

Pamukkale Üniversitesi, Mühendislik Fakültesi  
Bilgisayar Mühendisliği Bölümü

*Lisans Bitirme Tezi*

**Elektrik Sistemlerinin   
Uzaktan Denetimi ve Yönetimi: intelliPWR**

*Tarafından*

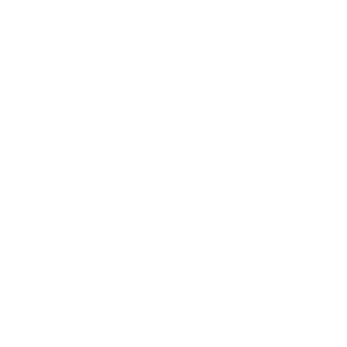
**Veysel Berk ALTUN   
13253004**

*Denetleyen*

**Yrd. Doç. Dr. Elif HAYTAOĞLU**

Mayıs 21, 2018

Denizli



Pamukkale Üniversitesi, Mühendislik Fakültesi  
Bilgisayar Mühendisliği Bölümü

*Lisans Bitirme Tezi*

**Elektrik Sistemlerinin   
Uzaktan Denetimi ve Yönetimi: intelliPWR**

*Tarafından*

**Veysel Berk ALTUN  
13253004**

*Denetleyen*

**Yrd. Doç. Dr. Elif HAYTAOĞLU**

Mayıs 21, 2018

Denizli

**LİSANS TEZİ ONAY FORMU**

Veysel Berk ALTUN tarafından Yrd. Doç. Dr. Elif HAYTAOĞLU yönetiminde hazırlanan “**Elektrik Sistemlerinin Uzaktan Denetimi ve Yönetimi: intelliPWR**” başlıklı tez tarafımızdan okunmuş, kapsamı ve niteliği açısından bir lisans tezi olarak kabul edilmiştir.

Yrd. Doç. Dr. Elif HAYTAOĞLU  
*Danışman*

|  |  |
| --- | --- |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Jüri Üyesi | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Jüri Üyesi |

Pamukkale Üniversitesi Mühendislik Fakültesi Bilgisayar Mühendisliği Bölüm Kurulu’nun   
\_\_\_\_\_ / \_\_\_\_\_ / \_\_\_\_\_ tarih ve \_\_\_\_\_\_\_\_\_\_ sayılı kararıyla onaylanmıştır.

**Prof. Dr. Sezai Tokat***Bölüm Başkanı*

Bu tezin tasarımı, hazırlanması, yürütülmesi, araştırmalarının yapılması ve bulgularının analizlerinde bilimsel etiğe ve akademik kurallara özenle riayet edildiğini; bu çalışmanın doğrudan birincil ürünü olmayan bulguların, verilerin ve materyallerin bilimsel etiğe uygun olarak kaynak gösterildiğini ve alıntı yapılan çalışmalara atfedildiğine beyan ederim.

Veysel Berk ALTUN

ÖNSÖZ

Tez çalışmam sırasında kıymetli bilgi, birikim ve tecrübeleri ile bana yol gösterici ve destek olan değerli danışman hocam sayın *Yrd. Doç. Dr. Elif HAYTAOĞLU*’na, ilgisini ve önerilerini göstermekten kaçınmayan Bilgisayar Mühendisliği Bölümü Ana Bilim Dalı Başkanı sayın *Prof. Dr. Sezai Tokat*’a sonsuz teşekkür ve saygılarımı sunarım.

Her türlü maddi ve manevi destekleriyle beni hiçbir zaman yalnız bırakmayan aileme sonsuz teşekkürler ederim.

Yardımlarını hiç esirgemeyen değerli arkadaşlarım *Gülşah ARAS*’a, *Osman YILMAZTÜRK*’e, *Egemen AYHAN*’a ve *Ahmet YILMAZ*’a da teşekkürü bir borç bilirim.

Veysel Berk ALTUN  
 Mayıs 2018

**ÖZET**

Elektriğin ulaştığı her alanda hayata geçirilmesi planlanan bu projede günümüzün priz temelleri yeniden ele alınmış ve tasarımı konusunda inovatif bir fikir ortaya atılmıştır. Yeniden tasarlanan bu prizler ile gelecekte üretilecek olan yeni nesil akıllı ev aygıtlarının bu prizler ile haberleşmesi sağlanmış ve bu akıllı ev aygıtlarının prizler yardımıyla uzaktan denetimi ve yönetimi gerçekleştirilmiştir.

Ortaya atılan bu yeni priz tasarımı fikrinde, priz soketlerinin faz ve nötr pimlerini altına 5 adet küçük bağlantı pimi eklenmiş ve bu pimler yardımı ile prize bağlanan aygıtların geçerli priz ile haberleşmesi amaçlanmıştır.

Tezin Ar – Ge sürecinde kablolu haberleşme süreçlerinde kullanılması amacıyla *RaspBerry Pi 3*, *Arduino Mega*, *Arduino Pro Mini* ve *NodeMCU* platformları ile uyumlu 3 farklı kütüphane yazılmıştır. Bütün bu kütüphanelerin yazım aşamasında *C*, *C++* ve *Phyton* dilleri tercih edilmiş; daha kolay bir Ar – Ge süreci için ise *GIT* versiyon kontrol sistemi kullanılmıştır. Tezin kablosuz haberleşme süreçlerinde ise açık kaynak kodlu olan ve halen geliştirilmeye devam edilen *openHAB* sistemi tercih edilmiştir.

Sonuç olarak bütün proje kapsamında temel bilgisayar mühendisliği etiğine ve kurallarına uygun olarak fonksiyonel ve modüler toplam 4400 satır kod yazılmış ve çağımıza uygun yeni nesil bir priz tasarımı ortaya konmuştur.

**GİRİŞ**

Günümüzde insanlığın elektriğe olan ihtiyacını bir bitkinin suya olan ihtiyacı ile benzer şekilde tanımlayabiliriz. Bir bitki su olmadan nasıl en temel faaliyetlerini gerçekleştiremiyorsa, insanlar da artık elektrik olmadan yaşayamıyor ve gündelik faaliyetlerinin çoğunu gerçekleştiremiyorlar. Elektriğin insan hayatında bu derece önemli olduğu 21. Yüzyılda, kullanım alanları ve popülaritesi gün geçtikçe artan *“Nesnelerin Interneti – Iot (Internet of Things)”* ile yapılar içerisindeki elektrik kaynaklarına artık uzaktan erişebilmekte ve yönetebilmekteyiz.

İngilizce karşılığı *Internet of Things* olan *Nesnelerin İnterneti*, ilk kez *1999* yılında Britanyalı teknoloji öncüsü *Kevin Ashton* tarafından ortaya atılmış bir terimdir. En genel tanımıyla fiziksel aygıt, sensör, araç ve elektronik aletlerin, sahip oldukları yazılımsal ve donanımsal kaynaklar ile bir ağ üzerinden birbirleri ile haberleşmeleri olarak tanımlanabilir. Benzer biçimde çeşitli haberleşme protokolleri sayesinde birbirleri ile haberleşen ve birbirine bağlanarak akıllı bir ağ oluşturmuş aygıtlar sistemi olarak da tanımlamak mümkündür.

Nesnelerin İnternetine alternatif olarak günümüzde birçok benzer kavramlar vardır; ancak Nesnelerin İnterneti, bu fenomeni açıklamak için en popüler terimdir.

Günümüzde basit bir IoT aygıtının internet ile haberleşmesi aşamasında kullanılan en yaygın yöntem Wi – Fi teknolojisi olarak tanımlanabilir. Herhangi bir üreticinin IoT kavramına uygun bir aygıt üretmeyi planlaması durumunda, Wi – Fi teknolojisi üzerine bir Ar – Ge yapması kaçınılmazdır. Benzer biçimde üretmeyi planladığı yeni nesil bir akıllı ev aygıtına Wi – Fi teknoloji destekleyen bir donanım eklemesi gerekmektedir. Bütün bu süreçler ise bir üreticiye ek bir kaynak tüketimi oluşturur.

Tam da bu noktada, üreticilerin kaynak tüketimini daha aza indirmek amacıyla inovatif bir çözüm arayışı doğmuştur. Literatürde yapılan uzun araştırmalar sonucu, donanımlar arası Wi – Fi haberleşme altyapısını gruplar halinde gerçekleştirecek bir öneri ortaya atılmıştır.

Bu aşama bir IoT aygıtının elektrik olmadan çalışamayacağı varsayılarak, gruplar halinde Wi – Fi haberleşme teknolojisini sağlayacak olan yeni nesil priz tasarımı ortaya atılmıştır. Bu prizler ile üreticilerin Wi – Fi teknoloji ya da benzer herhangi bir haberleşme altyapısına ihtiyaç duymadan, aynı haberleşme sürecinin daha düşük bir bütçe ile gerçekleştirebilmesi amaçlanmıştır.

**I²C**

*I²C (Inter-Integrated Circuit),* *1982* yılında *Philips* tarafından geliştirilmiş, hızlı veri aktarımına olanak tanıyan, *Half-duplex (yarı eş zamanlı çift yönlü çalışabilen)* olarak çalışabilen toprağa referanslı bir seri veri yoludur. *1990*’ların ortasından beri *Siemens*, *NEC*, *STMicroelectronics*, *Motorola*, *Intersil* gibi pek çok firma *NXP* geliştirdiği *I²C* sistem standartlıyla tam uyumlu ürünler piyasaya sürmüşlerdir.

*1 Ekim 2006* itibariyle *I²C* protokolünün lisans zorunluluğu ortadan kaldırılmış, ancak *NXP* tarafından tahsis edilen *I²C* birim adresine sahip olabilmek için üreticilerin ücrete tabii tutulmuştur.

*I²C* protokolünde *SCL* (*SerialClock)* ve *SDA* (*SerialData)* olmak üzere temel iki haberleşme hattı vardır. *SCL,* aygıtlar arası veri senkronizasyonunu gerçekleştiren darbe hattı iken *SDA*, aygıtların birbirleri ile haberleşmesini sağlayan veri yolu hattıdır.

*I²C* protokolünde aygıtlar arası haberleşme *start (başla)* komutuyla başlar ve *stop (bitir)* komutuyla biter. Bu protokolde yapılacak olan her haberleşme, saat darbesinin *lojik 0* durumunda gerçekleşir. Aygıtlar arasında haberleşmenin başladığını veya tamamlandığını belirten *start* ve *stop* durum komutları ise sadece *SCL* hattının, yani saat sinyalinin *lojik 1* olduğu durumlarda gönderilir. *SCL* hattı *lojik 1* iken *SDA* hattının *lojik 1*’den *lojik 0*’a geçişi *start* komutu anlamına gelir. Bu durumda aygıtlar arası haberleşme başlatılmış olunur. Benzer şekilde *SCL* hattı *lojik 1* iken *SDA* hattının *lojik 0*’dan *lojik 1*’e geçişi *stop* komutu anlamına gelir ve bu durum haberleşen iki aygıt arasındaki haberleşmenin tamamlandığını anlamına gelir.

*I²C* protokolünde gönderilen veri ile okunan veri aynı hat üzerinden, yani *SDA* hattı üzerinden gerçekleştirilir. *SDA* pimi, sürekli olarak bir *Pull Up* direnci ile güç kaynağına bağlı olduğundan hattaki *start* ve *stop* bitlerinin anlaşılması hattın *lojik 0* olup olmamasıyla anlaşılır. Ayrıca hattın sürekli *lojik 1* olarak kurulması bazı elemanların yüksek empedans konumlarında bile hattan *lojik 1* okunmasına neden olur.

**MQTT**

*MQTT (Message Queuing Telemetry Transport)*, *1999* yılında *IBM*’de çalışan *Andy Clark* *ile* Cirrus*’da çalışan Arlen Nipper* tarafından *OASIS* şirketinde geliştirilmeye başlanmış açık kaynak kodlu bir haberleşme protokolüdür. Kısıtlı kaynağa sahip ağ istemcilerinin telemetri bilgilerini dağıtmasını kolaylaştıran hafif bir iletişim protokolüdür. Yayın – abone iletişim modelini kullanan *MQTT*, *M2M* ve *IoT* platformlarında önemli bir rol oynar.

Bu protokolde istemciler, ağdaki bir yayıncıya bağlanabilir ve aracı istemciler arasındaki iletişimi düzenleyebilir. Ağdaki bir aygıt abone olduğu konuyla ilgili bir mesaj yayınladığında, geçerli aygıt ağdaki diğer tüm abone aygıtlara aynı mesajı iletebilir. Her aygıt belirli konulara abone olabilir veya aboneliklerden çıkabilir.

*MQTT* protokolünün hafif yüklü tasarımı (2 baytlık üstbilgi), yayınlama – abone olma modeli ve çift yönlü iletişim yetenekleri, elektronik tabanlı kontrol sistemlerin taleplerini karşılamak amacıyla en uygun altyapı modeli olarak tanımlanabilir.

**openHAB**

*openHAB (open Home Automation Bus)*, *Java* programlama dilinde yazılmış, farklı ev otomasyon sistemlerini, cihazlarını ve teknolojilerini tek bir çözümde birleştiren yazılım çözümüdür. Bununla birlikte herhangi bir işletim sistemine bağlı olmaksızın ek bağlantılarla yeni teknolojilerle ve protokollerle uyumlu hale getiriliebilir. *OpenHAB* yazılımı, günümüzde halen *Eclipse Kamu Lisansı (EPL)* altında açık kaynak kodlu olarak geliştirilmektedir. Günümüzde *iOS* ve *Android* temelli bütün aygıtlarda yazılımsal olarak desteklenmektedir.

Tek tip kullanıcı arayüzüne sahiptir. Üreticilerin ve alt sistemlerin sayısına bakılmaksızın tüm sistem otomasyon kurallarına ortak bir yaklaşım sunar.

**ARDUINO**

*Arduino*, *Processing/Wiring* dilinde yazılmış bir geliştirme ortamından oluşan fiziksel bir programlama platformudur. Tek başına çalışan interaktif nesnelerin geliştirilmesinde kullanılabileceği gibi bilgisayar üzerinde çalışan yazılımların geliştirilmesinde de kullanılabilir. Tüketiciler, ilgili mağazalardan hazır üretilmiş *Arduino* kartlarını satın alabilir veya donanım tasarımı ile ilgili gerekli bilgilere *Arduino*’nun resmî web sitesinden erişerek bireysel *Arduino* devre tasarımlarını gerçekleştirebilirler.

*Arduino* kartları, bir *Atmel AVR* mikrodenetleyici (Eski kartlarda *ATmega8* veya *ATmega168*, yenilerinde *ATmega328*) ile programlama ve diğer devre bağlantıları için gerekli olan yan devre elemanlardan oluşur. Her kart, üzerine entegre edilmiş en az bir adet *3.3* voltluk regüle devresi ile bir adet *16 MHz* kristal osilator devresi bulundurur. *Arduino* kartlarının mikrodenetleyicisine önceden bir *bootloader* programı yazılı olduğundan programlama için harici bir programlayıcıya ihtiyaç duyulmaz.

Geliştirme ortamı, sanatçıları programlamayla tanıştırmak için geliştirilmiş *Processing* yazılımından yola çıkılarak geliştirilmiştir.

**YAZILIM GEREKSİNİMLERİ**

Tezin uygulama süreçlerinde yapılan testlerin ve uygulamaların tamamı *Ubuntu 14.04* sürümü üzerinde başarılı bir şekilde test edilmiştir. Bu nedenle, tezin uygulama sürecinde yapılacak olan uygulamaların bu işletim sisteminin daha eski sürümlerindeki kararlılığı konusunda kesin bir bilgi teyit edilemez.

Bir diğer önemli konu, *MQTT* haberleşme sürecinde zorunlu olarak gereken aktif internet bağlantısı üzerinedir. Bu aşamada *port blocking* özelliği olmayan *WPA2 Personal* kimlik doğrulamalı bir bireysel ağ yapısı tercih edilebilir. Tezin uygulama süreçlerinde güvenlik üzerine yeterli bir Ar-Ge çalışması yapılamadığı için kamuya açık olan ağların bu tez sürecinde kullanılması önerilemez. Tercihen kablolu internet bağlantısı da alternatif olarak kullanılabilir.

**openHAB**