



**YENİLENEBİLİR ENERJİ SİSTEMLERİNDE
AÇIK KAYNAK UZAKTAN İZLEME ÜNİTESİ ve MALİYETLERİN
ASGARİYE İNDİRİLMESİ
YÜKSEK LİSANS TEZİ**

Erdal KAPLAN

Danışman

Dr. Öğr. Üyesi Ahmet YÖNETKEN

ELEKTRİK MÜHENDİSLİĞİ ANABİLİM DALI

Haziran 2019

Bu tez çalışması 17.Fen.Bil.73 numaralı proje ile BAP tarafından desteklenmiştir.

AFYON KOCATEPE ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

YÜKSEK LİSANS TEZİ

**YENİLENEBİLİR ENERJİ SİSTEMLERİNDE
AÇIK KAYNAK UZAKTAN İZLEME ÜNİTESİ ve
MALİYETLERİN ASGARİYE İNDİRİLMESİ**

Erdal KAPLAN

Danışman

Dr. Öğr. Üyesi Ahmet YÖNETKEN

ELEKTRİK MÜHENDİSLİĞİ ANABİLİM DALI

Haziran 2019

TEZ ONAY SAYFASI

Erdal KAPLAN tarafından hazırlanan “**Yenilenebilir Enerji Sistemlerinde Açık Kaynak Kodlu Uzaktan İzleme Ünitesi ve Maaliyetlerin Asgariye İndirilmesi**” adlı tez çalışması lisansüstü eğitim ve öğretim yönetmeliği ile ilgili maddeleri uyarınca 21 / 06 / 2019 tarihinde aşağıdaki jüri tarafından oy birliği ile Afyon Kocatepe Üniversitesi Fen Bilimleri Enstitüsü **Elektrik Mühendisliği Anabilim Dalı’nda**

YÜKSEK LİSANS TEZİ olarak kabul edilmiştir.

Danışman : Dr. Öğr. Üy. Ahmet YÖNETKEN

İkinci Danışmanı :

Başkan : Dr. Öğr. Üy. Süreyya KOCABEY
Sağlık Bilimleri Üniversitesi Hamidiye
Sağlık Hizmetleri MYO,

Üye : Dr. Öğr. Üy. Ahmet YÖNETKEN
Afyon Kocatepe Üniversitesi Mühendislik
Fakültesi,

Üye : Dr. Öğr. Üy. Emre AKARSLAN
Afyon Kocatepe Üniversitesi Mühendislik
Fakültesi,

İmza

İmza

İmza

Afyon Kocatepe Üniversitesi

Fen Bilimleri Enstitüsü Yönetim Kurulu’nun

...../...../..... tarih ve

..... sayılı kararıyla onaylanmıştır.

Enstitü Müdürü

.....

Prof. Dr. İbrahim EROL

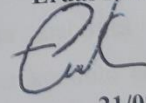
BİLİMSEL ETİK BİLDİRİM SAYFASI
Afyon Kocatepe Üniversitesi

**Fen Bilimleri Enstitüsü, 17.Fen.Bil.73 numaralı BAP tarafından desteklenen,
yazım kurallarına uygun olarak hazırladığım bu tez çalışmada;**

- Tez içindeki bütün bilgi ve belgeleri akademik kurallar çerçevesinde elde ettiğimi,
- Görsel, işitsel ve yazılı tüm bilgi ve sonuçları bilimsel ahlak kurallarına uygun olarak sunduğumu,
- Başkalarının eserlerinden yararlanılması durumunda ilgili eserlere bilimsel normlara uygun olarak atıfta bulunduğumu,
- Atıfta bulunduğum eserlerin tümünü kaynak olarak gösterdiğimi,
- Kullanılan verilerde herhangi bir tahrifat yapmadığımı,
- Ve bu tezin herhangi bir bölümünü bu üniversite veya başka bir üniversitede başka bir tez çalışması olarak sunmadığımı

beyan ederim.

Erdal KAPLAN



21/07/2019

ÖZET
Yüksek Lisans Tezi

YENİLENEBİLİR ENERJİ SİSTEMLERİNDE
AÇIK KAYNAK UZAKTAN İZLEME ÜNİTESİ VE MALİYETLERİN ASGARİYE
İNDİRİLMESİ

Erdal KAPLAN

Afyon Kocatepe Üniversitesi

Fen Bilimleri Enstitüsü

Elektrik Mühendisliği Anabilim Dalı

Danışman: Dr. Öğr. Üyesi Ahmet YÖNETKEN

Elektronik cihazların internet vasıtasıyla birbirleriyle ve kullanıcılarla tek veya çift yönlü iletişime geçmesine olanak sağlayan nesnelerin internetinin kullanımı günden güne artmaktadır. Elektronik ekipmanların doğru zamanda doğru işi yapabilmesi ve daha verimli çalışabilmesine olanak sağlayan bu veri akışı hayatı kolaylaştırmakla birlikte, daha fazla zaman, daha az iş gücü ve daha az kaynak kullanımına olanak sağlar. Büyük ölçekli enerji santrallerinde PLC üniteleri ve makine insan arabirimi olarak tanımlayabileceğimiz HMI'ler ile üretilen enerjinin izlenilmesi ve kontrolü işlemleri birebir veya uzaktan sağlanabilirken, küçük ölçekli üretim tesislerinde, örneğin; ev, bağ, bahçe gibi, üretimin uzaktan izlenmesi pahalı ve çok zahmetlidir. Bu projenin uygulanması ve geliştirilmesi çalışmaları da işte bu yüzdendir. Uyguladığımız bu projede çatısında iki bin watt solar panel bulunan bir kullanıcı saat, gün, ay hatta yıllık olarak panellerinin ne kadar aktif güç ürettiğini ve evindeki cihazların ne kadar elektrik tükettiğini görebilmektedir.

2019, x + 90 sayfa

Anahtar Kelimeler: Enerji uzaktan izleme, Yenilenebilir enerji uzaktan izleme

ABSTRACT
M.Sc Thesis

OPEN SOURCE REMOTE MONITORING UNIT ON RENEWABLE ENERGY
SYSTEMS AND REDUCING COSTS

Erdal KAPLAN

Afyon Kocatepe University

Graduate School of Natural and Applied Sciences

Department of Electrical Engineering

Supervisor: Assoc. Prof. Ahmet YÖNETKEN

The use of the internet of objects, which allows electronic devices to communicate with each other and with users via the Internet, is increasing day by day. This data flow enables electronic equipment to do the right job at the right time and work more efficiently, while making life easier, allowing for more time, less labor and less resources. In large scale power plants, monitoring and control of the energy produced by PLC units and HMIs, which can be defined as machine human interfaces, can be provided either directly or remotely, while in small scale production plants, for example; Remote monitoring of production, such as home, vineyard, garden, is expensive and very laborious. This is why the implementation and development of this project. In this project, a user with two thousand watt solar panels on the roof can see how much active power his panels produce in hours, days, months and even years and how much electricity is consumed by the devices in his house.

2019, x + 90 pages

Keywords: Remote monitoring of renewable energy systems, Remote monitoring

TEŞEKKÜR

Yenilenebilir Enerji Sistemlerinde Açık Kaynak Uzaktan İzleme Ünitesi ve Maliyetlerin Asgariye İndirilmesi araştırmasında, deneysel çalışmaların yönlendirilmesi, sonuçların değerlendirilmesi ve yazımı aşamasında yapmış olduğu büyük katkılarından dolayı tez danışmanım Sayın Dr. Öğr. Üyesi Ahmet YÖNETKEN 'e, araştırma ve programlama süresince yardımlarını esirgemeyen Sayın Bernard GIESECKE 'ye, her konuda öneri ve eleştirileriyle yardımlarını gördüğüm sevgili Tuğçe ÇETİN 'e teşekkür ederim. Bu araştırmayı 17.Fen.Bil.73 numarasıyla destekleyen Afyon Kocatepe Üniversitesi Bilimsel Araştırma Projeleri bölümüne ayrıca teşekkür ederim. Ayrıca manevi desteklerinden dolayı aileme teşekkür ederim.

Erdal KAPLAN
AFYONKARAHİSAR, 2019

İÇİNDEKİLER DİZİNİ

Sayfa

ÖZET	i
ABSTRACT	ii
TEŞEKKÜR	iii
İÇİNDEKİLER DİZİNİ.....	iv
KISALTMALAR DİZİNİ	vii
ŞEKİLLER DİZİNİ	viii
ÇİZELGELER DİZİNİ.....	x
1. GİRİŞ.....	1
2. LİTERATÜR BİLGİLERİ	2
2.1 Enerji ve Enerji Çeşitleri	2
2.2 Uzaktan İzleme Sistemlerinin Gelişimi	3
2.2.1 İletişim Yöntemleri Gelişiminin Uzaktan İzleme Sistemlerini Başlatması .	3
2.2.2 Uzaktan İzleme ve Uzaktan Kontrol İhtiyacı	5
2.2.3 Uzaktan İzleme Yöntemi	6
2.2.4 Enerjinin Uzaktan İzlenmesi	6
2.3 Nesnelerin İnterneti Nedir ?	8
2.3.1 Nesnelerin İnternetinin Endüstriyel Uygulamaları.....	10
2.4 Ac Güç Sistemini Anlamak	15
2.4.1 Omik Yükler	15
2.4.2 Kısmen Reaktif Yükler	16
2.4.3 Gerçek Güç, Reaktif Güç ve Görünen Güç	17
2.4.4 Güç Akışı Yönünün Belirlenmesi	20
2.4.5 Sonuç	20
3. MATERYAL METOD	21
3.1 Anlık Gerilim & Akım	21
3.2 Arduino'daki Gerçek Gücü Hesaplama	21
3.3 Kök-Ortalama-Kare (RMS) Gerilimi	22
3.4 Görünür Güç	22
3.5 Güç Faktörü	23
3.6 Gelişmiş Arduino Matematiği	23
3.6.1 Gerçek Güç	23
3.6.2 RMS Gerilim ve Akım Ölçümü	24

3.6.3 Görünen Güç ve Güç Faktörü.....	24
3.7 Akım Trafoları AT Sensörleri	25
3.7.1 Akım Trafoları Çeşitleri	26
3.7.2 İzolasyon, Güvenlik, Kurulum ve Kullanımı.....	27
3.7.3 AT 'nin Kablo Yönü Önemli Mi ?	29
3.7.4 AT 'nin Çalışma Mantığı	30
3.8 Yenilenebilir Enerjide Uzaktan İzleme Yöntemi	30
3.9 Solar Panel Uzaktan İzleme Donanımı.....	31
3.10 Solar Panel Elektriksel Bağlantısı	32
3.11 Arduino Yun Kartına AT Sensörlerinin Bağlanması	34
3.12 Uygun Yük Direnci Hesaplamasının Yapılması	35
3.12.1 Ölçmek İstedğimiz Mevcut Güç Aralığı Seçimi	35
3.12.2 DC BIAS Ekleme	36
3.12.3 Dirençler (R1 ve R2) İçin Uygun Değer Seçimi	37
3.12.4 Başlangıç Arduino Yazılımı	37
3.13 Yalnızca Şebeke Akımını Ölçen Arduino Enerji Monitörü Kurulumu.....	38
3.13.1 Malzeme Listesi.....	38
3.13.2 Kurulum.....	39
3.13.3 Yazılım	39
3.13.4 Ölçümlerin Görüntülenmesi	40
3.14 Düşük Akım Değerlerinde ADC Çözünürlüğünün Ölçüm Etkileri	40
3.15 Uzaktan İzleme İçin Arduino Yun Kartının Hazırlanması.....	43
3.15.1 Arduino Yun SD Kartın Hazırlanması	45
3.15.2 OpenWRT-Yun 'a Yükseltme	47
3.15.3 WIFI Üzerinden Arduino Yun 'a Bağlantı.....	47
3.15.4 Arduino Yun 'u WIFI Ağına Bağlamak	48
3.15.5 Arduino Yun Disk Alanının SSD Üzerine Genişletilmesi	50
3.15.6 Ek Yazılımların Yüklenmesi	54
3.15.7 PHP Dosyaları	55
3.16 Arduino Yazılımı.....	58
3.16.1 Ekizleme_a_Tanımlar.....	58
3.16.2 Ekizleme_b_İşlevler	60
3.16.3 Ekizleme_c_IşıkSensörü	65
3.16.4 Ekizleme_x_Ölçümleri.....	69

3.16.5 Ekizleme_y_Kurulum Tüm Kod Parçası.....	70
3.16.6 Ekizleme_z_Döngü	72
3.16.7 Zamanlayıcı ve Bekçinin Başlatılması	74
3.16.8 Işık ve AT Sensörlerinin Başlatılması	75
3.16.9 Ağ İletişimi ve Linux Dosya Sistemine Erişimin Başlatılması	76
3.16.10 Çıkış Pinlerinin Ayarlanması	76
3.16.11 Kalibrasyon Değerleri.....	76
3.17 Arduino Kartına Statik IP Atanması.....	77
3.18 İzleme Sistemine İnternet Üzerinden Erişimin Sağlanması	80
3.18.1 Modem Üzerinden NAT Ayarlamasının Yapılması.....	80
3.18.2 Statik İp Adresine Alt Alan Sunucu Eklenmesi	81
4. BULGULAR	83
4.1 Veriler.....	85
5. TARTIŞMA VE SONUÇ.....	87
6. KAYNAKLAR.....	88
6.1 İnternet Kaynakları	88
ÖZGEÇMİŞ.....	90

KISALTMALAR DİZİNİ

Kısaltmalar

OpenWRT	Ağ Tabanlı Aygıtlarda Kullanılan, Linux Tabanlı İşletim Sistemi
AT	Akım Ölçüm Trafosu
SCT-013-000	Akım Ölçüm Trafosu Modeli
AC	Alternatif Akım
A.B.D	Amerika Birleşik Devletleri
ADC	Analog Dijital Dönüştürücü
HTTPd	Apache Web Metin Aktarım Sunucusu
Yún	Arduino Yun Kartı
ARPANET	ARPA tarafından geliştirilen dünyanın ilk paket dağıtım ağı ve evrensel İnternet'in öncüsüdür.
TCP/IP	Bilgisayarlar ve cihazlar arasında iletişimi sağlamak için belirlenmiş kurallar dizisidir.
DC	Doğru Akım
GNU/GPL	Genel Kullanım Lisansı
WAN (Wide Area Network)	Geniş Alan Ağı
SD (Secure Digital Memory Card)	Hafıza Kartı Tipi
HMI	İnsan – Makine Arabirimi
İnt.Kynk	İnternet Kaynağı
IPV6 (Internet Protocol Version 6)	İnternet İp Adresi Versiyon 6
WIFI	Kablosuz Ağ İnternet Protokolü
KWH	Kilowatt Saat
RMS (Root Mean Square)	Kök Ortalama Kare
MIT	Massachusetts Teknoloji Enstitüsü
IOT (İnternet of Things)	Nesnelerin İnterneti & İnternetin Şeyleri
NCSA HTTPd	NCSA 'da Geliştirilen Bir Web Sunucusu
GD	PHP 'nin resim işlevleri
NAT (Network Adress Translation)	TCP/IP ağındaki bir bilgisayarın yönlendirme cihazı ile başka bir ağa çıkarken adres uzayındaki bir IP ile yeniden haritalandırma yaparak IP paket başlığındaki ağ adres bilgisini değiştirme sürecidir.
NSFNET	Ulusal Bilim Vakfı Ağı
Trifaze	Üç Fazlı Besleme
PHP	Yazılım Dili
V	Volt
W	Watt

ŞEKİLLER DİZİNİ

Sayfa

Şekil 2.1 Enerji çeşitleri	3
Şekil 2.2 İletişim araçları gelişiminin zaman çizelgesi	5
Şekil 2.3 Uzaktan izleme işlevleri ilişkileri.....	5
Şekil 2.4 Uzaktan izleme yöntemi.....	6
Şekil 2.5 Örnek enerji tüketimi uzaktan izleme sistemi arayüzü	7
Şekil 2.6 Gelişmekte olan teknolojilerden beklentiler	9
Şekil 2.7 Nesnelerin internetinin endüstriyel uygulamaları	10
Şekil 2.8 Nesnelerin interneti akıllı ev sistemleri.....	11
Şekil 2.9 Nesnelerin interneti enerji sistemleri	11
Şekil 2.10 Nesnelerin interneti giyilebilir sağlık uygulamaları.....	12
Şekil 2.11 Nesnelerin interneti akıllı trafik kontrolü.....	12
Şekil 2.12 Nesnelerin interneti hava tahmin istasyonu	13
Şekil 2.13 Nesnelerin interneti scada mimarisi	14
Şekil 2.14 Nesnelerin interneti akıllı ölçüm saatleri	15
Şekil 2.15 Omik yüklerde akım gerilim ilişkisi	16
Şekil 2.16 Kısmen reaktif yükte gerilim ve akım ilişkileri	16
Şekil 2.17 Güç faktörü aktif ve görünür güç ilişkisi	18
Şekil 2.18 Doğrusal olmayan yüklerin akım-gerilim eğrisi	19
Şekil 2.19 Güç üretimi esnasında gerilim, akım ve güç ilişkileri.....	20
Şekil 3.1 Anlık akım & gerilim grafiği	29
Şekil 3.2 SCT-000-013 akım trafosu sensörü	29
Şekil 3.3 AT sensörü kablo bağlantısı.....	29
Şekil 3.4 Solar uzaktan izleme devre şeması	32
Şekil 3.5 Ev tasarımı solar üretim tüketim ölçer	32
Şekil 3.6 AT sensörü ve Arduino	32
Şekil 3.7 Arduino kartı AT bağlantısı	35
Şekil 3.8 Yalnızca şebeke akımını ölçen enerji monitörü devresi	35
Şekil 3.9 Arduino AT sensörü basit ölçüm devresi.....	40
Şekil 3.10 Küçük bir ac gerilimi	41
Şekil 3.11 Gerilimin dalga boyu değişimi.....	41
Şekil 3.12 Arduino kartının yapısı	44

Şekil 3.13	Arduino kartının pin girişleri	45
Şekil 3.14	Arduino SD kartının formatlanması	46
Şekil 3.15	OpenWrt yazılımının yüklenmesi	46
Şekil 3.16	Arduino Yun WIFI bağlantısı	49
Şekil 3.17	SD kartın genişletilmesi birinci adım	49
Şekil 3.18	SD kartın genişletilmesi ikinci adım	49
Şekil 3.19	SD kartın genişletilmesi üçüncü adım	49
Şekil 3.20	SD kartın genişletilmesi dördüncü adım	52
Şekil 3.21	SD kartın genişletilmesi beşinci adım	53
Şekil 3.22	Mevcut kalibrasyon verileri görüntüleme ekranı	53
Şekil 3.23	Configure (Konfigürasyon) > Wan0 üzerinden ağ ayarlarına giriş	79
Şekil 3.24	Yun statik IP atama	79
Şekil 3.25	Port yönlendirme ayarları	80
Şekil 3.26	Alt alan adı sunucusu oluşturma	81
Şekil 3.27	Alt alan adı sunucusu üzerinden arduino ile iletişim sağlanması	82
Şekil 4.1	Solar izleme açılış sayfası	83
Şekil 4.2	Proje konteynırı üzerine monte edilen solar panel	83
Şekil 4.3	Konteynır iç kısmında yer alan 800W evirici & devre kartı	83
Şekil 4.4	16.04.2018 tarihi 14:33 – 17:41 arası ölçüm verileri	85
Şekil 4.5	19.04.2018 tarihi 14:41 – 23:59 arası ölçüm verileri	83
Şekil 4.6	Aylık & yıllık önizleme sayfası	86

ÇİZELGELER DİZİNİ

Sayfa

Çizelge 3.1 Başlangıç devresi için gerekli elektronik kart.....	39
Çizelge 3.2 Başlangıç devresi için gerekli elektronik ekipman ve sensörler	39
Çizelge 4.1 16.04.2018 tarihi 14:33 – 17:41 arası ölçüm verileri.....	Hata! Yer işareti tanımlanmamış.
Çizelge 4.2 19.04.2018 tarihi 14:41 – 23:59 arası ölçüm verileri.....	Hata! Yer işareti tanımlanmamış.



1. GİRİŞ

Bu tez çalışması yenilenebilir enerji sistemlerinin uzaktan izlenebilmesi ve maliyetlerin düşürülmesi amacıyla hazırlanmıştır.

Ülkemizin artan nüfusu ve her yıl düzenli olarak büyüyen ekonomisi ile birlikte enerji arzına olan ihtiyacı da ekonomik büyümeye paralel olarak artmaktadır. Söz konusu husus enerji talebinin uygun maliyetle ve istenildiğinde karşılanabilme gerekliliğini doğurmaktadır. Bu çerçevede enerji üretiminin ve takibinin yapılması, bunun mali ve çevresel etkilerinin minimum seviyede oluşturulması gerekmektedir. Yenilenebilir enerji konusunda ülkemizin zengin olması ve kaynakların yeterince kullanılmıyor olması neticesinde, bu husus daha fazla önem arz etmektedir.

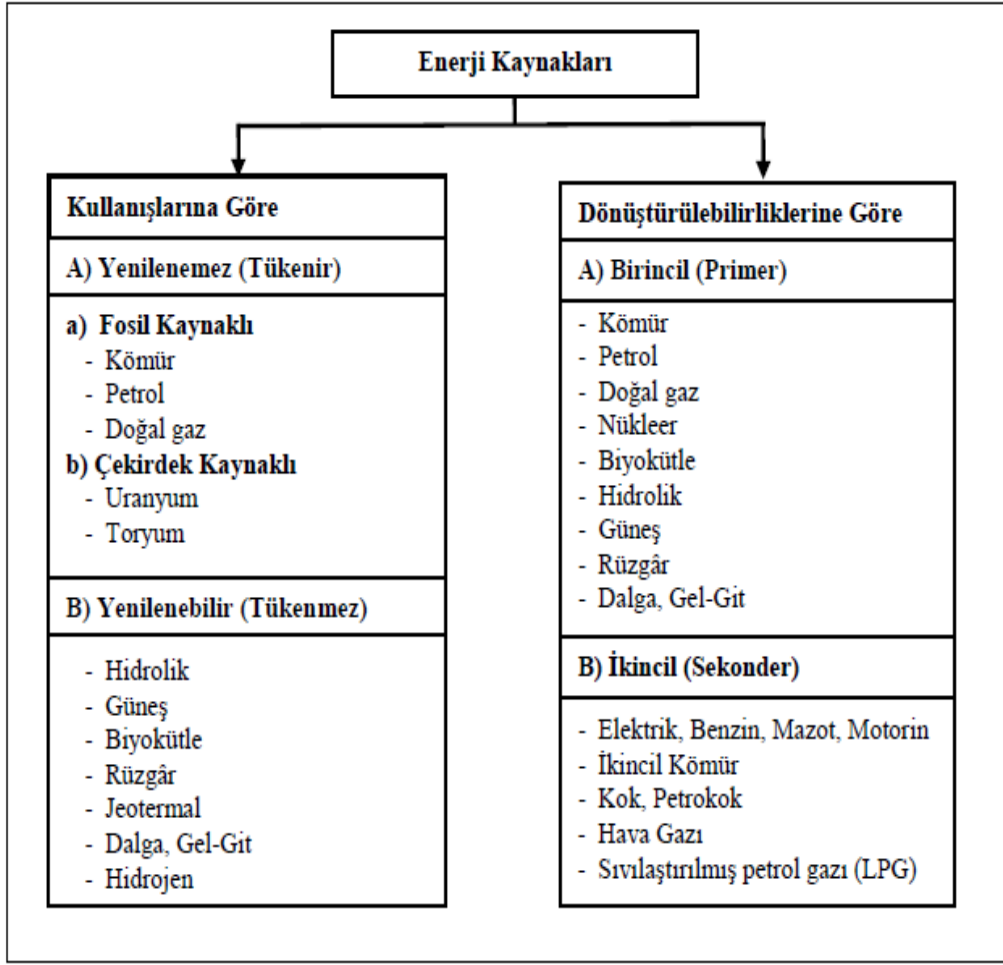
Kaynaklarımızın en doğru şekilde kullanılması prensibiyle yola çıkıldığında, mühendislik tabanlı çalışmalarla yeterli gelişme sağlanması ve bu doğrultuda ilerlenmesi, en az kaynak ve materyal kullanılarak en çok verimin elde edilmesi temel amacımız olmalıdır. En az maliyet kullanılmasıyla çıkarılması gereken anlam, uzun vadeli yatırımların hazır sistemler kullanılarak elde edilen verim ile gerçekleştirilmesi yerine başlangıç düzeyinde bile olsa yerli üretim ve öz gelişim ile sonuçlandırılması kastedilmektedir. Hazır sistemlerin uzun vadeli bakım, yedek parça, teknik destek, eğitim gibi giderleri de hesaba katıldığında ve bu sistemlerin gelişimini destekleyemeyeceğimiz de düşünülürse, alınan verimi minimum da olsa yerli üretim yaparak gelişim sağlamanın ve yenilenebilir enerji sistemleri kullanımında bu bakış açısı ile ilerlemenin en doğru yöntem olduğu kanaatindeyiz.

2. LİTERATÜR BİLGİLERİ

2.1 Enerji ve Enerji Çeşitleri

Ülkemiz gibi gelişmesini devam ettiren ülkelerde sanayileşmenin hızla artıyor olması enerjiye olan talebinde buna paralel olarak artmasına sebep olmaktadır. Enerji diğer ülkelerde olduğu gibi ülkemizin de temel kalkınma potansiyelini etkileyen ana unsurlardan bir tanesidir. Enerji tüketimi sosyal refah düzeyiyle doğrudan bağlantılıdır. Refah düzeyi arttıkça buna paralel olarak gelişmişlik seviyesi ve enerjiye olan talep de artar. Enerji; kimyasal, nükleer, mekanik (potansiyel ve kinetik), termal (ısı), jeotermal, hidrolik, güneş, rüzgâr, elektrik enerjisi gibi farklı şekillerde bulunabilir ve bazı yöntemler kullanılarak birbirlerine dönüştürülebilirler. Birbirinden farklı yöntemlerle enerji elde edilen enerji temelli kaynaklar, enerji kaynakları olarak adlandırılır. Enerji kaynakları kullanılış biçimlerine göre ise yenilenebilir ve yenilenemez enerji kaynakları olarak ikiye ayrılır. Enerji kaynakları dönüşüm biçimlerine göre ise birincil veya ikincil enerji kaynakları olarak adlandırılırlar. Yenilenemez enerji kaynaklarının kısa süre içerisinde tükeneceği öngörülmektedir, bu türdeki enerji kaynakları ise fosil temelli ve çekirdek temelli kaynaklar olmak üzere ikiye ayrılırlar.

Yenilenebilir enerji kaynakları ise kendini belirli süreçlerle tamamen yenileyen döngüsel ve uzun ömürlü kaynaklar olarak nitelendirilirler. Eğer enerji herhangi bir değişime yahut dönüşüme uğramadıysa bu tür kaynaklara primer yani birincil enerji kaynakları denilmektedir. Birincil enerji kaynaklarını sıralayacak olursak bunlar; kömür, nükleer, hidrolik, petrol biokütle, doğal gaz dalga-gelgit, güneş ve rüzgârdır. Birincil enerjinin dönüşümünün ardından seconder yani ikincil enerji elde edilmektedir. Benzin, mazot, motorin, elektrik, kok kömürü, ikincil kömür, petro kok, hava gazı, sıvılaştırılmış petrol gazı (LPG) ikincil türden enerji kaynaklarıdır. Hazırda olan her malzemenin kullanım sonrası tükeneceği bilinmektedir. Geri dönüşümü olmayan yenilenemez enerji kaynaklarının kullanım sonrası mevcut durumunu belirlemek ve yenilenebilir enerji kaynaklarının işlevselliğini artırarak daha kullanılabilir hale getirmek amacıyla tüm enerji kaynakları için bir durum tespitinin yapılması gerekmektedir (Şekil 2.1).



Şekil 2.1 Enerji çeşitleri.

2.2 Uzaktan İzleme Sistemlerinin Gelişimi

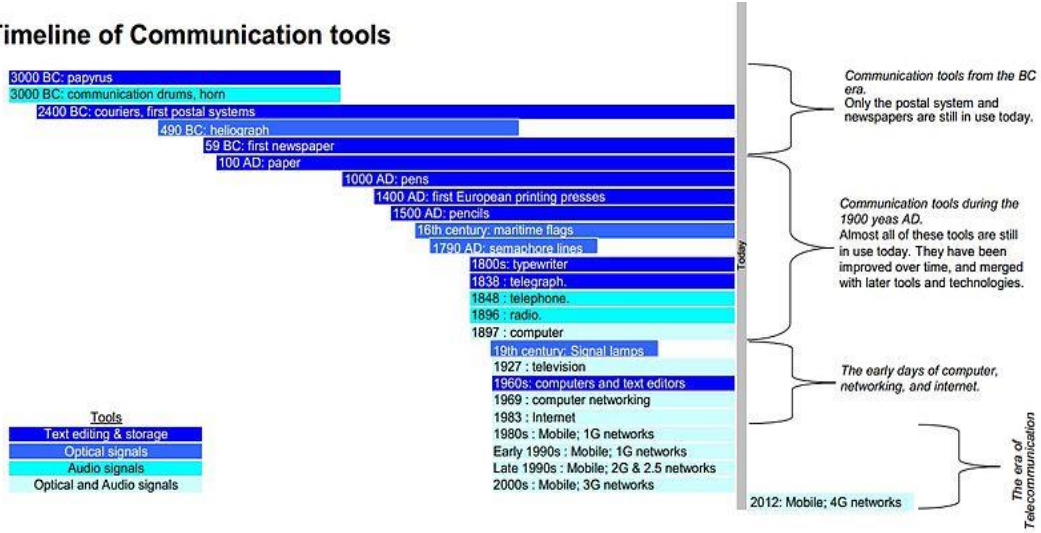
Uzaktan izleme sistemlerinin gelişimi, iletişim ve haberleşme ağlarının gelişimiyle doğrudan bağlantılıdır. Veri iletişim ağlarının küresel bağlamda yayılması ile birlikte yeni ihtiyaçlar doğrultusunda sırasıyla önce ses iletişimi, ardından bir yerden başka bir yere toplu görüntü iletimi sağlanmış bunu yeni sistemler takip etmiştir.

2.2.1 İletişim Yöntemleri Gelişiminin Uzaktan İzleme Sistemlerini Başlatması

İnsanlık mağara resimleri ile başlayan iletişim yolculuğuna, papirüsler ile devam etti, bunu ses ile uzak noktalara taşımak amacıyla önce davulları sonra ateş ve dumanı

kullanarak bir nevi sesli ve görsel iletişim sağlandı. Sesli ve görsel iletişim uzak noktalara yeterince taşınmadığı için posta teşkilatı gelişti. Bu arada yeniden ışıkla iletişim denenerek, mors alfabesi ve helyograf ile kısa mesafelerde anlık iletişim sağlandı. İnsanlık günümüzde olduğu gibi geçmişte de daha sürekli daha fazla iletişime gereksinim duyuyordu bu yüzden gelişim hiçbir zaman yeterli gelmedi, ilk gazeteler basıldı. Yazı iletişim için uygun bir yöntemdi ancak yazı yazılacak materyalin kullanımı zor ve bulunması da güçtü. Ardından kâğıdın bulunmasıyla yazının gelişimin önündeki büyük bir engel ortadan kalkarak yeni bir çağ başladı. Yazmayı kolaylaştırmak amacıyla kalem geliştirildi. Her insan için farklı konuları ayrı ayrı yazmak çok güçtü, bunun için matbaa sistemi geliştirildi. Derken, daktilolar kullanılmaya başlandı. Yazının elektrikle bir noktadan başka bir noktaya çok hızlı biçimde erişmesini sağlayan telgraf devrim niteliğindedir. Telgrafın ardından insanlık bilginin daha hızlı aktarımı konusunda, telefonları ve radyoyu icat etti. Ardından elimizdeki bilgileri işleyebilmek için bilgisayarlar, işlediğimiz bilgiyi paylaşabilmemiz için internet icat edildi. İnternetin gelişimi sürerken görüntüyü hızlı ve kolay bir şekilde aktarabilen televizyon sistemleri, televizyonun ardından cep telefonları ve günümüz. Günümüzde internet gelişimini halen devam ettirirken ışık ile verinin kablolar üzerinden çok hızlı aktarımını sağlayan fiber sistemlerin gelişimiyle internetin kullanımı günden güne artmakta, evimizde, işyerimizde, cebimizde kullandığımız cihazlar birer birer internete bağlanabilir şekilde geliştirilmektedir. Gelecekte, elektrik ve elektronik tüm sistemlerin internete bağlanacağı göz önüne alındığında, uzaktan izleme sistemlerine doğacak ihtiyacın kaçınılmaz olduğu da görülmektedir. Bu sistemlerin her sektörde ve her alanda kullanılacağı da açık bir şekilde bilinmelidir (Şekil 2.2).

Timeline of Communication tools



Şekil 2.2 İletişim araçları gelişiminin zaman çizelgesi.

2.2.2 Uzaktan İzleme ve Uzaktan Kontrol İhtiyacı

Uzaktan izleme ve kontrol sistemleri, bireylerin bir yerdeyken başka bir yere ait olan bilgileri bilme ve müdahale etme isteğiyle doğmuştur. İnsancıl ve içgüdüsel olan bu istek, bireylere daha önce hiç görmediği yerleri göstermiş, hiç tanımadıkları insanlarla tanışma, konuşma ve genel olarak iletişim imkânı tanımıştır (Şekil 2.3).

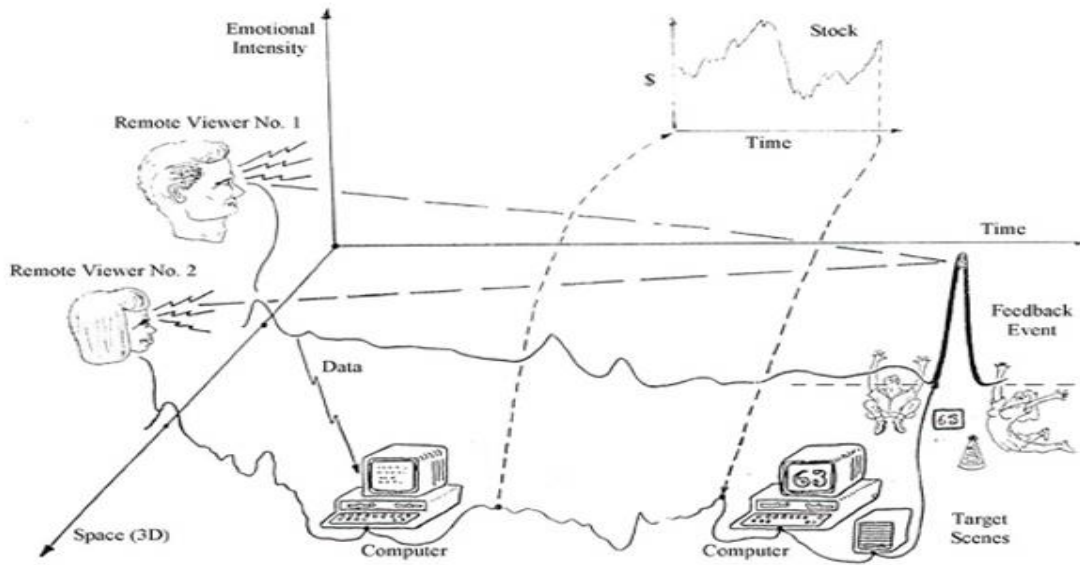


Figure 1. Associative Remote Viewing Process

Şekil 2.3 Uzaktan izleme işlevleri ilişkileri.

2.2.3 Uzaktan İzleme Yöntemi

Uzaktan izlenmesi istenen materyalden alınan ölçüm, veri iletimi sayesinde doğrudan veya dolaylı olarak kullanıcıya iletilir. İletilmesi istenen veri, ne şekilde iletilecekse (elektrik, ışık, frekans vb.), bunun için veri alımı ve veri okunması bölümlerinde dönüştürücü ve tersine dönüştürücü kullanılır. Son kullanıcı tarafında ise bu veriyi işleyen görsel veya işitsel işleyici kullanılır. Uzaktan kontrol sistemlerinde de çalışma prensibi aynı olup uzaktan izleme işlevinden farklı olarak veri iletimi çift yönlü olarak gerçekleştirilir (Şekil 2.4).

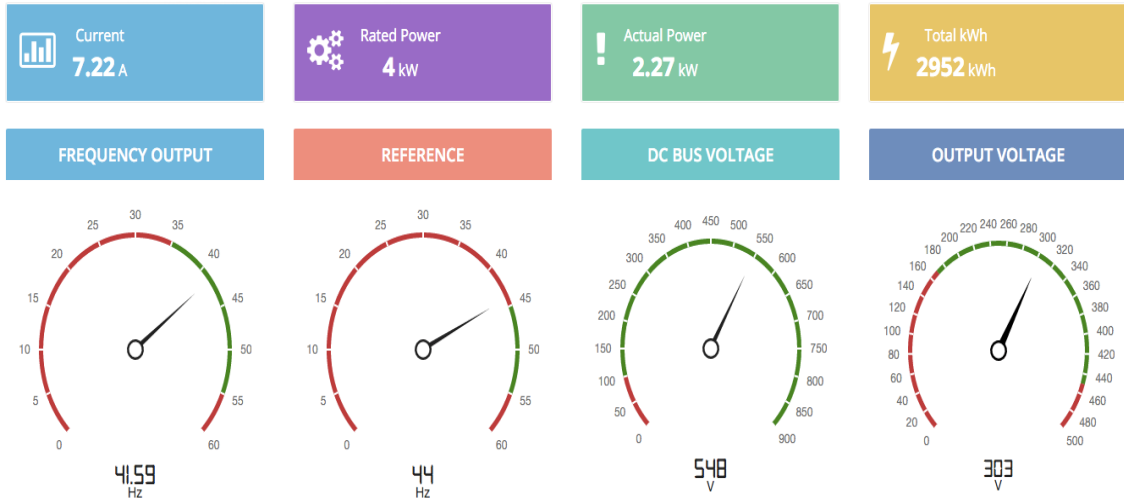


Şekil 2.4 Uzaktan izleme yöntemi.

2.2.4 Enerjinin Uzaktan İzlenmesi

Teknolojinin ve beraberinde internetin gelişimiyle birlikte sırasıyla bilgisayarlar, bilgisayarların gelişimi ve şekil, boyut ve yazılım değiştirmeleriyle birlikte cep

telefonları, televizyonlar ve görüntüleme sistemleri, ardından yaşam alanlarımızda hayatı kolaylaştırması amacıyla akıllı ev teknolojileri yaygınlaşmakta ve kullanım alanları hızla artmaktadır. Uzaktan kontrol ve uzaktan izleme konularında oluşan arz neticesinde bu teknolojiler hayatımızdaki bazı alanlara nüfus etmiş olmakla birlikte her alanda gelişimlerini devam ettirmektedirler. Devletlerin enerji ihtiyaçlarını yenilenebilir teknolojiden sağlamak için yaptığı yatırımların yanında bireyler de tak & kullan sistemlere sürdürülebilir yatırım yapmakta, bu sistemleri uzaktan izlemek ve kontrol etmek istemektedirler. Günümüzde enerji üretimi, orta ve büyük ölçekli santrallerden sağlandığı için genel olarak scada (veri tabanlı kontrol ve gözetleme) sistemleri ile ölçülen ve internet kanalıyla kontrolör ve yöneticilere iletilen, uzaktan izlenen enerji döngüsü, yenilenebilir sistemlerin ve akıllı ev sistemlerinin hızla yaygınlaşıyor olmasıyla birlikte kullanıcıların ihtiyaç duyacağı türden sistemlerdir. Hali hazırda elektrik dağıtım şirketlerinin iletim hatları üzerinden son kullanıcılara sağladığı enerjiyi zamanla kendisi üretmek isteyecek olan son kullanıcılar, enerji üreten sistemlerin sağlıklı çalışıp çalışmadığını ve akım, gerilim ve mevcut güç durumlarını da bilmek isteyeceklerdir. Bu durumda enerjiyi uzaktan izleme teknolojileri devreye girecektir. (Şekil 2.5) (<http://www.logicladder.com/energylogiciq-solar-energy-monitoring>. 2019).



Şekil 2.5 Örnek enerji tüketimi uzaktan izleme sistemi arayüzü.

2.3 Nesnelerin İnterneti Nedir ?

Karşılıklı bilgi alışverişinin en yoğun ve sağlıklı gerçekleştirildiği veri iletişim hattı internettir. Temeli 1962 yılında J.C.R. Licklider 'in Amerika'nın en büyük üniversitelerinden biri olan Massachusetts Institute of Tecnology'de (MIT) tartışmaya açtığı "Galaktik Ağ" kavramında bulabiliriz. J.C.R. Licklider galaktik ağ kavramında, isteyen herkesin, istediği herhangi bir yerden programlara ve istediği tüm verilere erişebileceği bir ağ kastedilmişti. Bunun devamı olarak 1965' te araştırmacı olarak görev yapan Lawrence Roberts ve Thomas Merrill dünyada ilk kez bilgisayarların birbirleriyle iletişimi sağladı. Bilgisayarların birbirleriyle iletişiminin gerçekleştirilmesi, internetin ilk adımının atılmasını sağladı ve bunu aşağıdaki gelişmeler takip etti;

- 1965, Mass, California, Telnet
- 1969, İlk ARPANET Nodel UCLA
- 1972, Küresel Gösterim, Email, DOD
- 1972-80, Yeniden Dizayn, TCP/IP Standartları
- 1984-85, Janet/NFSNET/Junet
- 1991, Gopher, Wais, WEB
- 1993, Mosaic, NCSA HTTPd
- 1960, Bush, Hypertext
- 1995, A.B.D, İnternetin Özel İşletim Süreci Başladı
- 1999, Nesnelerin İnterneti Kavramı İlk Kez Ortaya Atıldı
- 2006-2008, Avrupa'da İlk Kez Nesnelerin İnterneti Konferansı Gerçekleşti
- 2008-2009, Nesnelerin İnterneti Doğdu
- 2011- IPV6, Global Olarak Yayınlandı

İnternetin küresel bağlamda yaygınlaşması ve istikrarlı olarak büyümeye başlamasının ardından, gelişmiş ülkelerde ortalama veri iletim hızlarına ulaşıldı. Bilgisayarları takiben cep telefonları ve televizyonlar da internete bağlanmaya başladılar. Günümüzde kahve makinelerinden, çatallara kadar internete bağlanan cihazlar günden güne çeşitlenmektedir. Akıllı teknolojilerdeki gelişimin temel amacı; cihaz yazılımlarının küresel ağdaki ilerleyişi takip ederek güncellenmesi, tüm elektrikli cihazların birbirleriyle

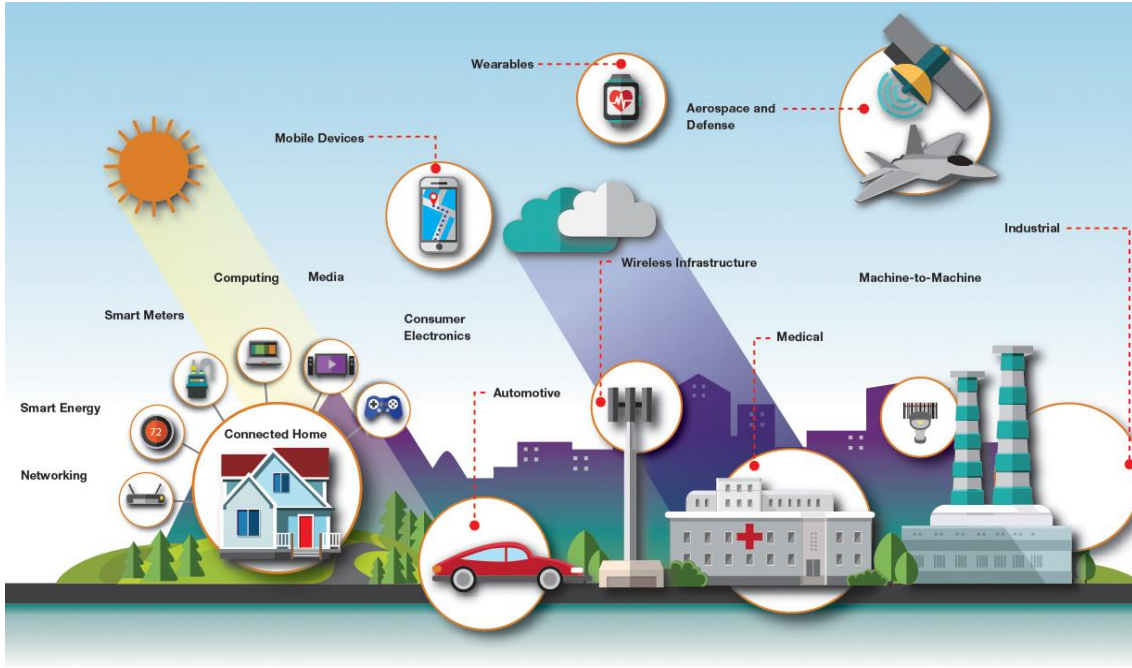
ve kullanıcılarla doğrudan ya da dolaylı iletişiminin sağlanmasıyla hayatın daha yaşanabilir hale gelmesidir. Günümüzde, akıllı kavramını kazanması en basit materyaller dahi yazılım ve çeşitli sensörler kullanan elektronik cihazlara dönüştürülmekte ve bu bağlamda birçok çalışma devam etmektedir. Birkaç örnekten bahsetmek gerekirse; akıllı bir termostat artık sadece ısıyı ölçmek yerine, kullanıcı alışkanlıklarını öğrenerek evin sıcaklığının ne olması gerektiğine karar vermekte ve bunu bildirmektedir. Akıllı çatal, kullanıcının beslenme alışkanlıklarını öğrenerek, gün içerisinde gereğinden fazla yemek yediğinde, kullanıcının mevcut beslenme şeklinin doğru olmadığını bildirmekte, daha az ve sağlıklı yemek yemesi hususunda kullanıcıyı bilgilendirmektedir. Akıllı toplar, topa kaç kez vurulduğunu, vuruşların gol olup olmadığını, topa hangi şiddette vurulduğunu, topun hangi yüksekliğe çıktığını ölçerek bildirebilmektedir. Akıllı evler, evin sıcaklığını uzaktan kontrol edebilmeyi, gün ışığına göre perdelerin kendinden açılıp kapanabilmesini, ışıkları ve ışık şiddetinin ayarlanabilmesini sağlayabilmektedir. Akıllı kilit kullanıcıları evlerine anahtar olmadan cep telefonları ile girip çıkabilmektedir (Şekil 2.6).



Şekil 2.6 Gelişmekte olan teknolojilerden beklentiler.

2.3.1 Nesnelerin İnternetinin Endüstriyel Uygulamaları

Nesnelerin interneti ve cihaz yazılımlarının gelişimiyle, daha az insan gücüyle daha çok verimin sağlanması amacıyla, endüstrinin 3.0 ‘dan 4.0 ‘a geçişi üzerine çalışmalar devam etmekte, bununla birlikte endüstrinin ihtiyaçları farklılık arz edeceği için ekonomik ve beşeri açıdan, dünyada pozitif ve negatif yönlü büyük değişimlerin yaşanacağı öngörülmektedir (Şekil 2.7) (İnt.Kynk.12).



Şekil 2.7 Nesnelerin internetinin endüstriyel uygulamaları.

Akıllı ev sistemlerinde kullanılan endüstriyel uygulamalar günden güne gelişerek, bizlere çok daha rahat ve modern bir yaşam alanı sağlamaktadır. Bunları genel hatlarla sıralamak gerekirse (Şekil 2.8);

- Çevreye adapte olabilen akıllı aydınlatmalar
- Kablosuz web ve mobil ağ üzerinden çalıştırılabilen grafiksel arayüzlü ışıklar
- Akıllı ekipmanların yönetimi ve kontrolü
- İzleme, güvenlik ve alarm ekipmanları
- Duman ve gaz algılama sensörlü internet tabanlı güvenlik sistemleri

- Görüntü, ses, projeksiyon gibi ev eğlence sistemleri



Şekil 2.8 Nesnelerin interneti akıllı ev sistemleri.

Nesnelerin interneti enerji sektörü uygulamaları ana hatlarıyla (Şekil 2.9);

- Gelişmiş ölçülendirme sistemi (Advanced Measuring Instruments)
- SCADA (Çalışma denetimsel kontrolü ve veri işleme sistemleri)
- Akıllı Eviriciler
- Enerji Üreten veya Tüketen Cihazların Uzaktan İzlenmesi



Şekil 2.9 Nesnelerin interneti enerji sistemleri.

Alıcıların gelişmesi, analog - dijital sağlık uygulamaları ve internet aracılığıyla, medikal cihazlarda ölçümleme ve veri iletişimi başladı. Sağlık sektöründe kullanılan IOT sistemleri (Şekil 2.10);

- Uzaktan Sağlık İzlemesi

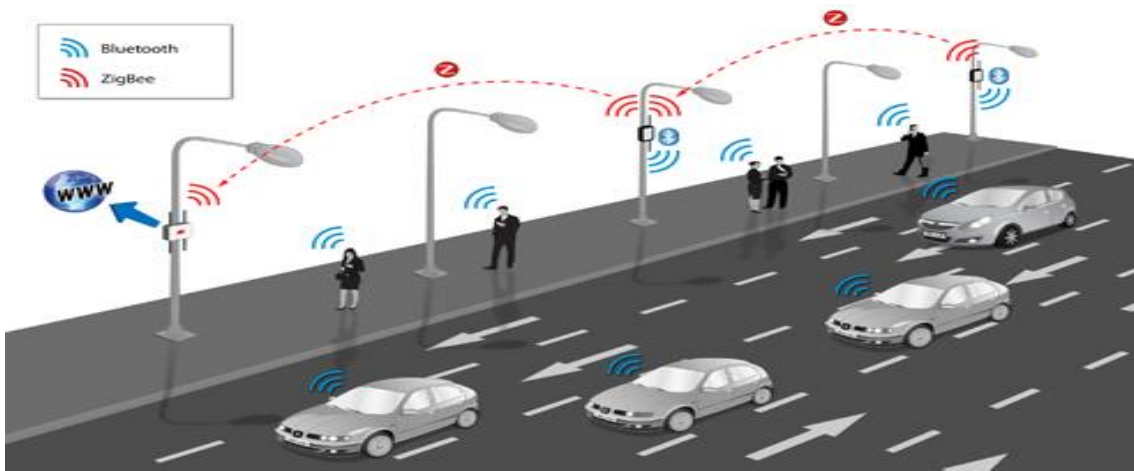
- Acil Uyarı Sistemleri
- Beden İçerisine Enjekte Edilen Akıllı Cihazlar
- Beden Üzerine Giyilebilir Cihazlar



Şekil 2.10 Nesnelerin interneti giyilebilir sağlık uygulamaları.

Trafik denetimi ve kazaların önüne geçebilmek adına geliştirilen yol ve yolcu IOT uygulamaları (Şekil 2.11);

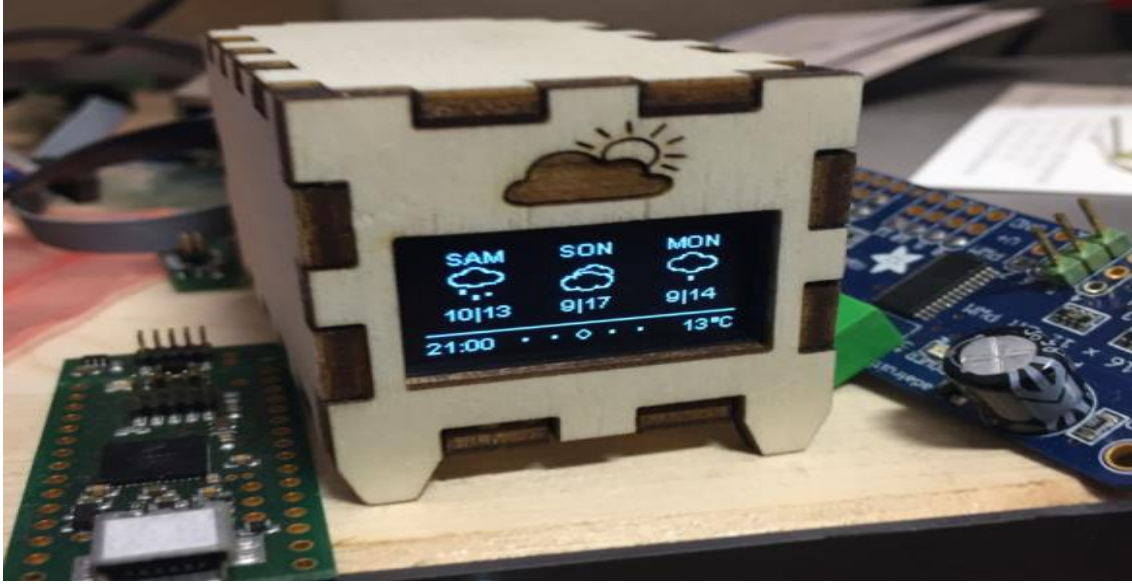
- Akıllı trafik kontrolü
- İnsansız özerk yol belirleme
- Yerel ve global araç iletişimi
- Acil kurtarma için otomatik şanzıman
- Güvenlik ve yol yardımı
- Akıllı park



Şekil 2.11 Nesnelerin interneti akıllı trafik kontrolü.

Deprem, yangın, tsunami gibi doğal afetler oluşmadan bildirim almak, hava ve çevre kirliliğinin önüne geçebilmek adına geliştirilen IOT uygulamaları (Şekil 2.12);

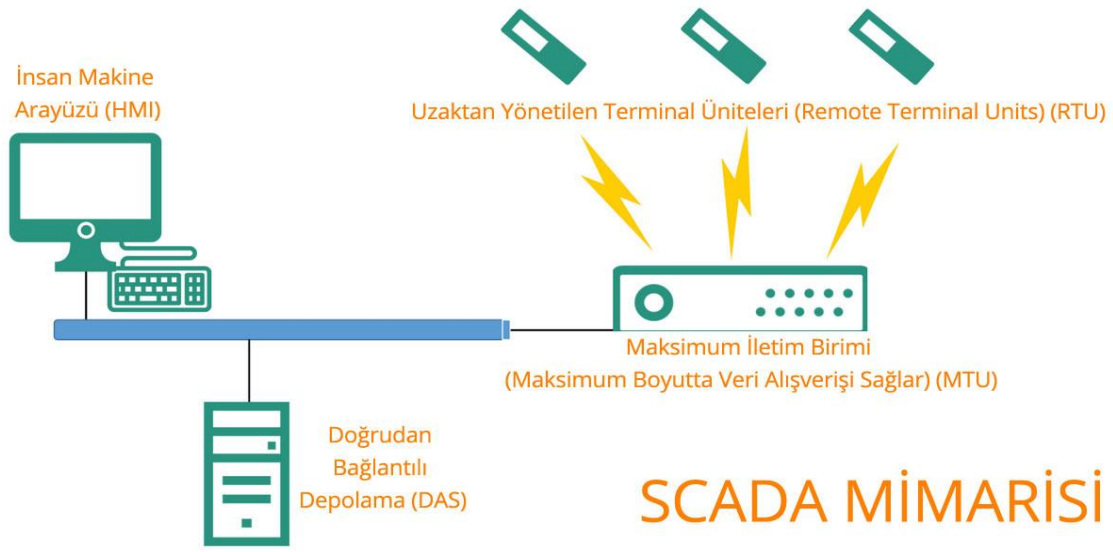
- Bulut tabanlı hava izleme
- Gürültü ve hava kirliliği izleme
- Yangın algılama sistemleri
- Deprem ve tsunami erken uyarı sistemi
- Toprak durum izleme



Şekil 2.12 Nesnelerin interneti hava tahmin istasyonu.

Elektrik enerjisiyle ilgilenen IOT uygulamaları ise scada sistemleri ve akıllı ölçüm sistemleridir.

► **Scada;** uzakta bulunan endüstriyel sistemlerin izleme, kontrol ve iletişimini sağlamaktadır. Sensörlerin bilgi akışı iletişim merkezi istasyonlarına iletilerek, HMI denilen insan arabirim cihazlarından bu verilerin takibi ve kontrolü sağlanır. Geçmiş veriler bulut sistemlerde saklanarak analiz programları üzerinden verilerin değişimi ve daha verimli çalışmalar için yapılması gereken geliştirme süreçleri üzerinden çalışmalar sürdürülmektedir (Şekil 2.13).



Şekil 2.13 Nesnelerin interneti scada mimarisi.

► **Akıllı Ölçümler;** Elektrik iletim ve dağıtım şirketleri her ay abone faturalarına işletim bedeli yansıtmaktadır. Bu işletim bedelinin içerisinde ise büyük ölçüde abonelere pay edilen sayaç okuma, açma kesme bedelleri yer almaktadır. Şimdi bir düşünün, bu şirketler tarafından periyodik olarak abonelerin enerji tüketimini ölçmek için yerleştirilen ölçüm cihazlarına internet üzerinden veri alışverişi sağlanırsa, elektrik dağıtım şirketleri, anlık olarak aktif üretim ve tüketimi, reaktif üretim ve tüketimi, ölçüm sayaçlarının sağlıklı çalışıp çalışmadığını görebileceklerdir. Faturasını ödemeyen abonelerin enerjilerini uzaktan kesebileceklerdir. Bu akıllı sayaçlar yaygınlaşırsa, elektrik dağıtım firmaları öncelikle sayaç okuma, açma kesme işlemleri çalışanlarını işten çıkaracak, üzücü şekilde yüzlerce belki binlerce kişi işsiz kalacaktır. Konuya diğer bir yönden bakacak olursak, bu şirket abonelere uyguladığı işletim ücretinden vazgeçmek zorunda da kalacaktır. Aynı zamanda çeşitli masraflar ve bakım maliyetleri de azalacağı için abonelerin elektrik enerjisi kullanım maliyetlerinde de indirimlerin de meydana gelmesi söz konusudur. Görüldüğü üzere akıllı bir elektrik saati dahi hayatımızda büyük değişimler meydana getirebilmektedir. Bu teknolojiler bir taraftan fayda sağlarken, diğer taraftan daha az insan gücüne gereksinim duyulacağı için işsiz sayısında oldukça fark edilir düzeyde artışa sebep olacaktır. Aynı zamanda beklenti gelecek neslin bilgi teknolojileri alanına uzmanlaşmak ve çalışmak hususunda daha fazla baskı altında hissedeceği ve akıllı sistemlerin kendine hiç adını duymadığımız iş dalları üreteceği yönündedir (Şekil 2.14).



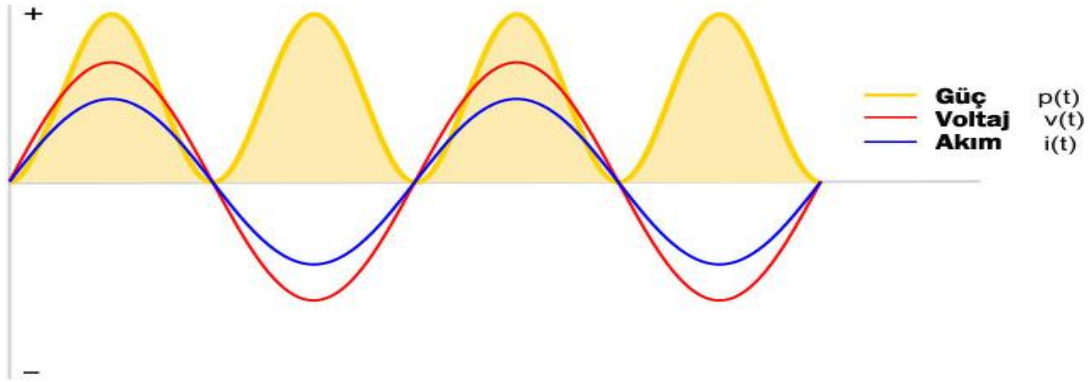
Şekil 2.14 Nesnelerin interneti akıllı ölçüm saatleri.

2.4 Ac Güç Sistemini Anlamak

Elektrik saati ev şebekesine bağlı cihazların kullandığı enerjiyi ölçer. Bunu nasıl yaptığını anlamak için, cihazların elektrik sistemi ile nasıl etkileşime girdiği hakkında bir şeyler bilmek gerekmektedir. Tüm cihazlar elektrik sistemiyle aynı şekilde etkileşime girmez. Bu konu başlığında ilk önce dirençli yükler ve kullandıkları gücün nasıl hesaplandığı üzerinde durulacaktır. Daha sonra reaktif yüklerden ve ardından doğrusal olmayan yüklerden bahsedeceğiz. Son olarak bu, eğer enerji tüketildiği şekilde üretilirse güç akışının yönünü nasıl ölçtüğümüzü gösterecek.

2.4.1 Omik Yükler

Akkor ampuller, ütüler, elektrikli su ısıtıcıları, elektrikli ve elektrikli ocakların çalışma mantığı oldukça basittir. Kısa bu cihazlar kendilerine verilen tüm enerjiyi kullanırlar. Bunlar dirençli yüklerdir, yani mevcut güçleri dirençleriyle voltajın çarpımına eşittir (Ohm Kanunu). Tamamen dirençten oluşan bir yük, bize aşağıdakine benzer bir akım gerilim dalga formu verir (Şekil 2.15).

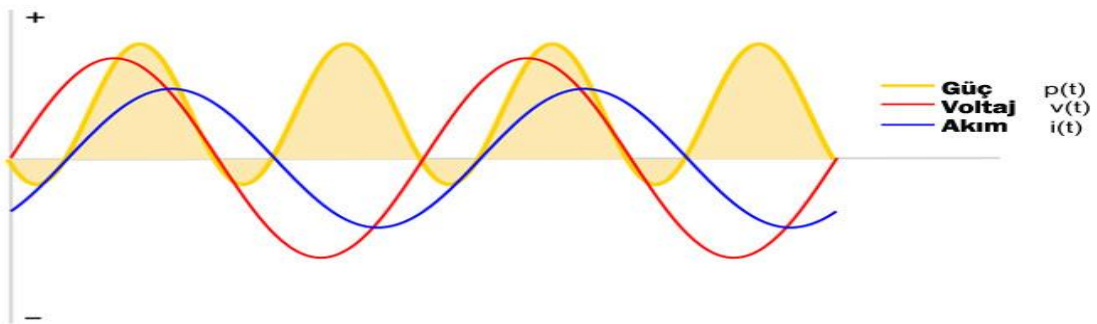


Şekil 2.15 Omik yüklerde akım gerilim ilişkisi.

Sarı çizgi belirli bir zamanda güçtür (herhangi bir anda anlık güç olarak adlandırılır), anlık güç, belirli bir zamanda voltaj ve akımın çarpımına eşittir. Şekilde gücün her zaman pozitif olduğuna dikkat edin. Bu durumda, pozitif yön yüke akan enerjidir. Bu aynı zamanda prensipte reaktif gücün olmadığını göstergesidir.

2.4.2 Kısmen Reaktif Yükler

Bununla birlikte, buzdolapları, çamaşır makineleri, beton matkapları ve ark kaynağı gibi cihazlar belirli miktarda enerji aldıkları için hesap o kadar kolay değildir. Bu cihazlar kullandıkları enerjinin bir miktarını elektrik şebekesine geri gönderirler. Bunlar, dirençli bileşene ek olarak endüktif (örneğin motorlar) veya kapasitif (örneğin ark kaynakçıları) bileşenlerine sahiptir. Kısmen endüktif bir yük, aşağıdakine benzer bir akım gerilim dalga formu verir (Şekil 2.16).



Şekil 2.16 Kısmen reaktif yükte gerilim ve akım ilişkileri.

Sarı çizginin şimdi bir süre boyunca negatif olduğuna dikkat edin, bu şekilde pozitif kısım yüke akan enerji ve negatif kısım yükten geri akan enerjiyi temsil eder. Dikkate alınması gereken diğer bir şey ise, kırmızı ve mavi çizgiler “Şekil 2.16” ile karşılaştırıldığında, voltaj ve akım dalga biçimlerinin birbirinden kaydırılmış olduğudur. Dirençli ve oldukça büyük bir kapasitör şarj ettiğimizi hayal edelim (Dirençli olduğu için anında şarj olamaz): Şarj etmek için kondansatörü önce boşaltalım. Besleme voltajı başlangıçta kapasitördeki voltajdan daha yüksektir, akım kapasitörün içine doğru akar (grafikteki pozitif yön), bu durum kapasitör voltajının yükselmesine neden olur. Besleme gerilimi düşer. Ardından, yüklenen kapasitördeki voltaj, besleme voltajından yüksek olur. Akım, bu sefer zıt yönde geriye doğru akmaya başlar (grafikteki negatif yön)[2.16]. Bu, geçerli dalga formunun grafikte gösterildiği gibi kaydırılmış gibi görünmesine neden olur. (Bu duruma faz kayması denir). Grafikten anladığımız üzere reaktif gücün oluşmasındaki etkenin bu faz kayması olduğundan bahsedebiliriz.

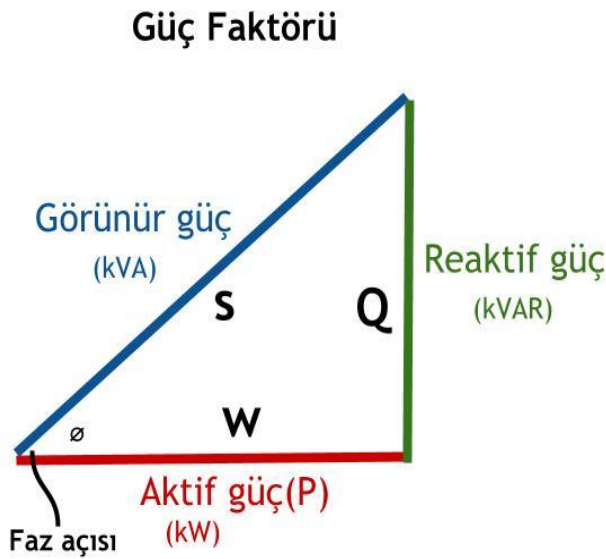
2.4.3 Gerçek Güç, Reaktif Güç ve Görünen Güç

Şebeke frekansındaki voltajı inceleyecek olursak, elektrik akımı saniyede 50/60 kez dalgalanır. Bu hızdaki güç değişimine ayak uyduramıyoruz, bu nedenle güç ölçümü için daha faydalı bir değere sahibiz: gerçek veya aktif güç olarak adlandırdığımız anlık gücün ortalaması.

Gerçek güç, genellikle bir cihaz tarafından faydalı işler üretmek için kullanılan güç olarak tanımlanır. Yukarıdaki grafiğe bakıldığında, sarı dalgalı pozitif kısım, güç kaynağından gelen güçtür ve sarı dalgalı negatif kısım ise güç kaynağına geri giden güçtür. Aslında yük tarafından kullanılan gerçek güç hattan gelen pozitif güçten, hatta geri giden negatif gücün mutlak değerlerinin çıkarılmasına eşittir.

Reaktif veya hayali güç, yük ile faydalı bir çalışma yapmayan besleme hattına geri giden güçtür.

Gücün diğer bir yararlı ölçütü, Kök-Ortalama-Kare (RMS) Geriliminin ve RMS Akımının ürünü olan Görünen Güç'tür. Tamamen dirençli yükler için (omik), gerçek güç görünür güce eşittir. Ancak tüm diğer yükler için, gerçek güç görünür güçten daha küçüktür. Görünür güç, gerçek ve reaktif gücün ilişkisinden doğar, görünür gücü gerçek ve reaktif güçleri direk toplayarak bulamayız. Çünkü ikisinin arasında bir faz farkı mevcuttur ve vektörel toplama yapmak gerekmektedir (Şekil 2.17).



$$P = VI \cos \phi$$

$$\cos \phi = \frac{P}{VI}$$

$$\cos \phi = \frac{kW}{kVA}$$

$$\cos \phi = \frac{\text{Aktif güç}}{\text{Görünür güç}}$$

$$\cos \phi = \frac{kW}{kVA} = \text{Güç faktörü}$$

Ayrıca rezistans ve empedans arasındaki oran da güç faktörü olarak adlandırılır.

$$\cos \phi = \frac{R}{Z}$$

Şekil 2.17 Güç faktörü aktif ve görünür güç ilişkisi.

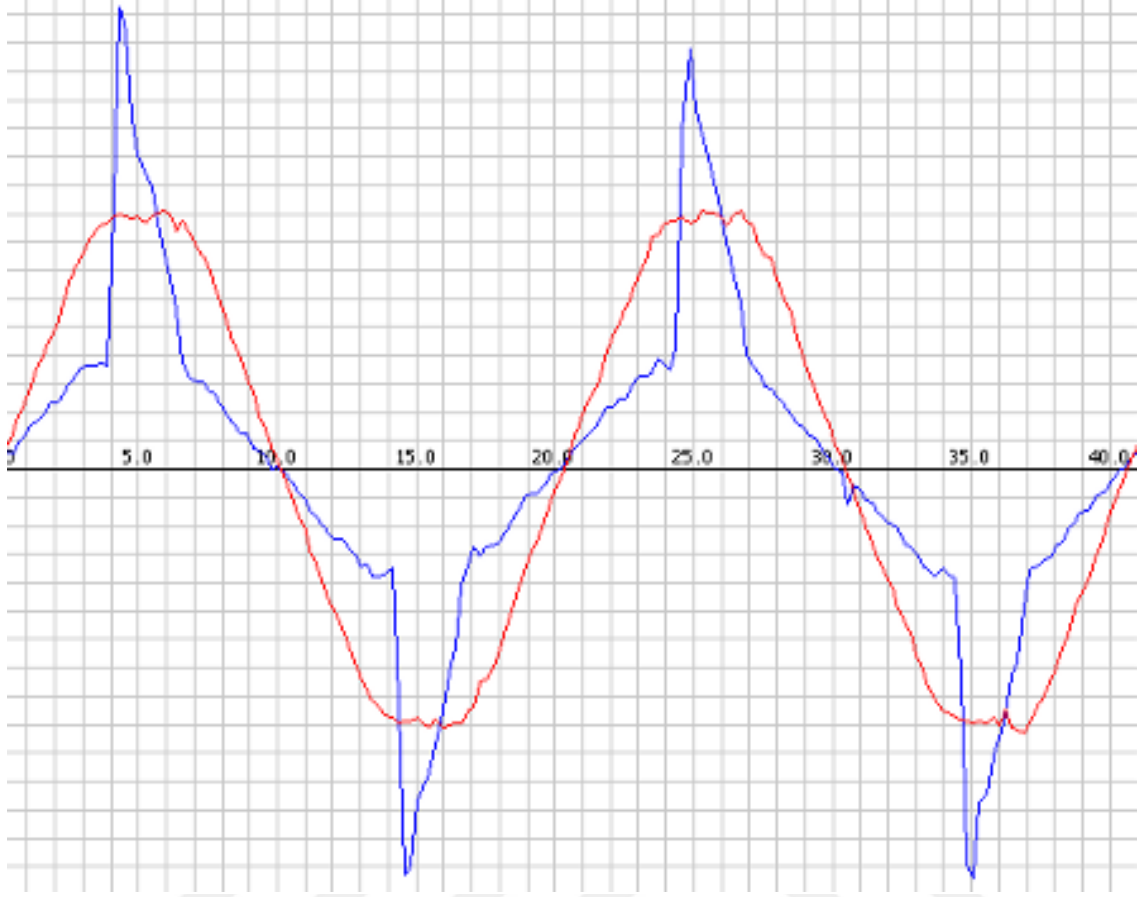
İdeal yani sinüzoidal yükler için gerçek, reaktif ve görünür güç arasındaki ilişki aşağıdaki gibidir:

- Aktif Güç = Görünen Güç x $\cos \phi$ (2.1)

- Reaktif Güç = Görünen Güç x $\sin \phi$ (2.2)

$\cos \phi$ aynı zamanda güç faktörü olarak da bilinir.

Bu güç faktörü ilişkisi yalnızca sinüzoidal yükler için geçerlidir. Örneğin; Dizüstü bilgisayarlar gibi doğrusal akım kullanan aygıtlar için çoğu güç kaynağı, şebekeye gerisingeri doğrusal olmayan bir yük sunar. Akım gerilim eğrileri de şu şekildedir [2.18]:



Şekil 2.18 Doğrusal olmayan yüklerin akım-gerilim eğrisi.

Güç faktörünü hala aşağıdaki denklemden hesaplayabiliriz:

- $Güç Faktörü = \frac{(Gerçek Güç)}{(Görünür Güç)}$ (2.3)

Fakat saf sinüs dalgaları için doğru olan:

- $(Görünür Güç)^2 = (Aktif Güç)^2 + (Reaktif Güç)^2$ (2.4)

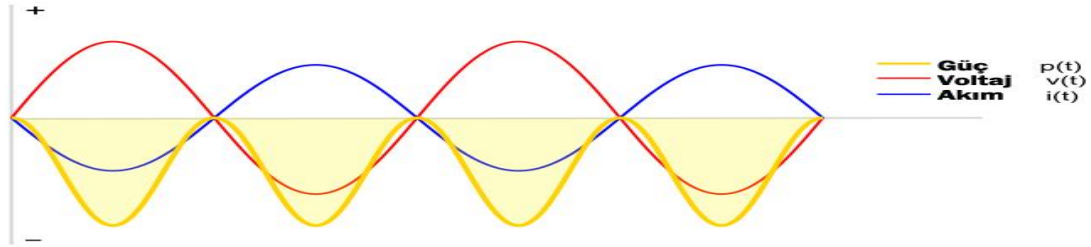
Formül 2.4 deki denklemini artık doğru doğrusal olmayan yükler için kullanamayız. Güç faktörü = $\cos\Phi$ de tam olarak doğru değildir, çünkü hem yüksek gerilim hem de akım dalga formlarındaki yüksek harmoniklerin etkileri dikkate alınmalıdır.

Güç faktörü değeri, şebeke verimliliğinin hem faz gecikmesinden hem de giriş akımının harmonik içeriğinden ne kadar etkilendiğini belirler.

2.4.4 Güç Akışı Yönünün Belirlenmesi

Şimdiye kadar yükün güç harcadığından bahsettik. Güç harcayan bir cihaz enerji tüketiyorsa güç üretirken de şebekeye enerji vermelidir, o zaman akımın akış yönü de tersine çevrilmelidir. Ancak akım dalgalı olduğundan, yön yine de saniyede 50 (veya 60) kez tersine dönmelidir. Bu yüzden geçerli yönü karşılaştırmak için bir referansa ihtiyacımız var. Referans için gerilim dalgalarını kullanabiliriz. Şekil 2.15 'de, gerilim ve akım dalgaları birlikte yükseldi ve birlikte alçaldılar. Gerilim pozitif olduğunda (X ekseninin üstünde) akım da pozitif idi ve gerilim negatifken (x ekseninin altında) akım da negatifti. Şekil 2.15 'de güç omik olduğundan, voltaj ve akımın direk çarpımına eşitti ve bu nedenle güç her zaman pozitifti yani tüm güç eğrisi X ekseninin üstündeydi.

İkinci durumda eğer cihaz enerji üretiyorsa, önceki örneğimize kıyasla akımın yönünü tersine çeviririz. Bu durumda gerilim pozitif olduğunda akım negatif (Güç X ekseninin altında) ve gerilim negatif olduğunda akım pozitif (Güç X ekseninin altında) olmalıdır. Sonuç olarak cihaz güç üretiyorsa eğri daima negatif yani tüm güç eğrisi X ekseninin altında olmalıdır (Şekil 2.19) (İnt.Kynk.6).



Şekil 2.19 Güç üretimi esnasında gerilim, akım ve güç ilişkileri.

Gücün X ekseni boyunca işareti gücün tüketildiğini veya üretildiğini göstermektedir.

2.4.5 Sonuç

AC sistemlerinde enerji kullanımıyla ilgili dikkate alabileceğimiz birçok parametre var ve her birinin kendine özel esasları vardır. Sonuç olarak bir evde enerji ölçümü için, tüm cihazlarınızın gerçekte ne kadar güç harcadığını bilmek ve bu cihazların giderlerine ne kadar para harcadığını ölçmek için gerçek güç ölçümü en doğru yöntemdir.

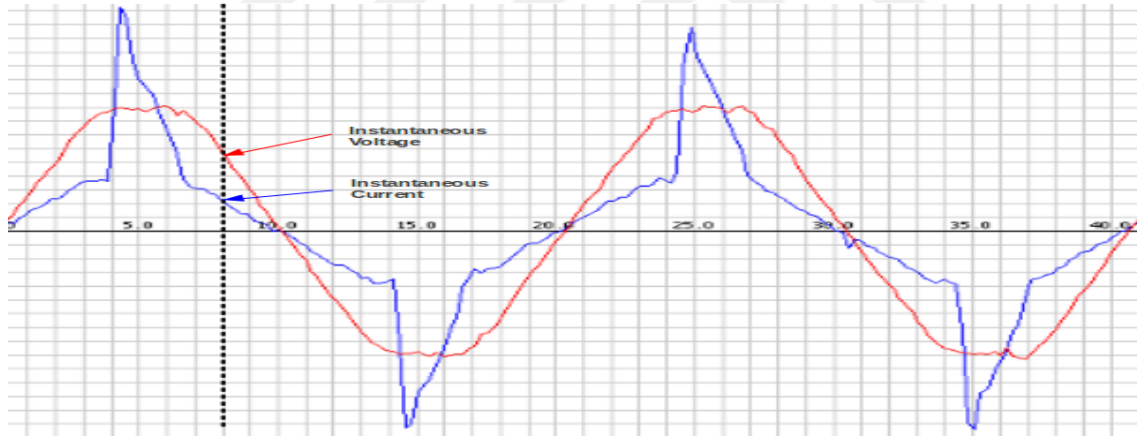
3. MATERYAL METOD

3.1 Anlık Gerilim & Akım

Adından da anlaşılacağı gibi, AC gerilimi ve akım sürekli alternatif zaman içinde gösterilmektedir. Gerilim ve akımın dalga formunun bir resmini çizersek, aşağıdaki şekle benzeyecektir. [3.1] Güç tüketen yük türüne bağlı olarak bu dalga formu değişiklik gösterir. Mevcut dalga şekli (Aşağıdaki şemada mavi) tipik bir dizüstü bilgisayar güç kaynağını incelediğimizde elde ettiğimiz görüntüdür. (Bir akkor ampul de var).

Görüntü, şebeke voltajını ve akımını yüksek frekansta örnekleyerek alınmıştır, bu da tam olarak Emontx veya Arduino 'da ölçümleme yaparken yaptığımız şeydir. Her 20 milisaniyede bir 50 ile 100 arasında ölçüm yapılmaktadır. Örnekleme sadece akım varsa 100' dür. Akım ve gerilim aynı anda örnekleniyorsa bu sınır 50'dir. Arduino işlemcisinin analog okuma komutu ve hesaplama hızı ile bu sayıda sınırlıdır.

Her bir bireysel numune anlık bir voltaj veya akım okumasıdır (Şekil 3.1) (İnt.Kynk.18).



Şekil 3.1 Anlık akım & gerilim grafiği.

3.2 Arduino'daki Gerçek Gücü Hesaplama

Gerçek güç, anlık gücün ortalamasıdır (İnt.Kynk.3). Hesaplama oldukça kolaydır. Öncelikle anlık gerilim ölçümünü anlık akım ölçümü ile çarparak anlık güç hesaplanır. Anlık güç ölçümünü belirli sayıda örnek üzerinde toplanır ve alınan örnekleme sayısına bölünür.

Kod:

```
for (n=0; n// anlik_voltaj ve anlik_akim hesaplama girisi

    anlik_guc = anlik_voltaj * anlik_akim;

    topla_anlik_guc += anlik_guc;
}

gercek_guc = toplam_anlik_guc / toplam_ornek_sayisi;
```

3.3 Kök-Ortalama-Kare (RMS) Gerilimi

Kök-ortalama karesi, adından da anlaşılacağı şekilde hesaplanır. Önce miktarı kareleriz, sonra ortalamayı ve son olarak, karelerin ortalamasının karekökünü hesaplarız, şu şekilde yapılır:

Kod:

```
FOR (n = 0; n // anlik_voltaj ham ADC girişinden hesaplama burada girilir.

    Voltaj_kare = anlik_voltaj * anlik_voltaj;

    toplam_voltajin_karesi + = voltaj_kare;
}

ortalama_voltaj_kare = toplam_voltaj_kare / toplam_ornek_sayisi;

kok_ortalama_voltaj_kare = sqrt(ortalama_voltaj_kare);
```

3.4 Görünür Güç

Kod:

```
gorunur_guc = kok_ortalama_kare_voltaj * kok_ortalama_kare_akim;
```

RMS gerilimi genellikle sabit bir deęerdir: 230V (Türkiye'de% + 10 -% 6). RMS voltajını 230V'a ayarlayarak voltaj ölçümü yapmaksızın görünür gücü yaklaştırmak mümkündür. Bu, piyasada satılan enerji monitörleri tarafından kullanılan yaygın bir uygulamadır (İnt.Kynk.5).

3.5 Güç Faktörü

Kod:

```
guc_factoru = gercek_guc / gorunur_guc;
```

Bunlar, bir Arduino 'daki AC güç ölçümünün temelleridir.

Bu sayfa, tek fazlı AC elektriğın anlık gerilim ve akım ölçümlerinden gerçek gücü, görünen gücü, güç faktörünü, RMS voltajını ve RMS akımını hesaplamaların arkasındaki matematiğı kapsar. Hesaplamalar dijital alanda Arduino 'da yapıldığından ayrık zaman denklemler detaylandırılmıştır.

3.6 Gelişmiş Arduino Matematiğı

3.6.1 Gerçek Güç

Gerçek güç (aynı zamanda aktif güç olarak da bilinir), bir cihaz tarafından faydalı işler üretmek için kullanılan güç olarak tanımlanır. Matematiksel olarak, gerilimin, $u(t)$, zaman akımı, $i(t)$ 'nin belirli integralidir:

$$P = \frac{1}{T} \int_0^T u(t)i(t)dt \equiv UxIxcos(\delta) \quad (3.1)$$

Denklem 1. Gerçek Güç Tanımı.

- U - Kök-Ortalama-Kare (RMS) gerilimi.
- I - Kök-Ortalama-Kare (RMS) akımı.
- $cos(\delta)$ - Güç faktörü.

$$P \equiv \frac{1}{T} \int u(t) \times i(t) dt \equiv U_{Xlxcos}(\varphi) \quad (3.2)$$

Denklem 2. Ayırık Zamanda Gerçek Güç Tanımı.

- $u(n)$ - örneklenmiş $u(t)$ örneği
- $i(n)$ - $i(t)$ 'nin örneklenmiş örneği
- N - örnek sayısı.

Ayrık zaman eşdeğeri:

$$P \equiv \frac{1}{N} \sum_{n=0}^{N-1} u(n)xi(n) \quad (3.3)$$

Gerçek güç, basitçe N gerilim akımı ürünlerinin ortalaması olarak hesaplanır. Bu yöntemin hem sinüzoidal hem de çarpık dalga formları için geçerli olduğu gösterilebilir.

3.6.2 RMS Gerilim ve Akım Ölçümü

Bir RMS değeri, periyodik olarak değişken bir miktardaki anlık değerlerin karelerinin ortalama değerinin, bir tam çevrim boyunca ortalaması alınmış karekökü olarak tanımlanır. Gerilim RMS hesaplaması için ayırık zaman denklemi aşağıdaki gibidir:

Denklem 3. Ayırık Zamanlı Alanda Gerilim RMS Hesabı.

$$U_{RMS} = \sqrt{\frac{\sum_{n=0}^{N-1} u^2(n)}{N}} \quad (3.4)$$

RMS akımı, aynı örnekler kullanılarak hesaplanır, mevcut örneklemeler için $u(n)$ voltaj örneklerinin yerine, $i(n)$ kullanılır.

3.6.3 Görünen Güç ve Güç Faktörü

Görünen güç şu şekilde hesaplanır:

$$\text{Görünen güç} = \text{RMS Voltajı} \times \text{RMS Akımı} \quad (3.5)$$

Ve güç faktörü:

$$\text{Güç Faktörü} = \text{Gerçek Güç} / \text{Görünen Güç}$$

(3.6)

3.7 Akım Trafoları AT Sensörleri



Şekil 3.2 SCT-000-013 akım trafosu sensörü.

Bir akım trafosunun yapısı hepimizin bildiği ve daha yaygın kullanılan gerilim trafosunun yapısına benzer. Demirden ya da ferritten meydana gelen çekirdek iki sargıya sahiptir. Ancak gerilim trafosunun aksine, ikincil tarafta yalnızca bir sargıyla gelir. Birincil sargı, transformatör çekirdeğinden geçen kablo ile beslenir. Ve adından da anlaşılacağı gibi, gerilimler yerine akımlar ile çalışır. Bu nedenle, birincil sargıda meydana gelen akımla orantılı olan ikincil sargı bir çıkış akımı üretecektir. AT sınırları dâhilinde, bu akımı sürmek için gerekli olan gerilimi meydana getirecektir. Akım trafoları (AT), alternatif akımı (AC) ölçen sensörlerdir. Ölçüm yapılması istenilen herhangi bir sistemde, elektrik tüketimini veya üretimini ölçmek için kullanışlıdır.

(Şekil 3.2)' de yer alan AT gibi bölünmüş çekirdek tipi, herhangi biçimde tehlikeli bir elektrik işçiliğine gerek kalmaksızın binaya gelen faz veya nötr kablolarına bağlanabilir. Diğer herhangi bir transformatör gibi, bir akım transformatörünün de bir birincil sarımı, manyetik bir çekirdeği ve ikincil bir sarımı vardır. Herhangi bir binanın akım değerleri ölçülmek istenildiğinde, akım trafosunun birincil sargısının çevrelediği açıklıktan kablo geçirilir. Ancak asla iki faz yada faz nötr kabloları tek açıklıktan geçirilmez. İkincil sargı, transformatör kasasına yerleştirilmiş birçok ince tel dönüşünden oluşur. Birincilde meydana gelen alternatif akım, çekirdekte başka bir ikincil sarmal devresindeki akımı indükleyen bir manyetik alan üretir (3.7) (3.8) (İnt.Kynk.8) (İnt.Kynk.16).

$$İkincil = CTDönüşOranı \times İbirincilsargı \quad (3.7)$$

$$CdönüşOranı = \frac{Birincil\ Dönüş}{İkincil\ Dönüş} \quad (3.8)$$

İkincil sargıda akan akım, birincil sargıdaki akan akımla orantılıdır:

Örneğin yukarıda gösterilen AT'deki ikincil dönüşlerin sayısı 2000 olsun, bu nedenle ikincildeki akım birincildeki akımın 2000'inden biridir.

Akım trafolarındaki bu orantısal hesap, amper cinsinden akımın sayısal değeri biçiminde yazılır, örneğin; 100:5 (100' e 5'ölçülü 200 oranlı akım trafosu). Yukarıdaki AT içinse bu oran 100: 0,05 şeklindedir. (2000 oranlı)

3.7.1 Akım Trafoları Çeşitleri

Akım trafolarının iki tipi vardır ve tam olarak aynı şekilde çalışmaktadırlar aralarındaki tek fark ise yapım şekillerindedir.

3.7.1.1 Halka Çekirdekli Akım Trafosu

Çekirdek tek parça halindedir ve AT'yi monte etmek için takmak istediğiniz kablonun, bağlantısının kesilmesi gerekir. Devrenin kesilmesi gerektiğinden yalnızca pratik, güvenli ve yasal olduğunda kullanılır. Halka çekirdekli akım trafosunun avantajı kablo üzerine temas etmesi sebebiyle iletim arttığı için ölçüm konusunda daha yüksek doğruluk sağlamasıdır.

3.7.1.2 Bölünmüş Çekirdekli Akım Trafosu

Halka çekirdekli akım trafosundan birinci farkı çekirdeğin iki parça halinde klipsler veya vidalarla tutturulmuş olmasıdır. Birbirinden bağımsız bu iki parça izolasyonlu kablo içerisinden geçirilerek monte edilir. Devre bağlantısının kesilmesine gerek yoktur. Avantajı elektrik işçiliğinin kolay olması, dezavantajı halka çekirdekli akım trafolarına göre daha az hassasiyet sağlamasıdır.

3.7.1.3 Yük Direnci

Akım trafoları diğer bir adla, trafolu akımölçerler mutlaka bir yük direnci ile birlikte kullanılmalıdır. Yük direnci, AT'nin ikincil devresini tamamlar veya kapatır. Yük değeri, ikincil akımla orantılı bir voltaj sağlamak amacıyla seçilir. AT çekirdeğinin doymasını önlemek için yük değerinin yeterince düşük olması gerekir.

3.7.2 İzolasyon, Güvenlik, Kurulum ve Kullanımı

3.7.2.1 İzolasyon

İkincil devre birincil devreden galvanik olarak yalıtılmıştır.(Kısacası metalik bir temas yoktur).

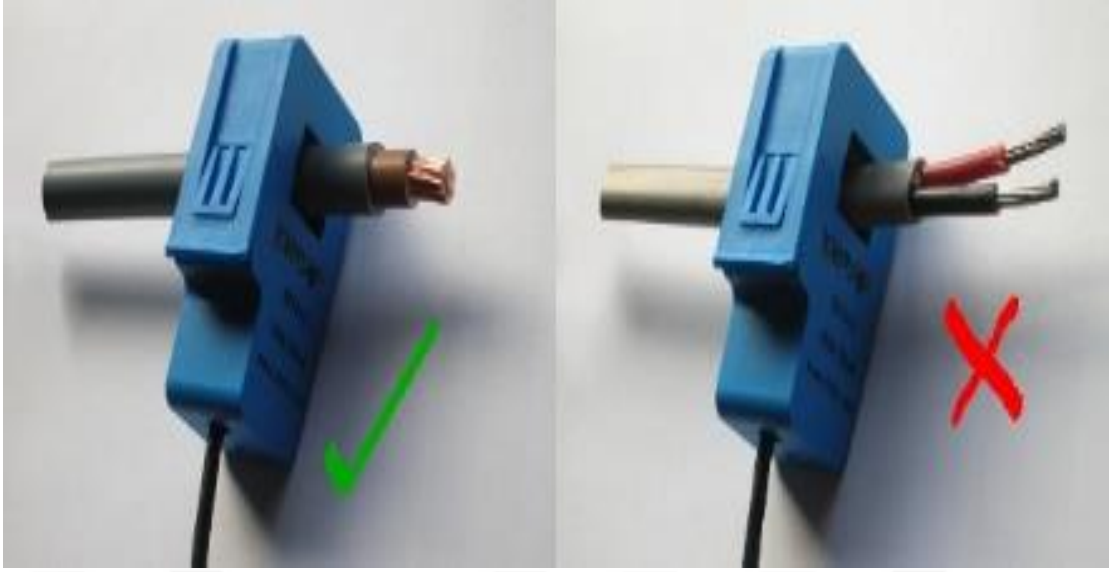
3.7.2.2 Güvenlik

Genel olarak, bir AT akım taşıyan bir iletkene bağlandığında asla açık devre olmamalıdır. Açık devre varsa, bir AT potansiyel olarak tehlikelidir. Eğer birincil sargıda akım akarken açık devre olursa, ikincil sargı transformatör sürüş akımını sonsuz bir empedans olana kadar devam ettirmeye çalışacaktır. Bu, ikincil sargıda yüksek ve potansiyel olarak tehlikeli bir voltaj üretecektir. Bazı AT'lerin yerleşik koruması vardır. Bazıları bizim projede kullandığımız gibi (SCT-013-000) koruyucu zener diyotlarına sahiptir. AT, voltaj çıkışı tipindeyse, yerleşik bir yük direncine sahiptir. Böylece açık devre yapılamaz. Projede kullandığımız klipsli akım sensörleri non-invasive tipindedir (temassız, manyetik) ve AC şebekeyle doğrudan (metalik) teması olmamalıdır. Bununla birlikte, sensörlerin monte edilmesi, insan hayatı için tehlikeli boyutta enerji taşıyan kablolarla yakın çalışma gerektirmektedir. Montaj yaparken, kabloların tamamen yalıtıldığından ve sensörleri takmadan önce gücün kapalı olduğundan emin olunması gerekir. Eğer yük dirençsiz bir akım trafosu, akım taşıyan bir kabloya bağlanırsa AT zarar görebilir, yanabilir, patlayabilir.

Projemizdeki YHDC transformatör, voltaj çıkışını güvenli bir seviyeye sınırlamak için dâhili zener diyotlarına sahiptir ve maksimum yüklenmemiştir. Dâhili yük dirençlerine sahip AT'ler ("voltaj çıkışı" tipi), tehlikeli voltajlara karşı da korunur. Projemizde kullanılan YDCH asla çıplak bir iletkenle takılmamalıdır. Kullanılan akım trafosu, bakır iletken üzerine monteli veya temassız manyetik, nasıl monte edilmesi gerekiyorsa türüne göre bağlantı yapılmalıdır. Elektrik çarpması tehlikesinden ayrı olarak, dikkate alınması gereken iki faktör vardır: AT'nin yalıtımının gücü ve üzerinde çalışma yapılan iletkenlerin normal sıcaklıklarına dayanma kabiliyeti.

3.7.2.3 Kurulum & Kullanımı

AT'nin birincil sargısı, ölçmek istediğiniz akımı taşıyan teldir. AT aynı akımı taşıyan fakat ters yönlerde telleri olan iki veya üç çekirdekli bir kabloya klipslenirse, ters yönlü vektörel akımlar tarafından oluşturulan manyetik alanlar birbirini iptal eder. AT bu durumda yanlış ölçüm yapar veya hiç yapmaz. Özellikle projemizde kullandığımız tip ferrit çekirdeğe sahip olan (YHDC tarafından yapılanlar gibi) bölünmüş çekirdekli bir AT, hiçbir zaman herhangi bir paketleme malzemesi kullanılarak kabloya kelepçelenmemelidir, çünkü ferrit çekirdeğin kırılkan yapısı sebebiyle kolayca kırılabilir, böylece AT bozulabilir. AT'yi yalnızca, mahfaza özel olarak tasarlandıysa, kabloya veya baraya kenetlenmelidir. Benzer şekilde, bir halka çekirdekli AT, merkezden serbestçe geçemeyecek kadar büyük bir kabloya asla zorlanmamalıdır. AT açıklığı içindeki kablonun konumu ve yönü ölçüm değerini mutlak anlamda etkilemez. Aşağıda sağdaki resimdeki gibi iki veya çok damarlı kabloya bağlantı yapılırsa, iletkenlerin her birindeki akımların toplamalarını ölçer. İki veya çok damarlı kablo ve toprak kablo olması durumunda, aynı akım zıt yönlerde akacak ve artı ve eski değerlerin toplamı sıfır olacaktır. Bu durum, doğru ölçüm yapmamızı engellediği için istenilen bir durum değildir (Şekil 3.3).



Şekil 3.20 AT sensörü kablo bağlantısı.

3.7.3 AT 'nin Kablo Yönü Önemli Mi ?

Eğer gerçek gücü okumak istiyorsak ve gücün hangi yönde gücün aktığını bilmek istiyorsak, örneğin; kendi elektriğimizi ürettiğinizde, alınan ve verilen enerjiyi bilmek istediğimizde AT doğru yöne bakmalıdır. Projemiz üretilen ve tüketilen güçler üzerine hazırlanmıştır. AT 'ler yanlış şekilde aynı yönlerde konumlandırılırsa, bu, pozitif olmasını beklediğimiz gücün negatif olarak gösterileceği anlamına gelir.

Türkiye’de AC-AC adaptörü ve YHDC AT, hat iletkeni üzerindeyken, AT’nin pozitif güç akışı soldan sağa doğru olmalıdır. AT nötr iletkenin üzerindeyken (bu sadece tek fazlı kurulumda geçerlidir) ters yöne bakmalıdır. Diğer AC-AC adaptör ve AT kombinasyonları için (AT ve AC-AC trafolarının çalışma şekillerine göre) doğru yönlendirmeyi belirlemek mümkündür, bu şekilde bir kurulum yapmak için en kolay ve hızlı yöntem muhtemelen deneme yanılmadır. AT'yi içerisinden geçen kablo yönünde ters çevirmek o girişin işaretini değiştirir. AC-AC adaptörünü (fiş tasarımının buna izin verdiği yerlerde) (İnt.Kynk.17) ters çevirmek tüm girişlerin işaretini değiştirecektir.

3.7.4 AT 'nin Çalışma Mantiğı

Elektrik akımı taşıyan bir tel, kendi etrafında manyetik bir alan oluşturur. Telden geçen akım sonucu telin etrafında oluşan manyetizma AT içerisindeki transformatörün birincil sargısını indükler. Demir (veya ferrit) transformatör çekirdeği alanı yoğunlaştırır, ikincil sargıya bağlar ve manyetik alanın sürekli değişmesi koşuluyla, bu sargıda da bir akımın akmasına neden olur. Bu akım, projemizdeki devrenin kullanabileceği bir voltaj üreten yük direncine akar. Fakat bir akım trafosu doğrudan akımları toplamaz. Akımların toplamını (daha spesifik olarak vektör toplamını) bilmek istiyorsak, bir akım trafosunun açıklığından birden fazla kablo geçirebiliriz.

Bunun yararlı olduğu iki yaygın durum vardır;

Örneğin; Küçük bir akımımız var ve kesin bir okuma elde etmesi zor. Bu durumda, tel çapı da küçüktür. Bir bobin şeklinde teli sararak ve telin etrafından AT'yi geçirerek akım ölçümünü daha kararlı hale getirmek mümkündür. AT akımı çekirdeğin içinden geçen tur sayısı ile etkili bir şekilde çarpar bu şekilde girişin kalibrasyonunu değiştirerek okuma yapmış oluruz.

Veya farklı devrelerde birçok küçük yükümüz var ve bu devrelerin kullandığı toplam akımı ölçmek istiyoruz. Tüm devreler aynı fazda olmalı ve tüm teller AT'den aynı yönde (Örneğin; Tüm teller güç kaynağından yüke doğru) geçmelidir. Aksi halde bir tel AT içerisinde ters yönde geçerse, o teldeki akım toplamdan çıkarılır.

3.8 Yenilenebilir Enerjide Uzaktan İzleme Yöntemi

Rüzgâr türbinleri, jeotermal enerji, biokütle enerjisi vb. sistemlerin laboratuvar veya ev ortamlarında deneme yapılma amaçlı uzaktan izlemesinin güneş enerjisinden elektrik üretimi sisteminin uzaktan izlemesine göre uygulaması daha zor olacağı için yenilenebilir enerji sistemlerinde uzaktan izlemesi için seçtiğimiz yöntem güneş enerjisi sisteminin uzaktan izlenmesi tarafında olmuştur.

Bunun için kullanılan materyaller;

- Arduino Bazlı Ölçüm Sistemi

- Gerekli Alıcılar
- Linux Tabanlı Web Sunucusu

Bir ve üçüncü maddeler için tek materyalin yeterli olması bizim için şanslı bir durumdu. Arduino işlemlerini yürütürken aynı zamanda Linux tabanlı bir web sunucusu üzerinde çalışabilme potansiyelini sağlayan Arduino Yun kartı, tasarlayacağımız ölçüm sistemi için bu açıdan biçilmiş kaftandır.

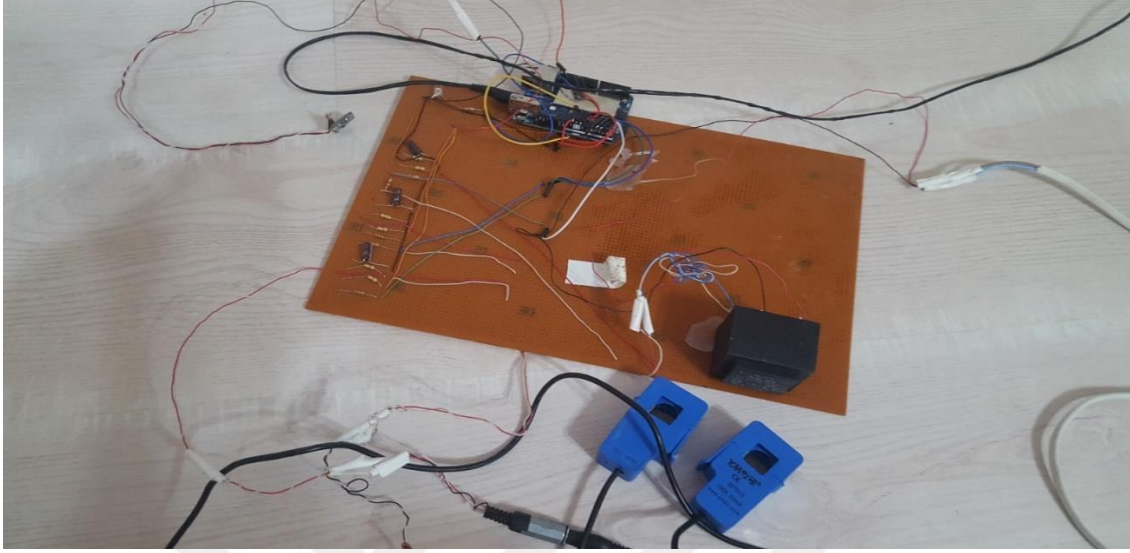
3.9 Solar Panel Uzaktan İzleme Donanımı

Bu adımda kullanılan donanımın detayları anlatılmıştır. Öncelikle farklı alıcıları bağlamak ve elektronik ekipmanları bir arada kullanmak için devre kartının kullanılması mecburidir. Arduino dünyasında bir devre kartı kartın monte edilebildiği bir uzatma materyalidir. Arduino 'nun birçok analog giriş portu, dijital giriş çıkış portu ve kartın yan tarafındaki başlıklarda haberleşme portu vardır. Fakat farklı devre elemanlarını bağlamak için doğrudan bir yer olmadığından devre kartı üzerinden iletişim sağlanması en uygun yöntemdir. Akım ölçümü için 2 adet AT (akım trafosu) gereklidir. Biri güneş panellerinden gelen yani üretilen gücü ölçer, ikincisi ise kullandığımız gücü yani tüketilen gücü ölçer. Gücün hesaplanması içinse bir adet AC-AC gerilim trafosu kullanılır. Ayrıca güneş panellerinin yanında da bir ışık alıcısı planlanmıştır. Işık alıcısı bir i2c seri bağlantısı ile bağlanacaktır. Ancak devre kartının üzerinde bağlı olsa da ışık alıcısı ile ilgili detaylı çalışma yapılmamıştır. Arduino analog giriş portları maksimum beş volt gerilim kabul eder. Bunun için analog sinyallerin bu özelliklere uygun bir devre tarafından uyarlanması gerekir. Arduino analog arabirimi için piyasada bulunan tek AT alıcısı 100A tipi (YHDC SCT-013-000). Genel ev kullanıcılarının kullandığı akımlar 100A'e ulaşmadığından, yük dirençleri gereksinimlere uyacak şekilde hesaplandı. Ana hat için 165 Ohm 'luk bir yük direnci kullanıldı, bu da maksimum ortalama ev tüketimi olan 20A içerisinde ölçüme izin vermektedir. Güneş panelleri için 250W 'lık bir sistem için fazlasıyla 10A'e izin veren 330 Ohm 'luk bir yük direnci seçildi. Yük dirençlerinin hesaplanması bu projede ayrıca anlatılmıştır. Gerilim ölçüm devresi için AC - AC güç adaptörü, 220V / 9V gerilim trafosu kullanıldı ve devre kartı vasıtasıyla Arduino donanımıyla birbirine entegre edildi. Işık alıcısı için Adafruit 'ten bir modül seçildi. Bu parça, ışığın görünen ve IR kısımlarını ölçen iki sensöre sahip olma avantajına sahiptir.

edilmesine gerek yoktur, sadece sensör tek faz bir kablo etrafına tutturulur. Türkiye’de ana sigorta beslemeleri genelde NYM cinsi tek damar kablolarla yapıldığından bu işlem kolay olacaktır. Sadece bir adet kablonun klipslenmesi önemlidir, çok damarlı kablo klipslenirse sensör çalışmaz yahut yanlış çalışır. Daha önceki konularda anlatılan AT kurulum konusunda bahsedilen tavsiyelere uyulması önemlidir.

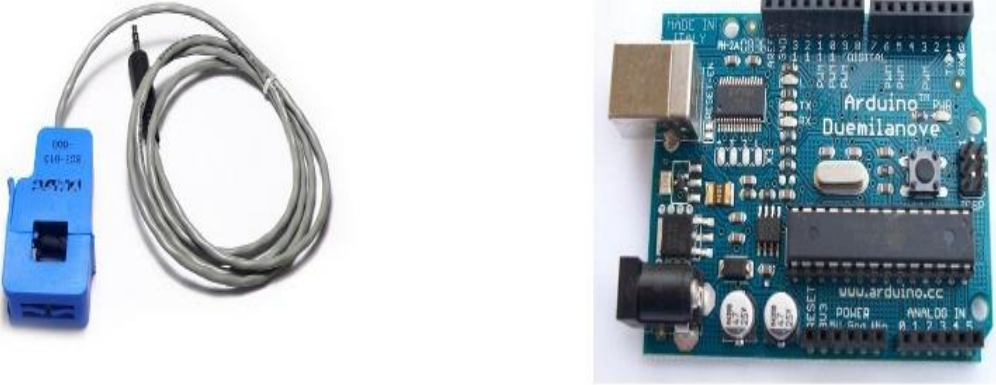
Güneş paneli sisteminden gelen kablolardan bir tanesi etrafına üretim ölçümü yapılacak olan AT klipslenir. Tüketim yapılan ana hat kablolarından bir tanesi etrafına ise tüketim ölçümünü yapan ikinci AT klipslenir. Bu projede kullanılan sistem üç fazlı besleme için kullanılamaz, yalnızca bir faz güneş enerjisi dönüşüm evirici beslemesi ve bir faz şebeke ölçümü için tasarlanmıştır. Ana kablo etrafında sadece bir AT'ye ve güneş enerjisi kaynağında bir AT'ye ihtiyaç bulunmaktadır. Bir sonraki adım Arduino kartı, sensör bağlantı devresi ve voltaj transformatörü için bir yer bulmaktır. Proje deneme yanılma ilerlemesiyle yapıldığından sigorta kutusunun içerisine monte etmek yerine üzerinde çalışması kolay hareket ettirilen bir devre kartı kullanıldı ve sabitleme yapılmadı. Kablolar AT sensörlerinden sigorta kutusunun dışına çıkarıldı. Daha sonra AT sensörlerinden gelen teller devre kartına bağlandı. Gerilim ölçümü için kullanılan transformatör de devre kart üzerine tellerle lehimlenmiştir. AC-AC transformatörün, 9V tarafının bağlantı kartına ve 220V tarafındaki bir fişe bağlanması gerekir. Projemizde standart fişli bir kablo doğrudan transformatöre lehimlenmedi. Arduino kartının beslemesi standart bir mikro USB fişiyle yapıldı. Giren ve çıkan tüm teller ve kablolar yerleştirildikten sonra devre kartı herhangi bir plastik kutu veya yalıtkan herhangi bir kutu ile kapatılabilir. Ölçüm sisteminin gücü 5V'luk bir USB güç adaptörü ile sağlanır. Fakat bu tarz bir projede en ucuz USB şarj cihazını kullanılması önerilmez. Projede düşük gürültü seviyeli istikrarlı bir 5V besleme sağlayan bir güç adaptörü bulmadan önce 4 farklı USB güç kaynağı denenmiştir. Piyasadaki birçok USB şarj cihazı zayıf şekilde tasarlanmıştır. Örneğin; akü şarjı için yüksek kaliteli, düşük gürültülü 5V'ye ihtiyacınız yoktur. Ancak düşük akımları ölçmek için (AT 'ler de yaptığımız gibi) besleme hattındaki gürültü, düşük ölçümlerde ölçümleri çok güvenilmez hale getirebilir çünkü gürültü ölçmek istediğimiz sinyalden daha büyük olabilir. Bu sorun için OpenEnergyMonitor 'de araştırma yapabilirsiniz (İnt.Kynk.14). Bu konuda forumlarında birçok araştırma yaptım. Işık sensörü modülünün kurulumundan ve bağlantısından bahsetmedim. Modül hazır, fakat devre kartı açık bir şekilde beklediğinden herhangi bir hava olayı projeye zarar

verebileceği için şuan için dâhil edilmemiştir. Sistemin kurulumu için tüm şematik yukarıdaki verilmiştir. Evde yapılmış üretim tüketim ölçer resmi aşağıdadır (Şekil 3.8).



Şekil 3.5 Ev tasarımı solar üretim tüketim ölçer.

3.11 Arduino Yun Kartına AT Sensörlerinin Bağlanması



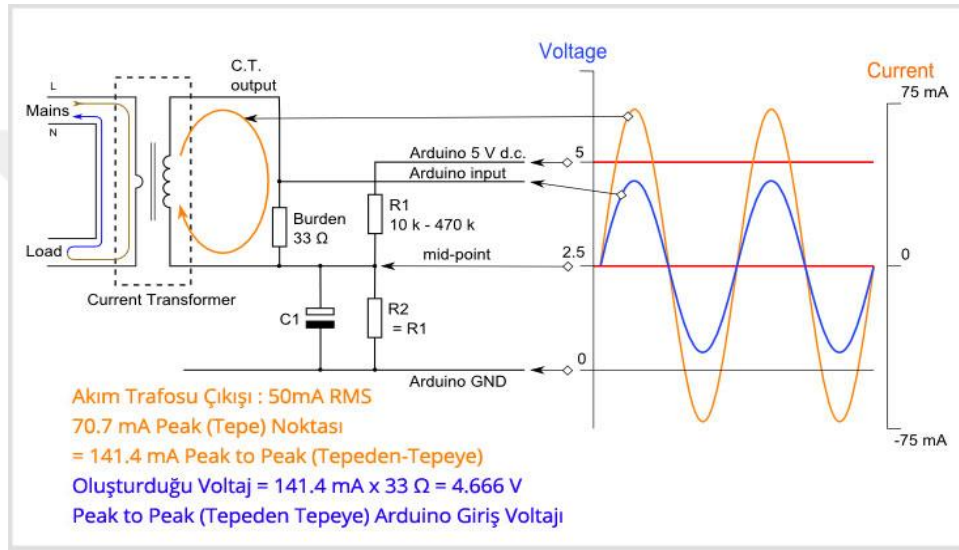
Şekil 3.6 AT sensörü ve Arduino.

Bir AT sensörünü bir Arduino 'ya bağlamak için, AT sensöründen gelen çıkış sinyalinin, Arduino analog girişlerinin gereksinimlerini, yani 0V ile ADC referans voltajı arasındaki pozitif voltajı karşılaması için koşullandırılması gerekir (Şekil 3.6).

Proje 5V 'da çalışan bir Arduino ve devre kartı üzerine tasarlanmıştır. Hesaplamalarınızda ve kurulumunuza uygun doğru besleme voltajını ve ön gerilim voltajını kullandığınızdan emin olun.

Bağlantı iki ana bölümden oluşan aşağıdaki devre ile sağlanabilir (Şekil 3.7):

- AT sensörü ve yük direnci
- Giriş ön gerilim bölücüsü (R1 ve R2)



Şekil 3.22 Arduino kartı AT bağlantısı.

3.12 Uygun Yük Direnci Hesaplamasının Yapılması

AT sensörü, YHDC SCT-013-000 gibi bir "akım çıkışı" türüyse, akım sinyalinin, bir yük direnci olan bir voltaj sinyaline dönüştürülmesi gerekir. Eğer AT sensörü bir voltaj çıkışlı tipi ise, bu adımı atlayabilir ve yük direnci AT'ye yerleştirildiği için devredeki yük direnci kısmını pas geçebilirsiniz (Devredeki burden direnci).

3.12.1 Ölçmek İstedığımız Mevcut Güç Aralığı Seçimi

YHDC SCT-013-000 AT 'nin akım aralığı 0 ila 100A arasındadır. Hesaplamamız için, maksimum akımımız olarak 100 A seçelim.

- 1) R2 ile çarparak maksimum RMS akımını tepe akıma dönüştürelim.

$$\text{Birincil tepe akımı} = \text{RMS akımı} \times \sqrt{2} = 100 \text{ A} \times 1.414 = 141.4\text{A} \quad (3.9)$$

2) İkincil bobindeki tepe akımını vermesi için, tepe akımını AT'deki tur sayısına bölelim.

$$\text{İkincil tepe akımı} = \frac{\text{Birincil Tepe Akımı}}{\text{Tur Sayısı}} = \frac{141.4\text{A}}{2000} = 0.0707\text{A} \quad (3.10)$$

3) Ölçüm çözünürlüğünü en üst düzeye çıkarmak için, tepe akımındaki yük direnci üzerindeki voltaj, Arduino analog referans voltajının yarısına eşit olmalıdır (AREF / 2).

5V'da çalışan bir Arduino kullanıyorsanız: AREF / 2 = 2,5 Volt olacaktır. Böylece ideal yük direnci:

$$\text{İdeal yük direnci} = \left[\frac{\text{AREF}}{2} \right] / \text{İkincil tepe akımı} = 2.5 \text{ V} / 0.0707 \text{ A} = 35.4 \Omega \text{ olmalıdır.} \quad (3.11)$$

Hesapladığımız direnç olan 35 Ω , yaygın olarak kullanılan bir direnç değeri değil. 35 Ω 'un her iki tarafındaki en yakın değerler 39 ve 33 Ω 'dur. Her zaman daha küçük bir değer seçmeliyiz, aksi takdirde maksimum yük akımı AREF' ten daha yüksek bir voltaj oluşturacaktır. Yani 33 $\Omega \pm \%1$ direnç değeri devrede kullanılacak en mantıklı seçim olacaktır. Bazı durumlarda seri halde iki direnç kullanmak ideal yük değerine daha yakın bir değer elde etmeyi sağlayabilir. İdeal olan değer ne kadar düşükse, doğruluk o kadar düşük olur. İşte daha kompakt bir şekilde yukarıdakiler ile aynı hesaplamalar:

$$\text{Yük Dirençleri (Ohm)} = (\text{AREF} * (\text{AT. TUR SAYISI})) / (2\sqrt{2}) * \text{maksimum birincil akımı} \quad (3.12)$$

3.12.2 DC BIAS Ekleme

AT tellerinden birini toprağa bağlayacak ve ikinci telin gerilimini toprağa göre ölçecek olursak, gerilim toprağa göre pozitif ile negatif arasında değişir. Ancak, Arduino analog girişleri pozitif voltaj gerektirir. AT kablosunu toprağa bağlayarak, besleme geriliminin yarısında bir kaynağa bağlamış olduk, bu şekilde AT çıkış gerilimi 2.5 V'nin üstünde ve altında olacak ve böylece pozitif kalacaktır.

Yukarıdaki devre şemasındaki R1 ve R2 dirençleri, 2.5V kaynağı sağlayan bir voltaj bölücüdür. Kondansatör C1 düşük bir dirence sahiptir birkaç yüz Ohm ile birlikte alternatif akımın direnci atlaması için bir yol sağlar. 10 μ F değeri kondansatör için uygundur.

3.12.3 Dirençler (R1 ve R2) İçin Uygun Değer Seçimi

Yüksek direnç, Arduino enerji tüketimini azaltır. Uygulamasını yaptığımız projede, güç tüketimini minimumda tutmak için 470k Ω direnç kullandık. Bu şekilde ölçüm cihazı birkaç ay boyunca pille çalışılabilecek şekilde tasarlandı.

3.12.4 Başlangıç Arduino Yazılımı

Yukarıdaki devredeki RMS akımını sabit bir RMS voltajı ile ölçerek (örneğin 240V) yaklaşık görünür gücü göstermek için, bu Arduino yazılımını kullanabilirsiniz (İnt.Kynk.13), (İnt.Kynk.15).

```
// Görünür Güç Ölçümü

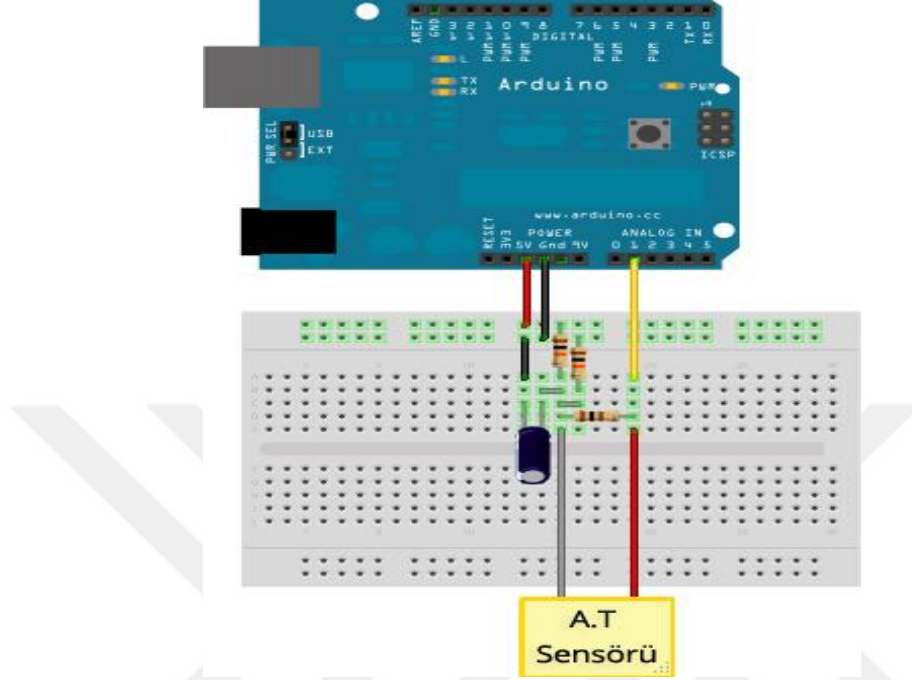
#include "EmonLib.h"                // Emon Kütüphanesini Kullan
EnergyMonitor emon1;               // Bir örnekleme oluştur

void setup()
{
  Serial.begin(9600);

  emon1.current(1, 111.1);          // Akım: Giriş Pini,
  kalibrasyon.
}

void loop()
{
  double Irms = emon1.calcIrms(1480); // Sadece RMS akımını hesapla
  Serial.print(Irms*230.0);          // Görünür Güç
  Serial.print(" ");
  Serial.println(Irms);              // RMS Akımı
```

3.13 Yalnızca Şebeke Akımını Ölçen Arduino Enerji Monitörü Kurulumu



Şekil 3.8 Yalnızca şebeke akımını ölçen enerji monitörü devresi.

Bu kısımda evinizde ne kadar elektrik enerjisi kullandığınızı ölçmek için kullanılabilecek bir devre kartı üzerinde basit bir enerji monitörünün nasıl oluşturulacağı anlatılmaktadır. Bu sistem akımı ölçer, ancak voltaj için varsayılan bir sabit değer kullanır (Türkiye'de iseniz 230V kullanabilirsiniz ya da ölçü aleti ile prizden bir değer alarak, aldığınız değeri kullanabilirsiniz) ve görünür gücü hesaplar. Gerilimin yanı sıra akımı ölçen bir monitör kadar hassas olmasa da, ticari olarak elde edilebilir tüm ev enerji monitörlerinde basitlik ve maliyet nedenleriyle yaygın olarak kullanılan bir yöntemdir. Monitör, akım sensöründen (şebeke akımıyla orantılı bir sinyal üreten) ve sinyali Arduino' nun kullanabileceği bir formata dönüştüren sensör elektroniklerinden oluşur (Şekil 3.8).

3.13.1 Malzeme Listesi

3.13.1.1 Ölçüm Kartı

Çizelge 3.1 Başlangıç devresi için gerekli elektronik kart

1 Adet	Arduino Nano
--------	--------------

3.13.1.2 Elektronik Ekipmanları & Sensörler

Çizelge 3.2 Başlangıç devresi için gerekli elektronik ekipman ve sensörler

1 Adet	Akım Trafosu: YHDC SCT-013-000
1 Adet	Yük Direnci Eğer 3.3V İler Beslenirse 18 Ohm, 5V ile Beslenirse 33 Ohm Olmalıdır.
2 Adet	10k Ohm direnç (veya aynı direnç değerini bulmak için uygun sayıda 470k direnç)
1 Adet	10uF Kondansatör

3.13.2 Kurulum

Bir breadboard ve bağlantılar için birkaç çift kabloyla şemadaki gibi devreyi birleştirebiliriz. Devre şeması için (Şekil 3.8) kullanılabilir. AT sensörünü üzerinden akım geçen yalıtımlı bir monofaze kabloya, kablo AT nin içerisinden geçecek şekilde faz kısmına klipsledikten sonra, Emonlib kütüphaneleri Arduino kütüphanelerine eklenecek, ardından aşağıdaki yazılımın Arduino yazılım yükleme bölümünden çalıştırılması yeterli olacaktır.

3.13.3 Yazılım

```
#include "EmonLib.h"           // Emon Kütüphanesini Kullan
EnergyMonitor emon1;           // Bir Örnekleme Oluştur

void setup()
{
  Serial.begin(9600);
  emon1.current(1, 111.1);      // Akım, Giriş Pini, Kalibrasyon
}

void loop()
{
  double Irms = emon1.calcIrms(1480); // Yalnızca RMS Akımını Hesapla
```

```

Serial.print(Irms*230.0);           // Görünür Güç (230 değeri değiştirilebilir)
Serial.print(" ");
Serial.println(Irms);               // RMS Akımı
}

```

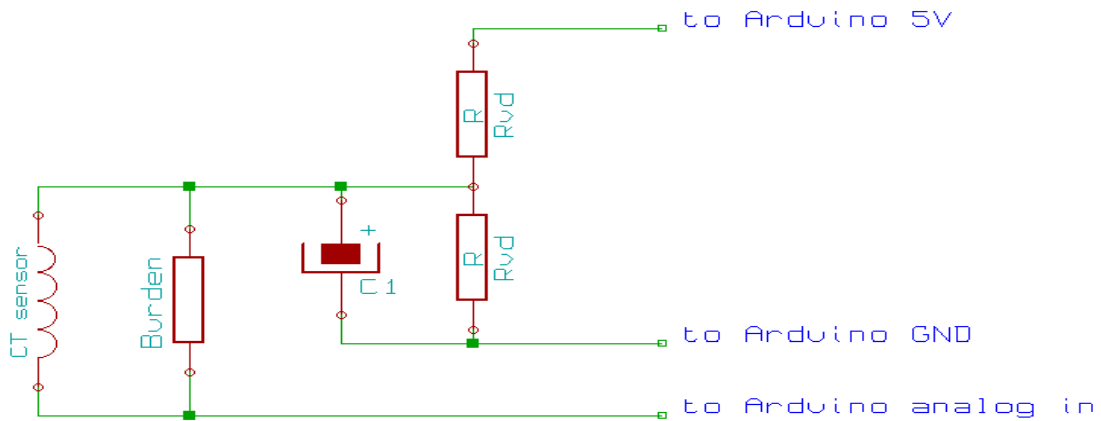
3.13.4 Ölçümlerin Görüntülenmesi

Program yüklenip çalıştırılmasının ardından Arduino 'da ki seri ekran penceresi açıldığında iki ölçüm sütunu görülecektir. Sol kısımda ki görünen güç, sağda ki ise RMS akımıdır.

3.14 Düşük Akım Değerlerinde ADC Çözünürlüğünün Ölçüm Etkileri

- Ölçüm yapılan kabloların üzerinde akım olmadığından neden akım sensörü (X) Amperini okuyor?
- Hiçbir cihaz bağlı değilken neden (Y) Watt değerini okuyorum?

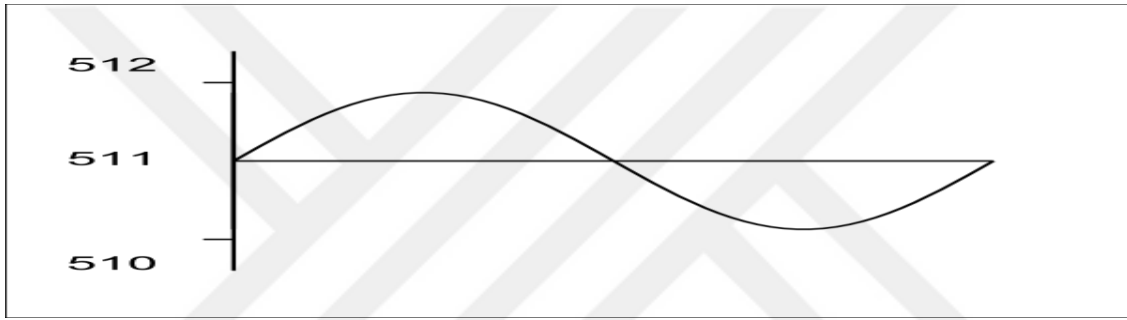
Bunun nedeni, Arduino işlemcisindeki analog dijital dönüştürücünün (ADC) mükemmel çalışmamasıdır (Şekil 3.9).



Şekil 3.23 Arduino AT sensörü basit ölçüm devresi.

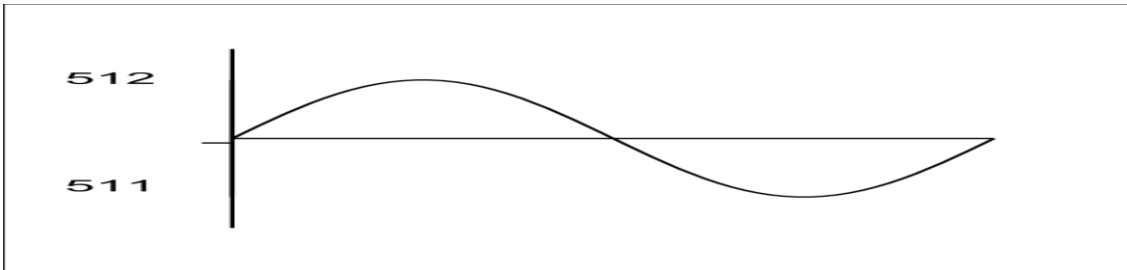
Küçük bir akım olduğunu varsayalım, ön gerilim voltaj bölücü dirençlerinin ortalaması, voltajın tam olarak Arduino ADC giriş aralığının ortasında durmasını sağlar. Bu noktada, analog ile dijital dönüştürücü arasındaki her sayının bir voltajı değil bir voltaj bandını temsil ettiğini bilmek gerekir. Analog dijital dönüştürücü, dalga biçimini birçok noktada ölçecektir. Pratikte yaklaşık 50 dalganın genliği yeterince küçükse, dalganın her örnekleme 511 değerine sahip olacaktır (Şekil 3.10) .

Yüksek geçiş filtresi, dalgayı, ortalama değer sıfır olacak şekilde kaydırır. Bu, her örneklemeden ortalama değer çıkarılmasına eşdeğerdir. Ortalama 511'dir, yani filtre edilen her numune şimdi 0 olacaktır.



Şekil 3.24 Küçük bir AC gerilimi.

Ön gerilim voltajı biraz kayarsa ne olacağını görelim. Şimdi, ortalama değer 511 'den 512 'ye değişimin hemen üstündedir (Şekil 3.11). Bu kez, analog dijital dönüştürücü dalgayı ölçerken, ilk yarı döngüsündeki örnekler (bir veya iki tane) 512 değerini alır ve geri kalanı 511 değerine sahip olacaktır. Örnekleme işlemi sinüs dalgalarımızı kare dalgaya çevirmiştir.



Şekil 3.11 Gerilimin dalga boyu değişimi.

Yazılımdaki yüksek geçiş filtresi tekrar dalgayı değiştirecektir, bu nedenle ortalama değer sıfırdır ve yine bu, her numuneden ortalama değer çıkarılmasına eşdeğerdir. Ancak bu

kez, ortalama 511.6'dır, bu nedenle ilk yarıdaki her örnek $512 - 511.6 = 0.4$ olacak ve ikinci yarıdaki her örnek 511 olacaktır. ($511 - 511.6 = -0.6$)

En önemli kısım yazılım RMS değerini ve dalganın gücünü hesapladığında ortaya çıkmaktadır. Gücü hesaplamak için karşılık gelen numuneler için akım ve voltaj değerleri birlikte çarpılır ve bunun ortalaması güçtür. RMS akımını hesaplamak için, her bir akım numunesi kendisiyle çarpılır, ortalama değer hesaplanır ve RMS akımı bu sayının kareköküdür.

Birinci durumda, her bir akım numunesi sıfır olduğu için, güç ve RMS akımının her ikisi de sıfırdır. İkinci durumda, gerilim ve akım dalga formlarının tam olarak fazda olduğunu varsayarak, ortalama güç [her bir pozitif yarı çevrim voltaj numunesi x 0.4] ve [her bir negatif yarı döngü voltaj numunesi x -0.6] veya yaklaşık 0.45 birim yapar. RMS akımı, ortalama veya $0,4^2 + (-0,6)^2$ 'dir yani 0,72 birimdir. Birim değerleri daha sonra tanımlanacaktır. Uygulamada beklediğiniz rakamları görmek için voltajın 240 olduğunu ve mevcut alıcının standart 18Ω yük dirençli 100A AT olduğunu varsayalım. 100A RMS, yaklaşık 800'lük bir tepeden tepeye sayımı temsil eder, bu nedenle, 1 sayımda tepeden tepeye genliğe sahip olan dalgamız, yaklaşık 0.125 A değerinde bir RMS değerine sahiptir. Bu, 30W 'lık bir gerçek gücü temsil eder. İkinci durumda, 13,5W ve 0,09 A değerlerini okuyabiliriz ve birim güç faktörümüz olmasına rağmen, 21,6 VA görünür gücü ve 0.625 güç faktörünü görürüz.

Ancak, analog dijital dönüştürücü iki değeri çevirmek için yeterli durumda bile olsa akım çok küçük olsa dahi yine aynı sonucu elde ederiz. Yani sonuç olarak pratikte sıfır güç ve akımdan 13,5 W 'lık bir güç ve 0,09 A 'lık bir akıma kadar okuma yapabilmekteyiz. Öte yandan, 30 W 'a kadar herhangi bir yük için de sıfır güç ve sıfır akım okuyabilmekteyiz.

Bu noktaya kadar problemin, Arduino Analog Dijital Dönüştürücü (ADC) girişinin mevcut sensöründen ileri geldiğini varsaydık. Sorun Arduino 'nun kendisi tarafından üretilen dijital parazitlerden veya PCB ya da kablolama tarafından toplanan dış seslerden de ortaya çıkabilmektedir.

3.15 Uzaktan İzleme İçin Arduino Yun Kartının Hazırlanması

Arduino kullanıcılarının en yaygın şikâyetlerinden biri, bağlantı eksikliğidir. Bu donanım üzerinde harika birçok bağlantı yapsak dahi Arduino 'ya bir WIFI bağlantı kartı takmazsak, Internet of Things (IOT) bağlantısını gerçekleştiremeyiz.(İnt.Kynk.4)

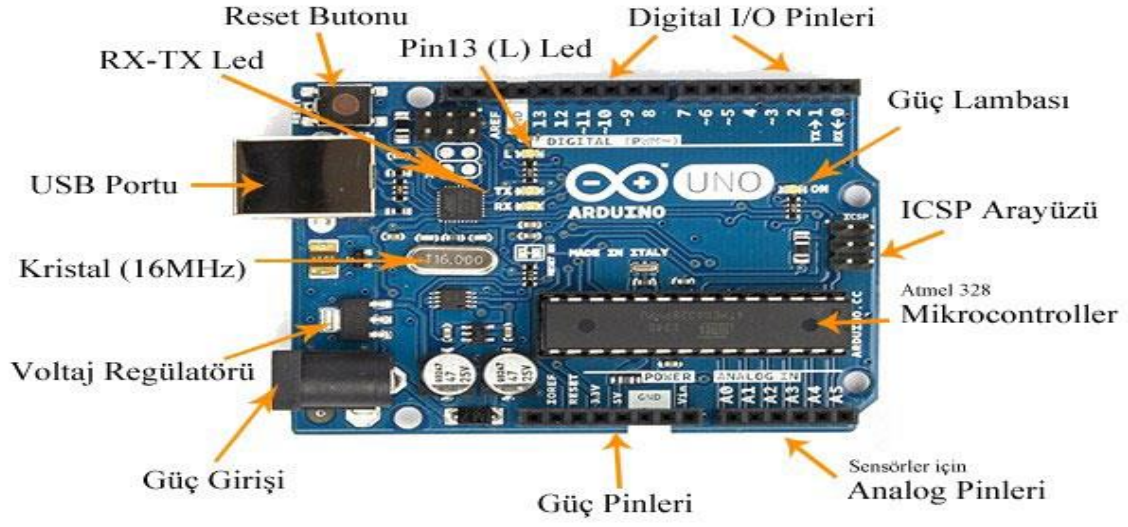
Arduino Yun bu sorunu çözdü. Yun, içerisinde dâhili WIFI kartı bulunan bir Arduino 'dur. Ek olarak, Yun 'da Linux 'un hafif bir sürümünü çalıştıran ve önceden yüklenmiş Python ile birlikte gelen ikinci bir mikroişlemci vardır. Bu, 75 \$ maliyetle sensörler, düğmeler Python komut dosyalarını kullanabileceğimiz ve Python komut dosyalarının LED'leri, motorları ve diğer aktörleri tetikleyebileceği anlamına gelir. Python 'un da beraberinde getirdiği dille birlikte projelerde Ruby, Node veya PHP kullanılabilir.

Ancak değişikliklere başlamadan önce bazı hazırlıklar yapmak gerekir. Bu kısımda bu hazırlıkların detayları anlatılmaktadır:

- Arduino Yun 'da çalışmak için bir SD kartın biçimlendirilmesi
- Yun üzerinde çalışan Linux sürümü olan OpenWRT 'nin yüklenmesi
- Arduino Yun 'un WIFI ağına bağlanması
- SSH bağlantısının yapılması
- Arduino IDE 'nin bilgisayara yüklenmesi
- Arduino programlarının “Merhaba Dünyası” Blink 'in çalıştırılması

Hazırlıkların tamamlanması ortalama 30-45 dakika sürecektir. Hazırlıkları tamamlamak için, bir Arduino Yun 'a (Spark Fun veya Adafruit ~ 75 \$), bir mikro USB kablosuna (~ 5 \$) ve bir mikro SD kartına (~ 8 \$) ihtiyaç olacak.

ARDUİNO KARTI DEVRE ELEMANLARI



Şekil 3.25 Arduino kartının yapısı.

Atmel ATmega32U4, pimleri kontrol eden ve Arduino ile ilişkilendirdiğiniz tüm donanım geliştirmelerini sağlayan Arduino işlemcisidir. Arduino IDE aracılığıyla Yun 'a bir çizim (Arduino kodu) gönderdiğinizde, bu işlemci üzerinde çalışır (Şekil 3.12).

Atheros AR9331: Bu Linux ve WIFI yongasıdır. OpenWRT isimli işletim sistemini çalıştırır, OpenWRT gömülü cihazlar için bir Linux dağıtımdır.

Bağlantı Noktaları: Standart bir USB, bir mikro USB ve bir ethernet bağlantı noktası

Düğmeler: WIFI sıfırlama, Arduino işlemcisi sıfırlama ve Linux işlemci sıfırlama. En sık WIFI sıfırlama kullanılmaktadır.

Hafıza: Harici depolama için mikro SD yuvası

Standart Arduino 'nun aksine, Yun 'un güç için özel olarak adanmış bir güç portu yoktur. Bunun yerine, gücünü mikro USB portundan alır. Yun bilgisayara takılıyken çok fazla geliştirme döngüsü harcanır, ancak aslında WIFI üzerinden veri gönderilmektedir. Satın alınan Arduino 'lar genellikle mikro USB kablosuna gelmezler. Bunun için genellikle mikro USB kablosu ve güç adaptörünü ayrıca almak gerekmektedir.

Arduino 'nun WIFI sıfırlama düğmesini 30 saniyeden uzun süre basılı tutulursa, Yun fabrika ayarlarına sıfırlanır. (OpenWRT yüklüyse yeni sürüm kalır.) WIFI düğmesi 5 saniyeden uzun süre fakat 30 saniyeden kısa süre basılı tutulursa WIFI ayarlarını sıfırlar.

Örneğin; Afyon'da bir ofiste “Yun 'a Başlarken” kılavuzu yazıldığını varsayımsal olarak söyleyelim ve o kış 19 cm kar yağmış olsun, yazarın o gün evden çalışmaya karar verdiğini varsayalım. Yun öncesinde Ofis interneti için ayarlanmış olduğundan, yazar WIFI sıfırlama düğmesini 5 ila 29 saniye arasında basılı tutar, ardından ev WIFI ağına bağlanmak için aşağıdaki adımları uygulaması gerekmektedir.

Yun ayrıca bir Arduino 'da görmeyi beklediğiniz tüm ipuçlarına sahiptir (Şekil 3.13):



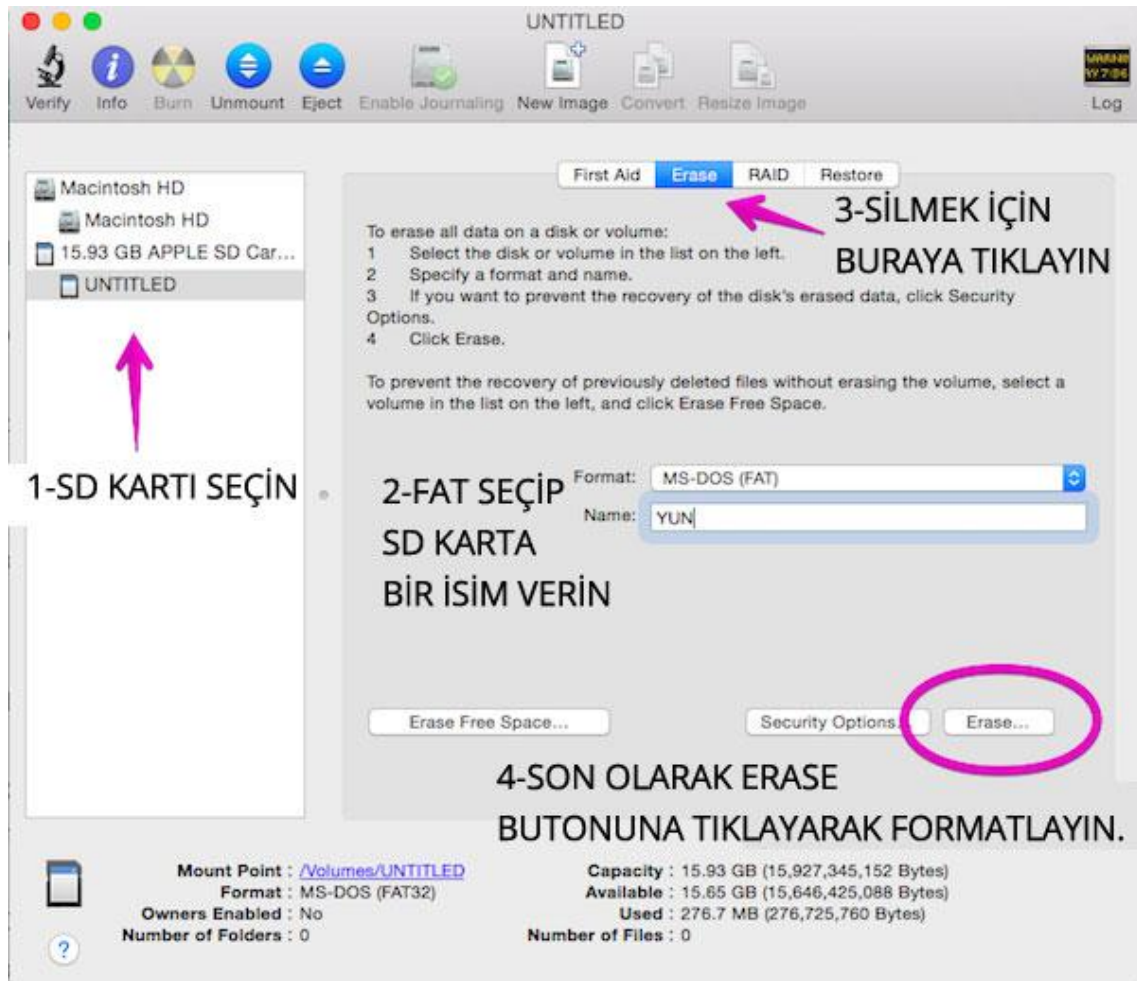
Şekil 3.26 Arduino kartının pin girişleri.

3.15.1 Arduino Yun SD Kartın Hazırlanması

Yun 'unuz ile çalışmak için bir mikro SD karta ihtiyaç olacak. Resmi başlangıç kılavuzunun söylediğine göre, “Yun 'un yerleşik belleğinde belirli sayıda işlem çalışmaktadır ve ekstra işlemler için ekstra belleğe ihtiyaç bulunmaktadır.” Demektedir. Bu yüzden, Yun 'un kendisi ile bir şey yapmadan önce, mikro SD kartı hazırlayalım. Hali

hazırda bir SD kartı kullanıyorsanız, muhtemelen başka bir cihazda kullanılmaktadır. Örneğin; SD kartını bir kameradan çıkartıyorsanız, onun öncelikle FAT dosya sistemine biçimlendirilmesi gerekir (FAT 32 değil!).

OSX 'te bir SD kartı biçimlendirmek için bilgisayara takın (İnt.Kynk.2) (Micro SD Kartı Bilgisayar Üzerinde Formatlamak İçin Bir Micro SD Adaptörüne İhtiyaç Olacak), Disk yardımcı programını açılıp sırasıyla aşağıdaki adımlar takip edilmelidir (Şekil 3.14).



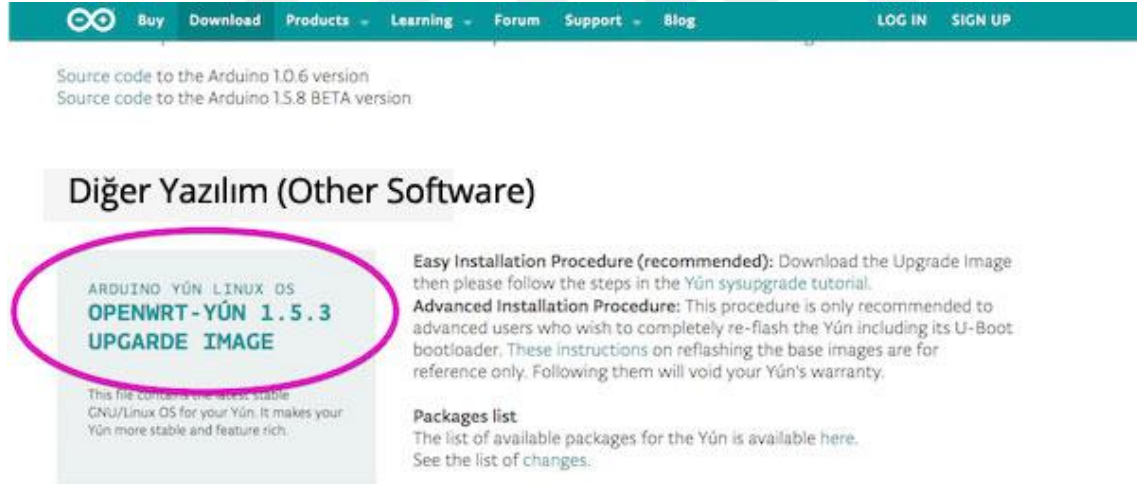
Şekil 3.27 Arduino SD kartının formatlanması.

SD kartı bilgisayardan henüz çıkarılmamalıdır. Yun 'a bağlanmadan önce üzerine bazı yazılımların yüklenmesi gerekmektedir.

3.15.2 OpenWRT-Yun 'a Yükseltme

Yun, OpenWRT-Yun adı verilen hafif bir Linux sürümüyle birlikte gelir. Yun 'un fabrika hattından çıkış yapması ve kullanıcıya teslim süresinde OpenWRT-Yun 'un daha yeni ve daha iyi bir sürümünün çoktan çıkmış olması muhtemeldir. OpenWRT-Yun 'un en son sürümünü kullanmak birçok hatayla karşılaşılmasının önüne geçecektir. (Nedeninin tüm ayrıntılarını bilmek istiyorsak, OpenWRT sürüm notlarına bakmak gerekir (İnt.Kynk.1). Bu işlem genellikle on dakikadan daha az sürer. Son sürüme güncellemek için aşağıdaki adımları takip etmek yeterlidir (Şekil 3.15).

1. Arduino Yazılım sayfası ziyaret edilir
2. Sayfa yarıya kadar kaydırılır
3. OpenWRT-Yun yükseltme imajı için linke tıklanır
4. Dosyayı açılır ve “bin uzantılı dosya” SD kartın kök dizinine kopyalanır.
5. SD kartı çıkartılır ve Yun 'a takılır



Şekil 3.15 OpenWrt yazılımının yüklenmesi.

SD kartında beklemede olan güncel yazılımın yüklenmesi için Yun başlatılır.

3.15.3 WIFI Üzerinden Arduino Yun 'a Bağlantı

Yun daha önce WIFI ağına bağlanmadıysa, kendi WIFI bağlantı noktasını oluşturacaktır. Yun 'un ilk kurulumda oluşturacağı WIFI ağına doğrudan bağlantı sağlanmasının ardından, modem bağlantı ayarları yapar gibi internet tarayıcı üzerinden Yun ip adresine bağlantı sağlandıktan sonra yerel ağ seçimi ve şifre girilmesi yapılır. Bu ayarlar aşağıda adım adım anlatılmıştır.

1. Mikro USB kablusunun küçük ucu Yun 'a ve diğer ucu bilgisayara bağlanır.
2. Yun 'un başlaması için yaklaşık 60 saniye beklenir. Yun hazır olduğunda, beyaz USB LED 'in yandığı görülecektir.
3. Yun WIFI ağı oluşturacaktır. Bu WIFI ağına bağlandıktan sonra, bir tarayıcı açıp tarayıcıya *arduino.local* adresini ziyaret ederiz. Bu işe yaramazsa, 192.168.240.1 adresini girip enter tuşuna basarız (arduino.local bazen bilgisayardaki güvenlik duvarından çalışmamazlık edebilir, ip girerek bağlanmak başlangıç için en doğru yöntemdir). Giriş şifresi içinse arduino veya linino denenmelidir.
4. Geri döneceğimiz birçok ayar göreceksiniz. Şimdilik sadece OpenWRT 'i yükseltme seçeneği göreceğimiz sayfanın en altına kaydırın ardından RESET 'i tıklayın.
5. Yeniden yüklenip yeniden başlatılması için sabırlı kalınarak birkaç dakika beklenmesi gerekmektedir. Bu aşamada kartın USB girişi çekilmemeli herhangi bir işlem yapılmamalıdır.

3.15.4 Arduino Yun 'u WIFI Ağına Bağlamak

Arduino Yun yeniden başladıktan sonra;

1. Arduino YUN-ABCDEFGH, WIFI ağına hala bağlı olduğundan emin olun
2. Arduino.Local veya 192.168.240.1'i tekrar ziyaret edin
3. Sağ üstteki yapılandır düğmesine tıklayın

Ardından bunları yapabiliriz;

Öncelikle Yun 'un adını ve şifresini değiştirebiliriz. Arduino 'yu bir WIFI ağına bağlayın. Açılır pencereden WIFI ağı seçilmeli ve şifre dikkatlice girilmelidir çünkü şifre yanlış girilirse bir hata mesajı alınmaz. WIFI ağı seçilerek şifre girilmesinin ardından Yun yeniden başlatılır, ancak yeniden başladıktan sonra Örneğin: Arduino Yun-ABCDEFGH

gibi bir WIFI ağını bir seçenek olarak görüyorsak, yanlış WIFI şifresi girilmiş olması ihtimali yüksektir (Şekil 3.16).

YÜN BOARD CONFIGURATION ⓘ **Arduino İsminizi Girin**

YÜN NAME *

Şifreyi Girin PASSWORD

Şifreyi Tekrar CONFIRM PASSWORD

Girin TIMEZONE * Zaman Dilimini Girin
Türkiye için UTC + 3

WIRELESS PARAMETERS ⓘ

CONFIGURE A WIRELESS NETWORK ☒

DETECTED WIRELESS NETWORKS Refresh **WİFİ AĞINI SEÇİN**

WIRELESS NAME * **Arduino Wifi Adını Girin**

SECURITY

PASSWORD *

Yeni Arduino Wifi Şifresini Girin

Şekil 3.28 Arduino Yun WIFI bağlantısı.

Tüm ayarlar istenildiği gibi olduğunda, Yapılandır ve Yeniden Başlat 'ı (CONFIGURE & RESTART) tıklayın. Beklerken, dizüstü bilgisayarınızı Arduino Yun 'a az önce girilen WIFI ağına getirin ve orada size katılmasını bekleyin. Bir kez tekrar çalışmaya başladıktan sonra, ayarlar tamamlandı demektir.

Bir terminal açıp,

```
ssh root@arduino.local
```

Yazılır (veya Yun adını değiştirdiyseniz, root@YeniArduinoYunAdı.local)

Yun' u bilinen bir ana bilgisayar olarak eklemek için “YES” yazılmalıdır.

Şifre girildiğinde ve işlem tamamlanacaktır.

3.15.5 Arduino Yun Disk Alanının SSD Üzerine Genişletilmesi

OpenWRT tabanlı ve Yún üzerine kurulu GNU / Linux işletim sistemi OpenWRT-Yun, dâhili flash belleğin mevcut 16 MB 'ının 9 'unu alır. Bu, çok sayıda yazılım paketi veya uygulaması yüklüyorsanız, alanın hızlıca tükenebileceği anlamına gelir.

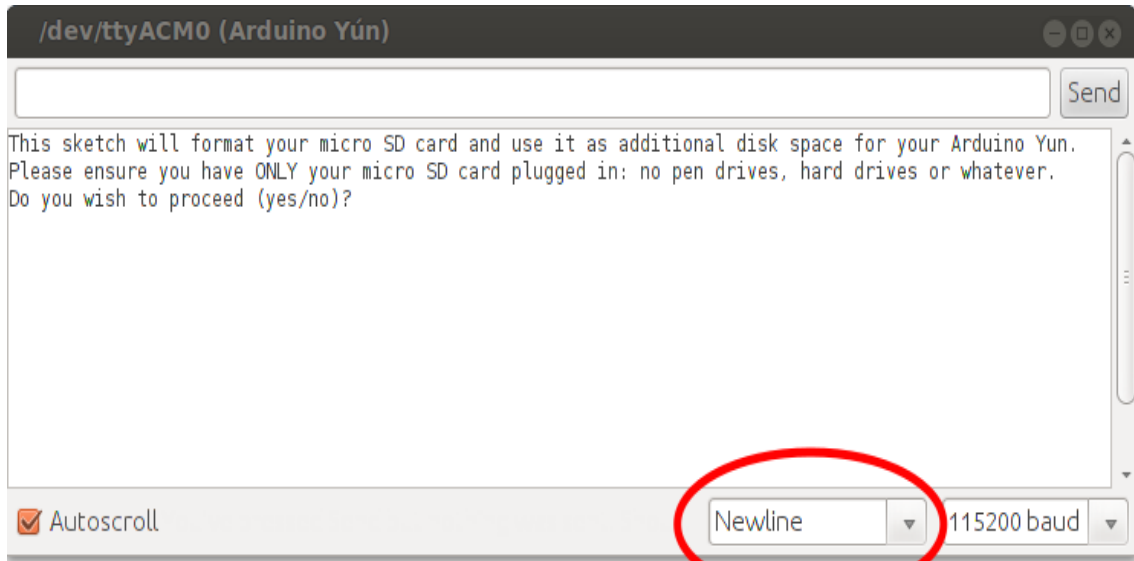
Depolama alanını artırmak için, Arduino Yún 'a bir mikro SD kart takıp yeni Linux dosya sisteminiz olacak şekilde yapılandırabilirsiniz. Yapılandırıldıktan sonra, Yún 'un Linux tarafında yapacağınız her şey SD 'de saklanacaktır.

Yún 'u SD kartı bir kök dosya sistemi olarak tanıyacak şekilde yapılandırma prosedürü karmaşık değildir, ancak yine de temel bir Linux bilgisine ihtiyaç duyulmaktadır. Bu yüzden tüm adımları otomatik hale getirdik ve bir çizim yaptık. Çizim, Yún ve SD kartı yapılandırmak için Bridge kitaplığını kullanır. Sadece SD kartı takın ve Seri Monitördeki etkileşimli adımları izleyin.

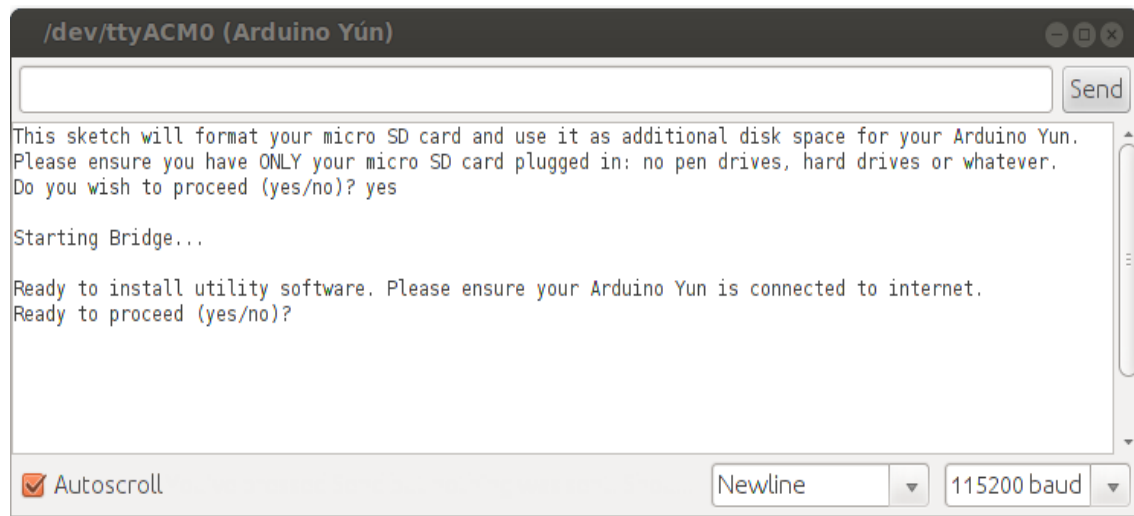
Çizim SD kartı iki parçaya böler. İlk bölüm FAT32 olarak biçimlendirilecek ve herhangi bir bilgisayardan okunabilecektir: bilgisayarınızla arasında dosya paylaşmak için kullanabilirsiniz. İkinci bölüm Linux dosya sistemini içerir. Yüklemiş olduğunuz her yazılım ve her bir yapılandırma dosyası SD karta kopyalanacaktır, bu nedenle herhangi bir şeyi yeniden kurmanız veya yapılandırmanız gerekmeyecektir: Bu işlemden sonra Yún ilk halinden farklı olarak, çok daha fazla disk alanına sahip olacaktır. Windows veya Mac kullanıyorsanız, bu ikinci bölümün bilgisayarda görünmeyeceğini ve erişilemeyeceği unutulmamalıdır.

Yún 'u bilgisayarınıza bağlayın ve bir dakika kadar bekleyin. Yún 'un internete bağlı olduğundan emin olun, çünkü prosedür bazı ek yazılımlara ihtiyaç duyar. Yún 'a takmış olabileceğiniz diğer tüm cihazları çıkarın. Bir mikro SD kart takın. Çizimin bulunduğu zip dosyasını indirin (<https://www.turktiger.com/yungenislet.zip>), açın ve sıkıştırılmış klasörü Arduino sketchbook klasörünüze kopyalayın. Arduino IDE 'yi açın ve

YunDiskAlanıGeniřletici taslađını y kleyin. Arduino IDE ‘den seri izleyiciyi a ın ve a ılır men den "Newline" se ildiđini iki kere kontrol edin. Seri monit r  a tıđınızda, bir mesaj sizi Y n 'a bađlamıř olabileceđiniz diđer herhangi bir yıđın depolama aygıtının fiřini  ekmeniz konusunda uyarır. Y n 'a sadece mikro SD kart takılmalıdır. "YES" yazın ve devam etmek istiyorsanız “ENTER” tuřuna basın. İřlemin ilk adımı, SD kartı bi imlendirmek i in gereken yazılımı kurmaktır. "YES" yazmadan  nce, Y n 'un internete bađlı olduđundan emin olun, aksi takdirde kurulum bařarısız olacaktır (řekil 3.17).

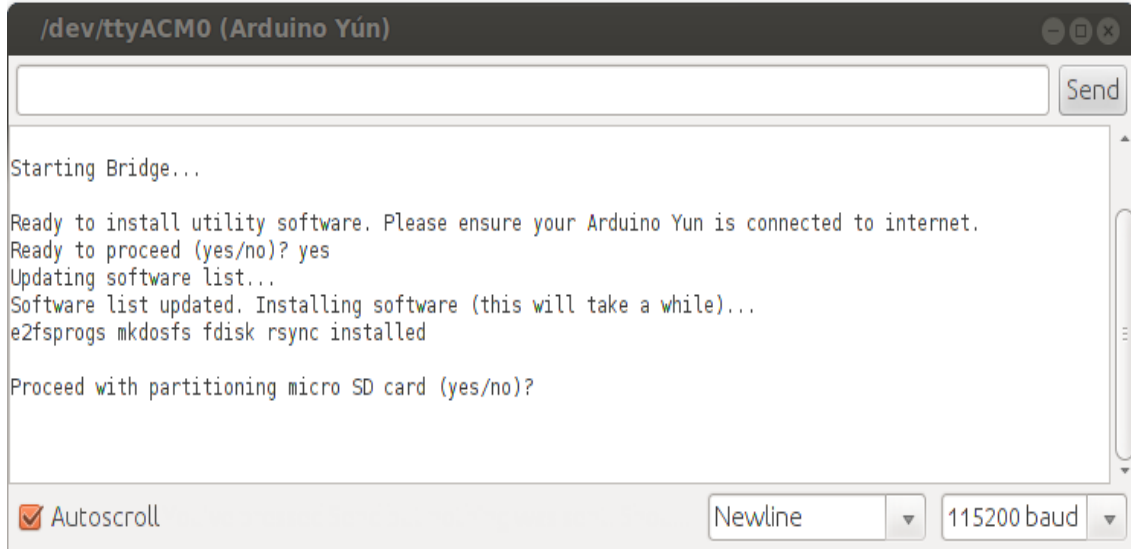


řekil 3.17 SD kartın geniřletilmesi birinci adım.



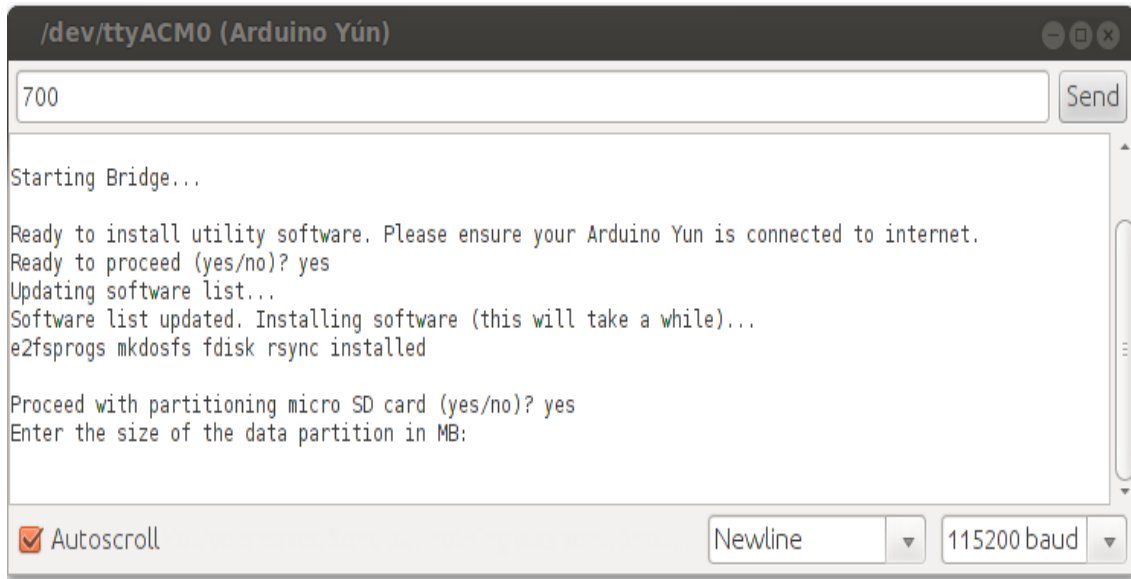
řekil 3.18 SD kartın geniřletilmesi ikinci adım.

Yukarıdaki ekranı görüyorsanız ek yazılım doğru yüklenmiştir. SD kartını biçimlendirmeye başlamak için "YES" yazın ve "ENTER" tuşuna basın (Şekil 3.19).



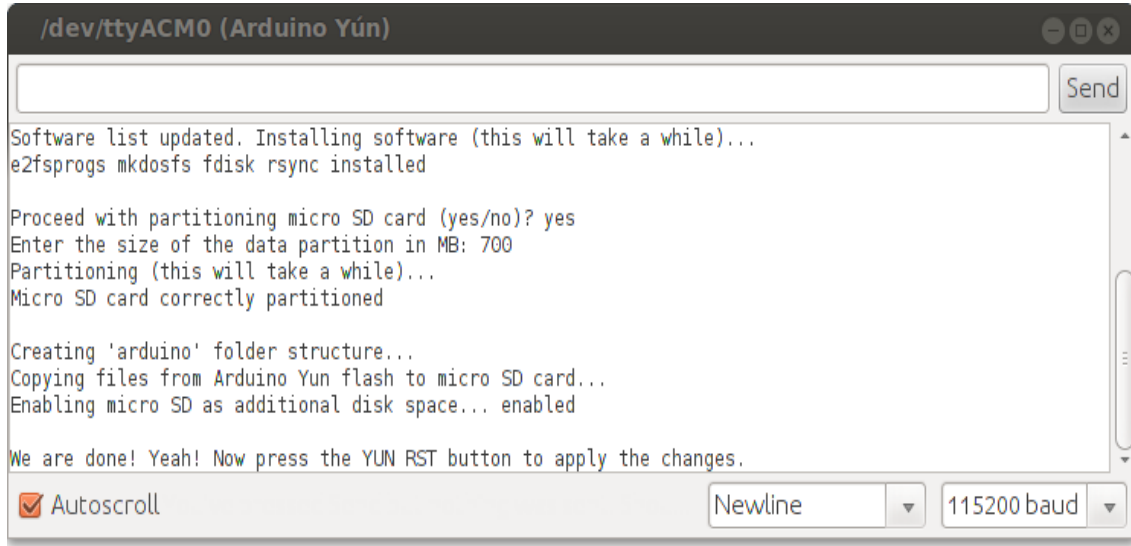
Şekil 3.19 SD kartın genişletilmesi üçüncü adım.

Bilgisayarınızla dosya paylaşmak için kullanacağınız SD kartın ilk kısmının boyutunu belirtin. SD 'nin kalan kısmı Linux dosya sistemine tahsis edilecektir. Bu örnekte 2 GB SD kart kullanıldı ve ilk bölüme 700 MB bırakılmasına karar verildi. Kalan 1.3 GB, Linux 'a adanmıştır (Şekil 3.20).



Şekil 3.29 SD kartın genişletilmesi dördüncü adım.

Bir kez yapıldığında, önceki tüm konfigürasyonunuz ve dâhili flash bellekte depolanan dosyalar SD kartın ikinci kısmına kopyalanacaktır. Tüm değişiklikleri uygulamak için Yún 'u yeniden başlatmanız gerekir. “YUN RST” düğmesine (Ledlerin yanındaki) basılmalıdır. (Şekil 3.21).



Şekil 3.30 SD kartın genişletilmesi beşinci adım.

Çizim, Yun Linux kabuğunu göstermeye devam edecek ve Yun 'un başladığını belirten çok sayıda mesaj görülecektir. Önyükleme bittiğinde şu yazılmalıdır:

```
df -h /mnt/sda1
```

Çıktısı şu şekilde olmalı:

Dosya Sistemi	Boyut	Kullanılan	Kullanılabilir	Kullanım%	Mounted
rootfs	1.2G	51.7M	1.0G	5%	/
/dev/sda1	698.6M	12.0K	698.6M	0%	/mnt/sda1

Görüldüğü üzere 1,0 GB kullanılabilir boş disk alanı var. Eğer OpenWRT 'de iseniz, kurduğumuz şey “EXTROOT” olarak adlandırılır. Bu konuyla ilgili ek bilgiler OpenWRT 'de bulunabilir.

3.15.6 Ek Yazılımların Yüklenmesi

OpenWRT-Yun'un standart kurulumunda bu uygulama için ihtiyaç duyduğumuz bazı işlevler eksik olarak gelmektedir. Verilerin depolanması için Sqlite, iletişim için PHP5 kullanıyoruz ve kayıtlı verileri görüntülemek için GD kullanıyoruz. Bu modüllerin bazıları eksik ve kurulması gerekiyor. Bunu yapmak için Linux sistemiyle SSH bağlantısına ihtiyacımız var. Bazı dosyaları aktarmak için bir SFTP bağlantısı da yardımcı olacaktır. Geliştirme sistemimiz Windows olduğu için, hem SSH terminalini hem de SFTP istemcisini sağlayan Bitwise SSH istemcisini kullanıyoruz. Linux veya MacOS kullanıyorsanız, benzer bir araca ihtiyacımız vardır.

İlk önce Arduino Yun ile bir SSH bağlantısı açıp, ardından gerekli paketler OPKG ile kurulmalıdır:

```
opkg update
opkg install openssh-sftp-server

opkg install php5 php5-cgi php5-mod-gd php5-mod-json php5-mod-pdo php5-mod-pdo
-sqlite php5-mod-sqlite3

opkg install libsqlite3 sqlite3-cli
```

Kurulumdan sonra PHP ve UHTTDP sunucusu için bazı ayarları değiştirmek gerekir. /ETC/PHP5.ini dosyasını düzenleyip ve aşağıdaki uzantıları etkinleştirilmelidir:

```
extension=gd.so
echo "extension=json.so"

echo "extension=pdo_sqlite.so"

echo "extension=sqlite3.so"
```

/Etc/config/uhttpd komutunu da düzenleyin ve aşağıdaki satırları değiştirin:

```
list interpreter ".php=/usr/bin/php-cgi"
echo "option max_requests 10"
```

Bundan sonra aşağıdaki komutu kullanarak web sunucusunu yeniden başlatın.

```
/etc/init.d/uhttpd restart
```

Veritabanı oluşturmak:

En kolay yol, “CREATE.SQL” dosyasını SFTP ile SD kartın kök dizinine kopyalamaktır. Dosya sitede yayınlanmıştır (turktiger.com/ekizleme). Ardından SSH kullanılarak aşağıdaki komutları girin:

```
cd /mnt/sda1  
sqlite3 yeni.db < create.sql
```

Bu, SD kartın kökünde boş bir SQLITE veri tabanı oluşturur. Veri tabanında “yeni” adında bir tablo ve adında dört sütun vardır.

- id => kayıt kimliği
- d => kaydın tarih ve saat damgasını içeren metin alanı
- s => güneş paneli tarafından üretilen güç
- c => evin harcadığı güç
- l => ışık değeri

Bu veri tabanı AVR (Yun İşlemcisi) mikro kontrolöründen gelen kayıtlarla doldurulacak, ardından veri tabanına kayıtlar aktarılacak veya görüntülenmek için PHP sayfaları tarafından erişilecektir.

3.15.7 PHP Dosyaları

Arduino taslağı panele yüklendiğinde, yerel bir ağdan verilere erişmek veya verileri bir web tarayıcısında görüntülemek için PHP sayfaları otomatik olarak yüklenir. Bunu başarmak için PHP sayfalarının Arduino taslak dosyalarının kaydedildiği yolda www.turktiger.com/ekizleme klasörü kopyalanırsa dosyalar otomatik olarak doğru klasöre yerleştirilecektir. Yerel ağdan gelen verilere erişmek için Linux tarafında yapılması gerekenler bu kadar. PHP sayfalarında neler yapıldığını görmek için bir sonraki adıma geçelim (Int.Kynk.7).

Sistem, veri iletişimi ve gösterimi için 3 PHP sayfası ve bir grafik kütüphanesi kullanır.

3.15.7.1 Php Info

Bu PHP sayfası sadece Linux sunucusundaki PHP konfigürasyonunun doğru kurulup kurulmadığını kontrol amaçlı çalışmaktadır. Günlük işlem için gerekli değildir. “Yun IP Adresi/phpinfo.php” açıldığında (Yun IP Adresi Arduino kartına atanan IP adresi olacaktır) standart PHP bilgilerini görebilirsiniz. Önemli olan, aşağıdaki paketlerin kurulu ve mevcut olmasıdır:

- PHP Sürüm 5.6.6 (sürüm 5.5' ten daha yeni sürüm olmalıdır)
- GD Desteği - etkin
- GD Sürümü - (2.1.0 uyumlu)
- JSON desteği - etkin
- PDO sürücüleri - sqlite
- SERVER_SOFTWARE – uHTTPd

3.15.7.2 Index

Bu projeye başladığımızda, asıl hedefimiz kayıtlı verileri görüntülemek için Android telefonlar / tabletler kullanmaktı. Ancak ilk testler için kaydedilmiş verileri bir web tarayıcısında gösteren basit bir web sayfası oluşturuldu. Bu okumayı gerçekleştiren index.php dosyasıdır. Web sayfasının yapısı çok basittir. En üstte, güne göre düzenlenmiş tüm mevcut kayıtları gösteren bir seçim kutusu bulunur.

Her gün için 2 ayrı seçenek vardır:

Yıl-Ay-Gün-Dolgulu => Gün, o günün tüm kayıtlarını, Yıl, yılı Ay, ayındaki bir grafikte görüntüler. Yıl-Ay-Gün-yakınlaştırılabilir => Herhangi bir günde ay veya yıldaki tüm kayıtları yatay ve dikey olarak yakınlaştırılabilen ve uzaklaştırılabilen bir grafikte gösterir. (Farklı grafik tipleri için resimlere bakınız).

Listedeki bir linke tıklandıktan sonra görüntülenecek tarih örneğin; PHP = 18-08-19 parametresi ile gönderilir. İkinci parametre türü = yakınlaştırılabilir (veya dolgu), hangi grafik türünün görüntüleneceğini tanımlar. Grafiği çizmek için Fusion Charts

kütüphanesini PHP işleyicisiyle birlikte kullanıyoruz. Fusion Charts ücretsiz (sınırsız) deneme sürümü sunar. Her grafikte yalnızca bir filigran gösterilir. Ancak ticari uygulamalarda kullanmak istiyorsanız bir lisans satın almanız gerekir. Kütüphanenin nasıl kullanılacağına dair ayrıntılara girilmeyecektir. Grafiği nasıl oluşturduğumuzu görmek için Index PHP dosyasını kontrol edebilirsiniz.

3.15.7.3 Query

Bu PHP sayfası uygulamadan Arduino kartındaki SQLITE veritabanına bir sorgu göndermek ve çıktıyı JSON dizesi olarak almak için kullanılır. Sorgu “IPAdresi / query.php? Date = param1 & get = all” çağrısı ile başlatılır (IPAdresi 'ni Arduino kartınıza atadığınız IP adresi ile değiştirin). “Date” ilk parametresi sorgu sınırlandırıcısını ayarlar:

- yy- ~> Tüm yıl
- yy-mm -> Tüm yıl ve aylar
- yy-aa-dd => Tüm Yıl Ay ve Gün
- yy-mm-dd-hh => Tüm Yıl Ay Gün ve Saat
- yy-mm-dd-hh:MM=> Tüm Yıl Ay Gün Saat ve Belirli Dakika

İkinci parametre “get = all”, betiğe “tarih” te tanımlanan yeni girişlerin de gönderilmesi gerekip gerekmediğini bildirir.

PHP betiğinin kilit noktaları şunlardır:

```
$ query = $ query. "Where d Like '". $ dateSelect. "% ";
```

"d" nin “\$dateSelect” alt dizinini içerdiği tüm satırları alır.

ilk sorguda bulunan son satırdan daha yeni olan tüm satırları alır.

Veritabanından bu iki sorgu tarafından gönderilen veriler daha sonra bir JSON dizisi olarak kodlanır ve talep eden kullanıcıya geri gönderilir.

```
echo json_encode($json);
```

Daha fazla ayrıntı görmek için Query PHP dosyasını inceleyebilirsiniz.

3.16 Arduino Yazılımı

Arduino Yun, yazılım için yalnızca 28,627 bayt ve global değişkenler için 2560 bayt veri alanı sunar. Yun için hızlı biçimde öğrenilmesi gereken şey, kod belleğinin % 99,9'a kadar kullanılabildiğidir. Ancak genel değişkenler için belleğin % 60'ından fazlası kullanılırsa eskiz artık güvenilir bir şekilde çalışmaz. Underlaying kodunun (bootloader...) global değişkenler belleği için biraz boş alan gerektirdiği düşünülmektedir. Bu yüzden kodu optimize etmek ve mümkün olduğunca global değişkenlerin kullanımı sınırlanmak zorunda kalındı. Kod için Arduino taslağı birkaç dosyaya bölündü. Kod olabildiğince iyi bir şekilde yorumlanmaya çalışıldı, bu yüzden anlatımın açıklayıcı olduğu kanaatindeyiz.

- Ekizleme.ino => Diğer dosyaların açıklaması ile info dosyası
- Ekizleme _a_ Tanımlar => Include (İçerik) ve global değişkenler
- Ekizleme _b_ İşlevler => Yardımcı işlevler
- Ekizleme _c_ Isiksensoru => TSL2561 ışık sensörüne arayüz
- Ekizleme _x_ Olcumleri => Ölçüm fonksiyonları
- Ekizleme _y_ Kurulum => Arduino kurulum fonksiyonu
- Ekizleme _z_ Dongu => Arduino döngü işlevi

Dosyaların adları belirli bir şema izlemektedir. Taslak için derleyici, kullanılan alt işlevlerin kullanılmadan önce bildirilmesini gerektirir. Yani dosya isimleri DosyaAdı_Index_Modül ile derlenecek dosyaları doğru sırada derleyici tarafından işlenmesi gerekmektedir.

3.16.1 Ekizleme _a_ Tanımlar

- Birinci bölümde, farklı fonksiyonlar için gerekli tüm harici dosyalar yer almaktadır.
- Wire.h => I2C iletişim kütüphanesi
- Adafruit_TSL2561_U.h, PgmSpace.h, Adafruit_Sensor.h => Işık sensörü için,
- EmonLib.h => AT ve voltaj sensörleri için,

- FileIO.h => Linux bölümünün SD kartına erişmek için
- BridgeServer.h ve BridgeClient.h => Web iletişimi için
- Avr / wdt.h => Bekçi uygulaması için

Kod Parçası:

```
#include <Wire.h>

#include <Adafruit_Sensor.h>
#include <Adafruit_TSL2561_U.h>
#include <pgmspace.h>
#include <EmonLib.h>
#include <FileIO.h>
#include <YunServer.h>
#include <YunClient.h>
#include <avr/wdt.h>

/** Ölçümler İçin Geri Sayım */
unsigned long lastMeasure;
/** Kayıt İçin Geri Sayım */
unsigned long lastSave;
/** Yeniden Başlatma İçin Geri Sayım */
//unsigned long lastReset;
/** Veritabanı İçin Kullanılan Mevcut Ay */
unsigned int thisMonth = 0;
/** Etkinlik LED portu için sabit değer */
#define activityLED 8
/** Ms cinsinden ölçüm frekansı için sabit değer */
#define measureFreq 5000
/** Adafruit TSL2561 Sensörü Mevcut Durumu */
Adafruit_TSL2561_Unified tsl = Adafruit_TSL2561_Unified (
TSL2561_ADDR_FLOAT, 1 );
/** Işık sensörü için şu anda kullanılan entegrasyon süresi, 0 = 13.7ms, 1 =
101ms, 2 = 402ms */
int lightInteg = 2;
```

```

/** Dakikada ortalama ıřık sensörü için kollektör*/
long collLight = 0;
/** CT sensörleri için emon'a bir örnek oluřtur*/
EnergyMonitor emon[2];
/ ** CT 1 için iCal tanımı (güneř enerjisi) * /
#define iCal1 5.7
/** CT 2 (řebeke) için ical tanımı */
#define iCal2 11.5
/** Gerilim (faz) için faz kayması tanımı*/
#define pShift1 1.3
/** Gerilim için faz kayması tanımı (řebeke) */
#define pShift2 6.1
/ ** Voltaj ölçümü için vCal tanımı * /
#define vCal 255
/ ** Dakikada ortalama CT sensörleri için kollektör
[0] = solar CT sensor
[1] = mains CT sensor */
double collPower[2];
/ ** Sensörlerden dakikadaki ölçümler için sayaç
[0] = solar CT sensor
[1] = mains CT sensor
[2] = light sensor */
int collCount[3];
/ ** Yún web sunucusu olan varsayılan 5555 numaralı baėlantı noktasını dinle
/** göndereceėiniz tüm HTTP isteklerini iletir * /
YunServer server;
/ ** Gelen ve giden HTTP istekleri için müşteriinin örneėi * /
YunClient client;

```

3.16.2 Ekizleme_b_İřlevler

Bu kod kaydedilmiş deėerleri saklamak için kullanılan 2 fonksiyon ięerir.

GetTimeStamp(), sistemin geçerli tarih ve saatini döndürür. Bunu elde etmek için Linux tarafında bir komut çalıştırmak için “köprü” işlevini “.run” kullanıyoruz ve çıktıyı “köprü” üzerinden alıyoruz.

Void SaveData(), Bir Linux kabuğundaki komutları ve komut dosyalarını başlatma olanağı, Arduino kartının özelliklerini genişletir. AVR kontrol cihazının sınırlı program alanına karmaşık fonksiyonlar eklemek yerine, bunlar Linux kısmına devredilebilir.

Ölçümler her saniye yapılır, ancak değerler sadece her dakika saklanır. Ölçülen değerler geçici hafızada depolanır. Kod parçası:

```
String getTimeStamp () {  
    /** Dönülecek Tarih İçin Dize */  
    String result;  
    /** Linino işlevinin durumu */  
    Process time;  
  
    /* date, tarih ve saati almak için kullanılan bir komut satırı yardımcı  
programıdır*/  
    /* ek parametreye bağlı olarak farklı formatlarda*/  
    time.begin ( "date" );  
    time.addParameter ( "+%y,%m,%d,%H:%M" );  
    wdt_reset();  
    time.run(); /* komutu çalıştır */  
    /* komut çıktısını oku */  
    while ( time.available() > 0 ) {  
        /** Linino 'dan döndürülen verinin karakteri */  
        char c = time.read();  
  
        if ( c != '\n' ) {  
            result += c;  
        }  
    }  
}
```

```

        return result;

    }

    /* Bu işlev, ortalama 1 dakikalık ölçümleri SD karttaki günlük
dosyasına kaydeder. */

    void saveData () {
        /** Kaydedilecek sensör log dosyası */
        String dataString;
        /** Sensor logU için zaman damgası */
        String timeString;
        /** Son kaydedilmeden bu yana toplanan ışık değeri */
        long light = 0;
        if ( collCount[2] != 0 ) {
            light = collLight / collCount[2];
        }
        /** Son kaydedilmeden bu yana toplanan güneş enerjisi değeri */
        double solar = collPower[0] / collCount[0];
        /** Son kaydedilmeden bu yana toplanan tüketilen güç değeri */
        double cons = collPower[1] / collCount[1];
        Bridge.put ( "L", String ( light ) );
        Bridge.put ( "S", String ( solar ) );
        Bridge.put ( "C", String ( cons ) );
        /* Mevcut veriyi sqLite veritabanına yaz */
        /** Linino işlevleri durumu */
        Process dataSave;
        timeString = getTimeStamp();
        timeString.replace(',', '-');
        /** Rakam Olarak Ay */
        int nowMonth = timeString.substring(3, 5).toInt();
        /* Ay değişikliği veya yeni uygulama başlangıcı olup olmadığını
kontrol et*/
        if (thisMonth != nowMonth) {
            // uygulamanın yeni başlangıcı veya yeni ay başladı

```

```

thisMonth = nowMonth;
// yeni ay için yeni veritabanı oluştur
dataString = "sqlite3 /mnt/sda1/"
    + timeString.substring(0, 5)
    + ".db < /mnt/sda1/create.sql";
dataSave.runShellCommand ( dataString );
}

dataString = "sqlite3 /mnt/sda1/"
    + timeString.substring(0, 5)
    + ".db 'insert into s (d,s,c,l) Values ('"
    + timeString + "\", "
    + String ( solar ) + ", "
    + String ( cons ) + ", "
    + String ( light ) + " );' &";
dataSave.runShellCommand ( dataString );
wdt_reset();
/* Mevcut Veriyi emonCMS ye gönder*/
dataString = "curl
\"http://emoncms.org/api/post?apikey=e778b92fc1f06d7e94a94bcc2a969664&j
son={s: ";
dataString += String ( solar );
dataString += ",c: ";
dataString += String ( cons );
dataString += ",l: ";
dataString += String ( light );
dataString += "}\" &";
dataSave.runShellCommand ( dataString );
/* Mevcut Veriyi Mysql Veritabanına Gönder */
dataString = "curl \"http://solar.turktiger.com/ekizle";
dataString += timeString + "&s=";
dataString += String ( solar );
dataString += "&c=";

```

```

dataString += String ( cons );
dataString += "&l=";
dataString += String ( light );
dataString += "\" &";
dataSave.runShellCommand ( dataString );
collPower[0] = collPower[1] = 0.0;
collCount[0] = collCount[1] = collCount[2] = 0;
collLight = 0;
}
/* Bu fonksiyon içine bir hata ayıklama bilgisini /mnt/sda1/hatATxt
içerisine yazar*/
*
* @param hataMesajı
*   Hata ayıklama dosyasına yazılacak mesaj içeren dize
*/
//void writeDebug(String message) {
/** Hata İçerik Dosyası Adı */
// String fileName = "/mnt/sda1/hatATxt"; //
/** İşaretçi dosyasında hata ayıklama */
// File dataFile = FileSystem.open ( fileName.c_str(), FILE_APPEND
);
//
// if ( dataFile ) {
//   dataFile.println ( message );
//   dataFile.close();
// } }

```

Yukarıdaki kod bölümü ile birlikte açıklanan verileri bu JSON dizesinde görebilirsiniz. L, S ve C değeri her dakika güncellenir ve Android uygulamasında kullanıcı arayüzünü güncellemek için kullanılır. C, CV, CR, .. değerleri her ölçüm döngüsünden (~ 1s) sonra güncellenir ve kalibrasyon modunda kullanılabilir.

sqlite.runShellCommand (dataString);

SQLITE veritabanına ölçülen değerleri yazmak için “bridge” fonksiyonunu “.runShellCommand” kullanır. Oradan query.php betiği ile yerel ağ üzerinden veri alınabilir.

3.16.3 Ekizleme_c_IşıkSensörü

Bu modül ışık sensörünü kullanır. Işık sensörü ile iletişim Adafruit Sensör ve Adafruit_TSL2561_U kütüphanelerinde yapılır. Bu kütüphaneler, ışık sensörü modülünün üreticisi olan Adafruit 'ten açık kaynak olarak temin edilebilir. Kod, sensörden bir değer okur ve eğer sensör bir değer verirse, kod daha yüksek bir entegrasyon süresine geçer. Ancak sensör zaten doygunsa, entegrasyon süresini kısaltır ve sensörden okumayı tekrar dener (İnt.Kynk.9) (İnt.Kynk.10).

void configureSensor (), ışık sensörünün otomatik aralık fonksiyonunu başlatır ve entegrasyon süresini en yüksek değere ayarlar.

void readLux (), ışık sensörü modülünde bir ölçüm başlatır. Farklı ışık yoğunluğuna uyum sağlamak için sensörün entegrasyon süresini adapte etmek gerekir. Başlangıçta, sensör düşük ışık durumlarında en iyi olan en yüksek entegrasyon zamanına ayarlanır. Ancak ışık artarsa, sensör doygun hale gelir ve daha kısa bir entegrasyon süresine geçmek gerekir. Sensör 3 farklı entegrasyon süresi sunar.

Kod parçası:

```
void configureSensor () {  
    /* Ayrıca, kazancı manuel olarak ayarlayabilir veya otomatik kazanım  
    desteğini etkinleştirebilirsiniz. */  
    tsl.enableAutoRange ( true );          /* Otomatik kazanç ... 1x ile 16x  
    arasında otomatik olarak geçiş yapar */  
    /* Entegrasyon süresini değiştirmek size daha iyi sensör  
    çözünürlüğü verir (402ms = 16-bit data) */  
    tsl.setIntegrationTime ( TSL2561_INTEGRATIONTIME_402MS ); /*
```

```

16-bit veri en yavaş dönüşümler */
}

/**
 * Güncel ışık ölçümü alın.
 * İşlev 5 Ölçüm Yapar ve Ortalama Değeri Döndürür.
 * Fonksiyon, sensörün aşırı yüklenmesi durumunda entegrasyon süresini
uyarlar
 * Sonuç, global sunLux değişkeninde saklanır
 */
void readLux () {
    /** Birikmiş sensör değerleri */
    long accLux = 0;
    /** Sensor event sensörden değer okur */
    sensors_event_t event;
    /** Entegrasyon süresini ayarlamak için kullanılan başarılı okumalar için
sayacı */
    int lightOk = 0; /* Doygunluk durumunda 5 kez yeniden dene*/
    for ( int i = 0; i < 5; i++ ) {
        wdt_reset();
        tsl.getEvent ( &event );
        /* Sonuçları görüntüle (ışık lüks olarak ölçülür)*/
        if ( event.light ) {
            /** Güneş ölçümü için AD conv'den okunan int değeri */
            accLux += event.light;
            lightOk++; /* Başarılı ölçümlerin sayacını arttır*/
            if ( lightInteg == 1 ) { /*orta entegrasyon zamanında, daha iyisini dene */
                tsl.setIntegrationTime ( TSL2561_INTEGRATIONTIME_402MS );
            }
        }
    }
    /* 16 bit veri ancak en yavaş dönüşümler */
    /* Yeni entegrasyon zamanı dene */
    tsl.getEvent ( &event );
    if ( event.light == 0 ) {
        /* Doymuş, orta entegrasyon zamanına geri dön */
    }
}

```

```

        tsl.setIntegrationTime (
TSL2561_INTEGRATIONTIME_101MS ); /* orta çözünürlük ve hız */
        } else {
            lightInteg = 2;
        }
    } else if ( lightInteg == 0 ) { /* en düşük entegrasyon süresindeyiz, daha
yüksek bir tane dene */
        tsl.setIntegrationTime ( TSL2561_INTEGRATIONTIME_101MS );
/* orta çözünürlük ve hız */

        /* Yeni entegrasyon zamanı dene */
        tsl.getEvent ( &event );

        if ( event.light == 0 )
        {
            /* Doymuş, düşük entegrasyon süresine geri dön */
            tsl.setIntegrationTime ( TSL2561_INTEGRATIONTIME_13MS );
/* Hızlı fakat düşük çözünürlük */

            } else {
                lightInteg = 1;
            }
        }
    } else {
        /*Eğer event.light = 0 lux ise, sensör muhtemelen doymuştur ve hiçbir
güvenilir veri üretilmedi! */
        if ( lightInteg == 2 ) { /* en yüksek entegrasyon süresindeyiz, daha
düşük bir tane dene */
            tsl.setIntegrationTime ( TSL2561_INTEGRATIONTIME_101MS );
/* ortalama çözünürlük ve hız */
            wdt_reset();
            tsl.getEvent ( &event );
            if ( event.light == 0 ) { /*Halen Doygun? */
                lightInteg = 0;
                tsl.setIntegrationTime ( TSL2561_INTEGRATIONTIME_13MS );
/* Hızlı fakat düşük çözünürlük */
                wdt_reset();
            }
        }
    }
}

```

```

        tsl.getEvent ( &event );
        if ( event.light != 0 ) { /*Bir sonuç aldınmı? */
            accLux += event.light;
            lightOk++; /* Başarılı ölçümlerin sayacını arttır */
        }
    } else {
        lightInteg = 1;
        accLux += event.light;
        lightOk++; /* Başarılı ölçümlerin sayacını arttır */
    }
} else if ( lightInteg == 1 ) { orta entegrasyon zamanındayız, daha düşük
dene */
    lightInteg = 0;
    tsl.setIntegrationTime ( TSL2561_INTEGRATIONTIME_13MS );
    /* fast but low resolution */
    wdt_reset();
    tsl.getEvent ( &event );
    if ( event.light != 0 ) { /* Şimdi bir sonuç al*/
        accLux += event.light;
        lightOk++; /* Başarılı ölçümlerin sayacını arttır*/
    }
}
}
}
}

if ( lightOk != 0 ) {
    collLight = collLight + (accLux / lightOk);
    collCount[2] += 1;
    Bridge.put ( "I", String ( accLux / lightOk ) );
} else {
    Bridge.put ( "I", "0" );
}
}
}

```

Değişen ışık durumlarına uyum sağlamak için bu döngü 5 kez tekrarlanır.

3.16.4 Ekizleme_x_Ölçümleri

Bu modül AT sensör ölçüm rutinlerini içerir. Akım ve gerilimin doğrudan ölçümü EmonLib kütüphanesi tarafından yapılır. EmonLib, OpenEnergyMonitor tarafından sağlanan açık kaynaklı bir kütüphanedir.

void getCTValues (int index), akım, voltaj ve güç ve güç faktörü hesaplamasını ölçmeye başlamak için EmonLib 'i çağırır. Parametre endeksi okunması gereken sensörü belirler. index = 0, güneş paneli kablosuna bağlı AT sensörünün bir ölçümünü başlatır, index=1 ana güç hattına bağlı AT sensörünün bir ölçümünü başlatır.

Toplanan değerler daha sonra bir akümülatörde depolanır. Şu anda sistem her saniye içerisinde bir ölçüm yapabiliyor. Geçici hafızada depolanan değerler, her 60 saniyede bir veri tabanına kaydedilmeden önce ortalama bir değeri hesaplamak için kullanılır.

Geceleri güneş paneli güç üretmiyor, ancak küçük bir bekleme akımı kullanıyor. Bu bekleme akımını aşağıdaki kodla ortadan kaldırırız.

```
/** Sensor 0 güneş panelini ölçüyor, güneş panelinden üretim 20W altındaysa bu genellikle invertörün harcadığı enerjidir ve ihmal edilmelidir */  
if ( emon[index].Irms < 0.55 ) {  
    power = 0.0;  
}
```

Bu şekilde, gece boyunca var olmayan bir enerji üretimini kaydetmediğimizden emin oluruz.

Ölçülen değerler daha sonra Anahtar / Değer yapısı olarak “Bridge.put” işlevi ile Linux tarafında kaydedilir. Anahtarlar, güneş sensöründen gelen değerler için ön eki ve ana hat sensöründen gelen değerler için “C” ön eki kullanılarak oluşturulur.

```
Bridge.put ( prefix, String ( emon[index].Irms ) ); // Akım
Bridge.put ( prefix + "r", String ( power ) ); // Gerçek güç
Bridge.put ( prefix + "v", String ( emon[index].Vrms ) ); // Voltaj
Bridge.put ( prefix + "a", String ( emon[index].apparentPower ) ); // Görünür güç
Bridge.put ( prefix + "p", String ( emon[index].powerFactor ) ); // Güç faktörü
```

Tüm Kod:

```
void getCTValues (int index) {
    /* Solar Panelden Ölçülen Akım Değerini Al */
    emon[index].calcVI(20, 2000);
    /** Ölçülen Değer Watt */
    double power = emon[index].realPower;
    /** Önek İçin Dize */
    String prefix = "c";
    if (index == 0) {
        prefix = "s";
    }
    /** Sensör 1 güneş panelini ölçüyor, eğer 20W'tan azsa, inverterler tarafından çekilen
    bekleme akımı */
    if ( emon[index].Irms < 0.55 ) {
        power = 0.0;
    }
    collPower[index] = collPower[index] + power;
    collCount[index] += 1;
    Bridge.put ( prefix, String ( emon[index].Irms ) );
    Bridge.put ( prefix + "r", String ( power ) );
    Bridge.put ( prefix + "v", String ( emon[index].Vrms ) );
    Bridge.put ( prefix + "a", String ( emon[index].apparentPower ) );
    Bridge.put ( prefix + "p", String ( emon[index].powerFactor ) );}
```

3.16.5 Ekizleme_y_Kurulum Tüm Kod Parçası

Bu rutin Arduino kartının sıfırlanmasından sonra bir kez çağrılır. Ağ iletişimi için başlatma rutinlerini ve etkinlik LED'i için bağlantı noktasını belirleyen yapıyı içerir;

```
void setup() {  
    /* Çıkış Pinini Ayarla */  
    pinMode ( activityLED, OUTPUT );  
    /* LED Açık */  
    digitalWrite ( activityLED, HIGH );  
    /* Köprü bağlantısını başlat */  
    Bridge.begin();  
    /* Yalnızca localhost'tan gelen bağlantıyı dinle*/  
    /* (harici ağdan kimse bağlanamadı)*/  
    server.listenOnLocalhost();  
    server.begin();  
    /* SD Karta erişimi başlat */  
    FileSystem.begin();  
    /* Adafruit TSL2561 ışık sensörünü yapılandır*/  
    /* Sensörü başlat */  
    if ( tsl.begin() ) {  
        /* Sensör kazancı ve entegrasyon zamanını ayarlayın */  
        configureSensor();  
    }  
    /* Sayaçları ve akümülatörleri sıfırla */  
    collPower[0] = collPower[1] = 0.0;  
    collCount[0] = collCount[1] = collCount[2] = 0;  
    /* YHDC SCT013-000 akım sensörlerini yapılandır*/  
    /* Mevcut sensör 1'i başlat*/  
    emon[0].voltage( 2, vCal, pShift1 ); // AD2, Vcal = 255, phase shift = 1.3  
    emon[0].current ( 0, iCal1 ); // AD0, Ical = 5.7  
    /* Initialise the current sensor 2 */  
    emon[1].voltage( 2, vCal, pShift2 ); // AD2, Vcal = 255, phase shift = 6.1  
    emon[1].current ( 1, iCal2 ); // AD1, Ical = 11.5
```

```

/* Düşük geçişli filtreyi ayarlamak için ilk okumayı alın */

/* Linino'nun açılmasını beklerken
internetten tarih / saati tamamla güncelle */
unsigned int i = 0;
while (i < 500) {
    emon[0].calcVI ( 20, 2000 );
    emon[1].calcVI ( 20, 2000 );
    i++;
}

/* Sadece Hata Durumunda Kullan */
//writeDebug( getTimeStamp() );
/* Sadece Hata Durumunda Kullan Son */

/* Her 5 saniyede / 60 saniyede bir getMeasures / saveData çağrısını
başlat */
lastMeasure = lastSave = millis();
/* LED Kapalı */
digitalWrite ( activityLED, LOW );

/* Linino'ya önyükleme yapmak ve doğru tarihi almak için daha fazla
zaman tanımak için 10 saniye daha */
delay(10000);

/* Bekçiyi Başlat */
wdt_enable(WDTO_8S);
}

```

3.16.6 Ekizleme_z_Döngü

Bu programdaki ana döngüdür. Her zaman çalışır. Döngü içinde millis () işlevini ölçümleri başlatmak ve belirli frekanslardaki değerlerin kaydedilmesini sağlamak için kullanırız. Bu şekilde ölçüm her saniye başlatılır. Her saniye ölçüm yapılır çünkü ışık sensörü henüz kullanılmamıştır. Işık sensörü takıldıktan sonra, ışık ölçümü için yapılan rutin en az 2 saniyeden daha uzun sürdüğü için ölçümün frekansı azaltılmalıdır.

```

/** ekIZLEME başlamasından bu yana milisaniye cinsinden gerçek zaman*/
unsigned long now = millis();
if ( now - lastMeasure >= measureFreq ) { /* her 1 saniyede bir ölçümü başlat */
    lastMeasure = now;
    wdt_reset();
    /* Aktivite LED'i açık */
    digitalWrite ( activityLED, HIGH );
    wdt_reset();
    /* Bir sensör takılıysa ışık ölçümünü al */
    readLux();
    /* Güneş panelinden ölçülen akımı al*/
    getCTValues(0);
    /** Şebekeden ölçülen akımı al*/
    getCTValues(1);
    /* Etkinlik LED'i kapalı */
    digitalWrite ( activityLED, LOW );
}

```

Bu kod, AT sensörleri ve ışık sensörü için ölçüm işlevlerini çağırır. Ölçüm döngüsü, sensör bağlantı kalkanına monte edilmiş bir LED ile görselleştirilir.

Aşağıdaki kod, değerleri her dakika veri tabanına kaydetme işlevini çağırır:

```

if ( now - lastSave >= 60000 ) { /* Her dakika veriyi kaydet */
    lastSave = now;
    wdt_reset();
    saveData();
}

```

Sonunda, doğrudan yerel ağ üzerinden bir bilgisayar veya Android cihazla iletişim kurmak için bir kod bölümümüz var (İnt.Kynk.11).Uygulamaya başlandığında, tüm iletişim bu Arduino kodu içinde çözümlenmeye çalışıldı ancak bellek sınırlamaları

nedeniyle bunun mümkün olmadığı anlaşıldı. Sonunda burada kalan tek bir fonksiyon var. İletişim bölümlerinin çoğu doğrudan Arduino kartının Linux bölümüne aktarıldığından, sadece durum raporu işlevi burada ele alınmaktadır.

```
/* Yeni bir kullanıcı bağlandı? */
if ( client.available() ) {
    wdt_reset();
    /** Gönderilen komutu tutan karakter */
    char command = client.read();
    if ( command == 'e' ) { /* Gerçek ayarları al */
        client.print ( "F " + String ( measureFreq ) + "s" );
        client.println ( " V " + String ( vCal ) );
        client.print ( "C1 " + String ( iCal1, 1 ) );
        client.println ( " C2 " + String ( iCal2, 1 ) );
    }
    /* Bağlantıyı Kapat. */
    wdt_reset();
    client.flush();
    client.stop();
}
```

Durum <http://IP-adresi/arduino/e> bağlantısı herhangi bir tarayıcı veya uygulama tarafından çağrılarak talep edilebilir. Bu bağlantıya tıklandığında alınan cevap şuna benzer olmalıdır.

```
F 1000s V 255
C1 6.2 C2 12.4
```

F ölçüm frekansı, V voltaj kalibrasyon değeri, C1 ve C2, AT sensörü kalibrasyon değerleridir. Bu işlev, yerel ağdaki Arduino cihazını bulmak için Android uygulaması tarafından kullanılacaktır.

3.16.7 Zamanlayıcı ve Bekçinin Başlatılması

```

/* Her 5 saniyede / 60 saniyede bir ölçümleri al ve verileri kaydet */
lastMeasure = lastSave = millis();
/* Bekçiyi Aktive Et */
wdt_enable(WDTO_8S);

```

Bekçi ile ilgili olarak, kaynak kodunda sıkça `wdt_reset()`; görebilirsiniz. Bekçi kullanarak, programın en geç 8 saniye sonra herhangi bir yere takıldığı zaman Arduino kartının sıfırlanmasının başladığından emin oluruz. Bu, programın uzun süre katılsız çalışabilmesini sağlar.

3.16.8 Işık ve AT Sensörlerinin Başlatılması

```

/* Sayaçları ve akümülatörleri sıfırla */
collPower[0] = collPower[1] = 0.0;
collCount[0] = collCount[1] = collCount[2] = 0;
/* YHDC SCT013-000 akım sensörlerini yapılandır*/
/* Akım Sensörü 1 i Ayarla*/
emon[0].voltage( 2, vCal, pShift1 ); // AD2, Vcal = 255, phase shift = 1.3
emon[0].current ( 0, iCal1 ); // AD0, Ical = 5.7
/* Akım Sensörü 2 i Ayarla*/
emon[1].voltage( 2, vCal, pShift2 ); // AD2, Vcal = 255, phase shift = 6.1
emon[1].current ( 1, iCal2 ); // AD1, Ical = 11.5
/* Düşük geçişli filtreyi ayarlamak için ilk okumayı al*/
unsigned int i = 0;
while (i<50) {
    /* LED on */
    digitalWrite ( activityLED, HIGH );
    emon[0].calcVI ( 20, 2000 );
    emon[1].calcVI ( 20, 2000 );
    /* LED off */
}

```

```
digitalWrite ( activityLED, LOW );  
i++;  
}
```

3.16.9 Ağ İletişimi ve Linux Dosya Sistemine Erişimin Başlatılması

```
/* Ağ İletişimini Ayarla */  
Bridge.begin();  
/* Sadece localhosttan gelen bağlantıları dinle */  
/* (Dış ağdan gelen bağlanılacak bağlantı yok) */  
server.listenOnLocalhost();  
server.begin();  
/* SDcard'a erişime izin ver */  
FileSystem.begin();
```

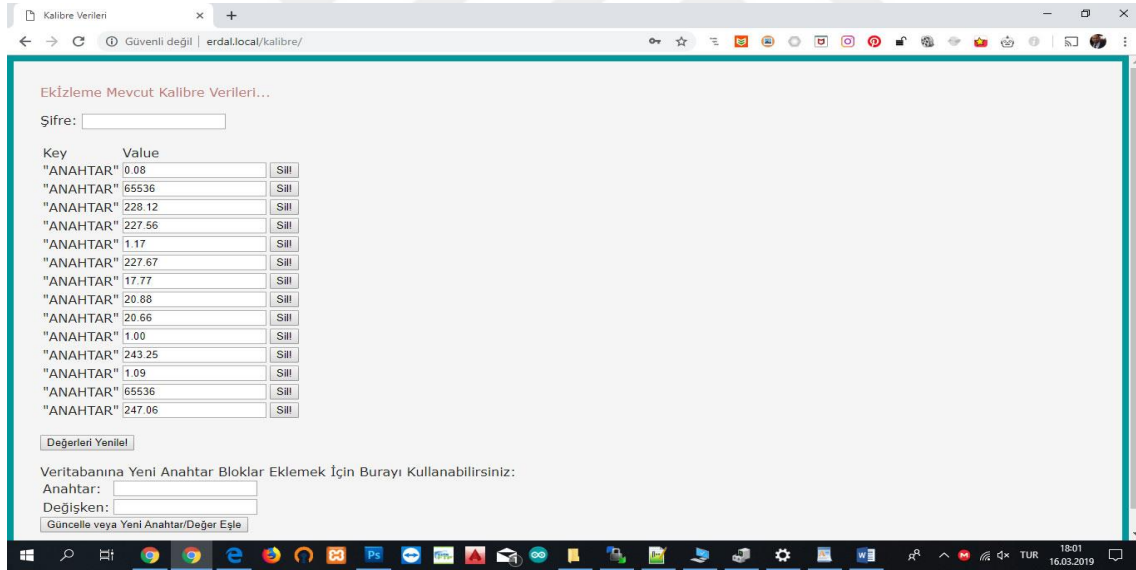
3.16.10 Çıkış Pinlerinin Ayarlanması

```
/* çıkış pinini ayarla */  
pinMode ( activityLED, OUTPUT );
```

3.16.11 Kalibrasyon Değerleri

```
/ ** CT 1 (solar) için iCal tanımı * /  
#define iCal1 5.7  
/ ** CT 2 (şebeke) için iCal tanımı * /  
#define iCal2 10.6  
/ ** Voltaj için faz kayması tanımı (güneş enerjisi) * /  
#define pShift1 1.8  
/ ** Gerilim için faz kayması tanımı (şebeke) * /  
#define pShift2 7.2  
/ ** Voltaj ölçümü için vCal tanımı * /  
#define vCal 245
```

Bu değerler ölçümler için çok önemlidir. Bu değerler kullanılan AT sensörlerine, kullanılan dirençlerin ve kapasitörlerin toleranslarına ve diğer değişkenlere bağlıdır. Bu değerleri doğru almak için, OpenEnergyMonitor yapı bloğu kalibrasyon prosedürünün kalibrasyon kılavuzu birkaç kez incelendi. Burada doğru değerleri elde edene kadar tüm ölçümler yanlış olacaktır! Özel bir senaryo yazılmadan kalibrasyonu mümkün kılmak için uygulamaya kalibrasyon modu eklendi. Kalibrasyon modu etkinleştirildiğinde, Android uygulamaları her 10 saniyede bir Arduino' dan veri çeker ve güç faktörü, gerçek ve görünür güç ile ölçülen voltaj hakkındaki ayrıntıları görüntüler. Kalibrasyon değerleri daha sonra değerler beklendiği gibi değişinceye kadar değiştirilebilir (örneğin, güç faktörü 1'e yaklaştırılabilir). Android uygulamasına tez projesinde yer verilmemiştir ancak kodlara (www.turktiger.com/ekizleme) adresinden ulaşılabilir. Mevcut kalibre değerleri ayrıca web arayüzü üzerinden <http://arduinoip/kalibre> bu projede <http://erdal.local/kalibre> veya <http://192.168.1.77/kalibre> üzerinden okunarak değer eşlemeleri yapılabilir (Şekil 3.22).



Şekil 3.22 Mevcut kalibrasyon verileri görüntüleme ekranı.

3.17 Arduino Kartına Statik IP Atanması

Modemin ana yerel IP 'sinin 192.168.1.1 olduğunu varsayalım ve Arduino Yun kartına 192.168.1.77 numaralı sabit yerel ip numarasını atayalım, bunun için önce SSH üzerinden aşağıdaki kodu Arduino 'ya girmeliyiz.

Kod:

```
root/etc/config/nano network
```

Ardından aşağıdaki değişiklikleri yapılmalıdır:

```
config interface 'lan'
option ifname 'wlan0'
option proto 'static'
option ipaddr '192.168.1.17'
option netmask '255.255.255.0'
option gateway '192.168.1.1'
option dns '192.168.1.1'
option metric '10'
config interface 'wan'
option ifname 'eth1'
option proto 'dhcp'
```

Sonra adresine gidip;

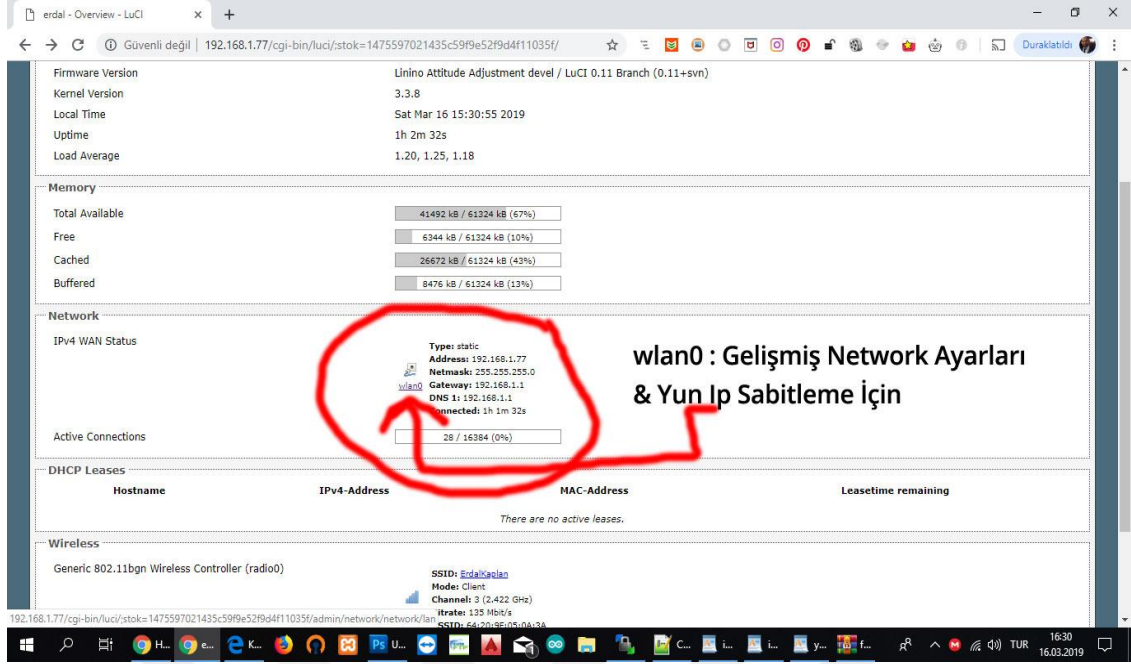
```
root/etc/init.d/
```

Çalıştırılmalıdır;

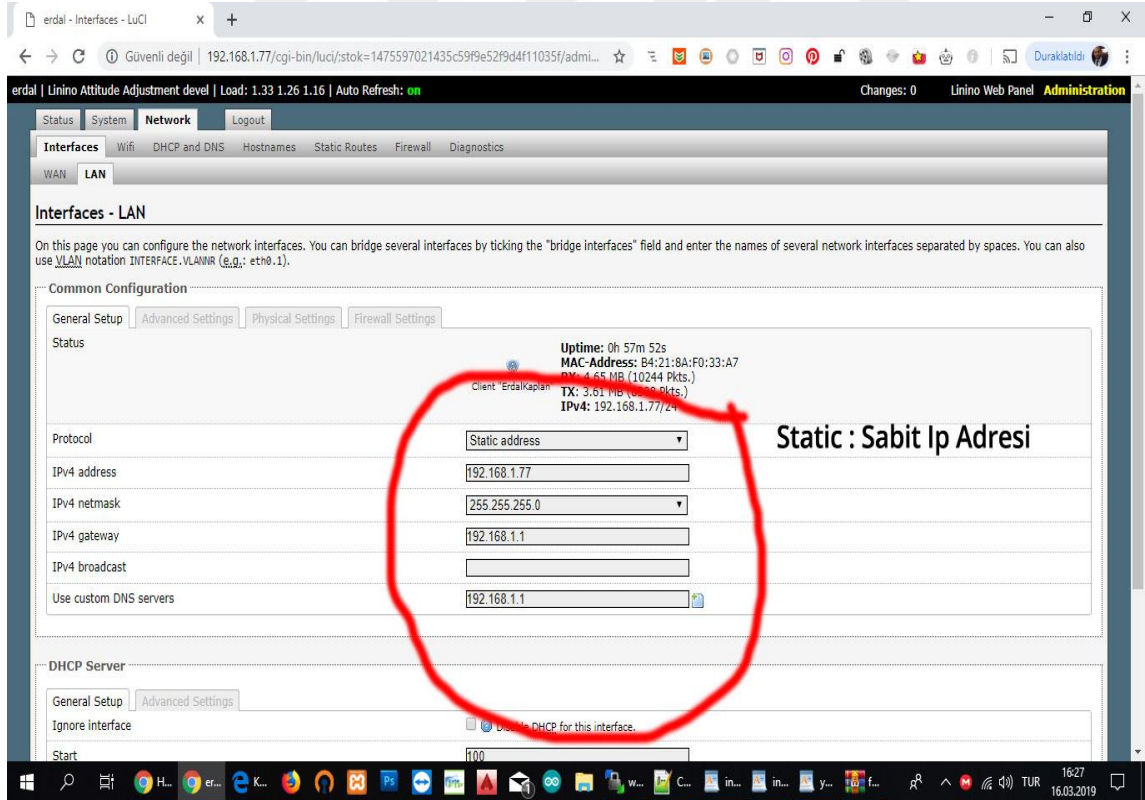
```
./network reload
```

Son olarak ip adresinin durumuna bakılarak, aşağıdaki komut girildiğinde, yeni yerel IP adresinin 192.168.1.77 olarak değiştiği görülecektir. İkinci bir yöntem olarak, Yun arayüzü üzerinden Configure ve Wan0 'dan statik ip seçimi yapılarak IP adresi sabitlenebilir (Şekil 3.23).

```
ifconfig
```



Şekil 3.313 Configure (Konfigürasyon) > Wan0 üzerinden ağ ayarlarına giri.



Şekil 3.24 Yun statik IP atama.

3.18 İzleme Sistemine İnternet Üzerinden Erişimin Sağlanması

Varsayılan ağ geçidi 192.168.1.1 olan yerel ağda, rastgele seçilen 192.168.1.77 IP adresi üzerinden ulaşılan izleme sistemine, internette herhangi bir yerden ulaşabilmesi için birkaç ayarlama yapılması gerekmektedir. Bu ayarlamaları kullanıcıların evlerinden yapacağını varsayarak gereksinimleri adım adım işleyeceğiz.

3.18.1 Modem Üzerinden NAT Ayarlamasının Yapılması

Dış internet ağından evdeki modeme bir istek geldiğinde herhangi bir port için o isteğin modeme bağlı cihazlardan hangisine iletileceği anlamına gelen diğer bir isimle, network address translation (Ağ Adres İletimi) işlevi, bir nevi evde modem ile sunucu kurabilmeyi sağlayan bir işlemdir. Bu işlev tamamlandığında dış IP adresinden çalışan bir web sunucusu kurulmuş olunacaktır. Modemin modeline bağlı olarak bu işlem port açma, port tetikleme, bağlantı noktası iletme gibi farklı isimlerle yer alabilir. Projede kullanılan modemde gelişmiş seçeneğinin ardından, bağlantı noktası iletme seçeneğine girildiğinde aşağıdaki gibi ayar sekmesi karşımıza çıkmaktadır. Lan istemcileri kısmından belirlenen IP Adresi, şablon kısmından web server (web sunucusu), bağlantı kısmından internet seçilerek ekle butonuna basılmasının ardından, 192.168.1.77 yerel IP adresinin gelen ve giden 80 numaralı portunu, dış IP adresinden gelen isteklere açmış oluruz. [3.28]

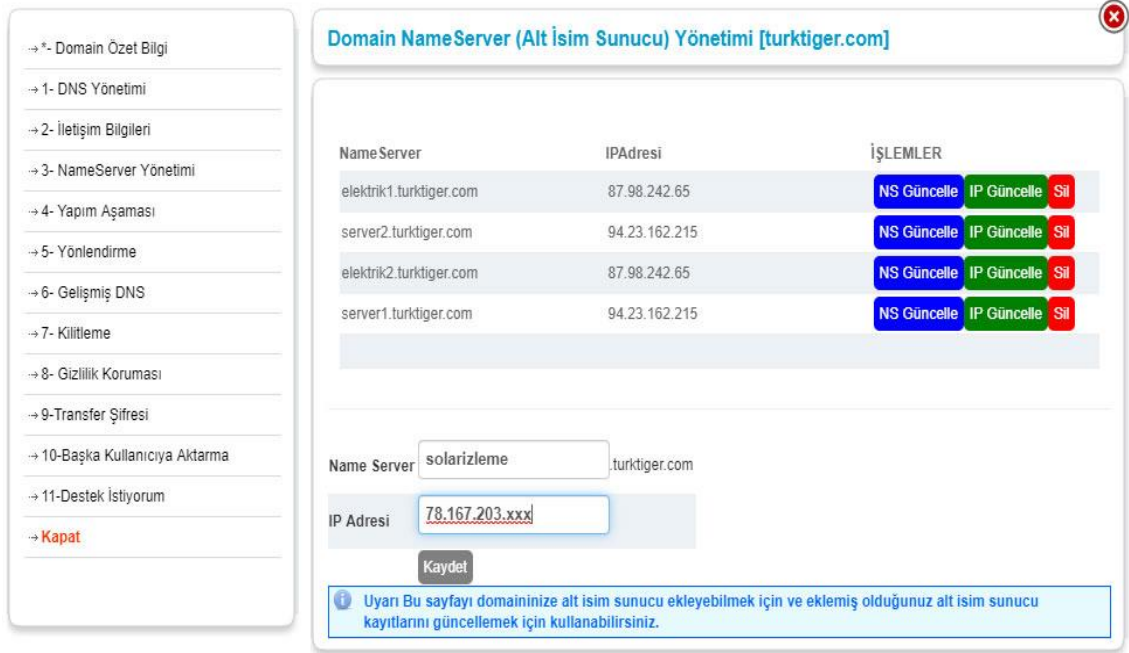
Şekil 3.25 Port yönlendirme ayarları.

3.18.2 Statik İp Adresine Alt Alan Sunucu Eklenmesi

Yerel internet sağlayıcıdan cüzi bir ücretle alınabilecek olan statik ip adresi, diğer bir isimle sabit internet IP adresi yukarıda birkaç kez bahsettiğimiz bizim dış IP adresimizdir. Yukarıdaki işlemleri yaptıktan sonra hali hazırda var olan örneğin; 206.182.32.xxx (xxx yerine herhangi bir rakam var) gibi bir dış IP adresinden solar izleme sunucusuna erişilerek uzaktan okuma sağlanabilir, ancak bu IP adresi akılda kalıcı olmayabilir. Bunun için dünyada alan adı sunucuları kullanılmaktadır. Kısacası bu işlem sizin belirleyeceğinizisminiz.com, sizin belirleyeceğinizisminiz.net gibi bir alan adını dış IP adresine bağlamaya yararmaktadır. Dış IP adresini, alan adı sunucusuna bağlamak için biz bu projede alt alan adı sunucusunu "A" yönlendirmesiyle kullandık. Belirlenen alan adı: turktiger.com

- ✓ Belirlenen alt alan adı: solarizleme
- ✓ Yönlendirme biçemi: "A"
- ✓ Tam Alan adı: solarizleme.turktiger.com

Yönlendirme ayarı ise aşağıdaki gibidir. (3.26) (isimtescil.net. 2019)



NameServer	IPAdresi	İŞLEMLER
elektrik1.turktiger.com	87.98.242.65	NS Güncelle IP Güncelle Sil
server2.turktiger.com	94.23.162.215	NS Güncelle IP Güncelle Sil
elektrik2.turktiger.com	87.98.242.65	NS Güncelle IP Güncelle Sil
server1.turktiger.com	94.23.162.215	NS Güncelle IP Güncelle Sil

Name Server: solarizleme.turktiger.com

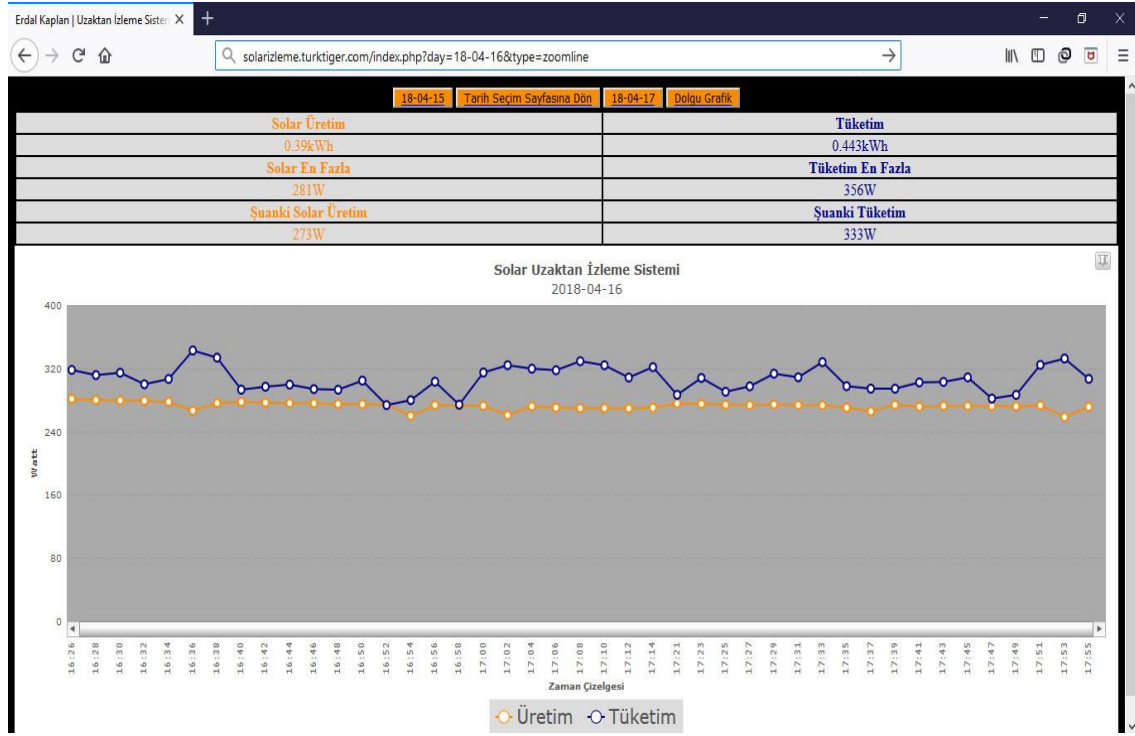
IP Adresi: 78.167.203.xxx

Kaydet

Uyarı Bu sayfayı domaininize alt isim sunucu ekleyebilmek için ve eklemiş olduğunuz alt isim sunucu kayıtlarını güncellemek için kullanabilirsiniz.

Şekil 3.26 Alt alan adı sunucusu oluşturma.

Kaydet seçeneğine basılmasının ardından dış IP adresinin alt alan adı sunucusuna oturması yaklaşık 24 – 48 saat arası sürecektir. Bu 24 – 48 saat arasında, sunucuya alt alan adı sunucusundan ulaşılamayabilir. Bu süre zarfında sabırlı olup beklemek gerekmektedir. Ya da, detayları uzun uzadıya bu projede anlatılmamış olsa da, varsa alan adı sunucusu sağlayıcısının gelişmiş IP ayarlarından IP adresinizi “A” yönlendiricisiyle direk alt alan adı sunucusuna yönlendirilebilir (Şekil 3.27).



Şekil 3.32 Alt alan adı sunucusu üzerinden arduino ile iletişim sağlanması.

4. BULGULAR

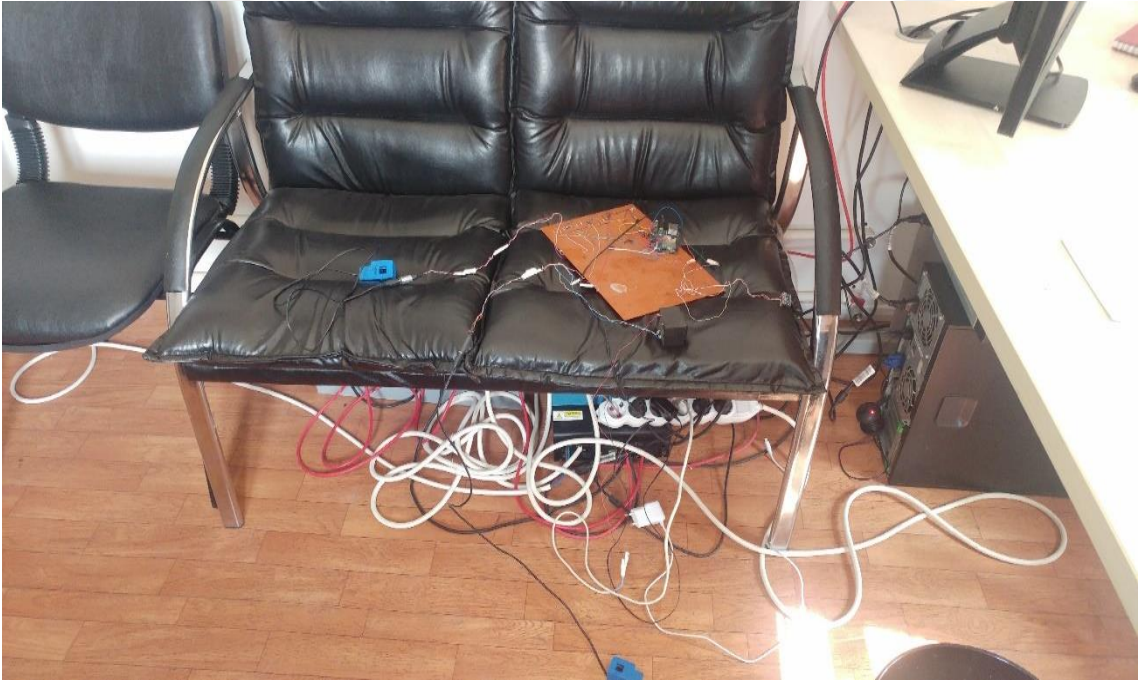


Şekil 4.33 Solar izleme açılış sayfası.

Bu proje temel anlamda en düşük bütçeyle var olan bir solar sistemin uzaktan üretim ve tüketim sonuçlarının izlenmesi amacıyla yapılmıştır. Proje çalışmaları esnasında 250 W solar panel bir barınma alanının üzerine monte edilmiş, 800 W evirici, tüketici ve izleyici olarak da yaklaşık 500 W masaüstü bilgisayar kullanılmıştır. Bununla birlikte kullandığımız evirici aynı anda 220 V şebeke beslemesi ve solar beslemesi yapabildiğinden ve kapasitesi üzerinde bir yük devreye girdiğinde kendini bypass ederek yükü şebekeye devredebildiğinden cihaz kalibresi ve ölçümler için bize alan açmıştır. Bu sebeple ölçümler yapılırken kalibre sırasında 1000 W ve 2000 W arasında omik yükler kullanılabilmiş, bu sebeple sonuçların ve cihazın çalışma kabiliyeti hakkında detaylı bilgi edinilmiştir. Cihaz yerleştirildiği kısmın kablo sarmalının ve bu kablolardan geçen akımın yoğunluğuna göre yaklaşık 30 W – 60 W arası gürültü algılamakta bu da bizim ölçüm sonuçlarımıza yansımaktadır. OpenEnergyMonitor 'de ayrı bir yer açılan 9V AC-AC trafo ve 5V güç adaptörü seçimi doğru yapıldığında bu gürültü seviyesi 20W 'a kadar düşürülebilir. Ve beklenmedik ölçüm sonuçları alınmasının önüne geçilmiş olunur.

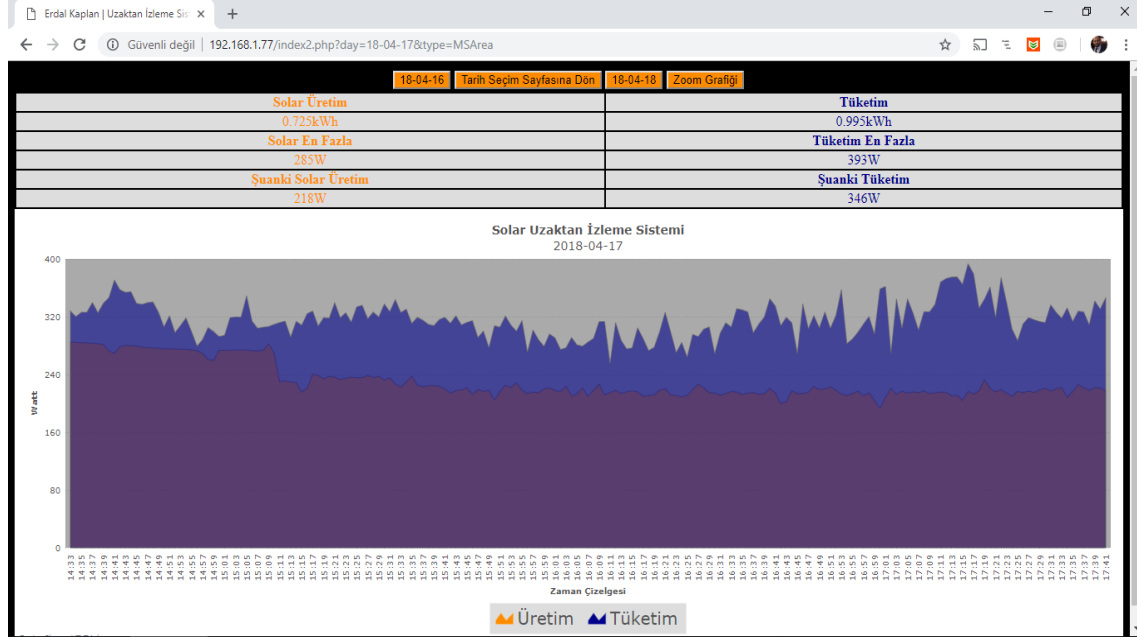


Şekil 4.2 Proje konteynırı üzerine monte edilen solar panel.



Şekil 4.3 Konteynır iç kısmında yer alan 800W evirici & devre kartı.

4.1 Veriler



Şekil 4.34 16.04.2018 tarihi 14:33 – 17:41 arası ölçüm verileri.

Çizelge 4.1 16.04.2018 tarihi 14:33 – 17:41 arası ölçüm verileri.

Üretilen Solar Güç: 0,725 KWS

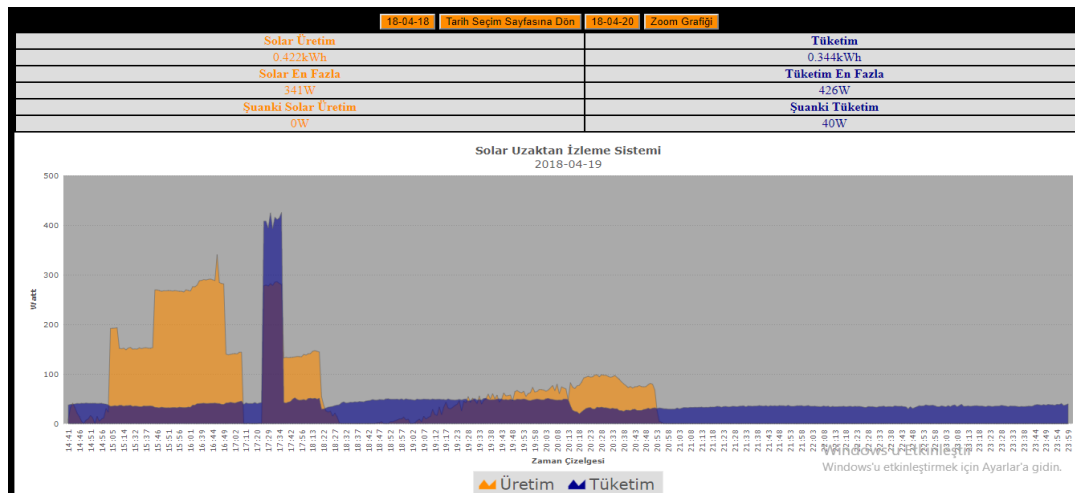
Tüketilen Güç: 0,995 KWS

Solar Maksimum Üretim: 285 W

Maksimum Tüketilen Güç: 393 W

Anlık Solar Güç: 218 W

Anlık Tüketim Gücü: 346 W



Şekil 4.5 19.04.2018 tarihi 14:41 – 23:59 arası ölçüm verileri.

Çizelge 4.2 19.04.2018 tarihi 14:41 – 23:59 arası ölçüm verileri.

Üretilen Solar Güç: 0,422 KWS

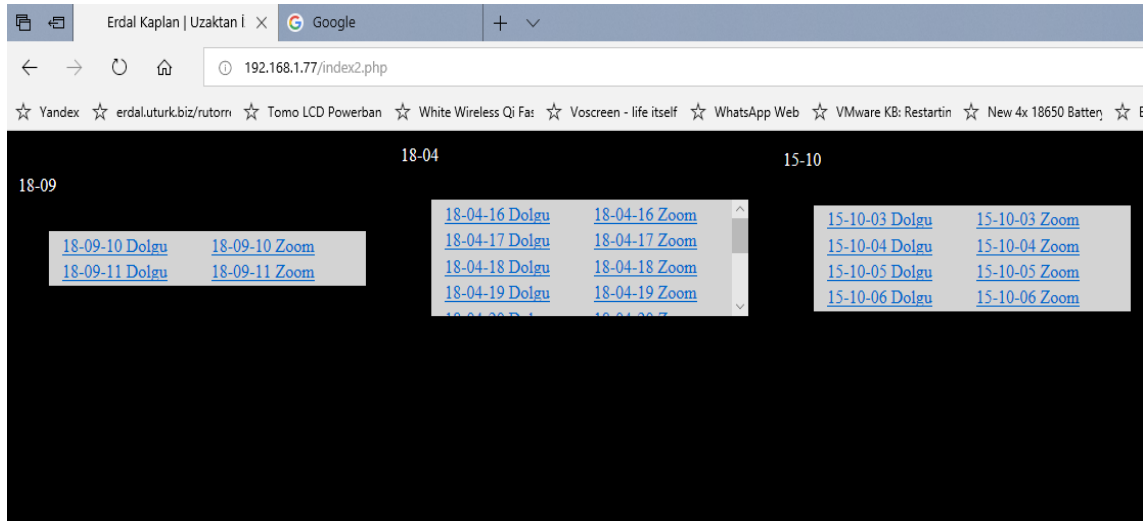
Tüketilen Güç: 0,344 KWS

Solar Maksimum Üretim: 341 W

Maksimum Tüketilen Güç: 426 W

Anlık Solar Güç: 0 W

Anlık Tüketim Gücü: 40 W



Şekil 4.35 Aylık & yıllık önizleme sayfası.

5. TARTIŞMA VE SONUÇ

Tezde şimdiye kadar anlatılanlar ve toplanan veriler ışığında yenilenebilir enerji sistemlerinin daha cazip hale getirilmesi ve uzaktan izleme amacıyla hazırlanan sistemin doğru kalibre edildiğinde 1000 W ve üzeri sistemlerde, yaklaşık 30 – 100 W yanılma payına sahip olabildiği görülmüştür. Bu yanılma payını oluşturan temel sebep Arduino işlemcisinin mükemmel çalışmamasından kaynaklanmakla birlikte, başka bir etki faktörü olarak temassız ölçüm yapan akım trafosu alıcılarının çevredeki cihazların manyetik gürültüsünden etkilenebilmesini de sayabiliriz. Ayrıca bu projede seçilen AC-AC akım gerilim trafosunun da ölçümlerin hassasiyeti noktasında büyük etkisi vardır. OpenenergyMonitor' ü bir kılavuz gibi kullanarak tasarladığımız bu projede, yine OpenenergyMonitor sitesinde anlatıldığı üzere özel üretim ya da belirli markalarda ait AC-AC akım gerilim trafolarından kullanılması gerekmekteydi. Eğer kendimize bir öz eleştiri yapacak olursak projenin çalışıp çalışmayacağı, çalışırsa da kararlı çalışıp çalışmayacağı konularındaki kafa karışıklığı ve süre kısıtlılığından dolayı özel bir AC-AC akım gerilim trafosu temin edilemedi ve ülkemizdeki benzer ürünlerden bir tanesi kullanılarak sistemin nasıl çalışacağını deneyimlendi. Sonuç olarak, elde ettiğimiz veriler ışığında, akım trafosu sensörleri manyetik kirlilikten yani kablo karmaşası ve manyetik alanlardan uzak arındırıldığında, cihaz kalibrasyonu, güç faktörü ölçümleri akım ve gerilim değerleri düzgün şekilde hesaplanarak programlama adımları olması gerektiği gibi yapılırsa ve OpenEnergyMonitor 'de ayrıca yer verilen özel üretim 9 V AC-AC akım trafolarından biri seçilirse, cihaz ölçümü yanılma payı 15 – 20 W aralığına indirgenebilir. Bu değer, hali hazırda yenilenebilir enerji sistemlerinin akım gerilim ölçümlerini de yapan ve piyasada kullanılan (ölçme ve evirici) DC-AC eviricilerin boşta çalışma güçlerine eşdeğerdir. Yani bu projede kullanılan cihaz mevcut haliyle endüstriyel geliştirmeye açık ev kullanıcıları içinse kullanılabilir düzeydedir.

6. KAYNAKLAR

6.1 İnternet Kaynakları

- 1) <https://www.arduino.cc/en/Main/Software> 03.05.2018
- 2) <https://www.arduino.cc/en/Tutorial/ExpandingYunDiskSpace> 03.07.2018
- 3) http://ww1.microchip.com/downloads/en/AppNotes/Atmel-2566-Single-Phase-Power-Energy-Meter-with-Tamper-Detection_Ap-Notes_AVR465.pdf
01.06.2018
- 4) <https://www.twilio.com/blog/2015/02/arduino-wifi-getting-started-arduino-yun.html> 07.06.2018
- 5) <https://www.allaboutcircuits.com/textbook/alternating-current/chpt-11/power-resistive-reactive-ac-circuits/> 02.07.2018
- 6) <https://www.allaboutcircuits.com/textbook/alternating-current/chpt-10/single-phase-power-systems/> 11.07.2018
- 7) <https://github.com/beegee-tokyo/spMonitor> 18.07.2018
- 8) https://www.elkor.net/pdfs/AN0305-Current_Transformers.pdf adresinden alındı
25.07.2018
- 9) https://github.com/adafruit/Adafruit_Sensor 01.02.2019
- 10) https://github.com/adafruit/Adafruit_TSL2561 19.02.2019
- 11) <https://github.com/PhilJay/MPAndroidChart> 03.03.2019
- 12) <https://www.endustri40.com/nesnelerin-interneti-ve-endustriyel-uygulamalari/>
04.03.2019
- 13) <https://github.com/openenergymonitor/EmonLib> 05.04.2019

14) https://wiki.openenergymonitor.org/index.php/EmonTx_Arduino_Shield
05.04.2019

15) <https://developer.android.com/studio> 06.04.2019

16) https://en.wikipedia.org/wiki/Current_transformer 10.04.2019

17) https://en.wikipedia.org/wiki/Mains_electricity_by_country 12.04.2019

18) <https://learn.openenergymonitor.org/electricity-monitoring/ac-power-theory/files/Wuidart.pdf> 12.04.2019



ÖZGEÇMİŞ

Adı Soyadı : Erdal KAPLAN
Doğum Yeri ve Tarihi : Afyonkarahisar - 12.10.1984
Yabancı Dili : İngilizce
İletişim (Telefon/e-posta) : 0507 0099 007 / erdalkaplan77@gmail.com

Eğitim Durumu (Kurum ve Yıl)

Lise : Milli Piyango Anadolu Lisesi, (1996-2003)
Lisans : Kahramanmaraş Sütçü İmam Üniversitesi, Elektrik-Elektronik Mühendisliği Bölümü, (2003-2008)
Yüksek Lisans : Afyon Kocatepe Üniversitesi, Fen Bilimleri Enstitüsü, Elektrik Mühendisliği Anabilim Dalı, (2016-2019)

Çalıştığı Kurum/Kurumlar ve Yıl : T.C Afyonkarahisar Valiliği - İl Afet ve Acil Durum Müdürlüğü, (2011 – Devam Ediyor)