesi irdelenmiş olup daha önce bu konu üzerine yapılan literatür araştırması yapılmıştır. Manuel jeneratörlerin fiyatlarının otomatik jeneratörlere göre çok ucuz olması ve buna benzer bir çalışmanın bulunmaması, daha düşük maliyetli otomatik jeneratör elde etme fikri çalışmanın amacıdır. Ayrıca uzaktan izleme programı olan bu çalışmada da kullanılan Thing Speak IoT'un yapılan bilimsel yazın araştırmalarında bu uygulamanın jeneratörler üzerinde uygulanmadığı görülmüş, başka uygulamalarda kullanıldığı alanlar ortaya çıkarılmıştır.

Thing Speak IoT'un uygulandığı alanlar ve jeneratörün uzaktan izlenmesi ve kontrol edilmesi ilerleyen alt başlıklarda verilmiştir.

1.1 Jeneratörü Uzaktan İzleme ve Kontrol

Yapılan bilimsel araştırmalarda bulunan ve kullanılan yöntemler hakkında birçok çalışma bulunmuştur.

Achkar vd. (2014) Jeneratörlerdeki değişkenlerin yağ, sıcaklık, voltaj ve akım günlük olarak izlenerek kontrol edilmesini sağlamaktadır. Bu çalışmada akıllı telefon

iřletim sistemi olan andorid kullanılarak elektrik jeneratörlerini kablosuz olarak uzaktan izlemeyi amaçlamaktadır. Bunun için verileri ölçmek, uygulanan sensörleri jeneratörün gerçek zamanlı verilerini analog sinyallere dönüřtürmektir. Bu veriler Arduino gibi bir mikro denetleyici kullanılarak programlanmış ve dijitale dönüřtürölerek çıkıřı android telefona yönlendirilerek seri sinyaller iletebilmektedir. Böylece jeneratörün durumu gerçek zamanlı canlı bir geri bildirim sağlamıřtır. Ayrıca bu projede jeneratörde açıp kapatabilecek buton kullanılarak kontrol düğmesi konulmuřtur.

Wróblewski vd. (2009) Güç jeneratörlerinin durumunun uzaktan izlemek için mobil uygulama geliřtirmişlerdir. Bu uygulama Symbian OS tabanlı cep telefonları üzerinde Python yazılım dili kullanılarak geliřtirilmiştir. Jeneratörlerin durumlarının ne olduėunu bulmak için önce sensörler aracılıėı ile titreřimler ölçölerek makinenin durumuna iliřkin göstergeler hesaplanmaktadır. Hesaplanan bu titreřim deėerleri genellikle hız ve ivme olarak ifade edilmiştir. Ölçölün bu deėerler uzaktan Wi-Fi ve GPRS kullanılarak eriřim saėlanıp veri tabanına saklanmaktadır. Bunun için orijinal uygulama olan LabWiev ortamı kullanılarak bir ara yüz geliřtirilmiştir.

Ezeanyagu vd. (2015) Baz istasyonlarına güç saėlamak için řebeke gerilimine ek olarak kullanılan dizel jeneratörlerin uzaktan izlenmesine olanak saėlamaktadır. Sistem XB-24ZB radyo ünitelerini ve bir de dizel yakıt tanklarının seviyelerini uzaktan izleyebilmek için ultrasonik sensör kullanılarak geliřtirilmiştir. Elde edilen veriler Arduino Atmega 328 mikroişlemcisine Xbee modölü kullanılarak GUI de hazırlanmış ara yüze aktarılmıştır. Bu aktarımda WSN adı verilen sensör aėları kullanılmıştır.

Federenciuc vd. (2010) Geniř çaptaki birbirine entegre olabilecek daėıtılmış enerji kaynaklarının sistemin azalmasına neden olabilecek elektronik ünitelerin mevcut kontrol ile uyumsuzlukla sonuçlanmasına neden olabilecek algoritmaları ayarlamaktadır. Bu kaynakların birbirleri ile birleřtiğinde küçük depolama üniteleri sanal senkronize çalışan jeneratörler üretilip ve bunların tasarımı ve gerekli izlemeleri yapılmaktadır. Kaynaklara ait veriler frekans, řebeke voltajı, akımlar, dalga řekli ve reaktif güç gibi deėerler Ethernet baėlantısı kullanılarak oluřturulan ara yüzde saklanmaktadır.

Tripathi vd. (2014) Internet of Thing sensörlerin RFID ile cihazların bilgilerini alma saklama gibi akıllı nesnelerle internetin birleşimini kastetmektedir. Günümüz teknolojinin en büyük vaadi olan Internet of Things tek başına yeterli gelmemektedir. Bu yüzden Semantik Füzyon Modeli (SFM) yardımıyla Internet of Things için bir mimari oluşturulmakta ve akıllı gömülü sistemler, RFID, sensör ağları için yeni uygulamalar geliştirilmektedir.

Evans (1990) Senkron bir jeneratörü bir elektrik sistemine senkronize etmede kritik delta faz açısı, delta frekansı, delta volta büyüklüğü gibi parametreler mevcuttur. Bu üç kritik parametre için manuel/otomatik senkronizasyon devre tasarlanmıştır. Hem manuel hem de otomatik devreler için koruma devresi ile beraber tasarlanmıştır.

Zhang vd. (2008) Turbo jeneratörler elektrik üretimi için önemli bir makinedir. Yapısal karmaşıklığı turbo jeneratörlerin yüksek arıza oranına sahip olmasına neden olmaktadır. Jeneratörlerin durumunun izlenmesi arıza durumu, elektriğin güvenli üretimi ve istikrarlı ve yüksek düzeyde etkili olması gerekmektedir. Sistemin başarılı bir şekilde çalışması ve uzaktan izlenmesini gerektirecektir. Bilgisayarın hızlı gelişimi ile birlikte B/S modeline dayanan turbo jeneratörler şebeke iletişimini Java, JSP gibi yazılım programları ile LAN ve WAN internet teknolojisi kullanılarak uzaktan durumlarının izlenmesine olanak sağlamaktadır.

Kumar vd. (2017) Günümüzdeki akıllı ev mekanizasyonu, akıllı ev alanları , araç park yerleri , trafik kontrolü, akıllı çevre, tarım alanları ve hasta sağlığı gibi izleme sistemin gibi farklı metodolojilerde Internet of Things araçları önemli bir rol üstlenmektedir. Bu rol modellerden biri olan hastanın sağlık durumunu izleme ve hastayı uzaktan kontrol etmektir. Hastanın sağlık durumunu kontrol edebilmek için solunum hızı, vücut sıcaklığı, kan şekeri, EKG gibi değerleri kontrol edebilmek için sensörler kullanılmıştır. Bu sensörler Arduino Uno kartı ile ilişkilendirilip Thing Speak adı verilen sunucuya aktarılarak uzaktan izleme sağlanmaktadır.

2. JENERATÖR

Jeneratörler, mekanik olarak üretilen enerjiyi elektrik enerjisine çeviren alternatör ve motordan meydana gelen makinelerdir.

Bu makineler, ışık enerjisini elektriğe çeviren, kimyasal enerjiyi elektrik enerjisine çeviren bataryalar, ısı enerjisini elektrik enerjisine çeviren termoelektrik jeneratörler ve dizel jeneratörleri de kapsamaktadır (Boduroğlu, 1962).

Mekanik enerjiyi elektrik enerjisine iki çeşit makine dönüştürebilmektedir.

- 1- Alternatif Akım (AC) jeneratörler
- 2- Doğru Akım (DC) jeneratörler

Günümüzde elektrik enerjisinin büyük bir bölümünü AC jeneratörler üretilmektedir. Bu makinelere alternatör adı verilmektedir.

Alternatörler, tüm su ve buhar enerjisi ile güç elde eden santrallerde ana jeneratörlerdir. Trafolar alternatif gerilimi kolay bir şekilde yükseltebilir veya düşürebilir. Elektrik enerjisinin nakli sırasında uzak mesafelere yüksek gerilim, dağıtım ve kullanım için düşük gerilim gerekmektedir.

2.1 Jeneratör Çalışma Prensibi

Jeneratörler prensip olarak Faraday Yasasına göre çalışmaktadır. Bir tel bobini çevreleyen manyetik akım çizgisi değiştirildiği zaman, bobinde manyetik akıya göre değişmekte olan sarım sayısı ile orantılı elektromotor kuvvet (EMK) meydana gelmektedir (Boduroğlu, 1962).

Elektrik enerjisi üretebilmek için hidroelektrik enerji, güneş enerjisi, rüzgâr enerjisi, buhar türbinleri, doğalgaz sıkıştırılarak elde edilen mekanik enerji kaynakları kullanılarak jeneratörlerin ilk hareketi sağlanarak, Bobin sayesinde manyetik alan içindeki indüksiyon çizgilerini keserek hareket sağlanır ve elde edilen elektriksel yük alıcılara iletilir.

Dinamoda elektromanyetikse indüksiyon akımı, mıknatıslı bir alan içerisinde kapalı bir devre hareket ettirilerek elde edilmektedir. Bobin N-S kutupları dönel bir alan oluşturulmak suretiyle bir alternatif gerilim meydana getirmektedir.

Jeneratörlerin çalışma prensibi olarak elektrik motorlarının tersinedir. Burada önemli olan nokta eksenin dönme hızının oluşturulan akımın büyüklüğünü etkilemesidir. Bu da verimin artmasını sağlamaktadır. Sipirlerde meydana gelen ve saat yönünde dönmekte olan N-S kutupları içerisinde dönmekte olan bobin kuvvet çizgilerini kesmektedir. Saat yönünde dönmekte olan bobin, alternatif bir akım indüklenenecektir. Üretilen gerilim, fırça ve kömür ve bakır dilimleri ile dışarı taşınmaktadır. Taşınan bu gerilim ve indüklenen bu akım sayesinde bağlanan yük çalışmaktadır (Altunsaçlı vd., 2000).

Jeneratörü oluşturan kısımlar;

1. Kontrol Üniteleri
2. Elektrik Sistemi
3. Yakıt Sistemi
4. Havalandırma Sistemleri
5. Yağlama Sistemleri
6. Soğutma Sistemleri
7. Alternatör

Jeneratörlerde tahrik kaynağı, bir dizel(motorin),benzin veya doğalgaz kullanılarak kimyasal enerjiyi mekanik enerjiye dönüştürmesi sağlanmaktadır. Alternatör ise elde edilen bu mekanik enerjiyi elektrik enerjisine çeviren kısmı oluşturmaktadır.

Jeneratörde üretilen elektrik enerjisini kontrol edebilmek için bir kontrol ünitesi gerekir. Bu ünite, sigorta, kontaktör, şalter ile kontrol edilebilir olup yüklerin beslenmesine yardımcı olmaktadır. Ayrıca gerilim, akım, faz sırası, vb. gibi parametreleri ölçen, izleyen bir kontrol alarm ünitesini kullanılmıştır.

2.2 Jeneratörlerin Yapısı

Jeneratörlerin yapısı genel olarak iki kısımdan oluşmaktadır. Bunlar;

1. Stator
2. Rotor

2.2.1 Stator

Stator, elektrik enerjisinin indükleyen, sargıların bulunduğu ve jeneratörün duran kısmını oluşturmaktadır. Büyük güçlü jeneratörlerde fazla akım oluşturmaktadır ve stator duran kısmı oluşturduğu için fırça ve bileziğe gerek duymadan oluşturulan bu akım kolaylıkla dışarı aktarılabilir. Fırça ve bilezikler dönel kısım olan rotorda kullanılmaktadır.

2.2.2 Rotor

Rotor jeneratörün dönel kısmını oluşturmaktadır. Büyük çaptaki jeneratörlerde N-S kutupları oluşturulan sargılar rotorda bulunmaktadır. İki çeşit rotor yapısı bulunmaktadır. Bunlar;

- Kutupları Çıkıntılı Rotorlar
- Kutupları Silindirik (Düz) Rotorlar

Kutupları çıkıntılı rotorların devir sayıları küçüktür. Bu yüzden düşük devirde çalışmaktadır. Demir nüve kayıpları azaltmak için özel saclardan yapılmaktadır. Sargı uçları rotor üzerinde bulunan bileziklere tutturulur. Çok kutuplu yapıda olup fırça, bilezik ve rotor sargıları doğru akımla beslenmektedir.

Kutupları düz olan rotorlarda sargılar rotor üzerine yerleştirilmektedir. Sargıların uçları rotor üzerindeki bileziklere montajı yapılmaktadır. Stator uçları üç fazlı olmakta U-V-W ve Z-X-Y olarak adlandırılmaktadır (Altunsaçlı vd., 2000).

2.3 Jeneratör Özellikleri

Jeneratörlerin genel olarak sekiz kısımda incelenmektedir. Bunlar;

1. Regülasyon
2. Gerilim Kararlılığı
3. Yüke Karşı Regülasyon
4. Ani Yüke Karşı Regülasyon
5. Frekans Kararlılığı
6. Aşırı Yük ve Kısa Devre Koruması
7. Toplam Harmonik Distorsiyon (THD)
8. Yüksek Verim

Regülasyon giriş gerilimindeki değişimin kontrol edilmesini sağlamaktadır. Türkiye’de şebeke gerilimindeki değişim -20 Volt ile +20 Volt arasında değişkenlik göstermektedir. Kesintisiz güç kaynaklarında olması gereken çıkış geriliminin hata payı yüzde artı eksi 1 seviyelerinde olmalıdır. Bu yüzden çıkış geriliminin değişen şartlarda sabit kalması istenmektedir. Bu sayede gerilim kararlı bir seviyede durması sağlanmaktadır.

Çıkıştaki yüklerin değişimine rağmen çıkış geriliminin sabit kalması yüke karşı regülasyonun göstergesidir. Kesintisiz güç kaynaklarında bu değer hata payının yüzde artı eksi 1 den düşük değerde olması istenmektedir. Frekans kararlılığı ise çıkış yükünün ani değişimlerinde ve şebeke geriliminin kesilmesi ve gelmesi durumlarında çıkış oluşacak değişimin yüzde 10 seviyelerinde olması istenmektedir.

Çıkışta oluşabilecek aşırı yük ve kısa devrelere karşı kesintisiz güç kaynaklarının kendini koruması ve çalışmayı sürdürmesi gerekmektedir. Kendi yükünün 1,5 katı kadar oluşabilecek aşırı yüklerde bile kesintisiz güç kaynaklarının gerilim regülasyonunun en az 1 dakika süre ile çalıştırılması gerekmekte ve hata ortadan kalktığında herhangi bir müdahaleye gereksinim duymadan çalışmasının sürdürülmesi gerekir.

Toplam Harmonik Distorsiyon (THD) ise çıkış gerilimindeki harmoniklerin bir ölçüsünü belirtir. Bu harmonikler frekansa dayalı çalışan cihazların çalışmasını etkiler. Bu yüzden bu değer doğrusal sistemlerde %3’ten, doğrusal olmayan

sistemlerde %5'ten küçük olması gerekir. Kesintisiz güç kaynaklarında PWM(Darbe Genişlik Modülasyonu) kullanılarak bu harmoniklerin değeri azaltılabilir.

Kesintisiz güç kaynaklarında da enerji kayıpları oluşmaktadır. Burada önemli olan nokta enerji kayıplarının düşük seviyede kalmasını sağlamaktır. Bu sayede yüksek verim alınabilecektir. Kesintisiz güç kaynaklarında gücün devamlılığını sağlamak için aküler kullanılır.

2.4 Manuel Jeneratör

Manuel jeneratörler enerji kesintisinde kullanıcı tarafından çalıştırılan jeneratörlerdir. Kumanda panosundan jeneratöre manuel bir kontak geldiğinde marş motoru akülerden aldığı elektrik enerjisini kullanarak dizel motorun ilk çalışması için yakıt pompalanmaya başlamaktadır. Motor çalıştıktan sonra Faraday Yasasına göre değişken manyetik alan içerisindeki stator sargılarında gerilim endüklenir. Endüklenen bu gerilim sarım sayısı ile doğru orantılı artar. Üretilen gerilimin sabit olması için stator sargılarına gerilim regülatörü bağlanır. Üretilen gerilim frekansı fazlar arası gerilim ve çekilen güç anlık olarak kumanda panosunda(veya transfer panosu) görülür. Bu şekilde elektrik enerjisi gelen kadar çalışma devam eder.

Jeneratörlerde, çıkış frekansı;

$$f_e = \frac{n_m \cdot P}{120} \quad (2.1)$$

Formülü ile hesaplanır. Burada rotor dönüş hızı ve kutup sayısı çıkış frekansını belirler. Stator fazında indüklenen gerilim genliği;

$$E_A = K \cdot \Phi \cdot \omega \quad (2.2)$$

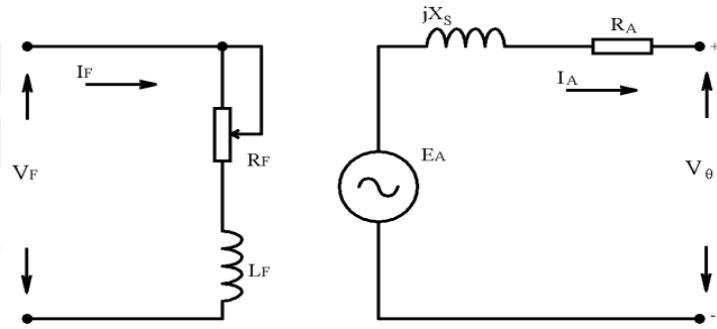
Denklemi ile ifade edilirken, indüklenen gerilimin doğrudan akı ve hız ile orantılı olduğu görülmektedir. Sargı uçlarındaki gerilim ise;

$$V_\phi = E_A - jX_s I_A - R_A I_A \quad (2.3)$$

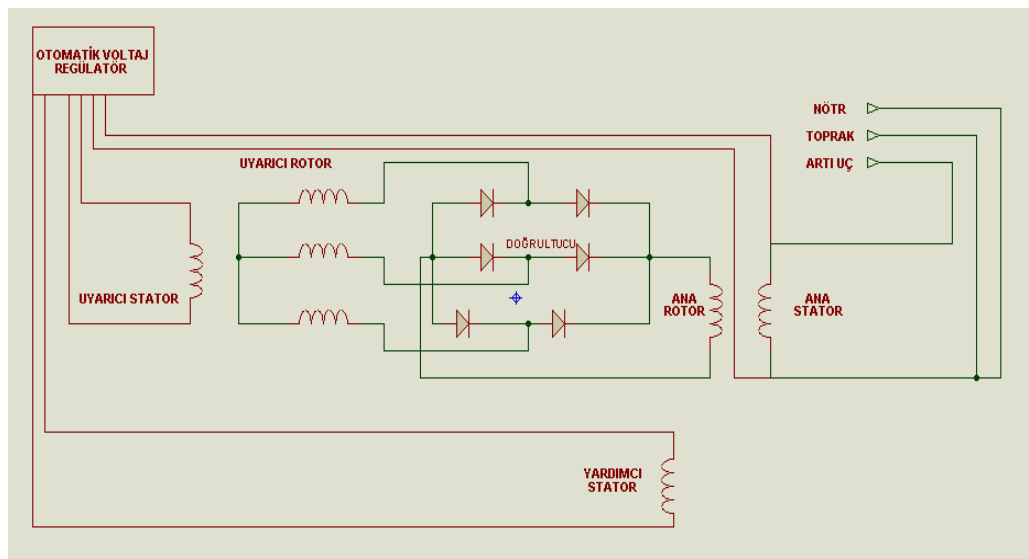
eşitliğine bağlıdır. Burada Çıkış gerilimini X_s (senkron reaktans) , I_A (endüvi akımı) ve R_A (endüvi sargı direnci) belirler.



Şekil 2.1. Manuel jeneratör.



Şekil 2.2. Jeneratör eş değer devresi.



Şekil 2.3. Jeneratör iç yapısı.

2.5 Transfer Panosu

Şebeke veya jeneratör enerjisini yük çıkışına yönlendiren ve anahtarlamasını sağlayan sistemdir. Anahtarlamalar kumanda panolarının şebeke ve jeneratör enerjisini kontrol ederek uygun durumda transferi gerçekleştirmesini sağlar. Transfer sistemi tüketilen güce göre kontaktörler veya motorlu şalterler ile gerçekleştirilir. Transfer panolarının gücü jeneratörün besleyeceği güce göre en az jeneratör gücünde olmalıdır.

Dağıtım panosunda jeneratörün besleyeceği yükler acil ve acil olmayan yükler olarak ayrılır. Bazı durumlarda dağıtım panolarında acil olan ve olmayan yüklerin ayrımı yapılmayarak şebeke enerjisi varken birebir yükleme, jeneratörle yapılan beslemede ise kısmi yükleme yapılmaktadır. Bu gibi durumlarda şebeke gücü jeneratör gücünden fazla olduğu için transfer panosunda kullanılan kontaktör veya motorlu şalter şebeke gücüne uygun seçilmelidir.

2.6 Manuel Jeneratör Çalışması

Manuel jeneratörün çalışması için gerekli adımlar aşağıdaki gibidir.

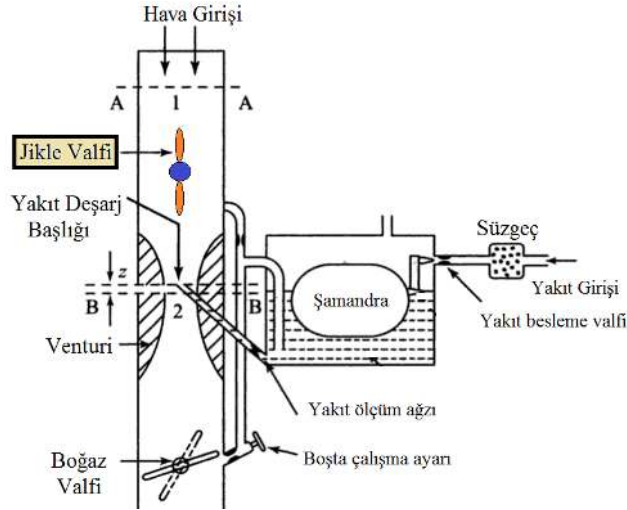
1. Manuel jeneratörün üzerinde start-stop butonu ve jigle bulunur.
2. Elektrik kesildiğinde (hava şartlarına bağlı olarak değişir) jigle kolu ve butonu çekilir ve marş butonuna basılır,
3. Marş butonuna basıldıktan sonra jeneratör çıkış verirse jeneratör çalışmış olur.
4. Eğer jeneratör çıkış vermemiş olursa stop butonuna basılır ve işlemler tekrar başa dönlür.

Jikle, soğuk havalarda motoru çalıştırmak için gerekli olan yakıt karışımının ayarlanmasını sağlar. Havalar çok soğuk olunca hava-yakıt karışımı motorda düzgün çalışmaz veya motor hiç çalışmaz. Bu yüzden jikle teli çekildiğinde yakıt hava ile daha iyi oranda karışarak yanmayı gerçekleştirir ve motor çalışır. Jikle, gerektiğinde karbüratörün ağzından içeri giren havayı kontrol eden bir mekanizmadır. Jiklenin plakası kapanınca vakum artarak daha hızlı yakıt pompalanır ve jigle karbüratörün üzerinde tel ile bağlı olup itilerek ve çekilerek kontrol edilir. Jikle, motor ısındıktan

sonra kapatılmalıdır. Aksi takdirde aşırı yakıt sarfıyatına ve motorun boğulmasına neden olur (<https://www.tasit.com/araba-sozlugu/jikle-2-nedir>).



Şekil 2.4. Jikle teli.



Şekil 2.5. Temel karbüratör ve jikle yapısı.

2.7 Manuel Jeneratörler Otomatik Olarak Çalıştırılması

Bu projede gerçekleştirilen işlemler Arduino Mega 2560 R3 kullanılmıştır. İlerleyen bölümlerde yazılan kodlarla beraber bu adımları nasıl gerçekleştirildiği anlatılmaktadır. Manuel jeneratörlerin otomatik çalıştırılması ilgili adımlar aşağıda belirtilmiştir.

- 1- Jeneratör, şebeke enerjisini her daim izleyecektir.
- 2- Elektrik kesildiğinde 10 sn bekler.10 sn sonunda jigle vanasını çekilip marş butonuna 3 sn basılıp Jeneratör çıkış verirse sisteme otomatik çalışmış olacaktır.

- 3- Eğer jeneratör çıkış vermezse stop butonuna basılıp bir süre beklenir. Ve 2 no'lu adım tekrar gerçekleştirilir.
- 4- Bu döngü sizin yazacağınız yazılımınızda istediğiniz kadar devam edecektir. Eğer isterseniz istediğiniz deneme sayısına gelindiğinde jeneratör çalıştırmayı denemeyi bırakacaktır.
- 5- Tekrar bir çıkış alınamazsa akü durumu ve yağ durumu kontrol edilmelidir.
- 6- Şebeke tekrar geldiğinde 10 saniye (yazılımın durumuna göre değişiklik gösterilebilir eğer istenirse 1 dakika kadar beklenebilir.) beklenmelidir. Bu süre sonunda stop butonuna basılmalıdır. Burada amaç şebekenin aniden gelip gitmesini kontrol edilmesi olacaktır. Böyle bir durumda jeneratör korunmuş olacaktır.

3. ARDUINO

Arduino; bir Giriş/Çıkış kartı ve basit bir programlama dili olan Processing/Wiring denilen bir Java tabanlı uygulamasını içeren geliştirme ortamından oluşan bir fiziksel programlama platformudur. Processing/Wiring yapı olarak C diline benzemektedir. Açık kaynaklı kodlar bulunduğu için programlamayı hiç bilmeyen biri bu dili rahatlıkla çözebilir.

Arduino kartlarının donanımında bir adet Atmel AVR mikro denetleyici (ATmega328, ATmega2560, ATmega32u4 gibi) ve programlama ve diğer devrelere bağlantı için gerekli yan elemanlar bulunur (Güray, 2013).

Her Arduino kartında en azından bir 5 voltluk regüle entegresi ve bir 16MHz kristal osilatör (bazılarında seramik rezonatör) vardır. Arduino kartlarında programlama için harici bir programlayıcıya ihtiyaç duyulmaz, çünkü karttaki mikro denetleyiciye önceden bir bootloader programı yazılımı yüklüdür (Güray, 2013).

3.1 Arduino Temel Bileşenleri

Arduino 'nun temel bileşenleri

- 1- Arduino geliştirme ortamı (IDE),
- 2- Arduino bootloader (Optiboot),
- 3- Arduino kütüphaneleri, AVR Dude (Arduino üzerindeki mikro denetleyici programlayan yazılım)
- 4- Derleyiciden (AVR-GCC)'den oluşur.

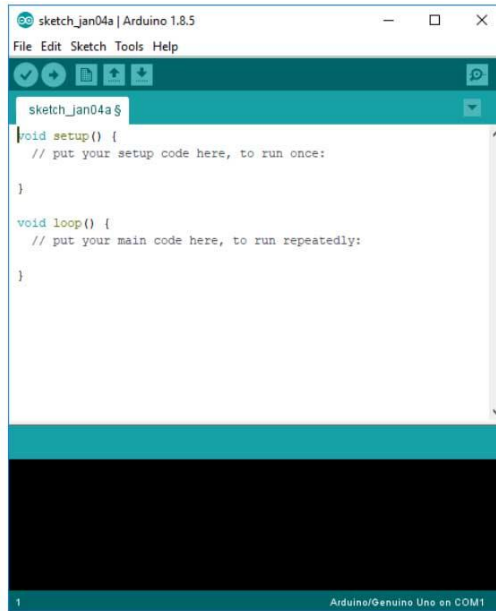
Arduino yazılımı bir geliştirme ortamı (IDE) ve kütüphanelerden oluşur. IDE Java dilinde yazılmıştır ve Processing adlı dilin ortamına dayanmaktadır.

Kütüphaneler ise C ve C++ dillerinde yazılmıştır ve AVR-GCC ve AVR Libc. ile derlenmektedir.

Optiboot bileşeni Arduinonun bootloader bileşenidir. Bu bileşen Arduino kartlarının üzerindeki mikro denetleyicinin programlanmasını sağlayan bileşendir. Arduino hafızasının bir kısmını kendine ayırmaktadır. Yani cihaz başlatıldığı hazır hale getirebilmek için Arduinonun kendi için ayırmış olduğu sistem kısmını oluşturur.

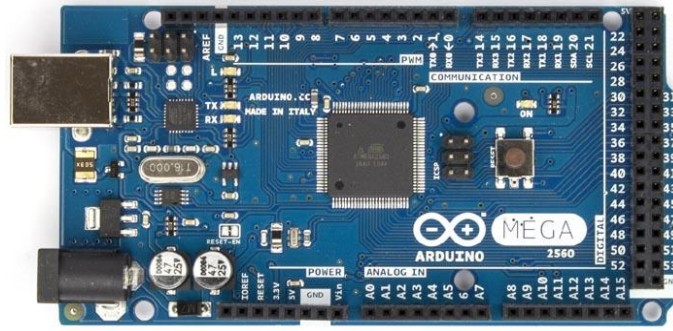
Arduino'nun bu kadar tercih edilmesini sağlayan en önemli bileşen ise mikro denetleyici konusunda detaylı bilgi sahibi olmayı gerektirmeden herkesin programlama yapabilmesini sağlayan Arduino kütüphanelerinin ve yazım dilinin kolay bir dille yazılmış olmasıdır.

Arduino kütüphanelerinin listesine Arduino kendi sitesinden veya bazı kişilerin www.github.com adlı sitede paylaşmış olduğundan kolaylıkla ulaşılabilir. Arduino kütüphaneleri, geliştirme ortamı ile birlikte "Libraries" klasörünün altında bulunmaktadır. Kodları inceleyerek mikro denetleyicilerin nasıl programlandığını ve kütüphanelerin yapısını görmemize mümkün kılmaktadır. AVR-Dude bileşeni ise yazdığımız kodları derledikten sonra programlamak için kullanılır (<https://github.com/arduino-libraries/SD>).



Şekil 3.1. Arduino IDE gösterimi.

Bu çalışmada Arduino Mega 2560 kartı kullanılmıştır.



Şekil 3.2. Arduino Mega 2560 R3 versiyon.

Karta ait özellikler şu şekilde sıralanabilir.

1. Mikro denetleyicisi Atmega2560
2. Çalışma gerilimi 5V DC (Ancak önerilen çalışma gerilimi 7-12 V DC)
3. Harici besleme gerilimi 6-12 V
4. Digital giriş ve çıkış pinleri 54 tanedir. Bunlardan 15 tanesini PWM (pulse width modulation) için kullanılabilir.
5. Analog pinleri 16 tanedir.
- 7- Giriş-Çıkış pin başına düşen akım DC 40 mA(miliamper) 'dir.
6. Arduinonun hafızası 256 kb (kilobayt) 8 kb'ını bootloader kullanılmaktadır.
7. SRAM (Static Random Access Memory) 8 kb'lık kısmı oluşturmaktadır
8. Küçük boyuttaki verileri kalıcı olarak saklamak için kullanılan EEPROM (Electrically Erasable Programmable Read-Only Memory) hafızası 4 kb tır.
9. Saat frekansı yani elektrik devresinin saniyede ürettiği darbe sayısı 16 MHz(Mega Hertz)'dir.

3.2 Arduino Kütüphane Eklenmesi ve Kurulması

Arduinonun IDE'si indirildiğinde standart bazı kütüphaneler yüklü gelmektedir. Ancak projeleri geliştirdikçe yüklü gelmeyen kütüphaneleri daha önce belirtmiş olduğumuz siteden indirilip kurulması gerekmektedir. Bu çalışmada da bazı kütüphane ve modülleri eklememiz gerekmektedir. Bunları internette indirdiğimiz kütüphane dosyasını bilgisayarın "C:\Program Files (x86)\Arduino" klasöründeki

libraries dosyasına eklenmelidir. Veya Arduino IDE'deki taslak sekmesinin altındaki zip ekle bölümünden eklenebilmektedir. Ayrıca indirmiş olduğumuz kütüphaneleri veya kullanacağımız modüllere ait dosyaları yine aynı taslaklar sekmesinde library düzenle sekmesine tıklanıp gerekli kütüphaneler düzenlenip kullanılacak olan modüllerin dosyaları güncellenip kurulmalıdır (Çamoğlu, 2014).

Bu çalışmada da ENC2860J Ethernet modülü kullanıldığından bu modüle ait dosyalar ve Ethernet Kütüphanesi ile kullanılan servo motor kütüphanesi kurulmuştur.



4. TRANSFER PANOSU ÜZERİNDEKİ DEĞERLERİN İZLENMESİ

Enerji otomasyon sistemleri, genel olarak tesislerde bulunan trafoların, jeneratörlerin, AG panolarının ve şalterlerinin durum ile arızalarının izlenmesi ve kumanda edilmesi, OG kesici ve ayırıcılarının izlenmesi ve kumanda edilmesi, enerji analizörleri veya güç dönüştürücü cihazlar kullanılarak akım, gerilim, güç, güç faktörü, aktif-reaktif enerjinin izlenmesini, koruma rölelerinin izlenmesini ve kumanda edilmesini sağlamaktadır (<https://www.voltimum.com.tr/haberler/transfer-panosu-nedir-ve-nasil-calisir>).

Günümüzde kurulan enerji otomasyon sistemleri, bir tesisin sadece girişindeki akım, gerilim, güç ve enerji değerlerinin ölçümünden, bir organize sanayi bölgesi ve enerji santralindeki geniş bir lokasyon üzerinde kurulu tesislerde her türlü otomatik senaryoların uygulanması, senkronizasyon işlemlerinin yapılması, yük atma, alma işlemlerinin çok hızlı bir şekilde uygulanması, koruma fonksiyonlarının izlenmesi ve kontrol edilmesi, uzaktan sayaç okuma ve faturalama işlemlerinin yapılması ve enerji kalitesinin takip edilmesi ve raporlanmasına kadar ölçeklendirilebilmektedir (Güzelbeyoğlu, 1991).

4.1 Uzaktan İzlemenin Avantajları

Bir jeneratörün otomatik veya manuel çalışmasının birtakım avantajları ve dezavantajları vardır. Bunlar aşağıda sıralanmıştır.

1. Elektrikler kesilince devreye giren jeneratörler hayatın kesintiye uğramadan devam etmesinde önemli rol oynamaktadır.
2. Özellikle hastane, ilaç deposu, fabrika gibi kesintisiz elektriğin kritik önem taşıdığı yerlerde jeneratörlerin sağlıklı çalışması şarttır. Bu nedenle dizüstü bilgisayar, tablet veya akıllı telefon üzerinden kullanılabilen uzaktan izleme ve kontrol sistemleri ile jeneratörde meydana gelebilecek sorunlara anında müdahale imkânı elde edilebilir.

3. Çekilen yük akım değerleri, fazlardaki gerilim değerleri, buna bağlı olarak güç değerleri anında takip edilebilir.
4. Yakıt seviyelerini izleme ile, depo dolum zamanı için sistem gerekli uyarıları yaparak yakıt istenilen zamanda doldurulabilir.
5. Yağ basıncı ve akü durumunu takip edilebilir. Çünkü jeneratörün ilk çalışması için gerekli enerjiyi aküden alır. Bu yüzden akünün seviyesi önem arz etmektedir.
6. Soğutma suyu sıcaklığını görülebilir.
7. Konum sensörü kullanılarak uzaktan jeneratörün ve yakıt deposunun hırsızlığa karşı takip edilmesi sağlanacaktır.
8. Çalışma saat süreleri izlenebilir.
9. Bakım periyot zamanları takip edilebilmektedir.
10. Sıcaklık ve nem gibi ortam değişkenlerini izlenebilmektedir.
11. Opsiyonel olarak ortama yerleştirilecek bir kamera ile uzaktan görüntüleme yapılabilmektedir (<https://www.elektrikstok.com/jenerator-transfer-panosu-nedir>).

Bu çalışmada jeneratörü uzaktan izlemeyi gerçekleştirebilmek için ThingSpeak IoT (İnternet of Things) adı verilen Mathwork' un geliştirdiği site kullanılmıştır. Thing Speak'in yapısı ve kullanım aşamaları bir sonraki kısımda incelenmiştir.

5. THINGSPEAK INTERNET OF THINGS KULLANIMI

Evimizde, akıllı telefonlarımızda, otomobillerde, şehir altyapısında ve endüstriyel ekipmanlarda kısaca her tarafta sensörler mevcuttur. Sensörler ; sıcaklık, nem ve basınç gibi nesneler üzerindeki her türlü bilgiyi algılayarak ve ölçerek bu verileri sayısal bir değer veya elektriksel sinyal g oluşturup istediğimiz siteye iletmemizi sağlar.

Sensörler veya nesneler, verileri algılar ve genellikle yerel olarak hareket ederler. ThingSpeak, sensörlerin, enstrümanların ve web sitelerinin, özel veya kamu kanalında saklandığı bulutlara veri göndermesini sağlamaktadır. ThingSpeak verileri özel kanallarda varsayılan olarak depolar, ancak diğer kanallarla veri paylaşmak için ortak kanallar kullanılmalıdır. Veriler bir ThingSpeak kanalında olduğunda, onu analiz edebilir ve görselleştirebiliriz, bu da yeni verileri hesaplayabilmemizi veya sosyal medya, web servisleri ve diğer cihazlarla etkileşime geçebilmenize olanak sağlar (<https://thingspeak.com/>).

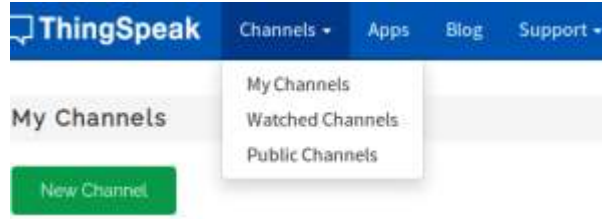
Nesnelerin interneti uygulamalarının artması, uygulamalardan elde edilen verilerin kaydedileceği veri kayıt merkezi uygulamaları ihtiyaçlarını da doğurmaktadır. Bu verilerin yerelde tutulmasına kimi zaman olanak sağlamamaktadır. Kimi zamanda veriye erişim ve inceleme gereksinimlerinden dolayı bir uygun ortam oluşturmamaktadır. Yani nesnelerin interneti kavramı büyük veri ve bulut bilişim ağına ve uygun bir ortama gereksinim duyulmaktadır.

Bu gereksinimin farkında olan birçok şirket kullanıcılara ücretsiz ve ücretli olarak seçenekler sunmaktadır. Bunlardan bir tanesi de arkasında Mathworks'un olduğu ThingSpeak'tir. Thingspeak'i diğerlerinden farklı kılan yanı Matlab ile entegrasyonudur. ThingSpeak gerek matlabteki yapılmış olduğumuz hesaplamaları, grafikleri ve çizimleri gerekse Matlab harici Arduino gibi ve diğer mikro denetleyici programlarla elde ettiğimiz verileri internette yayınlamak ve bu verileri saklamak için ideal bir ortam oluşturmaktadır.

Örnek olarak ilinizin içerisinde ayırdığınız bölgelerdeki hava sıcaklık, nem, rüzgâr ile hava kirliliği arasında bir ilişki konusunda bir araştırma yapıyorsanız sahadaki internet erişimi olan sensörlerinizin verilerini elde etmiş olduğu verileri ThingSpeak'e gönderebilmenize veya saklayabilmenize olanak sağlamaktadır. Bunun için https://thingspeak.com/users/sign_up adresinde çok kısa sürede kendinize bir hesap oluşturmanız gerekmektedir.

5.1 ThingSpeak Yapısı ve Kullanımı

Hesabınızı oluşturduktan sonra My Channels linkine girerek New Channel diyerek yeni bir channel – yani bir veri tabanı – tablo oluşturmanız gerekmektedir. Sonra burada kullanmanız gereken alanları belirlemeniz gerekmektedir. Bu durumlara ait resimler aşağıda mevcuttur (<https://thingspeak.com/>).



Şekil 5.1. ThingSpeak My Channel oluşturma ekranı.

The image shows the 'New Channel' form in the ThingSpeak web interface. The form is titled 'New Channel' and contains the following fields and options:

- Name:** A text input field.
- Description:** A larger text area for channel description.
- Field 1 to Field 8:** Eight input fields for data, each with a corresponding checkbox to the right. Field 1's checkbox is checked (indicated by a red checkmark icon).
- Metadata:** A text area for specifying the data format (e.g., JSON, XML, CSV).
- Tags:** A text area for adding tags to the channel.
- Latitude:** A text input field.
- Longitude:** A text input field.
- Elevation:** A text input field.
- Make Public?:** A checkbox to make the channel public.
- URL:** A text input field for the channel's URL.
- Video ID:** A text input field for a video ID.
- Save Channel:** A green button at the bottom to save the new channel.

Şekil 5.2. ThingSpeak New Channel oluşturma ekranı.

Karşımıza çıkan ekran yandaki şekilde olacaktır.

Burada yer alan alanlara ait kısımlar ve açıklamaları aşağıda verilmiştir.

- **Channel Name:** Kanal için kullanılacak isimdir.
- **Description:** Eğer istenirse kanal için açıklamadır.
- **Field#:** Kanalda kaydedilebilecek 8 adet veri alanı mevcuttur. Bu veri alanı için isim. Kullanabilmek için yandaki kutucuk seçilerek aktif edilmelidir. Bizim çalışmamızda 4 adet veri alanı açılmaktadır.
- **Metadata:** Kanal verisinin belli bir türde olması gerekmektedir. Bunlar JSON, XML, veya CSV olup olmadığı bilgisi gibi.

- Tags: Kanalı tanımlamak için kullanılacak anahtar kelimelerdir.
- Latitude: Kullanılacak konum sensörü ile kanal için enlem bilgisini ifade etmektedir.
- Longitude: Kullanılacak konum sensörü ile için boylam bilgisini ifade etmektedir.
- Elevation: Kullanılacak konum sensörü ile kanal için yükseklik bilgisini açıklamaktadır.
- Make Public: Verileri herkese açık hale getirerek paylaşım açmaktadır.
- URL: Kanal hakkında bilgi içeren bir web sitesi var ise onu ifade etmektedir.
- Video ID: Kanal hakkında bilgi içeren ve proje hakkında çekilen bir youtube veya vimeo videosu var ise bu videoları yayınlamak için gerekli olan bilgisini anlatmaktadır.

Her kanal içine 8 tane anlık veri kaydedebilmemize olanak sağlamaktadır. Bu formu doldurarak kaydettiğimizde Thing Speak bize bir kanal oluşturacaktır.

Oluşturulan her kanala ait bir ID ye sahip olacaktır. Verilere erişmek için bu ID (Kullanıcı adı) bilgisi gerekmektedir. Kanalı oluşturduktan sonra My Channels görünümü aşağıdaki şekilde olacaktır.

isim	düzenlendi	Güncellenmiş
JENERATÖR Özel Genel Ayarlar Paylaşma API Anahtarları Veri içeri Aktarma / Dışa Aktarma	2018/07/10	2018-07-16 15:29

Şekil 5.3. ThingSpeak kanal oluşturulduktan sonraki görüntüsü.

JENERATOR

Kanal ID: 535981
Yazar: seremet
Erişim: Özel

Özel Görünüm Halk görüşü Kanal Ayarları Paylaşım API Anahtarları Veri Alma / Verme

Görselleştirme ekle

Widget ekle

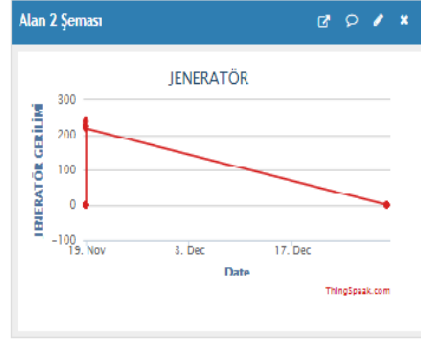
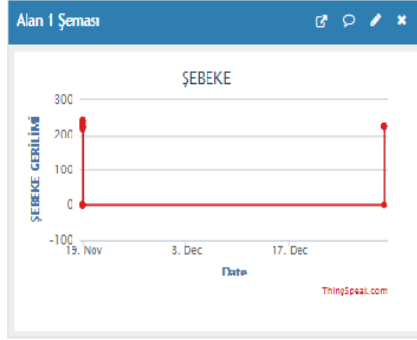
Son verileri dışa aktar

MATLAB Analizi

MATLAB Görselleştirme

Kanal İstatistikleri

Oluşturuldu: 10 ay önce
Son giriş: 5 ay önce
Görüşler: 1635



Şekil 5.4. ThingSpeak özel görünüm ekranı.

Özel görünümde kanalımıza ait bilgiler, kendimizin görebileceği bilgiler ve uzaktan bilgilerini görmek istediğimiz nesnelere ait verileri yukarıda anlatılarak oluşturduğumuz alanlara verileri aktarabilmemize olanak sağlamaktadır.

Halk görüşü kısmında ise bize ait verileri dışarıya paylaşım açabildiğimiz kısmı oluşturmaktadır. Burada herkesin görebileceği verilerin olduğu kısmı oluşturmaktadır (<http://www.iot.gen.tr/2015/12/13/nesnelerin-interneti-icin-veri-kayit-alanlari-thingspeak-temel-kullanim/>).

Kanal ayarları ise kanalı oluştururken kullandığımız alanları yönetebileceğimiz kısmı oluşturmaktadır.

API anahtarları altında ise ThingSpeak'e uzaktan göndereceğimiz verileri yazmış olduğumuz programda belirtmemiz gereken veri yazma ve okuma işlemi sırasında kullanacağımız anahtar kodlar yer almaktadır. Bu anahtar kodları birbirinden farklı olduğu için uygulamalar sırasında buna dikkat etmek gerekmektedir. Diğer sayfadaki resimde bizim çalışmamıza ait API anahtar kodu yer

almaktadır (http://www.iot.gen.tr/2015/12/13/nesnelerin-interneti-icin-veri-kayit-
alanlari-thingspeak-temel-kullanim/).

[Özel Görünüm](#) [Halk görüşü](#) [Kanal Ayarları](#) [Paylaşım](#) [API Anahtarları](#) [Veri İçer Aktarma / Dışa Aktarma](#)

API Anahtarını Yaz

anahtar

[Yeni Yazma API'si Anahtar Oluştur](#)

API Anahtarlarını Okuyun

anahtar

Not

[Notu Kaydet](#) [API Anahtarını Sil](#)

yardım et

API anahtarları, bir kanala veri yazmanızı veya özel bir kanaldan veri okumayı sağlar. Yeni bir kanal oluşturduğunuzda API anahtarları otomatik olarak oluşturulur.

API Anahtar Ayarları

- **API Anahtar Yaz** : Bir kanala veri yazmak için bu tuşa basın. Anahtarınızın ele geçirildiğini düşünüyorsanız, **Yeni Yazma API Anahtarı Oluştur'u** tıklayın .
- **API Anahtarlarını Okuyun** : Diğer kişilerin özel kanal yayınlarınızı ve listelerinizi görüntülemesine izin vermek için bu anahtarı kullanın. Kanal için ek bir okuma anahtarı oluşturmak için **Yeni Okuma API Anahtarı Oluştur'u** tıklayın .
- **Not** : Kanal okuma tuşları hakkında bilgi girmek için bu alanı kullanın. Örneğin, kullanıcıların kanalınıza erişimi olan kullanıcıları takip etmek için notlar ekleyin.

API İstekleri

[Kanal Beslemesini Güncelle](#)

```
GET https://api.thingspeak.com/update?api_key= AZEBL98HQFNIRH7W & al
```

[Kanal Besleme Alın](#)

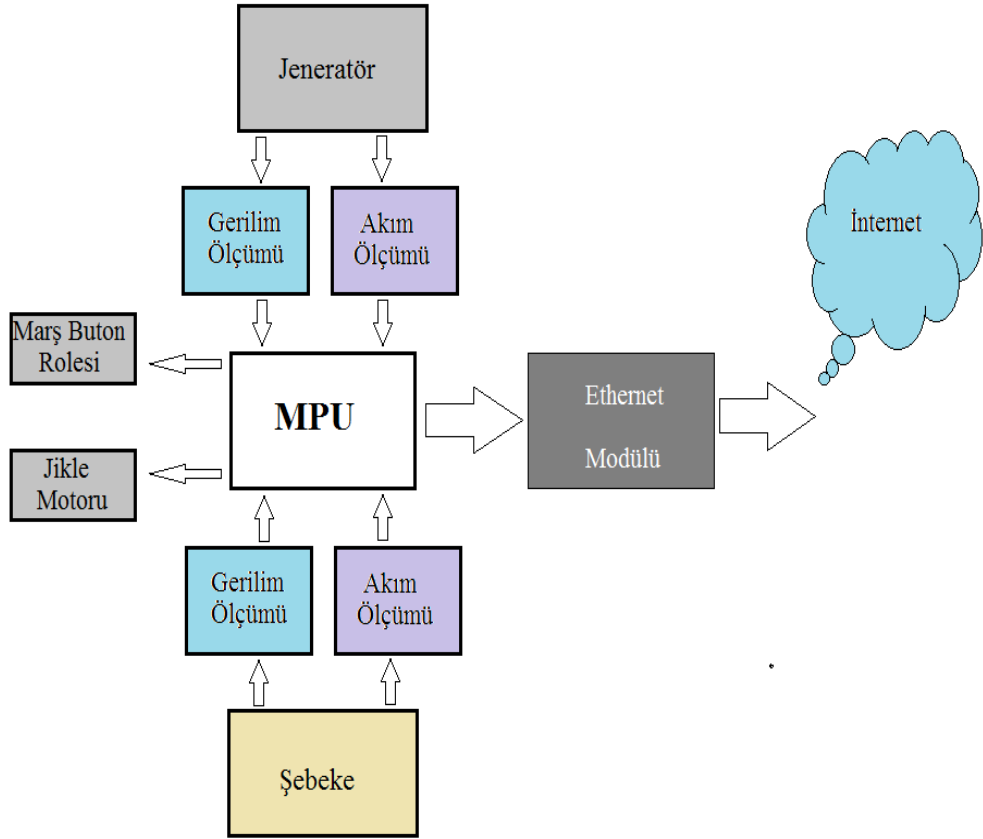
```
https://api.thingspeak.com/channels/ 535981 /feeds.json?api_key= 537
```

Şekil 5.5. ThingSpeak API anahtar kodu.

Data Import/Export altında ise kanal altında veri eklemenize ve veri almanıza yarayan taslak mevcuttur.. Export ile CSV formatında alınan dosya yapısı aşağıdaki şekildedir. Ayrıca oluşturulan alanlara gelen kaydedilen verileri Excel formatında bilgisayarınıza indirebilirsiniz. Bu bilgiler gelen verilerin zaman bilgisini oluşturmaktadır (http://www.iot.gen.tr/2015/12/13/nesnelerin-interneti-icin-veri-kayit-
alanlari-thingspeak-temel-kullanim/).

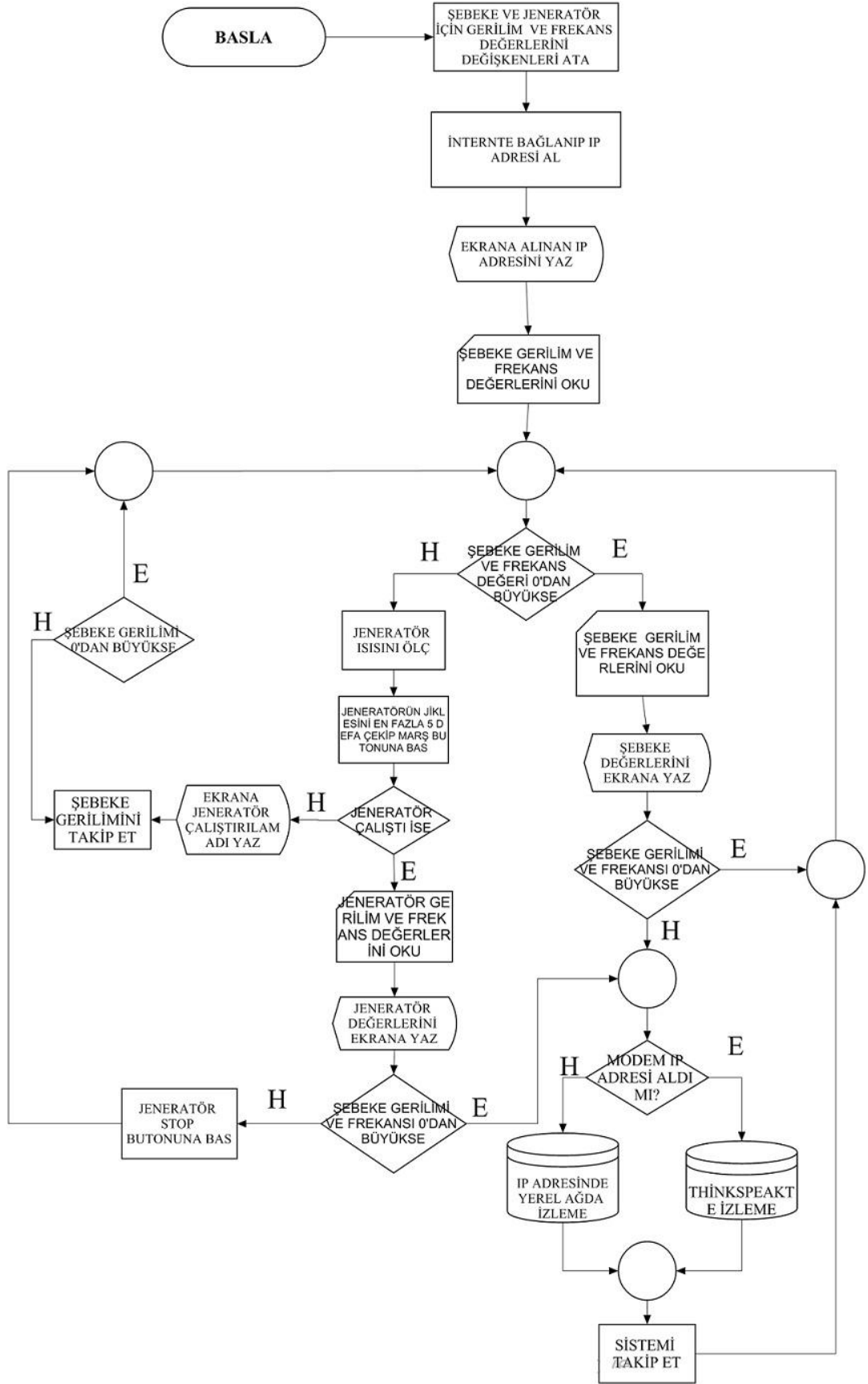
6. JENERATÖR ÇEVİRİMİ VE PROGRAMIN AÇIKLANMASI

Daha önceki bölümlerde manuel jeneratörün nasıl otomatik bir jeneratöre çevrilebileceğini aşama aşama anlatmıştık. Yazmış olduğumuz yazılımı akış diyagramı ile birlikte ve sistemin nasıl kurulduğu sistem bloğu aşağıdaki resimlerde gösterilmektedir. Yazılan yazılımı adım adım açıklayarak devrenin nasıl çalıştığı anlatılmıştır. Yazılımın tamamını ek kısmında verilmektedir.



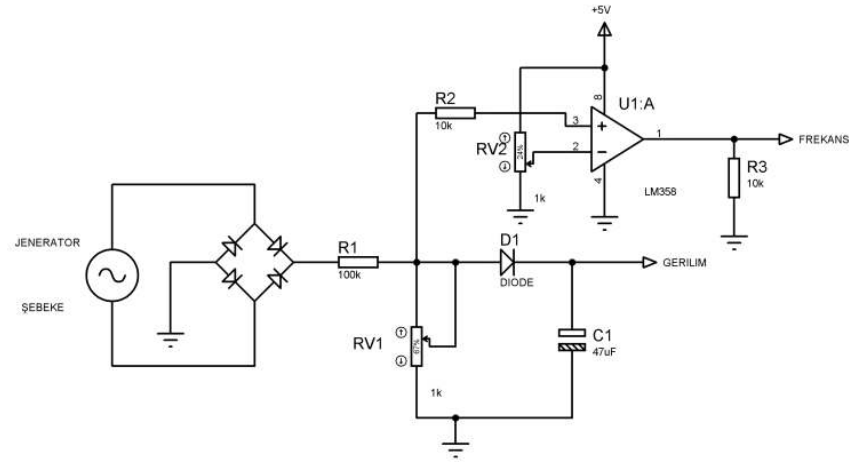
Şekil 6.1. Sistem bloğu.

Sistem için kullanılan sistem bloğu şekildeki gibidir. Bir tane mikro işlemci üzerinde şebekenin gerilim ve frekans değerlerini her daim takip edecek şebeke kesildiğinde jeneratörün jikle ve marş butonu aktif ederek jeneratörün ve şebekenin gerilim ve frekans değerlerini okuyarak internete aktarmaktadır.

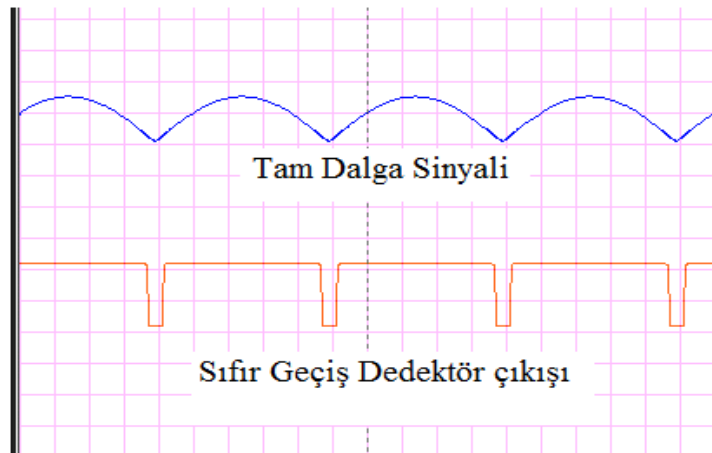


Şekil 6.2. Yazılımın akış diyagramı.

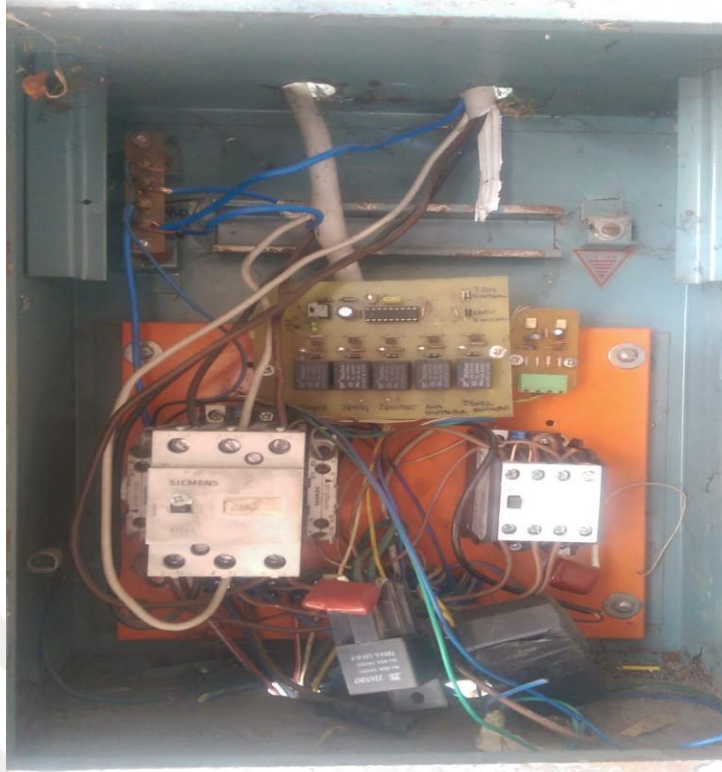
Şebeke ve jeneratör gerilimleri ile frekanslarının ölçülmesi için Şekil 6.3’de görüldüğü gibi bir devre yapısı tasarlanmıştır. Gerilim ölçümü için köprü tip doğrultmaç ile AC akım doğrultulmuş, dirençler ile düşürülüp kondansatör ile tam DC değere dönüştürülerek gerilim ölçümü yapılmıştır. Frekans ölçümü içinse köprü çıkışı kondansatörle tam DC değere çevrilmeden opamli karşılaştırıcıya verilmiştir. Burada DC değere dönüşmeden tam dalga olarak almak için D1 diyotu kullanılmıştır. Opampli devre ile sıfır geçiş dedektörü yapılmış ve AC sinyalin sıfır geçiş noktaları belirli zaman boyunca saydılarak frekans ölçümü gerçekleştirilmiştir. Şekil 6.4’de ise tam dalga çıkış ve sıfır geçiş dedektör çıkış sinyalleri gösterilmiştir.



Şekil 6.3. Gerilim-Frekans ölçüm devresi .



Şekil 6.4. Gerilim-Frekans ölçüm sinyalleri.



Şekil 6.5. Kurmuş olduğumuz devrenin bir görüntüsü.



Şekil 6.6. Kurmuş olduğumuz devre sisteme bağlıken görüntüsü.

Sistemde kullanılan devre elemanları ;

1. Arduino Mega 2560 R3;
2. LCD 16*2 ekran (yazdırdığımız değerleri ve deneme sayısını görmek için)
3. ENC2860J Ethernet modülü (internete bağlanabilmek için)
4. Röle (Elektrik gittiğinde marş butonuna basabilmek için)
5. Servo Motor (Jikleyi çektirebilmek için)
6. Diyot, direnç ve kondansatörler;

Sistemin çalışması için gerekli kütüphaneler Arduinoya eklenmektedir. Arduino ya kütüphanelerin nasıl yüklendiği daha önceki konularda anlatılmıştır.



Şekil 6.7. Sistemin kütüphane ve değişkenlerin akış diyagramı.

Jeneratör, şebekeyi her daim izlemesi gerekmektedir. Bu yüzden yazdığımız programda bir şebeke giriş voltajı ve giriş frekansı için ve jeneratör için de bir çıkış voltajı şebeke frekansı için bir isim tanımlanmalıdır.

Şebeke gerilimini şebeke frekansını ve jeneratör gerilimini ve frekansını izleyebilmek için arduino'nun analog girişlerinden iki tanesini kullanılmaktadır.



Şekil 6.9. Sistemin şebeke üzerindeki okumuş olduğu değeri yazdırması.

Yukarıda açıklanmış olan akış diyagramında; giriş gerilimi Arduinonun analog girişinin 1 nolu pininden, jeneratör gerilim değeri Arduinonun analog girişinin 0 nolu piminden okunmaktadır.

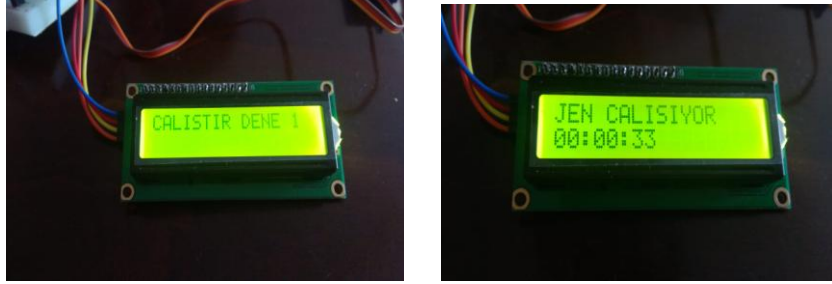
Frekans değerini ise şebekenin frekansı Arduinonun 6 nolu pininden, jeneratör frekansı ise Arduinonun 5 nolu pininden okunmaktadır. Burada geçen 1 saniyede gelen gerilimin pozitif olduğu ve negatif olduğu zamanlardaki frekans değeri ayrı ayrı bulunup toplanmaktadır. Bu sayede şebeke ve jeneratörün frekans bulunabilmektedir.

Şebeke gerilimini gittiğini sistemin anlayabilmesi için tanımlamış olduğumuz giriş voltajı değeri belli bir değerin altına indiğinde şebekenin ani gidip gelmelerinde jeneratörü koruyabilmek için sistem 5 saniye beklemektedir. Şebeke gittiğinde LCD ekranda “SBK YOK” ibaresi yer alır .Soğuk havalarda jeneratör çalıştırabilmesi için jikle çekilmesi gerekir. Jeneratörün çıkıştaki ısısını ölçecek buna göre karbüratördeki jikle telini ona göre çekecektir. Bu sayede jikle her daim kullanılmamış olacak gerektiği yerde kullanılacaktır. Bunun için jikleye çekebilmesi için küçük güçlü de olsa bir servo motor kullanılmaktadır.

Şebeke gittiğinde deneme sayısı LCD ekranda başlamaktadır. her deneme sayısı LCD ekranda “CALISTIR DENE” olarak gösterilir.Bu deneme sayısı ilerledikçe servo motor jikle çekmek için 180 derece döndürülür. Jeneratörü çalıştırmak için kullandığımız röleye Arduinonun digital 9 nolu piminden sinyal gönderilir, bu röleye bağlanmış olduğumuz marş kablosu ile jeneratörün aküsünden aldığı (+) 12 V ile marş dinamosunu 3 saniye aktif ederek jeneratörün çıkış vermesi

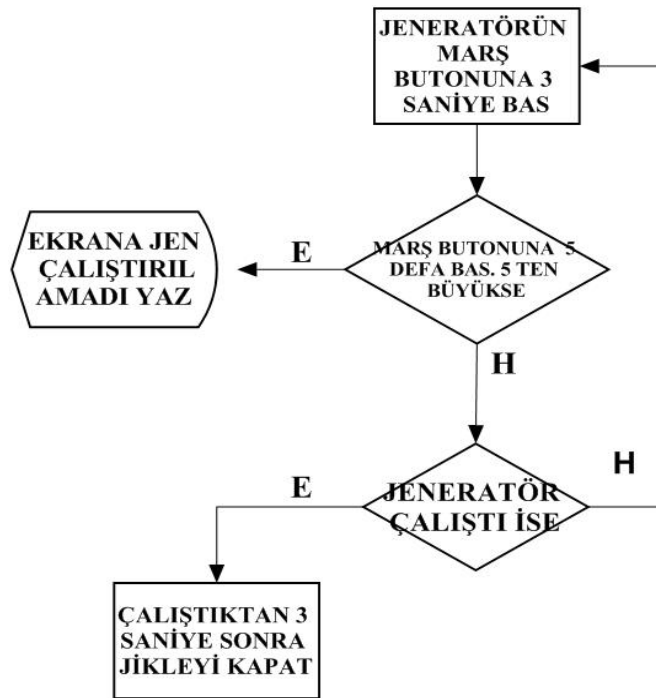
sağlanır. Eğer jeneratör çıkış vermezse 3 saniye bekleyip tekrar röleye sinyal göndererek marş kablosu ile aküden alınan (+) 12 V çektilerle jeneratörün çıkış vermesi beklenmelidir. Bu işlem jeneratör çalışması için toplam 5 defa denenmelidir. Eğer çıkış verirse servo motor geri eski haline dönecektir.

Deneme sayısı beşten küçükken jeneratör çıkış verirse LCD ekranda “JEN CALISIYOR” ibaresi yer alacaktır.



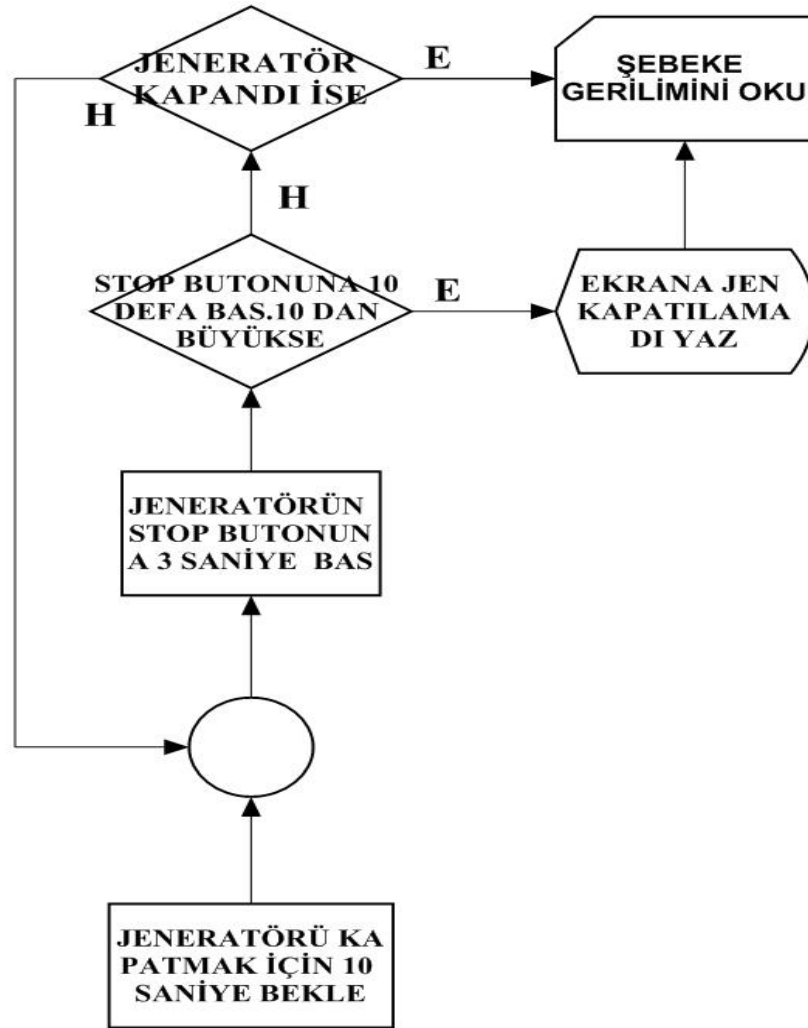
Şekil 6.10. Jeneratörü çalıştırmayı denerken ve çalıştığına dair Lcd yazısı.

Eğer deneme sayısı beşinci denemede de jeneratör çıkış vermezse LCD ekranda jeneratör çalıştırılmadığına dair ” JEN CALISTIRILAMADI” ibaresi çıkacaktır.



Şekil 6.11. Jeneratörün çalıştırılmadığına dair akış diyagramı.

Şebeke, geri geldiğinde jeneratörü kapatmadan önce 10 saniye kadar şebekenin ani gelip gitmesine karşı jeneratörü korumak için beklenmelidir. Arduionun digital girişinden 10 nolu pinine bağlı röle jeneratörün aküsünün (-) ucu ile jeneratörün stop kablosu 3 saniye kadar çektilirmeye çalışılmaktadır. Jeneratörün gövdesi (-) ucu gördüğü anda jeneratör kapanacaktır. Deneme sayısı birden başlayarak ona kadar sayacaktır. Bu işlemler devam ederken LCD ekranda “JEN KAPATILIYOR” ibaresi yer alır, ancak deneme sayısı on’u geçtiği anda jeneratörü kapatılamazsa LCD ekranda “JEN KAPATILAMADI” ibaresi yer alacaktır.



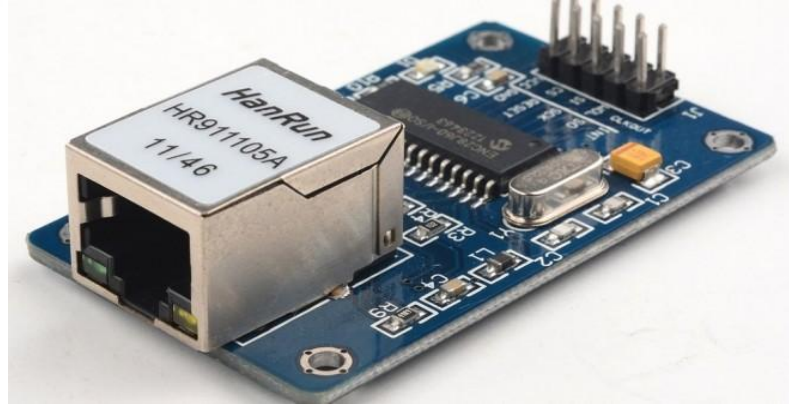
Şekil 6.12. Jeneratörün kapatılamadığına dair akış diyagramı.

7. JENERATÖRÜ UZAKTAN İZLEME

Bu kısımda şebeke gerilimini, frekansını ve jeneratör çalışırken üzerindeki gerilim ve frekansı iki farklı tipte uzaktan izlenebilir. Bunlardan ilki modem bilgisayarın alacağı IP ile yerel ağ bağlantısını kullanarak, ikincisi ise daha önceki bölümde anlatmış olduğumuz Thinkspeak IoT ile uzaktan izlenmesidir.

7.1 Modemden Alınan IP adresi ve Yerel Ağ Bağlantısı Kullanılarak

Bunun için ENC2860J Ethernet modülü kullanılır. Modül 5 Volt(V) ile çalışır. Bu modül internet modeminden almış olduğu IP (İnternet Protocol) adresi ile bağlanır. Modülün 10 tane bağlantı pini bulunmaktadır. Ancak bunlardan 7 tanesi kullanılarak modül çalıştırılabilmektedir. Bu bağlantı pinleri aşağıdaki çizelgede verilmiştir.

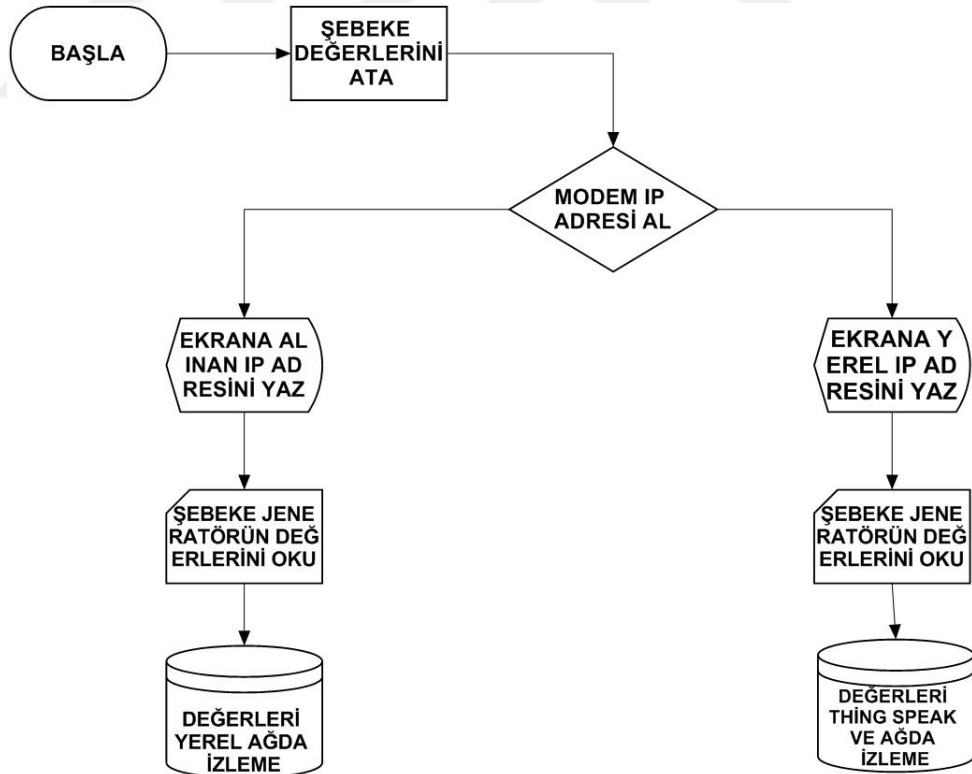


Şekil 7.1. Enc2860J ethernet modülü.

Çizelge 7.1. ENC2860J modül bağlantı pin numaraları.

PİN ADI	PİN NUMARASI
VCC	3,3 V veya 5V
GND	GND
CS	D53
SI	D51
SO	D50
SCK	D52
INT	D2

Sistem devreye girebilmesi için önce kullanılabilecek olan kütüphaneler daha önceki bölümde anlatılmıştı. ENC2860J Ethernet modülüne bağlanan LAN (Lokal Area Network) kablosu ile modül modemden IP alır. Burada LCD ekranda “AG BAGLANTISI BASLATILIYOR” ibaresi yer alır.



Şekil 7.2. Sistemin internet ağı akış diyagramı.



Şekil 7.3. Ağ bağlantısının başlatıldığını gösteren Lcd yazısı.

Modül modemden IP adresi alırsa almış olduğu IP adresi LCD ekranda görülür. Almış olduğu bu IP adresi bağlı olduğu bilgisayarlarda ve o ağın bağlı bulunduğu bütün bilgisayar, tablet ve telefonlarda alınan IP adresi girilerek jeneratörün ve şebekenin gerilimlerini, frekansını ve eğer jeneratör devredeyse jeneratörün ne kadar süre açık olduğu, durumu kontrol edilebilmektedir.

Eğer modül IP adresi alamazsa LCD ekranda “AG BAGLANTISI YAPILAMADI” ibaresi yer alacaktır. Yukarıda belirtmiş olduğumuz sistemde tanımlı olan statik IP adresini alacaktır. Bu sayede jeneratör ve şebeke üzerinde bilinmesi gereken değerler gerilim ve frekans değerleri uzaktan bilgisayar, tablet ve telefon aracılığı ile izlenebilecektir.



Şekil 7.4. Modülün modemden aldığı IP adresi ve elde edilen veriler.

7.2 ThingSpeak IoT ile Uzaktan İzleme

Thingspeak’e veri gönderebilmek için kayıt yapılmasını zorunlu kılmaktadır. Alınan kullanıcı adı ve şifre ile herkesin ayrı bir hesabı bulunur. Burada daha önce anlatılan Thingspeak yapısını göz önünde bulundurularak sistemin ENC2860J

modülü IP adresi alınması beklenir. Modül IP adresi alıp sistem internete veri gönderir hale gelir, bu verileri gönderebilmek için internet başlatılmalıdır. Başlatılan internette Thingspeak'e bağlanmalıdır. Thingspeak te açılan kanal adı ve API numarası sisteme tanıtılmalıdır.

Bu kanal da alınan ve sisteme tanıtılan API numarası :

X-THINGSPEAKAPIKEY: AZEBL98HQQFNIRH7W\ın dir.

Açılan bu kanalda alanlar belirlenmeli ve bu alanlara sistemden gelecek verilerin hangi alana ait olduğu belirlenmelidir. Bu kanal da sisteme ait 4 adet alan belirlenmiş ve bu alanlar;

- Şebekenin Gerilim
- Şebeke Frekansı
- Jeneratör Gerilimi
- Jeneratör Frekansı olarak belirlenmiştir.

Bundan sonra okuduğu her veriyi belli aralıklarla Thingspeak'e gönderir, gönderiler veriler kaydedilir ve belirtilen alanın hangi veriyi hangi saatte hangi dakikada aldığı belirlenebilir.

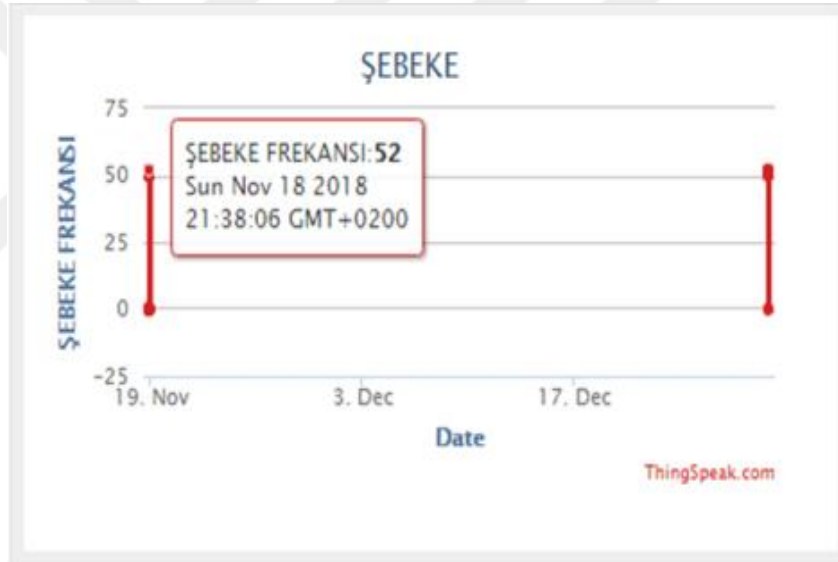
Diğer sayfada daha önceden gönderilen veriler bulunmaktadır.



Şekil 7.5. Şebeke gerilimi grafiği.



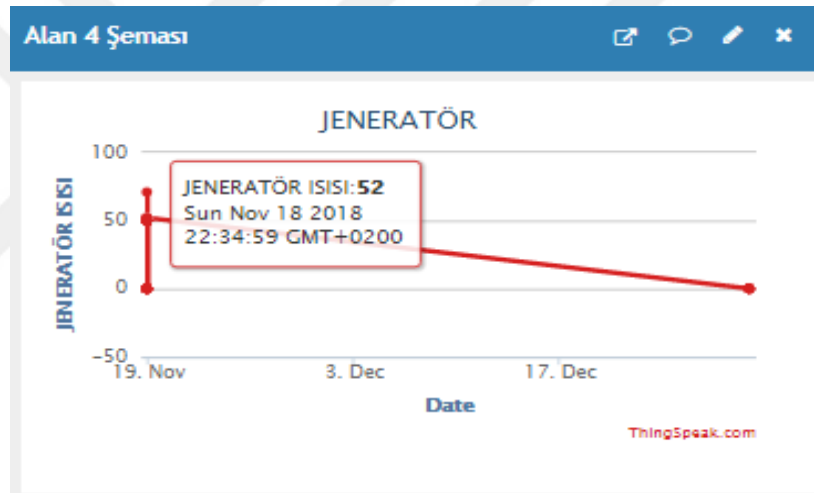
Şekil 7.6. Jeneratör gerilimi grafiği.



Şekil 7.7. Şebeke frekansı grafiği.



Şekil 7.8. Jeneratör frekansı grafiği.



Şekil 7.9. Jeneratör ısı grafiği.

8. SONUÇ VE ÖNERİLER

Günümüzde teknoloji ve endüstrinin gelişmesi ile birlikte elektrik enerjisine duyulan ihtiyaç her geçen gün artmaktadır. Sanayinin gelişmesi ve kullanılan cihazların maliyeti ve makinelerin güçlerin çok olması ve herhangi bir elektrik kesintisinde işletmenin durmaması için jeneratörlere ihtiyaç giderek artmaktadır. Piyasada elektrik kesintisinde otomatik devreye giren jeneratörler bulunmaktadır. Ancak bu jeneratörlerin pahalı olması ve periyodik bakımlarının düzenli olarak yapılması gerektiğinden bu maliyetleri düşürmek için başka yöntemler geliştirilmek zorundayız.

Bu çalışmada maliyet olarak daha pahalı olan otomatik jeneratör yerine elle çalıştırılan manuel jeneratör alınmış olup çalışma şekli otomatik jeneratöre dönüştürülmüştür. Ayrıca jeneratör ve şebekedeki gerilim ve frekans değeri uzaktan izlenmesi sağlanmıştır. Bu sayede hem maliyet olarak daha ucuza proje gerçekleştirilmiş hem de uzaktan izleme yapılmıştır. Eğer istenirse Periyodik bakımlar uzaktan izlenerek gerekli müdahale yapılabilecektir.

Yapmış olduğumuz projeyi geliştirmek adına jeneratörün akü durumu, ısı kontrolü, yakıt durumu, çıkıştaki yükün distorsiyonu, çekilen akımı, çekilen güç izlenebilir. Bunun için izlenmek istenen parameterlere ait sensör kullanmak ve bu mikrodenetleyici kullanarak sensörlerden ölçtüğümüz bu verileri uzaktan izleyebiliriz.

9. KAYNAKLAR

- Abou Haidar G, Achkar R, Abou Dayya R, Salloum A, Daoud K (2014) “Remote Generator Control Using Android Application”, 2014 8th Asia Modelling Symposium IEEE Conferences, 23-25 September 2014, Taipei, Tayvan.
- Albu M, Diaz J, Thong V, Neurohr R, Federenciu D, Popa M, Calin M (2010) “Measurement and Remote Monitoring for Virtual Synchronous Generator Design”, 2010 IEEE Workshop on Applied Measurements of International Power Systems IEEE Conferences, 22-24 September 2010 Aachen, Germany.
- Altunsaçlı A ve Alacacı M (2000) Alternatif Akım Elektrik Makinaları, Kendi Yayını, İskenderun-Hatay.
- Anarche (2017) Arduino Ethernet Enc28j60 Network Ağ Bağlantısı, Droid, <http://droid.com.tr/arduino-ethernet-enc28j60-network-ag-baglantisi/>, 1 Ağustos 2018.
- Arduino, Arduino download ve kütüphaneler, <https://www.arduino.cc/>, 06 Temmuz 2018.
- Miguel A Arduino EthernetShield ENC28J60 Configuration - Parte1, Youtube, <https://www.youtube.com/watch?v=d2toibPesS4> , 26 Haziran 2018 .
- Bal G (2012) Özel Elektrik Makinaları, SeçkinYayınları, Ankara.
- Banzy M and Shiloh M (2014) Getting Started With Arduino, Maker Media Inc, Sebastopol-CA.
- Başer O (2015) Elektrobot, Arduino ileİnternete Bağlanma, <http://www.elektrobot.net/hadi-arduino-ile-internete-baglananim/>, 28 Haziran 2018.
- Bekiroğlu E ve Bayrak A (2010) “Sanal elektrik makinaları laboratuvarı: Senkron jeneratör deneyi”, Gazi Univ. Muh.Mim. Fak.Dergisi, 25: 405-413.
- Blasol K, How to connect the ENC28J60 to an Arduino, Youtube, https://www.youtube.com/watch?v=5MHisFC-_dE, 25 Haziran 2018.
- Chapman J (Akın E., Orhan A. Çeviri) (2019) Elektrik Makinalarının Temelleri, Çağlayan Yayıncılık.
- Datakom, Jeneratör Hakkında Bilgi, http://www.datakom.com.tr/upload/Files/700_TAN.pdf , 14 Haziran 2018.
- Doğan E (2017) Arduinio Ethernet Modülü İle IP Üzerinden Röle Kontrolü, <https://www.projehocam.com/arduino-ethernet-modulu-ile-ip-uzerinden-role-kontrolu/>, 16 Temmuz 2018.

- Çakıroğlu H (2012) Arduino Uyumlu Enc28j60 Ethernet Modülü Ve EtherCard Kütüphanesi Kullanımı, <http://www.hakancakiroglu.com/wordpress/?p=785>, 30 Haziran 2018.
- Çelik Ç (2016) Ethernet Shield ve Nesnelerin İnterneti, Youtube, <https://www.youtube.com/watch?v=iS1q-NHHt9g>, 20 Temmuz 2018.
- Çamoğlu D (2014) İleri Seviye Arduino, İstanbul Dikeyksen Yayıncılık, İstanbul.
- Elektrik Port, Jeneratörlerin Yapısı Ve Çalışma Prensipleri, <https://www.elektrikport.com/universite/elektrik-uretimindeki-onemli-aktor-jeneratörler/8002#ad-image-0>, 18 Mayıs 2018.
- Elektrik Rehberiniz, Jeneratör Nedir, <http://www.elektrikrehberiniz.com/jenarator/jeneratör-nedir-1567/>, 10 Mayıs 2018.
- Elektrik Stok, Jeneratör Transfer Panosu, <https://www.elektrikstok.com/jeneratör-transfer-panosu-nedir>, 21 Haziran 2018.
- Elektrokits, Como fazer um WEBSERVER ENC28J60 Arduino Ethernet How to make, Youtube, <https://www.youtube.com/watch?v=MytVPxE2jH0>, 26 Haziran 2018.
- Evans RA (1990) “A manual/automatic synchronization circuit for a 37.5 MVA steam-turbine-driven generator”, IEEE Transactions on Industry Applications, IEEE Conferences, 1081-1085, November-December 1990 IEEE Industry Applications Society.
- Geng H (2017) Internet of Things and Data Analytics Handbook, John Wiley&Sons Co., New Jersey.
- Github, Arduino Kütüphaneleri, <https://github.com/arduino-libraries/SD>, 20 Haziran 2018.
- Gray CB (1989) Electrical Machines and Drive System, Copublished in the United States with John Wiley & Sons Inc., New York.
- Güray Y (2013) Arduino İle İlk Adım Başlangıç, <https://www.gurayyildirim.com.tr/arduino-ile-ilk-adim-baslangic-902.html>, 18 Haziran 2018.
- Kumar SP, Samson VRR, Bharath Sai U, Malleswera Rao PLSD, Kedar Eswar K (2017) “Smart Health Monitoring System of Patient Through IoT”, 2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) IEEE Conferences, 551-556, 10-11 February 2017, Palladam, India.
- Kyriakides E and Heydt GT (2004) “Estimation of synchronous generator parameters using an observer for damper currents and a graphical user interface”, Electric Power Systems Research, 69: 7-16.

- Molenaar DP, Bosgra OH and Hoeijmakers MJ (2002) “Time-Domain Identification of Synchronous Generator Transfer Functions”, Journal of Solar Energy Engineering, 124: 419-426.
- Nesnelerin İnterneti, ThingSpeak, <http://www.iot.gen.tr/2015/12/13/nesnelerin-interneti-icin-veri-kayit-alanlari-thingspeak-temel-kullanim/>, 01 Temmuz 2018.
- Okeke IO, Abu AE, Eze RC, Ezeanyagu ON (2015) “Wireless Sensor Network-based Remote Diesel-level Monitoring System”, 2015 IEEE International Conference on Industrial Technology IEEE Conferences, 2209-2213, 17-19 March 2015, Sevilla, Spain.
- Peng D, Zhang H, Yang L, Li H (2008) “Research of Remote Condition Monitoring System for Turbo-generator Unit Based on B/S Model”, 2008 International Situation Monitoring and Diagnosis Conference IEEE Conferences, 21-24 April 2008 Pekin, China.
- Podsiadly P, Swiercz B, Wroblewski S (2009) “Mobile Remote Control Application for Power Generators Vibration Monitoring”, 2009 MIXDES-16th International Conference Mixed Design of Integrated Circuits & Systems IEEE Conferences, 163-166, 25-27 June 2009 Lodz, Poland.
- Power Portal, Jeneratör Çalışma Prensipleri ve Özellikleri, <http://www.power.gen.tr/makaleler/icerik.asp?id=109>, 20 Mayıs 2018.
- Robotik Sistem, Arduino Mega2560 R3 özellikleri, www.robotiksistem.com, 15 Haziran 2018.
- Robotica A (2014) Módulo enc28j60, <https://www.youtube.com/watch?v=QUZhq5I0uYQ>, 12 Haziran 2018.
- Scwhartz M (2014) Arduino Networking, Packt Publishing, Sellschopp FS and Arjona LMA “A tool for extracting synchronous machines parameters from the dc flux decay test”, Computers and Electrical Engineering, 31: 56-68. Birmingham-England.
- Singh D, Tripathi G, Jara A (2014) “A survey of Internet-of-Things: Future Vision, Architecture Challenges and Services”, 2014 IEEE Internet of Things World Forum (WF-IoT) IEEE Conferences, 287-292, 6-8 March 2014, Seul South Korea.
- Şerifoğlu AN (2007) Elektrik Makinaları Senkron Makinalar ve Doğru Akım Makinaları, Nobel Yayın Dağıtım, 2. Cilt, Ankara.
- Taşdemir C (2013) Arduino, Dikeyksen Yayıncılık, Ümraniye-İstanbul.
- Taşit.com, Jikle Nedir, <https://www.tasit.com/araba-sozlugu/jikle-2-nedir>, 21 Ocak 2019.

Teknoloji Portalı Teknoloji.net, Jeneratörlerin Çalışma Prensibi, <https://teknoloji.net/jeneratörlerin-yapisi-ve-calisma-prensibi/amp/>, 10 Ekim 2018.

ThingSpeak, <https://thingspeak.com/>, 02 Temmuz 2018.

ThingSpeak, ThingSpeak Kanalım, <https://thingspeak.com/channels>, 05 Temmuz 2018.

Turkcell Geleceği Yazarlar, Arduino Millis Fonksiyonu, <https://gelecegiyazarlar.turkcell.com.tr/konu/arduino/egitim/arduino-201/millis-fonksiyonu-nedir-ve-nerelerde-kullanilir>, 05 Temmuz 2018.

Umans A (2012) Elektrik Makinaları, 6. Baskı, Palme Yayıncılık.

Upadhyay KG (2008) Desing of Electrical Machines, New Age International Limited Publishers, New Delhi.

Voltimum, Transfer Panosu Nedir Ve Nasıl Çalışır, <https://www.voltimum.com.tr/haberler/transfer-panosu-nedir-ve-nasil-calisir>, 20 Haziran 2018.



EKLER

10. EKLER

EK A

Kodlar Arduino IDE de yazılmıştır.

```
#include <UIPEthernet.h>
```

```
#include <utility/logging.h>
```

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

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

```
#include <Servo.h>
```

```
#include <SPI.h>
```

```
LiquidCrystal_I2C lcd(0x3F, 16, 2);
```

Buraya kadar olan kısımda kullanılacak kütüphaneler mevcuttur.

```
EthernetServer server(80);
```

Burada Ethernet portunun 80 olduğu belirtilmiştir.

```
const int updateThingSpeakInterval = 16 * 1000;    // Time interval in  
milliseconds to update ThingSpeak (number of seconds * 1000 = interval)
```

```
// Variable Setup
```

```
long lastConnectionTime = 0;
```

```
boolean lastConnected = false;
```

```
int failedCounter = 0;
```

```
float value = 0;
```

```
byte deneme = 1;
```

```
byte Gsay = 1;
```

```
byte Csay = 0;
```

```
byte Cdeneme = 0;
```

```
byte GHZ = 0;
```

```
byte CHZ = 0;
```

```
float GV220 = 0;
```

```
float CV220 = 0;
```

İnternet kısmından buraya kadar olan kısımda gerekli olan değişkenler tanımlanmış ilk değerlerinin 0 olması sağlanmıştır.

```
Servo SrvM1;
```

Jikle için servo motor kullanılacağı için servo motor kütüphanesi tanımlanmıştır.

```
boolean AgBagli = false;
```

```
String Durum;
```

```
unsigned long SonIslemZamani=0;
```

```
unsigned long SonUpdateZamani=0;
```

```
unsigned long JenCalismaZamani=0;
```

```
void LcdClear()
```

Burada arduinonun kendi zaman dilim

```
{
```

```
  lcd.setCursor(0,0);
```

```
  lcd.print("                    ");
```

```
  lcd.setCursor(0,1);
```

```
  lcd.print("                    ");
```

```
}
```

```
void LcdPrint(String Line1, String Line2)
```

```
{  
  
    lcd.setCursor(0,0);  
  
    lcd.print(Line1+"      ");  
  
    lcd.setCursor(0,1);  
  
    lcd.print(Line2+"      ");  
  
}
```

```
String MillisToTime (unsigned long Salise) {
```

```
    unsigned long allSeconds=Salise/1000;
```

```
    int runHours= allSeconds/3600;
```

```
    int secsRemaining=allSeconds%3600;
```

```
    int runMinutes=secsRemaining/60;
```

```
    int runSeconds=secsRemaining%60;
```

```
    char buf[21];
```

```
    sprintf(buf,"%02d:%02d:%02d",runHours,runMinutes,runSeconds);
```

```
    return buf;
```

Servo motordan buraya kadar olan kısımda arduinonun kendi zaman fonksiyonu olan mills fonksiyonundan yararlanılmıştır. Bu fonksiyon arduinoya enerji verildiği anda başlar ve geçen zamanı kaydetmektedir. Bu zaman kullanılarak bizim sistemimizin kalibrasyonu düzenlenmiştir.

Aşağıdaki kısımları daha önceki anlatılan bölümlerde değinilmiştir.

```
}
```

```

float VoltajOku(int Giris_Cikis)

{

float vout = 0.0;

float vin = 0.0;

float R1 = 100000.0;

float R2 = 2000.0;

    if (Giris_Cikis==1)

        value = analogRead(1);

    else if (Giris_Cikis==2)

        value = analogRead(0);

    vout = (value * 3.45) / 1024.0;

    vin = vout / ((R2/(R1+R2)-0.005));

    if (vin<20.0) vin=0.0;

return vin;

}

int FrekansOku(int Giris_Cikis)

{

int frekans = 0;

byte v = 0;

byte p = 0;

int i = 0;

int k = 0;

byte Saniye = 50;

```

```

if (Giris_Cikis==1)

p = digitalRead(6);

else if (Giris_Cikis==2)

p = digitalRead(5);

v = p;

for (int i=0; i <= Saniye; i++){

if (Giris_Cikis==1)

p = digitalRead(6);

else if (Giris_Cikis==2)

p = digitalRead(5);

if (p != v)

frekans += 1;

if (frekans > 1 and p == v)

{

k = i;

break;

}

delay(1);

}

for (int i=k; i <= Saniye; i++){

if (Giris_Cikis==1)

p = digitalRead(6);

else if (Giris_Cikis==2)

```

```

p = digitalRead(5);

if (p == v)

    frekans += 1;

    if (frekans > 1 and p != v)

        {

            k = i;

            break;

        }

    delay(1);

}

for (int i=k; i <= Saniye; i++)

    { delay(1); }

if (frekans >= 1 and frekans < Saniye)

    return (1000 / frekans);

else return 0;

} //int FrekansOku(int Giris_Cikis)

void printIPAddress()

{

    lcd.clear();

    lcd.setCursor(0,0);

    lcd.print("IP adresi: ");

    lcd.setCursor(0,1);

    lcd.print(Ethernet.localIP());

```



```

        delay(2000);

    }

    void AgaBaglan()

    {

        byte mac[] = {0x00, 0xAA, 0xBB, 0xCC, 0xDE, 0x02};

        IPAddress ip(192,168,10,11);    //Fall back IP address

        IPAddress myDns(192,168,10, 1);

        IPAddress gateway(192, 168, 10, 1);

        IPAddress subnet(255, 255, 255, 0);

        LcdPrint("AG BAGLANTISI","BASLATILIYOR");

        Ethernet.maintain();    // Keep looking for an IP address

        if (Ethernet.begin(mac) == 0)    // start the Ethernet connection, connect to
DHCP.

        {

            //Ethernet.begin(mac, ip);

            Ethernet.begin(mac, ip, gateway, subnet);

            //Ethernet.begin(mac, ip, myDns, gateway, subnet);

            LcdPrint("IP ADRESI","ALINAMADI");

        } else

        {

            LcdPrint("AG BAGLANTISI","YAPILDI");

            AgBagli = true;

        }

```

```

        printIPAddress();           // print your local IP address:

server.begin();

}

void WebHtmlServer() {

    EthernetClient client = server.available();

    if (client) {

        boolean currentLineIsBlank = true;

        while (client.connected()) {

            if (client.available()) {

                char c = client.read();

                if (c == '\n' && currentLineIsBlank) {

                    client.println("HTTP/1.1 200 OK");

                    client.println("Content-Type: text/html");

                    client.println("Connection: close"); // the connection will be closed after
completion of the response

                    client.println("Refresh: 12"); // refresh the page automatically every 5
sec

                    client.println();

                    client.println("<!DOCTYPE HTML>");

                    client.println("<html>");

                    client.print("Sebeke Gerilimi ve Frekansi      : "+ String(GV220,0)+"V
"+ String(GHz)+"Hz ");

```

```

        client.println("<br />");

        client.print("Jenerator Gerilimi ve Frekansi : "+ String(CV220,0)+"V "+
String(CHz)+"Hz ");

        client.println("<br />");

        if (AgBagli)

            client.print("Ag baglantisi var.");

        else

            client.print("Ag baglantisi yok.");

        client.println("<br />");

        client.print("Son Durum : "+ Durum);

        client.println("<br />");

        if (CV220 > 100) {

            client.print("Jenerator Calisma Suresi : "+ MillisToTime(millis() -
JenCalismaZamani));

            client.println("<br />");

        }

        client.println("</html>");

        client.stop();

        break;

    }

    if (c == '\n') {

```

```

        currentLineIsBlank = true;

    } else if (c != '\r') {

        currentLineIsBlank = false;

    }

}

}

delay(1);

client.stop();

}

} // WebHtmlServer

void UpdateData() {

    String field1="field1="+String(GV220);

    String field2="field2="+String(CV220);

    String field3="field3="+String(GHz);

    String field4="field4="+String(CHz);


    EthernetClient client;

    if (client.connect("api.thingspeak.com", 80))

    {

        client.print("POST /update HTTP/1.1\n");

        client.print("Host: api.thingspeak.com\n");

        client.print("Connection: close\n");

        client.print("X-THINGSPEAKAPIKEY: AZEBL98HQNIRH7W\n");

```

```
client.print("Content-Type: application/x-www-form-urlencoded\n");
```

```
client.print("Content-Length: ");
```

```
client.println(field1.length()+field2.length()+field3.length()+field4.length()+3); //
```

need to include the & seperator

```
client.println("");
```

```
client.print(field1);
```

```
client.print("&");
```

```
client.print(field2);
```

```
client.print("&");
```

```
client.print(field3);
```

```
client.print("&");
```

```
client.print(field4);
```

```
client.stop();
```

```
}
```

```
} //UpdateData
```

```
void setup() {
```

```
  lcd.begin();
```

```
  lcd.clear();
```

```
  digitalWrite(10, HIGH);
```

```
  digitalWrite(9, HIGH);
```

```
  pinMode(10, OUTPUT);
```

```

pinMode(9, OUTPUT);

SrvM1.attach(7);

SrvM1.write(0);

pinMode(0, INPUT);

pinMode(1, INPUT);

pinMode(6, INPUT);

pinMode(3, INPUT);

digitalWrite(3, HIGH);

AgaBaglan();
}

void loop() {

    if (millis() - SonIslemZamani >= 1000) {

        GV220 = VoltajOku(1);

        CV220 = VoltajOku(2);

        GHz = FrekansOku(1);

        CHz = FrekansOku(2);

    }

    if (digitalRead(3) == HIGH) {

        if (not AgBagli) AgaBaglan();

        printIPAddress();

    }

    if (millis() - SonIslemZamani >= 500)

```

```

if(GV220 >= 100 or CV220 >= 100) {

    LcdPrint("SBK " +String(GV220,0) +"V " +String(GHz) +"Hz"

    , "JEN " +String(CV220,0) +"V " +String(CHz) +"Hz");

}

if (millis() -SonIslemZamani >= 1000) {

    if (GV220 > 100) {

        if (CV220 > 100) {

            Csay += 1;

            if (Csay <= 10) {

                lcd.setCursor(13, 1);

                lcd.print(Csay);

                lcd.print("sn");

            }

            if (Csay>=10 and Cdeneme<=5) {

                Cdeneme += 1;

                Durum = "JEN KAPATILİYOR";

                LcdPrint(Durum, "Dene "+String(Cdeneme));

                digitalWrite(10, LOW);

                delay(1000);

                digitalWrite(10, HIGH);

            }

            if (Cdeneme>=5) {

                Csay = 0;

```

```

        Durum = "JEN KAPATILAMADI";

        LcdPrint(Durum, MillisToTime(millis() -JenCalismaZamani));

    }

}

deneme = 1;

Gsay = 0;

}

else {

    Gsay += 1;

    Cdeneme = 0;

    if (Gsay<=5) {

        LcdPrint("SBK yok "+String(Gsay)+"sn", "");

    } else {

        Durum = "CALISTIR DENE "+String(deneme);

        LcdPrint(Durum, "");

    }

    if (CV220>100) {

        Durum = "JEN CALISIYOR";

        LcdPrint(Durum, MillisToTime(millis() -JenCalismaZamani));

        Csay = 0;

        Gsay = 0;

        deneme = 0;

        Cdeneme = 0;

```



```

    }

}

if (CV220<100) {

    JenCalismaZamani = millis();

    Cdeneme = 0;

}

    if (CV220>100 and millis() -JenCalismaZamani < 1000)
JenCalismaZamani = millis();

    if (deneme>5) {

        Durum = "JEN CALISTIRILAMADI";

        LcdPrint("",Durum);

    }

    if(GV220 <= 100 and CV220 <= 100 and Gsay>=6 and deneme<=5)

    {

        deneme= deneme + 1;

        SrvM1.write(180);

        delay(1000);

        digitalWrite(9, LOW);

        delay(1000);

        digitalWrite(9, HIGH);

        delay(1000);

        SrvM1.write(0);

        if (CV220>100) JenCalismaZamani = millis();

```

```
}  
  
SonIslemZamani = millis();  
  
} //if (IslemSay %5 ==0) {  
  
    if (AgBagli and millis() -SonUpdateZamani >= 30000) {  
  
        UpdateData();  
  
        //WebHtmlServer();  
  
        SonUpdateZamani = millis();  
  
    }  
    WebHtmlServer();  
}
```

11.ÖZGEÇMİŞ

Adı Soyadı :İbrahim ŞEREMET

Doğum Yeri ve Tarihi :Akıncılar 19.04.1991

Lisans Üniversite : Abant İzzet Baysal Üniversitesi

Elektronik posta :yasin-seremet @hotmail.com

İletişim Adresi :Şehit Cevdet Özdemir Mahallesi
Perçem Caddesi Perçem Sokak 4/9 ÇANKAYA/ANKARA