

**T.C
MUĞLA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

ELEKTRONİK ve BİLGİSAYAR EĞİTİMİ ANABİLİM DALI

İNTERNET ÜZERİNDEN HAVUZ DENETİMİ

YÜKSEK LİSANS TEZİ

Öner GÖKER

HAZİRAN 2009

MUĞLA

**T.C
MUĞLA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

ELEKTRONİK ve BİLGİSAYAR EĞİTİMİ ANABİLİM DALI

İNTERNET ÜZERİNDEN HAVUZ DENETİMİ

YÜKSEK LİSANS TEZİ

Öner GÖKER

MUĞLA 2009

T.C.
MUGLA ÜNİVERSİTESİ
Fen Bilimleri Enstitüsü

Yrd. Doç. Dr. Mahmut TENRUH danışmanlığında Öner GÖKER tarafından hazırlanan İNTERNET ÜZERİNDEN HAVUZ DENETİMİ başlıklı tez, 16/06/2009 tarihinde aşağıdaki jüri tarafından Elektronik Bilgisayar Eğitimi Anabilim Dalı'nda yüksek lisans tezi olarak oybirliği/oyçokluğu ile kabul edilmiştir.

Başkan	: Yrd. Doç. Dr. Aydın ÇETİN	İmza :
Üye	: Yrd. Doç. Dr. Mahmut TENRUH	İmza :
Üye	: Yrd. Doç. Dr. İzzet PEMBEÇİ	İmza :

ÖNSÖZ

Uzaktan denetim sistemleri, endüstriyel uygulamalarda ve otomasyon sistemlerinde sıklıkla kullanılmaktadır. Cihazların verimliliğinin artırılması, teknik desteğin hızlı bir şekilde verilmesi, maliyetlerin azaltılması ve cihazların yönetiminin kolaylaştırılması gibi faydaları bulunmaktadır.

Bu çalışmada bir havuz sisteminin İnternet üzerinden denetimi için mevcut denetim sistemlerine alternatif, düşük maliyetli ve etkin bir sistemin tasarımı yapılmıştır.

Bu tez çalışması ile benzer yapıdaki yüzme havuzu, akvaryum, su deposu ile endüstriyel havuzların İnternet üzerinden denetimi yapılabilmekte ve benzer uygulamalar gerçekleştirmek için gerekli bilgiler verilmektedir.

Bu çalışma süresince beni destekleyen ve yardımlarını esirgemeyen danışmanım Yrd. Doç. Dr. Mahmut TENRUH'a, yardımlarından dolayı sevgili dostum Umut ÖNDİN'e, desteklerinden ötürü arkadaşlarıma ve bana her zaman güvenen aileme teşekkür ederim.

Öner GÖKER
MUĞLA 2009

İÇİNDEKİLER**Sayfa No**

ÖNSÖZ	I
İÇİNDEKİLER	II
ÖZET	IV
ABSTRACT	V
ŞEKİLLER DİZİNİ	VI
TABLolar DİZİNİ	VII
SEMBOLLER ve KISALTMALAR DİZİNİ	VIII
1. GİRİŞ	1
2. KAYNAK ÖZETLERİ	2
3. MATERYAL ve YÖNTEM	5
3.1 İnternet'e Bağlantı	5
3.1.1 Fiziksel katman	8
3.1.1.1 Ethernet	8
3.1.1.2 CSMA/CD	11
3.1.1.3 RS232	11
3.1.2 Ağ katmanı	12
3.1.2.1 İnternet protokolü (IP)	12
3.1.2.2 ICMP protokolü	14
3.1.2.3 Adres çözümleme protokolü (ARP)	16
3.1.2.4 Ters adres çözümleme protokolü (RARP)	16
3.1.3 Taşıma katmanı	16
3.1.3.1 Kullanıcı datagram protokolü (UDP)	17
3.1.3.2 Aktarım denetim protokolü (TCP)	19
3.1.4 Oturum katmanı	22
3.1.5 Sunum katmanı	22
3.1.6 Uygulama katmanı	23
3.1.6.1 Dinamik istemci ayarlama protokolü (DHCP)	24
3.1.6.2 HTTP protokolü	25
3.1.6.2.1 HTML	26
3.1.6.2.2 JavaScript	26
3.1.6.2.3 CGI Programlama	27
3.2 Gömülü Web Sunucu	28
3.2.1 SBC68EC Gömülü web sunucu	31
3.2.1.1 PIC 18F6680	35
3.2.1.2 RTL 8019AS	37
3.2.1.3 MAX 232	38
3.2.1.4 24LC512 EEPROM	39
3.2.2 TCP/IP Stack	40
3.2.2.1 TCP/IP Stack tanımlamaları	42
3.2.2.1.1 Ortam erişim kontrolü katmanı (MAC)	42
3.2.2.1.2 Adres çözümleme protokolü (ARP ve ARPTask)	43
3.2.2.1.2.1 ARPTask fonksiyonları	44
3.2.2.1.3 İnternet protokolü (IP)	44

3.2.2.1.4 İnternet denetim iletisi protokolü (ICMP)	45
3.2.2.1.5 Aktarım denetim protokolü (TCP)	45
3.2.2.1.6 Kullanıcı datagram protokolü (UDP)	47
3.2.2.1.7 Dinamik istemci ayarlama protokolü (DHCP)	48
3.2.2.1.8 Dinamik HTTP sayfa oluşturulması	48
3.3 Programlama	49
3.4 Havuz Ekipmanları	51
3.5 Uygulama	52
3.5.1 Denetim kartı ve sunum materyali	52
3.5.1.1 LM 35	58
3.5.1.2 Opto Kuplör	58
3.5.2 Yazılım	59
3.5.2.1 JavaScript kodları	60
4. ARAŞTIRMA BULGULARI	63
5. SONUÇ ve DEĞERLENDİRME	64
KAYNAKLAR	65
EKLER	68
Ek 1-a: SBC68EC Bağlantı Şeması	68
Ek 1-b: SBC68EC Bağlantı Pinleri ve Baskı Devresi	69
Ek 2: Havuz Kontrol Kartı Şematiği	70
Ek 3: Gömülü Web Sunucu Web Sayfası	73
Ek 4: PIC 18F4620 Blok Diyagramı	82
Ek 5: TCP/IP Stack Konfigürasyonu	83
Ek 6: Gömülü Web Sunucu Bellek Diyagramı	85
ÖZGEÇMİŞ	86

İNTERNET ÜZERİNDEN HAVUZ DENETİMİ

(Yüksek Lisans Tezi)

Öner GÖKER

MUĞLA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

2009

ÖZET

Teknolojinin ilerlemesiyle birlikte denetim sistemlerine olan ihtiyaç artmıştır. Birçok firma bu alandaki ihtiyacı karşılamak için çalışmalarını sürdürmektedir. Karşılaşılan sorunlardan bazıları sistemler arası uzaklık ve maliyetlerin yüksekliğidir. Firmalar, aralarında uzak mesafeler bulunan sistemleri birleştirmek için değişik protokoller ve iletim sistemleri geliştirmektedirler.

Mevcut ağ sistemlerinin en büyüğü İnternettir. Bu çalışma ile İnternet ağını kullanarak dünyanın herhangi bir yerindeki cihaza bağlanmak ve denetimini yapmak mümkündür. Kullanılan denetim sistemi, İnternet'in ulaştığı mesafeyi ve hızı kullanmaktadır. Arayüz tasarımının ise kullanıcıyı zorlamayacak şekilde yapılabilmesi arzulanmıştır. Bu çalışma ile denetimi yapılacak cihaza İnternet'e bağlanabilen standart bir bilgisayar ile bağlanmak mümkündür. Maliyetler ise oldukça düşüktür. Örnek uygulamada bir havuz sisteminin ekipmanlarının denetimi ve sistemin gözlemi, Ethernet ağı üzerinden gerçekleştirilmiştir.

Anahtar Kelimeler: Uzaktan Denetim, Gömülü Sistemler, TCP/IP Stack, HTML, JavaScript, Havuz Otomasyonu.

Sayfa adedi : 95

Tez yöneticisi : Yrd. Doç. Dr. Mahmut TENRUH

POOL SYSTEM CONTROL OVER THE INTERNET**(M. Sc. Thesis)****Öner GÖKER****MUĞLA UNIVERSITY****INSTITUTE of SCIENCE and TECHNOLOGY****2009****ABSTRACT**

With the improvement of technology, the need for control systems is increased. Many companies continue to work on this necessity. Some of the problems are the distance between the systems and high cost levels. Researchers are developing different protocols and communication systems to connect different systems over distances.

The largest network system is the Internet. With the technique introduced in this thesis, it is possible to connect, control and observe a device from wherever you are in the world. This control system uses the distance and the speed of the Internet. Additionally the use of the interface should be easy for the user. With this thesis, it is possible to connect the device with a standard PC. The cost is quite low. Sample application is the control and monitoring of a pool automation system over the Ethernet network.

Keywords: Remote Control, Embedded Ethernet, TCP/IP Stack, HTML, JavaScript, Pool Automation.

Page number : 95

Adviser : Asst. Prof. Dr. Mahmut TENRUH

ŞEKİLLER DİZİNİ

<u>Şekil No</u>	<u>Sayfa No</u>
Şekil 3.1 OSI Referans Modeli	6
Şekil 3.2 TCP-IP Protokolü Katmanları	7
Şekil 3.3 OSI ile TCP-IP Referans Modellerinin Karşılaştırılması	8
Şekil 3.4 Ethernet Çerçeve Formatı	9
Şekil 3.5 Ethernet'in OSI Referans Modeli'ndeki Yeri	10
Şekil 3.6 IP Paket Yapısı	13
Şekil 3.7 ICMP Mesaj Paket Formatı	15
Şekil 3.8 UDP Paket Formatı	18
Şekil 3.9 TCP Paket Yapısı	20
Şekil 3.10 DHCP Paket Yapısı	24
Şekil 3.11 SBC68EC Web Sunucu	32
Şekil 3.12 PIC18F6680 Pin Bağlantıları	37
Şekil 3.13 RTL8019AS	38
Şekil 3.14 MAX232 içyapısı ve bacak bağlantıları	39
Şekil 3.15 TCP/IP Stack'in İçerdiği Modüller	41
Şekil 3.16 Denetim Kartı	53
Şekil 3.17 Elektrot Yöntemiyle Seviye Ölçümü	54
Şekil 3.18 Altı Yollu Vana Kontrol Devresi	55
Şekil 3.19 Havuz Modeli	57
Şekil 3.20 Opto Kuplör	58
Şekil 3.21 Web Sayfası Ekran Görüntüsü	59

TABLolar DİZİNİ

<u>Tablo No</u>	<u>Sayfa No.</u>
Tablo 3.1 Gömülü Web Sunucu Pinleri.	33
Tablo 3.2 18F6680 Özellikleri.	36

SEMBOLLER ve KISALTMALAR DİZİNİ

ARP	:	Adres Çözümleme Protokolü (Address Resolution Protocol)
AUI	:	Ek Ünite Arayüzleri (Attachment Unit Interface)
CAN	:	Kontrol Alan Ağı (Controller Area Network)
CGI	:	Genel Geçiş Arayüzü (Common Gateway Interface)
DCE	:	Veri Haberleşme Ekipmanı (Data Communications Equipment)
DHCP	:	Dinamik İstemci Ayarlama Protokolü (Dynamic Host Configuration Protocol)
DTE	:	Veri Terminal Ekipmanı (Data Terminal Equipment)
EEPROM	:	Elektrikle Silinebilen Sadece Okunabilir Bellek (Electrically Erasable Programmable Read-Only Memory)
EPROM	:	Mor Ötesi Işıklı Silinebilen Sadece Okunabilir Bellek (Erasable Programmable Read-Only Memory)
FTP	:	Dosya Transfer Protokolü (File Transfer Protocol)
HTML	:	Zengin Metin İşaretleme Dili (Hypertext Markup Language)
HTTP	:	Zengin Metin Transfer Protokolü (Hypertext Transfer Protocol)
ICMP	:	İnternet Kontrol Mesaj Protokolü (Internet Control Message Protocol)
ID	:	Tanımlama Bilgisi (Identification)
IHL	:	İnternet Başlık Uzunluğu (Internet Header Length)

ISO	:	Uluslar Arası Standart Oluřturma Organizasyonu (International Standards Organization)
IP	:	İnternet Protokolü (Internet Protocol)
LAN	:	Yerel Alan Ağları (Local Area Network)
MPFS	:	Microchip Pic Dosya Sistemi (Microchip Pic File System)
OSI	:	Açık Sistem Arabağlantısı (Open Systems Interconnection)
PIC	:	Çevresel Ünite Denetleme Birimi (Peripheral Interface Controller)
PLC	:	Programlanabilir Sayısal Kontrolör (Programmable Logic Controller)
RAM	:	Rasgele Eriřimli Bellek (Random Access Memory)
RARP	:	Ters Adres Çözümleme Protokolü (Reverse Address Resolution Protocol)
ROM	:	Sadece Okunabilir Bellek (Read-Only Memory)
SMTP	:	Basit Posta Transfer Protokolü (Simple Mail Transfer Protocol)
TCP	:	Aktarım Denetim Protokolü (Transmission Control Protokol)
UDP	:	Kullanıcı Datagram Protokolü (User Datagram Protocol)
WAN	:	Geniř Alan Ağları (Wide Area Network)

1.GİRİŞ

Bu çalışmada bir sistemin uzaktan denetiminin, gömülü sistemlerle, İnternet üzerinden, modüler olarak ve güncellemeye ihtiyaç duymayacak şekilde, düşük maliyetlerle sağlanması amaçlanmıştır. Denetim; sistemin özellikleri, kullanılacak donanım, ayrılacak maliyet, gözlem yapılacak ana merkeze uzaklık, bulunduğu ortamın fiziki şartları gibi birçok değişkene bağlı olarak sağlanmaktadır. Değişik firmalar denetimin en etkili, hızlı ve güvenli şekilde sağlanması için protokoller ve donanım ürünleri geliştirmektedirler. Ancak günümüzdeki denetim sistemlerinin ortak sorunu maliyet ve gerekli teknik desteğin bulunmasındaki zorluktur.

Bir denetim sistemi gerçekleştirmek için gerekli temel unsurlar; denetimi yapacak ve yapılacak donanım ile yazılım ve ara bağlantıdır. Bu çalışmada ara bağlantı olarak İnternet'in kullanılması amaçlanmıştır. İnternet günümüzde tüm dünyayı bir ağ gibi sarmış ve düşük maliyetler ile yüksek hızlara ulaşmıştır. Bu sayede denetimi yapılacak birimin konumu dünya üzerindeki herhangi bir yer olabilecektir. Denetimi yapacak sistemin ise dünyanın herhangi bir yerindeki İnternet'e bağlı bir bilgisayar olması amaçlanmıştır. Bunun için iletişim hattı ile denetimi yapılacak birimi birbirine bağlayacak bir ara sisteme ihtiyaç duyulmaktadır. Bu sistem yine bir bilgisayar olabilir, ancak donanım ve yazılım maliyeti ile yüksek güç tüketimi alternatif arayışlara sebep olmuş ve gömülü sistemler tasarlanmıştır. Gömülü sistemler, istenilen amaca yönelik tasarlanmış, küçük boyutlu ve düşük maliyetli denetim kartlarıdır ve bir bilgisayarın ağa bağlanmak için gerçekleştirdiği tüm işlemleri yapabilecek kapasitededirler.

Bu tez çalışmasında bir sistemin İnternet üzerinden sadece web tarayıcısı kullanılarak denetimi ve gözlemlenmesi için gerekli yazılımın ve denetim kartının tasarlanması amaçlanmıştır.

Tezin ikinci bölümünde mevcut denetim sistemleri incelenmiş, avantaj ve dezavantajları belirtilmiştir. Üçüncü bölümde, çalışma esnasında kullanılmış teorik bilgilere, gömülü sistemlere ve uygulamaya ilişkin bilgilere yer verilmiştir. Dördüncü bölümde, araştırma bulgularına değinilmiştir. Beşinci bölümde ise ulaşılan sonuçlara yer verilmiştir.

2. KAYNAK ÖZETLERİ

Uzaktan denetim, çok çeşitli ekipmanlarla ve yöntemlerle yapılabilir. Teknolojinin geldiği noktada, ihtiyaçlar doğrultusunda ve mevcut sorunların giderilmesi konusunda birçok çalışma yapılmaktadır.

Yıldız Teknik Üniversitesi'nde Ovalı (2005), yaptığı çalışmada, endüstriyel denetim sistemlerinde denetleyici ve algılayıcılar arasında düşük bant genişliği gerektiren sayısal haberleşme için tasarlanmış bir kısa mesafe radyo haberleşme protokolünü gezgin robotlar için uyarlamıştır. Radyo ağı ve İnternet ağlarının bağdaştırılması için gerekli sistem tasarlanarak, donanım ve yazılım alt birimleri gerçekleştirilmiş ve tümleştirilmeleri ile sistemin çalıştırılabilir bir ilk örneği tamamlanmıştır. Bu tezin kapsamında gerçekleştirilen sistem, düşük güç tüketimi ve maliyet ile İnternet üzerinden sanal bir şekilde uçtan uca haberleşme imkanı sağlaması avantajıyla, kısıtlı bütçe ile gerçekleştirilecek İnternet erişimli, gezgin robot sistemlerinde haberleşme altyapısı olarak kullanılabilir.

Muğla Üniversitesi'nde Kapucu (2006), fotovoltaik sistem verilerinin gerçek zamanlı olarak ağ üzerinden bir veritabanında toplanabilmesi ve İnternet ve/veya intranet üzerinden izlenebilmesi için bir prototip sistem tasarımı ve uygulamasını gerçekleştirmiştir. Uygulamalar C++, Delphi, Java ve Javascript programlama dilleri kullanılarak IntraWeb ve Java Applet teknolojileri desteği ile tasarlanmıştır. Tasarlanan sistemde veriler, bir bilgisayara paralel bağlantı noktası üzerinden bağlı mikrodenetleyici tabanlı elektronik devreden bir yazılım aracılığı ile okunarak bir bilgisayar ağı üzerinden uzaktaki merkez bilgisayar üzerinde çalışan uygulama sunucusuna iletilmektedir. Uygulama sunucusu bu verileri, aynı merkez bilgisayarda çalışan bir veritabanı sunucusunun yönettiği veritabanında depolamaktadır. Depolanan veriler, geliştirilen bir web uygulaması üzerinden İnternet ve/veya intranet ortamında yayınlanmakta ve gerçek zamanlı olarak web üzerinden izlenebilmektedir.

Yüzüncü Yıl Üniversitesi'nde Özdağ (2006), bilgisayarın paralel portunu kullanarak, bir çeşit Akıllı Ev uygulaması gerçekleştirmiştir. Akıllı ev içerisindeki algılayıcıların ve cihazların bilgisayar ile iletişimini sağlamak için tasarlanan sürücü devreleri, kişisel bilgisayar ile paralel port'u kullanarak haberleşmektedir.

Akıllı ev içerisindeki algılayıcıların tetiklenmesi sonucunda oluşabilecek güvenlik tehditlerini bildirmek için GSM numaralarına mesaj göndermede kullanılan cep telefonu, kişisel bilgisayar ile Bluetooth üzerinden seri haberleşmektedir. Bu cep telefonuna tanımlı mesajlar gönderilerek de akıllı ev içerisindeki cihazların yönetilmesi sağlanmıştır. Cep telefonu ile sistem arasındaki haberleşme, Visual Basic programının ActiveX fonksiyonları kullanılarak gerçekleştirilmiştir. Akıllı ev içerisindeki sisteme İnternet ağı üzerinden istemci program ile bağlantı kurulması sağlanarak, cihaz ve algılayıcıların İnternet vasıtası ile de yönetilerek denetimi sağlanmıştır.

Selçuk Üniversitesi'nde Eşme (2006), uzaktan denetimi yapılabilen bir kalp cihazı tasarlamıştır. Tasarlanan cihaz modüler ve taşınabilir özelliktedir. Bu sebeple uç birimler arasında EKG işaretleri ve diğer iletişim verilerinin iletilmesi dial-up ve İnternet ile olabileceği gibi GSM modem ile haberleşme de tercih edilebilir. Bu alanda yapılan diğer çalışmalardan farklı olarak uzmandan geri bildirim yine bilgisayar ağı ile alınmıştır. Tüm donanımı yönetecek olan yazılım diğer programlama dillerine göre sayısız üstünlüğü olan LabVIEW ile gerçekleştirilmiştir. Ara bağlantı kartları ve diğer elektronik donanımlar için LabVIEW üreticisi olan National Instruments firmasının ürünleri tercih edilmiştir.

Marmara Üniversitesi'nde Doğan (2007), Marmara Üniversitesi Teknik Eğitim Fakültesi Robotik laboratuvarında bulunan Sıvı Seviye Eğitim Seti'ni (Sistem - GUNT RT512 Process Level Trainer), Delphi programlama dili kullanarak geliştirdiği bir yazılım ve tasarladığı web sayfaları aracılığıyla İnternet üzerinden denetimi sağlamıştır. Kullanılan IP (İnternet Protocol) Kamera yardımıyla Sistem'in çalışması, kullanıcının bilgisayarına eş zamanlı olarak aktarılmış ve böylelikle bir Uzaktan Erişimli Laboratuvar (UEL) uygulaması gerçekleştirilmiştir. Denetim sistemi yazılımı, Sunucu Bilgisayar üzerinde çalışmaktadır ve bu yazılım sayesinde kullanıcı, Sistem'i yerel olarak (Robotik Laboratuvarı'nda) veya uzaktan (Web üzerinden) denetimi yapabilmektedir. Web Arayüzü' nü oluşturan Web sayfaları ise HTML ve PHP script dilleri kullanılarak tasarlanmıştır. Bu çalışma sonucunda, kullanıcı, gerçek bir sisteme İnternet üzerinden erişerek eş zamanlı deneyler yapabilmekte, deneyin durumunu ve sonuçlarını eş zamanlı olarak

görüntüleyebilmekte, deney sonuçlarını bilgisayarına “csv” dosyası olarak kaydedebilmektedir.

Hong Kong Çin Üniversitesi’nde Yeung ve Huang (2003), uzaktan erişimli laboratuvar deneylerinde bir dc motorun uzaktan denetimini gerçekleştirmişlerdir. Öğrencilere, laboratuvarında yapılan çalışmaları, İnternet’e bağlı herhangi bir bilgisayar ile izleme şansı vermek için yapılan bu çalışmada; deneysel olarak bir dc motorun denetimi ve video görüntülerinin İnternet üzerinden iletimi sağlanmıştır. Bilgisayara eklenen bir veri işleme kartı ile dc motorun denetimi ve yine bilgisayara bağlı bir video kamera yardımıyla görüntülerin iletimi gerçekleştirilmiştir. Bilgisayar İnternet’e bağlı bir web sunucu görevinin yanında veri işleme görevi yapmakta ve aynı zamanda video sunucu olarak çalışmaktadır.

Southeast Üniversitesi’nde Song vd. (2004), ayrıık ölçüm yapan algılayıcıların (sensör) standart bir ağ yapısı ile İnternet üzerinden gözlemi deneyini yapmışlardır. Sunucu olarak kullanılan bir bilgisayara bağlanan algılayıcıların, İnternet üzerinden gözlemi bilgisayar yardımıyla yapılmıştır. Algılayıcıların bilgisayar ile haberleşmesi için ise CAN protokolü kullanılmıştır. Böylece farklı ağ yapılarını kullanan sistemlerin birbiriyle haberleşmesi gerçekleştirilmiştir. Algılayıcıların, CAN ağına bağlanması için ağa bağlanma kapasitesine sahip mikroişlemciler algılayıcılara eklenmişlerdir.

Uzaktan denetim sistemleri incelendiğinde, uzun mesafeler için İnternet ve GSM şebekelerinin öne çıktığı görülmektedir. İstemci ve sunucu tarafı yazılımları için ise oldukça kapsamlı ve değişik özelliklere sahip birçok program olduğu görülmektedir. Bunlardan öne çıkan yazılım dili JAVA’dır. İstemci sunucu bağlantısıyla çalışan tüm bu uygulamalarda her iki taraf için özel yazılımlar geliştirilmiştir. Denetim, gözlem ve güvenlik amaçlarıyla yapılan bu yazılımlar maliyeti arttırmakta ve tüm kullanıcılar için yeniden programlamayı zorlaştırmaktadır.

Bu çalışmada, İnternet’e bağlanabilen herhangi bir bilgisayarın ek bir yazılıma ihtiyaç duymadan denetimi yapılacak sisteme bağlanması amaçlanmaktadır. Sistem yazılımının yeniden tasarlanması kolay ve güncellemesinin de İnternet üzerinden yapılabilmesi hedeflenmektedir.

3. MATERYAL ve YÖNTEM

Bu bölümde İnternet'e bağlantı için bilinmesi gereken temel protokoller, gömülü sistemlerin yapıları, kullanılan sistem ve havuz denetimi için bilinmesi gereken temel bilgilere yer verilmiştir. Son olarak uygulama açıklanmıştır.

3.1 İnternet'e Bağlantı

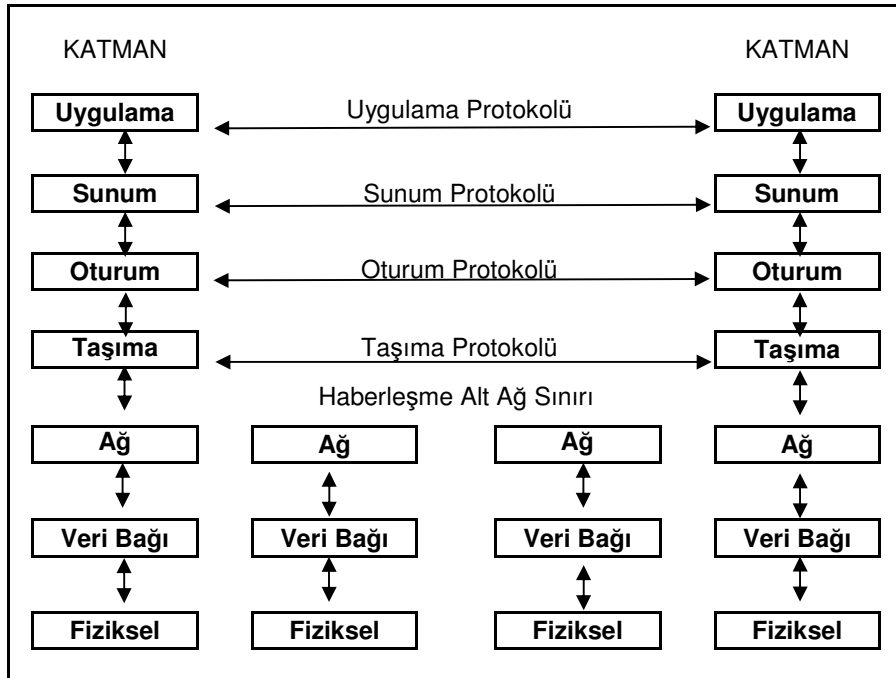
Bir sistemin veya cihazın İnternet üzerinden denetimi için İnternet'e bağlanabilen bir donanıma ihtiyaç duyulur. Bu donanım bir bilgisayar olabileceği gibi denetimi yapılacak cihaza özel, bağlantı ara elemanları olabilir. PLC'ler bu ihtiyaca yönelik ek modüller içermektedir. Ancak tüm bu donanımsal ürünler önemli bir maddi külfeti beraberinde getirmektedirler. Ayrıca sistem yazılımı ve güncellemesinin de yapılabilmesi için yeterli bilgi birikimine sahip teknik elemanın bulunması gerekmektedir. Tüm bu maliyetlerin yanında boyutları da küçültmek için mikrodenetleyici temelli bir elektronik kart kullanılabilir. Ancak bunun için İnternet haberleşmesi sırasında kullanılması gereken bazı protokolleri ve standartları bilmek, gerekli yazılımlar ve donanımsal ürünler hakkında bilgi sahibi olmak gerekir.

Modern bilgisayar ağları yapısal olarak tasarlanmıştır. Tasarım karmaşıklığını azaltmak için birçok ağ, her biri diğeri üzerine inşa edilmiş bir seri tabaka şeklinde organize edilmiştir. Sistemin belli bir standarda oturtulması gerektiğinden protokoller, birbiriyle ilintili tabakalara ayrılmıştır. Bu tabakalar OSI Referans Modeli adıyla anılmaktadır.

OSI Referans Modeli, International Standards Organization (ISO) tarafından sunulan bir modeldir. Bu model ISO OSI (Open Systems Interconnection) Referans Modeli olarak anılır ve diğeri sistemlerle haberleşmeye açık sistemlerin bağlantısı ile ilgilenir (Kaplan, 2000). OSI modeli yedi tabakadan oluşur. Bu tabakalar arası bağlantı Şekil 3.1'de gösterilmiştir.

Yedi Tabakalı Model

- 7) Uygulama : Uygulamalara deęişik servisler saęlar.
- 6) Sunum : Bilgi formatını çevirir.
- 5) Oturum : Haberleşme ile ilgili olmayan problemlerle ilgilenir.
- 4) Taşıma : Uçtan uca haberleşme denetimini saęlar.
- 3) Ağ : Ağ üzerinde bilgiyi yönlendirir.
- 2) Veri Baęı : Baęlı uçlar arasında hata denetimini saęlar.
- 1) Fiziksel : İletim ortamına baęlantıyı saęlar.



Şekil 3.1 OSI Referans Modeli

Kullanımda aę yapısı bu modele tam olarak uymaz. Bazı katmanlar tek bir katmanda birleştirilmiş, bazıları ise kullanılmamaktadır.

İnternet, OSI referans modelinden önce geliştirilmiştir. TCP/IP olarak adlandırılan protokol kümesi İnternet'in temelini oluşturmaktadır. OSI'nin yedi tabakasına rağmen İnternet protokolleri dört tabakadan oluşmaktadır (Şekil 3.2).

Bunlar uygulama (application), iletim (transport), İnternet ve ağ erişim (network access) tabakalarıdır.

UYGULAMA
İLETİM
İNTERNET
AĞ ERİŞİM

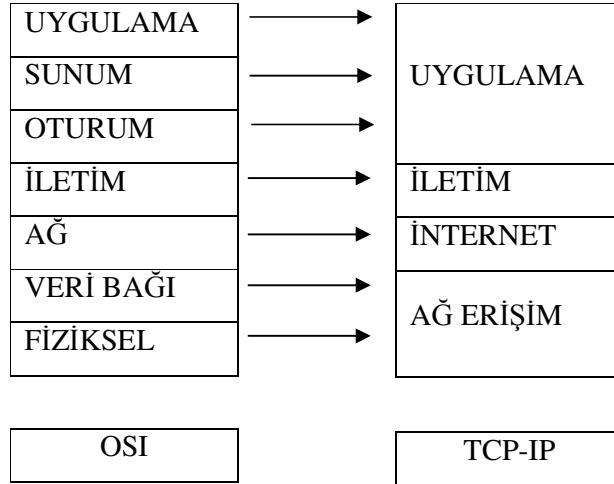
Şekil 3.2 TCP-IP Protokolü Katmanları

Uygulama katmanı; İnternet ortamında kullanılan TELNET, FTP, SMTP, HTTP gibi protokolleri kapsar. Kullanıcı veya makinelere erişim bu katmanda sağlanır. OSI modelinin üst üç katmanı olan oturum, sunum ve uygulama katmanlarına denk gelir (Şekil 3.3).

İletim katmanı; iletim kalitesi ve akış denetimi gibi görevleri üstlenir. Bu katmanda TCP veya UDP protokolleri çalışır. TCP, aktarım denetim protokolüdür ve iki nokta arasında güvenilir, hatasız, akış denetimli ve bağlantılı bir iletim sağlar. UDP ise kullanıcı datagram protokolü olarak adlandırılır ve bağlantısız bir iletim sağlar. Bilgi blokları kontrol edilmeden gönderilir.

İnternet katmanı; kaynak ağdan İnternet ortamına gönderilecek bilgilerin adreslenmesi ve gönderilmesi ile ilgilidir. Bu katmanda IP (İnternet protokolü) görev yapmaktadır. Ağa bağlı her cihazın yerini belirleyen bir IP adresi bulunmaktadır. IP ve bu katmanda çalışan ARP, RARP ve ICMP gibi diğer protokollerle adresleme ve iletim işlemleri gerçekleştirilir.

Ağ erişim katmanının bir diğer adı da ana bilgisayar - ağ katmanıdır (host to network). Bu katmanın görevi bir üst katmandan gelen bilgilerin İnternet ortamına iletilebilmesi için gerekli tüm fiziksel katmanların bilgilerini, yerel alan ağı ve geniş alan ağı bilgilerini, OSI protokolünün fiziksel ve veri bağı katmanı bilgilerini içermektir.



Şekil 3.3 OSI ile TCP-IP Referans Modellerinin Karşılaştırılması.

3.1.1 Fiziksel katman

Bu katman, fiziksel iletişimin elektriksel, mekaniksel, prosedürel ve fonksiyonel özelliklerinin, fiziksel iletişimin başlatıldığı, yönetildiği ve bitirildiği protokollerin tanımlandığı katmandır. Bu özelliklerden bazıları voltaj seviyesi, fiziksel veri gönderim hızı, maksimum transfer mesafesi ve konektörlerdir. Kullanılan standartlardan bazıları; EIA/TIA-232, V.35, EIA/TIA-449, V.24, RJ45, FDDI, Ethernet, NRZI, NRZ, B8ZS'dir. LAN ağları denilince akla ilk Ethernet ağları gelmektedir. Bunun yanında RS232, RS485, BlueTooth ve IRDA gibi teknolojilerle de ağ oluşturulabilmektedir. Bu çalışmada Ethernet protokolü ile ağa bağlanılmaktadır.

3.1.1.1 Ethernet

Ethernet ilk olarak, deneysel çalışmaların (1970 sonlarında Palo Alto Research Center'da yapılan çalışmalar) sonucu olarak ortaya çıkmıştır. İlk Ethernet LAN 2.94 Mbps hızında idi; ancak günümüzde bilgisayar haberleşmesine olan gereksinimin artması ve mikroelektronik teknolojinin gelişmesine paralel olarak daha yüksek hızlara; 10 Mbps, 100 Mbps ve 1000 Mbps gibi hızlara kadar çıkmıştır.

Ethernet standardında aktarılabacak veri, çerçeveler içinde alıcısına aktarılır. Çerçeveler içinde gönderilecek veri parçasının yanında alıcı-gönderici adresleri, hata sınama bitleri gibi birtakım denetim bilgileri de gönderilir. Aşağıda, Şekil 3.4’de çerçeve formatı görülmektedir.

Öntakı	Alıcı Adres	Gönderici Adresi	Tip	VERİ	FCS
8	6	6		N	4

Şekil 3.4 Ethernet Çerçeve Formatı

ÖNTAKI (PREAMBLE): Ethernet’in ilk 8 sekizlisi ön takı olarak senkronizasyon için kullanılır; bunun için gönderilen bit deseni 101010...11 şeklinde olup, alıcı saati ile gönderici saatinin senkronize olmasını sağlar.

ALICI ADRES (DESTINATION ADDRESS): Çerçeveyi alacak düğümün adresini içerir; varış adresi olarak da adlandırılabilir ve MAC adresi içermelidir. 48 bit uzunlukta olup, birebir, grup ve yayma (broadcast) şeklinde adresleme yapılabilir; adresin en anlamlı biti adresleme şeklini belirler, birebir adresler için 0, grup veya yayma adreslemesi için ise 1 ‘dir. En anlamlı bitleri aynı olan grup ve yayma adresleri geri kalan 47 bit ile ayrıştırılır; hepsi 1 ise yayma adresi olarak algılanır; grup adresleme için geri kalan 47 bit 1 ve 0 olabilir.

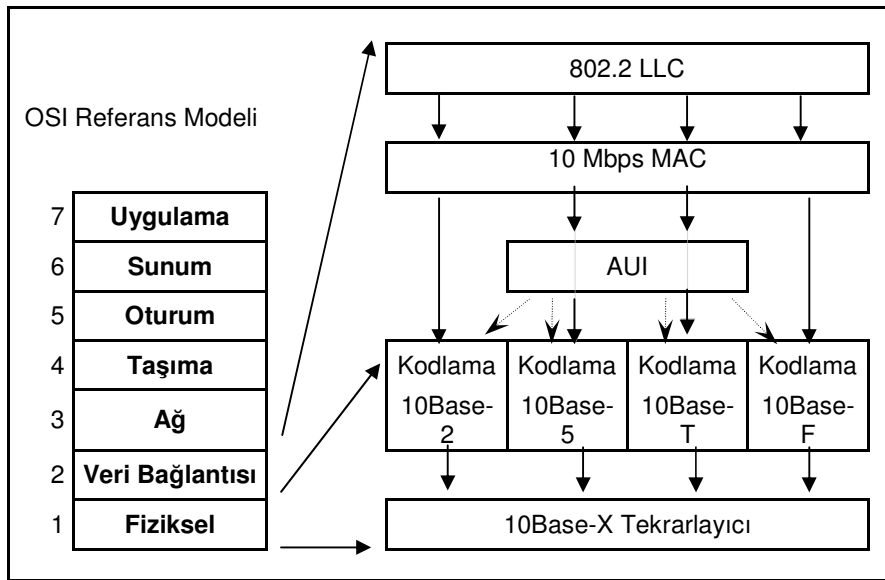
GÖNDERİCİ ADRESİ (SOURCE ADDRESS): Çerçeveyi gönderen düğümün adresini içerir; 48 bit uzunluktadır.

TÜR (TYPE): Bu 2 sekizli tür alanıyla, alınan çerçevelerin hangi üst katman protokolüne veya fonksiyonuna gönderileceği belirlenir. Örneğin üç tür servis sunan bir düğüm kendisine gelen çerçevelerin hangi servise ait olduğunu bu alana bakarak anlayabilir. 0x0800 verinin IP paketi olduğunu göstermektedir. Bu alan 0x0806 ise bu da çerçevenin bir ARP istek veya cevabı olduğunu gösterir.

VERİ (DATA): Aktarılabacak olan veri parçasını içerir; 46 ile 1500 sekizli arasında olabilir. Gönderilecek verinin en az ne kadar olacağı önemlidir; 10 Mbps’lik Ethernet’te 46 sekizliden daha küçük olmamalıdır. Aksi durumda paylaşılan yol üzerinde olan çatışmalar sezilemez.

ÇERÇEVE HATA SINAMASI (FRAME CHECK SEQUENCE): Bu alana çerçeve hata sınaması için hesaplanan 32 bitlik CRC değeri yerleştirilir. Hata sınaması ön takı dışında çerçevenin tüm bitleri için yapılır.

Ethernet içerisinde bulunan alt birimler Şekil 3.5’de OSI Referans Modeli’ne göre görülmektedir; OSI’nin ilk katmanına sahip olan Ethernet’te, en üstte LLC ve MAC birimleri, hemen altında kodlama birimi vardır. Şekilde görüleceği gibi, MAC birimiyle Kodlama birimi arasında AUI diye adlandırılan bir birim vardır. AUI birimi, Ethernet’e esnek bir fiziksel arayüz desteği sağlamak amacıyla araya koyulmuş arabirimdir. En altta ise tekrarlayıcı arabirim vardır.



Şekil 3.5 Ethernet’in OSI Referans Modeli’ndeki Yeri

Ethernet alıcısının görevi, çerçeveyi başlıktan arındırıp üst katmana iletmektir. Alıcıya gelen çerçevenin hedef adresine bakılır. Cihazın MAC adresiyle uyumsuzsa yayın adresi olma durumuna bakılır. İki şartında uyumsuz olduğu durumda çerçeve hafızadan silinir. Hedef adresi uyduğu durumda hata denetimi yapılır. Hata tespit edilirse çerçeve işlenmez. İlk iki aşamayı geçen çerçevenin

protokol türüne bakılır. Gelen çerçeve ARP veya RARP isteği ise bunlara uygun cevaplar gönderilir. Eğer IP paketi ise veri bir üst katman olan IP katmanına iletilir.

Ethernet ağına bağlanmak için Ethernet kartı kullanılır. Her Ethernet kartının MAC adresi olarak adlandırılan 6 sekizlik (48 bitlik) özel bir adresi vardır (00-23-c3-45-00-b3 gibi) ve bu adres tektir. LAN içerisindeki yerel erişimler, gerçekte bu adresler kullanılarak gerçekleştirilir. Değiştirilemeyen bu adresin ilk üç sekizlisi üretici firmaya ait satıcı kodudur. Son üç sekizli ise üretim sıra numarasıdır.

Ethernet ağına bağlantı için Ethernet kablosu kullanılır. Kullanılan kablo çevrede bulunabilecek manyetik alanlardan, iletilen verinin en az etkilenmesi için bükümlü çift ekranlı kablo olmalıdır. Fiziki şartların durumuna göre Ethernet kablosu yerine fiberoptik kablo veya kablosuz iletişim tercih edilebilir.

3.1.1.2 CSMA/CD

Ethernet standardında yola erişim yöntemi olarak CSMA/CD yöntemi kullanılır. Bu yöntemde bir Ethernet düğümü, veri aktarmadan önce yolu dinler, eğer yol, o anda diğer düğümler tarafından veri aktarmak için kullanılıyorsa, yolda bir taşıyıcı (carrier) olduğunu sezer ve kendi verisini yola çıkarmaz, bir süre bekler. O anda yolda aktarılan veri paketinin son bitinden itibaren en azından 9.6 µs bekler. Eğer bir düğüm o anda yolda taşıyıcı olduğunu sezdiği halde, yola verisini çıkarırsa çatışma (collision) oluşur; veri aktarımı gerçekleşmez. Çatışma gerçekleştikten sonra gönderici düğümler iletimi keserler ve yeniden gönderim için rasgele bir süre beklerler.

3.1.1.3 RS232

RS-232 (Recommended Standart), Veri Terminal Ekipmanı, *Data Terminal Equipment (DTE)* ve Veri Haberleşme Ekipmanı, *Data Communications Equipment (DCE)* arasında seri ikili veri alış verişi ile arabirim oluşturmak amacı ile tasarlanmıştır. DTE bir terminal veya bilgisayar, DCE ise modem veya benzeri haberleşme cihazıdır.

Seri haberleşmenin temelindeki prensip, verinin tek bir hat veya devre üzerinden aynı anda tek bir bit olmak üzere iletilmesidir. Seri port, bilgisayar hattından aldığı 8, 16 veya 32'lik paralel bit paketlerini, 8, 16 veya 32' lik seri bit akımlarına dönüştürür. Seri haberleşme ismini bu prensipten alır; iletilecek bilginin her biti, bir yerden diğerine seri olarak iletilir. Teorik olarak bir seri hat, bir yerden diğerine seri işareti iletmek için yalnızca bir işaret hattı ve toprak olmak üzere iki kabloya ihtiyaç duyar. Seri haberleşme ile 15 metre mesafeye kadar 19600bps hızıyla veri iletimi sağlanır. Lojik 1 için +3 _ +15, lojik 0 için -3 _ -15 Volt arası gerilim iletilir.

3.1.2 Ağ katmanı

Ağ katmanı, verinin ağ boyunca iletilerek doğru adreslere ulaştırılması görevini yerine getirmektedir. Bunun yanı sıra bilginin hedef adrese giderken hangi yolu takip etmesi gerektiğine karar verme, büyük veri paketlerinin parçalara ayrılıp, adreslenerek hedef adrese iletilmesi görevlerini de yerine getirir. Ağa bağlı her terminalin kendine özel atanmış bir ağ adresi vardır. İletişim bu adresler üzerinden gerçekleştirilir. Bu katmanda çalışan en önemli protokol İnternet protokolü yani IP protokolüdür (Çölkesen vd., 2003).

3.1.2.1 İnternet protokolü (IP)

İnternet yapısı, çalışma sistemi, topolojileri ve büyüklükleri birbirinden farklı bilgisayar ağlarının birbirine bağlanmasıyla oluşmuş "ağların ağı" diyebileceğimiz bir bilgisayar ağıdır. Bir ağın, sisteminin kurulması ve işletilmesi bile birçok problemi beraberinde getirirken bu ağların birleşmesiyle oluşan İnternet'in çalıştırılmasının zorluğu aşıkardır. İnternet'i bir arada tutan, çalışmasını sağlayan yapı TCP/IP Transmission Control Protocol - İnternet Protocol; ağlar arasında yönlendirmeyi yapan, ağların birbirleriyle haberleşmesini sağlayan da bu protokol içerisindeki IP İnternet protokolüdür. Şu an kullanılan İnternet protokolü versiyonu 4 tür ve IPv4 olarak da ifade edilir (Postel, J., RFC791).

İnternet protokolü başlık yapısı, 32 bitlik yapıdadır. IP başlığının yapısı aşağıdaki şekilde görüldüğü gibidir (Şekil 3.6). IP başlığı kaynak bilgisayardan

gönderilen datagramların her parçası önüne eklenerek her parçanın kaybolmadan hedef adrese gitmesi sağlanır.

0	4	8	16	19	31
VERSİYON	IHL	SERVİS TİPİ	TOPLAM UZUNLUK		
TANIMLAMA			BAYRAK	BÖLÜMLEME	
TTL	PROTOKOL		BAŞLIK KONTROL		
KAYNAK ADRESİ					
HEDEF ADRESİ					
OPSİYONLAR					

Şekil 3.6 IP Paket Yapısı

İnternet protokol başlığı yukarıdaki tabloda görüldüğü üzere şu bileşenlerden oluşmaktadır.

VERSİYON: 4 bitten oluşmuştur. İnternet protokolünün versiyonunu belirtir.

IHL: 32 bitlik bilgi içerisindeki İnternet başlık uzunluğunu (İnternet header length) verir. 4 bitten oluşmuştur. Değeri en az 5 byte en fazla 20 byte olabilir.

SERVİS TİPİ: Alt ağın istediği datagram tipini belirtir. Önem sırası, uyumluluk, gecikme gibi parametreleri belirler. 8 bitten oluşmuştur.

TOPLAM UZUNLUK: Byte cinsinden toplam datagram uzunluğunu verir. 16 bitten oluşmuştur. Bir datagram en fazla 65.536 bitten oluşabilir.

TANIMLAMA: Parçalara ayrılan verinin herbirinin tanımlanması için gereken tanımlama bilgisidir. 16 bitten oluşmuştur. Datagramda ayrı ayrı giden veri paketlerinin tekrar bir araya getirilmesi bu bilgiyle yapılır.

BAYRAK: 3 bitten oluşmuştur. İlk bit kullanılmamaktadır. İkinci bit parçalara ayrılan datagramın son parçası olup olmadığını belirtir. Son paket hariç tüm paketlerde devamı var bilgisi, son pakette devamı yok bilgisi yollar. Böylece paketin bittiği anlaşılır. Üçüncü bit ise verinin parçalara ayrılıp ayrılmayacağını belirtir. Çünkü bazı hedef bilgisayarlar parçalara ayrılan datagram paketlerini tekrar birleştiremeyebilirler.

BÖLÜMLEME: Bu parçanın datagramın neresinde olduğunu, datagramın hangi parçası olduğunu belirtir. 13 bitten oluşmuştur. 13 bitten oluştuğu için bir datagram en fazla $2^{13} = 8192$ parçadan oluşabilir.

TTL: Datagramın alıcısına belirli bir süre içinde ulaşmaması durumunda yok edilmesini sağlayan bir alandır. TTL alanına, başlangıçta 255 veya daha küçük bir tam sayı terleştirilir. Her yönlendiricide bu alandaki değer bir eksiltir. Ayrıca yönlendiricide paket bir bekleme kuyruğuna alınırsa her geçen saniye TTL alanındaki sayı bir eksiltir. Sayı sıfıra ulaşırsa paket çöpe atılır. Çöpe atan yönlendirici kaynağa bir uyarı paketi gönderir.

PROTOKOL: İnternete protokolde işlenen verinin daha sonra hangi katmana, protokole gideceğini belirtir. 8 bitten oluşmuştur.

BAŞLIK KONTROL: Başlığı oluşturan bilgilerin doğruluğunu denetler. 16 bitten oluşmuştur. Başlık kontrol bilgisi her yönlendiricide denetlenerek hata olmaması sağlanır.

KAYNAK ADRES: Verinin gönderildiği kaynak bilgisayarın ağ adresi ve host adresi bilgisini taşır. 32 bitten oluşur.

HEDEF ADRES: Verinin gönderildiği hedef bilgisayarın ağ adresi ve host adresi bilgisini belirtir. 32 bitten oluşmuştur.

OPSİYONLAR: Gönderilen datagramın güvenlik, yol bilgisi, yönlendirici kaydı gibi bilgilerini belirtir. Toplam 32 bitten oluşmuştur.

3.1.2.2 ICMP protokolü

İnternet kontrol mesaj protokolü, OSI referans modeline göre ağ katmanında çalışan, görevi kaynak bilgisayardan gönderilen verinin hedef bilgisayara gidip gitmediğini kontrol etmek olan, olası anormal durumlarda mesaj üreten bir protokoldür (Postel, J., RFC792).

ICMP gönderilen verilerin durumuyla ilgili birçok mesaj üretir. Bunlar hedef ulaşılamaz, yankı isteği ve cevap, yeniden yönlendirme, yönlendirici tanımlama, yönlendirici talep etme vb. mesajlardır. ICMP mesajları sonsuz mesaj trafiğini engellemek için sadece bir defa gönderilir. Aşağıda bazı ICMP mesajları listelenmiştir.

- Hedef ulařılmaz
- Vakıt geçti
- Parametre problemi
- Yeniden yönlendirme
- Yankı istek
- Yankı cevap
- Kaynak yavařlatma

Örneđin bir yönlendiriciden hedef ulařılmaz mesajı gelmiřse bu hedef adresin yönlendirici tarafından bulunamadıđı veya tanımlanamadıđı anlamına gelir. Bu durumda yönlendirici veri paketini yok eder. Hedef adresin bulunamamasının iki sebebi olabilir. Büyük ihtimalle kaynak bilgisayar olmayan bir adresi hedef olarak göstermiřtir, daha az ihtimalle yönlendirici adrese giden yolu bilmiyordur (yönlendirme tablosunda tanımlı deđildir).

Yankı istek mesajı, kaynak bilgisayarın hedef bilgisayarın yerinde olup olmadığını anlamak için gönderdiđi "ping" komutudur. Bu komuta verilen cevap ise yankı cevap mesajıdır. Kaynak bilgisayara gönderilen yankı cevap mesajı hedef adrese ulařılabildiđini belirtir. Cihazda bir arıza olma durumunda ping oldukça önem kazanmaktadır.

Yeniden yönlendirme (redirect) mesajı yönlendirici tarafından veri kaynađına daha verimli bir yol bulması için gönderilen uyarı mesajıdır.

Vakıt geçti mesajı kaynak sistemden gönderilen verinin hedef adresi bulamaması ve süresinin dolduđunu belirtir.

TÜR (1 byte)	KOD (1 byte)	SAĐLAMA TOPLAMI (2 byte)
VERİ		

řekil 3.7 ICMP Mesaj Paket Formatı

ICMP mesajında (řekil 3.7), mesajın türü, kodu ve sađlama toplamının ardından veri bölümü gelmektedir. Bu protokol veri iletimi için kullanılsa da taşıdıđı veri ile ađ hakkında gerekli bilgiler hesaplanır.

3.1.2.3 Adres çözümleme protokolü (ARP)

Ağa bağlı olan her bilgisayarın bir IP adresi vardır. Ağ üzerindeki bilgisayarlar, yönlendiriciler ve diğer yönlendirme mekanizmaları verileri IP numaralarına göre değil Ethernet adreslerine göre gönderirler; IP adreslerine bakmazlar. Veri gönderen kaynak bilgisayarın Ethernet kartı tarafından ağa ? numaralı IP'nin Ethernet adresi nedir diye soru sorulur. Bu veri tüm ağa broadcast yapılır. IP numarasının sahibi bu soruya cevap verir. Böylece verinin gönderileceği hedef bilgisayarın Ethernet adresi belirlenmiş olur. Tüm bu soruların sorulması ve cevapların verilmesi işlerini yöneten Adres Çözümleme Protokolüdür (Plummer, D. C., RFC826).

3.1.2.4 Ters adres çözümleme protokolü (RARP)

Ters adres tanımlama protokolü bir önceki protokolün tersi işlemi yapar. Ethernet adresi verilen bilgisayarın IP adresini tanımlar. Bu durum hard diski olmayan bazı makineler için geçerlidir. Bu makineler ilk açıldıklarında sadece Ethernet adreslerini bilirler ve diğer makinelere Ethernet adresini söyleyerek kendisinin IP numarasının ne olduğunu sorarlar. Bu sorunun muhatabı ters adres çözümleme protokolüdür. Bu protokol bazı durumlarda yeterli olmaz. Bu nedenle bu protokolün işini yapan fakat biraz daha geliştirilmiş bir protokol olan BOOTP protokolü de kullanılmaktadır.

3.1.3 Taşıma katmanı

Taşıma katmanının temel işlevi, oturma tabakasından veriyi alıp, ihtiyaç duyulduğunda küçük bileşenlere ayırıp ağ katmanına aktararak, diğer uca bu parçaların doğru bir şekilde ulaşmasını sağlamaktır. Bunun dışında, bu işlev oturma tabakasını, donanım teknolojisindeki kaçınılmaz gelişmelere karşı izole ederek, işlevini verimli olarak yerine getirmesini sağlar.

Normal şartlar altında, taşıma katmanı, oturma katmanı tarafından ihtiyaç duyulan her taşıma bağlantısı için bir sanal ağ bağlantısı oluşturur. Eğer taşıma

bağlantısı yüksek bir kapasite isterse, taşıma katmanı birçok ağ bağlantısı oluşturup, kapasiteyi artırmak için veriyi bu bağlantılara paylaştırır. Öte yandan, farklı ağ bağlantılarının oluşturulmasının maliyeti artırdığı durumlarda taşıma katmanı çeşitli taşıma bağlantılarını bir ağ bağlantısı üzerinde maliyeti azaltmak için birleştirebilir. Tüm durumlarda taşıma katmanı birleştirme işinin oturum katmanına yansımaması için gereklidir.

Taşıma katmanı ayrıca oturum katmanına ulaşacak ağ kullanıcılarına ne tip servisler sunulacağına karar verir. Taşıma bağlantısının en popüler tipi TCP olarak adlandırılan ve veriyi gönderildiği sıra ile hatasız uçtan-uca ulaştıran kanaldır. Ancak, diğer tip taşıma, UDP olarak adlandırılan; servis ve taşıma bilgisi ayrılmış mesajları değişik bölgelere ileten ve hedefine ulaştırma konusunda herhangi bir garanti vermeyenidir. Servis tipi bağlantı sağlandığında belirlenir. Taşıma katmanı, gerçek bir kaynaktan hedefe veya uçtan uca katmandır. Başka bir ifadeyle, kaynak sistemde çalışan bir program, mesaj başlıkları ve denetim mesajlarını kullanarak hedef sistemdeki benzeri bir programla konuşur.

Birçok bilgisayar, üstünde birden fazla programı çalıştırır, yani sisteme giren ve çıkan birçok bağlantı vardır. Bu yüzden hangi mesajın hangi bağlantıya ait olduğunun belirlenmesi için bir metoda ihtiyaç duyulur. Taşıma başlığı bu bilginin koyulabileceği bir yerdir.

Değişik mesajları bir kanal içinde birleştirmenin yanında, taşıma katmanı ağ boyunca bağlantıların kurulması ve kaldırılmasını da takip etmelidir. Bu, bir bilgisayar üzerinde kiminle konuştuğunu tarif edecek bir tür isimlendirme mekanizması gerekliliğini doğurur. Ayrıca hızlı bir bilgisayarın yavaş bir bilgisayarı aşmaması için bilgi akışını düzenleyecek bir mekanizmanın olması gereklidir. Her ne kadar ikisine de aynı prensipler uygulansa da uçlar arasındaki akış denetimi anahtarlar arası akış denetiminden ayrıdır.

3.1.3.1 Kullanıcı datagram protokolü (UDP)

Gelişmiş bilgisayar ağlarında paket anahtarlama bilgisayar iletişimi bir datagram modu oluşturabilmek için UDP protokolü yazılmıştır (Şekil 3.8). Bu protokol minimum protokol mekanizmasıyla bir uygulama programından diğerine

mesaj göndermek için bir prosedür içerir. Paketin teslim garantisi yoktur. Geniş alan ağlarında (WAN) ses ve görüntü aktarımı gibi gerçek zamanlı veri aktarımlarında UDP kullanılır. Ayrıca UDP bağlantı kurulum işlemlerini, akış kontrolü ve tekrar iletim işlemlerini yapmayarak veri iletim süresini en aza indirir (Postel, J., RFC768).

UDP'yi kullanan genel protokoller DNS, TFTP, ARP, RARP ve SNMP protokolleridir. Uygulama programcılarını birçok zaman UDP'yi TCP'ye tercih eder. Çünkü ağ üzerinde fazla bant genişliği kaplamaz.

UDP güvenilir olmayan bir aktarım protokolüdür. UDP protokolü ağ üzerinden paketi gönderir ve gidip gitmediğini takip etmez ve paketin yerine ulaşmış olup olmadığına onay verme yetkisi yoktur.

UDP, datagramların belirli sıralara konmasının gerekli olmadığı uygulamalarda kullanılmak üzere tasarlanmıştır. TCP'de olduğu gibi UDP'de de bir başlık vardır. Ağ yazılımı bu UDP başlığını iletilecek bilginin başına koyar. Ardından UDP bu bilgiyi IP katmanına yollar. IP katmanı kendi başlık bilgisini ve protokol numarasını yerleştirir (bu sefer protokol numarası alanına UDP'ye ait değer yazılır). Fakat UDP, TCP'nin yaptıklarının hepsini yapmaz. Bilgi burada datagramlara bölünmez ve yollanan paketlerin kaydı tutulmaz. UDP'nin tek sağladığı port numarasıdır. Böylece pek çok program UDP'yi kullanabilir. Daha az bilgi içerdiği için doğal olarak UDP başlığı TCP başlığına göre daha kısadır. Başlık, kaynak ve varış port numaraları ile kontrol toplamını içeren tüm bilgidir. Veri kısmı dördün katı olmalıdır. Eksik alan sıfır bilgisi ile doldurulur.

Kaynak Port (16bit)	Hedef Port (16bit)
Uzunluk (16bit)	Hata Kontrolü (16bit)
VERİ	

Şekil 3.8 UDP Paket Formatı

3.1.3.2 Aktarım denetim protokolü (TCP)

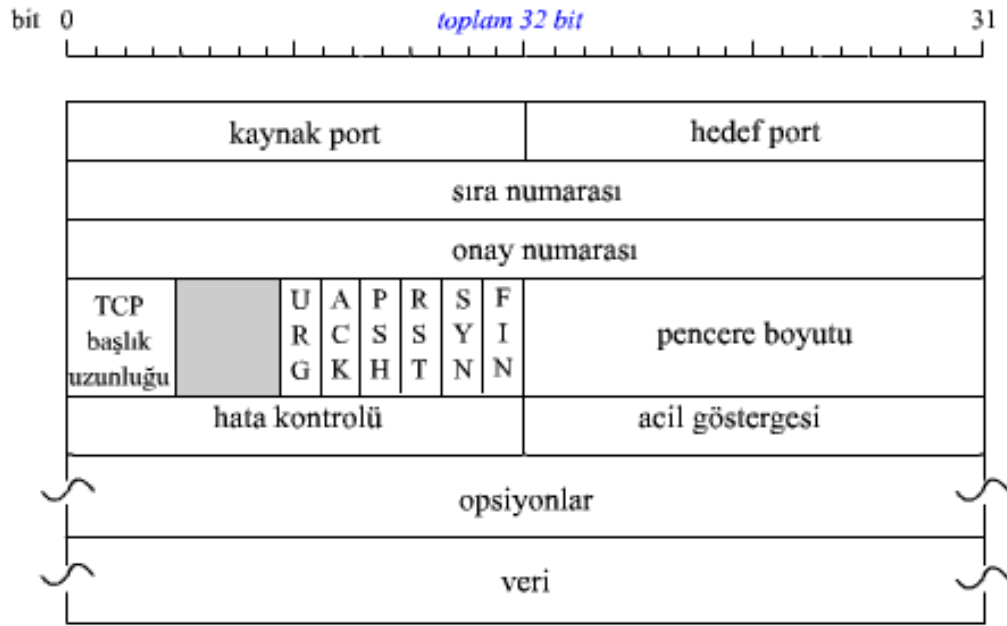
İnternet iletişim katmanında, bağlantı yönlendirmeli ve bağlantısız (datagram) olmak üzere iki tip protokol tanımlanmıştır. İnternette kullanılan bağlantı yönlendirmeli protokol Aktarım Denetim Protokolü (Transmission Control Protocol - TCP)'dür (Şekil 3.9). İnternet üzerindeki ağların yapıları, topolojileri, paket boyutları, hızları, gecikmeleri farklıdır. TCP bu farklı ağları birbiriyle uyumlu olarak çalıştırmak için tasarlanmıştır. TCP-IP protokolünde İnternet katmanı gönderdiği verinin istenilen adrese gitmesini garanti etmez. Bu garantiyi iletim katmanında çalışan TCP verir ve gerekirse yeniden iletilmesini sağlar. Ayrıca İnternet protokol tarafından bağlantısız datagram olarak gönderilen paketler hedef adrese farklı sıralarla ulaşabilirler. Bu paketleri gerçek sıralarına göre düzenleyen de TCP'dir (Postel, J., RFC793). TCP nin görevleri şöyle sıralanabilir:

- Bir bağlantının (connection) kurulması ve sonlandırılması
- Güvenilir (reliable) paket dağıtımının sağlanması
- Akış kontrolü (flow control) ile hostlarda veri taşmasının (overflow) önlenmesi
- Bozulmuş ya da iki defa gönderilmiş verinin düzeltilmesi (error recovery)
- Alıcı host içerisinde birçok uygulama arasında demultiplexing yapılması

İletişim kontrol protokolünde veri gönderen kaynak ve veriyi alan hedef bilgisayar socket adı verilen bir son nokta belirler. Her socketin bir socket numarası vardır. Bilgisayarın sahip olduğu IP numarası ve socket numarasının tümüne birden port adı verilir. Her bir port gerçekte üst katman tarafından sağlanan bir servisi simgeler. Örneğin Telnet için port 23, FTP için port 21, bir başka uygulama için port 3 gibi.

Tüm TCP bağlantıları çift yönlü (full duplex) ve noktadan noktaya bağlantı sağlarlar. TCP ile yapılan bağlantılarda hedef ile kaynak arasında çift yönlü iletişim yapılabilir fakat TCP çoklu gönderim (multicasting) ve yayın (broadcasting) haberleşmeye izin vermez.

TCP gönderilen veri yığını, mesaj şeklinde değil byte şeklinde gruplandırır. Örneğin dört adet 512 byte uzunluğundaki veri; dört adet 512 byte ile iki adet 1024 byte ile veya 1 adet 2048 byte ile gönderilebilir. Verinin uzunluğunu TCP ayarlar.



Şekil 3.9 TCP Paket Yapısı

TCP gönderdiği her paketin başına kendi başlık bilgisini ekler. TCP başlık formatı 32 bitlik yapıda sıralanmıştır, uzunluğu değişkendir.

KAYNAK PORT: Veriyi gönderen Host'un port numarasını tanımlar

HEDEF PORT: Veriyi alan Host'un port numarasını tanımlar.

SIRA NUMARASI: Gönderilen segmentin sıra numarasını gösterir. TCP 32 bitlik sıra numarası tanımlayabilir.

ONAY NUMARASI: TCP gönderilen her verinin onayını almak yerine birkaç veri için bir onay verisi alır. Bu alan gönderilen verinin onay numarasını göstermektedir.

TCP BAŞLIK UZUNLUĞU: 4 bitlik bu alan, TCP başlığındaki 32 bitlik kelime sayısını gösterir. Bu yolla segment içindeki verinin başladığı alan tespit edilir. Bu alana ihtiyaç duyulmasının nedeni opsiyon ve veri alanları içeriğinin değişken uzunlukta olmasıdır.

Başlık alanı içerisinde gri renkle gösterilen 8 bitlik alan gelecekte kullanılmak üzere saklanmıştır. Şu anda tümü 0 bitlerinden oluşmuştur.

URG - ACİL (URGENT): Acil bayrağının 1 olması acil göstergesi (urgent pointer) bölümünün kullanımda olduğunu gösterir

ACK - ONAY (ACKNOWLEDGEMENT): Onay bayrağının 1 olması Onay numarası alanının kullanımda olduğunu gösterir.

PSH - İTME (PUSH): İtme bayrağı; gönderen, TCP'ye gönderilecek veriyi hemen göndermesi için emir verir ve bu emir alt katmanlara iletilir. Benzer biçimde, alt katmandan gelen verinin hemen üst katmana aktarılması için TCP 'yi zorlar. Normalde TCP bu işlemi kendi önceliğine göre yapar ancak itme bayrağı ile bu öncelik değiştirilebilir.

RST - SIFIRLAMA (RESET): Sıfırlama bayrağı sorunlu veya kopmak üzere olan bağlantıları ilk duruma getirir.

SYN - SENKRONİZE (SYNCHRONIZE): Senkronize bayrağı ilk bağlantı kurulurken kaynak ve hedef Host'un aynı adımda çalışmasını sağlar.

FIN - BİTİŞ (FINISH): Bitiş bayrağı başka verinin kalmadığını; bağlantının sonlandırılabilceğini belirtir.

PENCERE BOYUTU (WINDOW SIZE): 16 bitlik pencere alanı onay (Acknowledgement) alanından, göndericinin iletmek istediği alana kadar iletilecek veri baytlarının sayısını verir. Sıra numarası ve onay numarası ile birlikte bu alan TCP'nin akış denetimini sağlar.

HATA KONTROLÜ (CHECKSUM): 16 bitlik bu alan veri transferinde başlığın bozulup bozulmadığını kontrol eder. Eğer hesaplanan hata kontrolü alanı, paketteki hata kontrolü alanı ile aynı değer olmaz ise paket atılır.

ACİL GÖSTERGESİ (URGENT POINTER): 16 bitlik acil göstergesi, paketin sıra numarasından hesaplanan değişken başlangıç değeridir. Bu alan veri içinde acil bilginin nerede bulunacağını belirtir. Gerçekte, bu alandaki değer acil verinin son baytından sonraki değeridir. Acil veri, üst katman protokolü için önemli olan veridir. Genelde bu alandaki değer klavye kesilme sinyali gibi bilgiyi denetler. TCP bu veri üzerinde herhangi bir işlem yapmaz.

OPSİYONLAR (OPTIONS): Değişken değerli seçenekler alanı, eğer varsa, acil gösterge alanından sonra gelir. En çok kullanılan seçenek olan "en uzun segment boyu" seçeneği bağlantı kurma sırasında TCP 'nin kabul edebileceği en büyük band içi segmenti belirtir.

3.1.4 Oturum katmanı

Oturum katmanı, değişik makinelerdeki kullanıcıların birbirleri arasında oturumlar açmasını sağlar. Bir oturum taşıma katmanının yaptığı gibi sıradan veri taşıma işini gerçekleştirdiği gibi, bazı uygulamalarda çok yararlı gelişmiş hizmetler de sunar. Bir oturum bir kullanıcının uzaktaki zaman-paylaşımlı bir sisteme bağlanmasını (Log on, log in) veya iki makine arasında dosya transferi yapmasını sağlar.

Oturum katmanının sunduğu hizmetlerden biri de sistemlerin karşılıklı iletimlerinin yönetimidir. Oturumlar aynı anda tek yönlü veya aynı anda çift yönlü veri akışına izin verebilirler. Eğer trafik tek yönlü ise oturum katmanı iletim sırasının kimde olduğu konusunda yardımcı olur.

İlgili diğer bir oturum hizmeti token yönetimidir. Bazı protokoller için, her iki tarafın aynı anda aynı işlevi yerine getirmeye çalışmaması çok önemlidir. Bu aktiviteleri yönetmek için oturum katmanı taraflar arasında değiştirilebilecek tokenlar sağlar. Token'a sahip taraf kritik uygulamayı çalıştırma hakkına sahip olur.

Diğer bir oturum servisi senkronizasyondur. Ortalama çökme süresinin 1 saat olduğu bir ağda, iki makine arasında iki saatlik bir dosya transferi sırasında ortaya çıkacak problemleri ele alalım. Her çökmeden sonra dosya transferi yeniden en baştan başlayacak ve bu yüzden dosya transferi belki de hiç gerçekleştirilemeyecektir. Bu problemi ortadan kaldırmak için, oturum katmanı veri akımının içine kontrol noktaları yerleştirir, böylelikle bir çökmeden sonra en son kontrol noktasından sonraki veri gönderilir.

3.1.5 Sunum katmanı

Sunum katmanı, kullanıcıların problemlere kendi başlarına çözüm bulmaları yerine onlara yeterli bir genel çözüm sunar. Kısaca, diğer alt katmanların aksine, bit'leri bir uçtan diğerine güvenilir bir biçimde iletimleri ile ilgilenmek yerine oturum katmanı iletilen bilginin söz dizimi ve semantiği ile ilgilenir.

Sunum servislerine örnek olarak; standart, üzerinde anlaşılan bir şekilde veriyi kodlamaktır. Birçok kullanıcı programları, rastgele bit dizilerinin kendi

aralarında deęişimini gerekleřtirmez. řahıs adları, tarih, para gibi řeyleri deęiřtirirler. Bu bařlıklar, karakter dizileri, tamsayılar, kayan nokta numaraları gibi daha basit veri yapıları olarak ifade edilirler. Deęiřik bilgisayarlar karakter dizileri ve tamsayıları ifade etmek iin deęiřik kodlar kullanırlar. Bu bilgisayarlar arasında veri deęiřimini standartlara uygun olarak yerine getirmek sunum katmanının iřidir.

Sunum katmanı ayrıca bilginin sunulmasının diđer ynleri ile de ilgilidir. rneęin veri sıkıřtırması iletilmesi gereken bit sayısını artırmak iin kullanılabildięi gibi kriptografi gvenlik ve kullanıcı doęrulaması iin sık sık kullanılır.

3.1.6 Uygulama katmanı

Uygulama katmanı oka ihtiya duyulan birok protokol ierir. Dnyada birbirine uyumsuz yzlerce terminal, ubirim tipi vardır. rnek olarak her biri farklı ekran dzenleri, metin silme ve dzenleme iin farklı escape dizileri, imle konumlandırması vs. kullanan deęiřik ubirimlerle alıřan bir tam ekran metin editrn ele alalım.

Bu problemi ozmenin yolu editrlerin ve diđer programların yazabildięi sanal bir aę ubirimi oluřturmaadır. Her ubirim tipini karřılamak iin, sanal ubirimin fonksiyonlarının gerek ubirim zerine eřleřmesini saęlamak iin bir yazılım yazılmalıdır. rneęin bu yazılım, editr sanal ubirimin imlecini sol st křeye konumlandırıđında, yazılım gerek ubirimde imlecin asıl konumuna yerleřimi iin dzgn komut dizisini iřlemelidir. Tm sanal ubirim yazılımları uygulama katmanındadır.

Uygulama katmanının diđer bir iřlevi ise dosya transferidir. Deęiřik dosya sistemleri, deęiřik dosya isimlendirme tanımlamalarına, metin bilgisinin temsili iin deęiřik yntemlere sahiptir. Deęiřik dosya sistemlerinden dosya transferleri bu uyumsuzlukları ortadan kaldırmayı gerektirir. Bu iř, yine, elektronik posta, dizin taraması ve diđer zel ve genel amalı iřlevlerde yapıldıęı gibi uygulama katmanına aittir.

Bu katmanda birok protokol alıřmaktadır. Bizim iin en nemli olanları HTTP, FTP ve DHCP'dir. HTTP, arayz oluřturmak iin ideal bir protokoldr. FTP, yazılımı gncellemek iin kullanılabılır. FTP, dosya iletim protokoldr ve

hazırlanan dosyaların gömülü web sunucunun IP adresi ile sunucuya gönderilmesi işlemleri için yürütülmesi gereken protokoldür (Postel, J., RFC959). DHCP ise ağa bağlanan cihazın IP adresini alması için gereklidir.

3.1.6.1 Dinamik istemci ayarlama protokolü (DHCP)

Dinamik ana bilgisayar biçimlendirme protokolü olarak adlandırılır (Şekil 3.10). Ağdaki IP adresi dağıtma ve verimli şekilde kullanımı için geliştirilmiştir. DHCP'li bir ağda istemci çalıştığında IP almak için istekte bulunur. Sunucuda veritabanını inceleyerek istemciye geçici süreyle bir IP adresi tahsis eder ve istemciye bir mesaj ile bildirir (Droms, R., RFC1541).

İstemci ilk çalıştığında 0.0.0.0 IP adresini alır. Daha sonra UDP 67 portundan UDP 68 portuna yayın yaparak ağa bağlanmak için ihtiyaç duyduğu IP adresini ister. DHCP sunucu, içerisinde istemcinin MAC adresinin de olduğu bu isteği alır ve uygun bir adresi DHCP paketi (Şekil 3.14) ile geri yollar. İstemci kendi MAC adresiyle gelen paketi alır ve IP adresini kaydederek ağa bağlanır. Bütün işlemler için başlık formatı aynıdır ve veriler UDP formatı ile taşınır.

İşlem (8 bit)	Başlık Türü (8 bit)	Başlık Uzunluğu (8 bit)	Başlık Seçenekleri (8 bit)
İşlem ID (32 bit)			
Saniyeler (16 bit)		Bayraklar (16 bit)	
İstemci IP adresi (32 bit)			
Sizin IP adresiniz (32 bit)			
Sunucu IP adresi (32 bit)			
Ağ Geçidi IP adresi (32 bit)			
İstemci Donanım Adresi (16 Byte)			
Sunucu Ev Sahibi İsmi (64 Byte)			
Ön Yükleme Dosya Adı (128 Byte)			
Seçenekler (312 bite kadar)			

Şekil 3.10 DHCP Paket Yapısı

3.1.6.2 HTTP protokolü

HTTP, WWW bileşenlerinin ağ üzerinden transferi için kullanılan temel protokoldür. HTTP, web dokümanlarını, ilişkili resimleri ve diğer nesnelere İnternet üzerinden tarayıcılara aktarır. FTP, Gopher ve diğer İnternet protokolleri gibi istemci-sunucu tabanlı bir protokoldür. Bir başka ifadeyle, kullanıcı rolündeki tarayıcı, bilgi isteğini HTTP sunucusuna (HTTP Sunucu, Web Sunucu) iletir. Bu isteği değerlendiren HTTP sunucusu, istenilen bilgiyi HTML formatında kullanıcıya yollar. HTTP protokolü, sunucu tarafında yapılan ayarlamalarla güvenli iletişim kanallarını da açabilir. HTTPs denilen bu sistemde, güvenli olarak iletilmesi istenilen dokümanlar şifrelenerek (encrypted) kullanıcıya gönderilir. Tarayıcı tarafında, bu şifrelemeyi çözen anahtarlar bulunmaktadır (Fielding, R., RFC2616).

HTTP, daha önce de belirtildiği gibi, sürekli bağlantı gerektirmeyen bir protokoldür. HTTP sunucular, istekte bulunan bilgiyi paketlere bölerek gönderir. İki paket arasında geçen zaman değişebilir. Bu özellik sayesinde, sunucu birçok isteğe birden yanıt verebilir.

HTTP sunucular, kullanıcı (client) tarafında hangi bilgilerin alındığından habersizdir. Eğer bilgiler alınamamışsa, kullanıcı bilgileri yeniden almak için istekte bulunmalıdır. Uygulamada, bu işlem tarayıcının Refresh (yenile) düğmesine basılarak yapılır.

HTTP protokolünde metin dosyalarının yanında .gif ve .jpg gibi resim dosyaları da taşınabilmektedir. Bu protokol her bilgisayarda bulunan İnternet tarayıcıları tarafından kullanılmaktadır. Bu sayede sisteme erişim herhangi bir yerden sağlanabilmektedir.

Tarayıcının adres satırına cihazın IP adresi yazılarak erişim sağlanır. Tarayıcı TCP 80 portundan bir soket açarak cihaza GET komutuyla başlayan düz metin şeklinde bir mesaj gönderir. Gönderilen mesaj

GET /index.htm http/1.0 <CR><LF>

şekindedir. Bu mesajı alan cihaz tarayıcının hangi protokolü kullandığını ve hangi dosyayı istediğini öğrenir ve kendinde bu dosya varsa tarayıcıya (istemciye) gönderir. Eğer istenen dosya kendinde yoksa 404 kodu ile hata mesajı gönderir. Tarayıcılarda GET komutunun yanında POST komutları da mevcuttur.

3.1.6.2.1 HTML

(Hyper Text Markup Language) Web sayfası hazırlama ve formatlama dilidir. Bu dil, daha çok, yazılı bir dokümanı formatlamak ve bir objeden başka bir objeye linkler sağlamak ile ilgili komutlar içerir. HTML dosyası ses, yazı, video, animasyon, script içerir (Berners-Lee, T., RFC1866). HTML, HTTP ve ilgili diğer protokolleri kullanabilmek için renkli ve görsel kullanıcı arayüzleri hazırlanmasını olanaklı kılar. Kodlar metin olarak yazılır, gönderilir ve tarayıcı tarafından yorumlanır. Sayfaya görsel öğelerin yanında tablolar ve başka sayfalara bağlar (link) yerleştirilebilir.

HTML statik bir dildir. Dinamik hale getirmek için JavaScript programlama dili kullanılır. Tarayıcıdan sunucuya veri gönderilecek ise PHP, ASP, JSP veya CGI programlama dillerinden biri kullanılabilir. Projemizde HTML, JavaScript ve CGI komutlarını kullanmaktayız.

Web sayfaları yazılırken programlamayı kolaylaştırmak ve hızlandırmak için Dreamweaver ve resim dosyaları yaratmak için Photoshop programı kullanıldı.

HTML yapısına sahip bir örnek aşağıda görülmektedir.

```
<head>
<title>ORNEK</title>
<meta name="description" content="HTML sayfası için bir
kullanım örneği">
</head>
<body bgcolor=white>
    //Sayfa içeriği
</body>
</HTML>
```

3.1.6.2.2 JavaScript

Netscape Navigator 2.0 ile birlikte geliştirilen JavaScript dili istemci taraflı (client-side) script dilidir. Web sayfalarında dinamik içerik sağlamak ya da kullanıcıyla iletişim kurmak için kullanılır.

İlk başta yalnızca Netscape tarayıcıları için tasarlanan JavaScript, daha sonra diğer tarayıcılar tarafından da desteklenen genel bir scripting dili haline gelmiştir. Şu anda Navigator 4.0 tarafından desteklenen JavaScript 1.2 sürümü vardır. Microsoft İnternet Explorer 3.0 versiyonu ile JavaScript 1.0'ı desteklemektedir (Anonim D).

JavaScript kendi başına bir script dilidir. Genellikle HTML sayfalarında kullanıcı denetimleri eklemek, formları kontrol etmek, hesaplamalar yapmak gibi işler için kullanılır.

JavaScript kullanımının iki ana alanı vardır: istemci taraflı ve sunucu taraflı script yazmak. Şu anda HTML sayfalara gömülü yazılan scriptlerin çoğu JavaScript kullanılarak yazılmaktadır. Bunlar da genellikle kullanıcı bilgilerinin doğru girilip girilmediği ya da kullanıcıya veri girişinde ya da menü kullanımında seçenekler sağlamaktır. Örnek bir JavaScript aşağıda görülmektedir.

```
<HTML><HEAD>
<SCRIPT LANGUAGE="JavaScript">
<!--
function AlertTest( ){
alert("Dikkat!");
}
// -->
</SCRIPT>
</HEAD><BODY>
```

JavaScript ile sayfaya dinamik bir yapı eklenir. Hesaplamalar ve hareketli yapılar eklenir.

3.1.6.2.3 CGI Programlama

CGI (Common Gateway Interface), web servisleri ile bu servislerin dışındaki programlar arasında etkileşim (ortak çalışma) platformu oluşturmak için geliştirilmiş bir standarttır. CGI, aslında bir programdır. Web'in statik yapısına, HTML kodu içinden çağrılan CGI programları dinamik bir nitelik kazandırmaktadır. CGI programları gerçek zamanlı çalışırlar. CGI'ların önemli uygulama alanlarından biri de, web üzerinde doldurup gönderilen formlar üzerindeki bilgileri sunucu tarafında değerlendirip kullanıcıya cevabı göndermektir (Anonim C).

Bir CGI programı genel olarak şu 4 adımdan geçer, ancak bunların bazıları her uygulamada olmayabilir:

- Kullanıcıdan bilgi alma
- Alınan verilerin bir veritabanına işlenmesi
- İstenen verilerin veritabanından bir sorgu sonucu alınması
- Kullanıcıya işlenmiş bilgilerin gönderilmesi

TCP/IP Stack içerisinde bulunan HTTP protokolü CGI dosya yapısına cevap verecek şekilde programlanmıştır. Sayfa içerisinde “%” simgesiyle başlayan dinamik içeriklerin işlenebilmesi için dosyanın CGI uzantısıyla kayıt edilmesi gerekmektedir. Stack tanımlamaları içerisinde anlatılmış olan HTTPGetVar ve HTTPExecCmd fonksiyonları, CGI uzantılı dosya içerisinde karşılaştığı dinamik içerikler için çağrılır ve ilgili pinin değeri dinamik içeriğin tanımlandığı yere yazılır. HTM uzantılı dosyalar statik içerikli olmalıdır. Dinamik içeriğe sahip olan dosyaların uzantılarının HTM yerine CGI olarak kayıt edilmesi yeterlidir.

3.2 Gömülü Web Sunucu

İnternet'e bağlantı için gerçekleştirilmesi gereken bu kurallar bilgisayarların tümünde desteklenmektedir. Gerekli yazılım ve donanım ürünleri ile günümüz bilgisayarları İnternet'e sorunsuzca bağlanabilmektedirler. İnternet üzerinden bir sistemin denetimi için bilgisayarlar rahatlıkla kullanılabilir. Ancak bazı sorunlara dikkat çekmek gerekmektedir. Bilgisayarların enerji ihtiyaçları fazladır. Şebeke geriliminin sürekli sağlanması gerekir. İnternet üzerinden gelebilecek tehditlere ve

özellikle virüslere karşı yüksek güvenlik önlemleri alınmalıdır. Bilgisayarın çökmesi ihtimalinin de göz önünde bulundurulması gerekir. Tüm bu sebeplerden ötürü denetim sisteminde sürekli ve sorunsuzca çalışması gereken bir biriminin bilgisayar olması birçok problem yaratacak ve maliyeti arttıracaktır. Bilgisayarın İnternet'e bağlanması için gereken tüm protokolleri gerçekleştirip, sistemi sorunsuzca İnternet erişimine açacak alternatif sistemler arasında "gömülü sistemler" mevcuttur (Ahmed, I., vd. 2004).

Gömülü sistemler istenilen amaca yönelik tasarlanmış, küçük boyutlu ve düşük maliyetli denetim kartlarıdır. Bir cihazın ya da aletin içinde bulunan ve bir program çalıştıran denetleyici veya karmaşık bir mantıksal sistemdir. Cihaz görülür fakat onu çalıştıran ve işini mantık çerçevesinde sorunsuzca yapan denetleyici görünmez. Gömülü sistemler nadiren çökerler ve herhangi bir güncellemeye gerek duymayacak şekilde bir bütün olarak üretilirler (Insam, E., 2003).

Mikrodenetleyici tabanlı gömülü sistemler, cep telefonundan çamaşır makinesine kadar çok çeşitli elektronik aletin içinde bulunmaktadır. Ancak mikrodenetleyicilerin bağlanabilirlikleri sınırlıdır. Bilgisayarın RS232 portu ile bağlantı yapılabilmektedir. Ancak bu bağlantı 10 mt.'lik mesafede kullanılabilir. Mikrodenetleyicilerin bağlanabilirlik özelliklerini arttırmak için Ethernet işlemcileri kullanılabilir. Ethernet işlemcileri ve gerekli protokolleri işleyebilecek yazılım ile mikrodenetleyiciler İnternet'e bağlanabilmektedirler. Mikrodenetleyici ve Ethernet işlemcisinden oluşan elektronik kart "gömülü web sunucu" olarak anılacaktır. Gömülü web sunucular bilgisayarlarla kıyaslandığında oldukça düşük maliyetlidirler. Ayrıca 5V ile çalışabilecek kart düşük enerji ihtiyacı ile sorunsuzca iletişime devam edebilecektir. Ancak enerji kayıplarının denetimi yapılacak sistemi etkileyeceği unutulmamalıdır.

Mikrodenetleyici içine yüklenecek programın hazırlanması için gerekli yazılım Microchip firmasının MPLab programıdır. C kodları ile hazırlanabilecek program MPLab programıyla mikrodenetleyiciye uygun hale dönüştürülecektir. Programın derlenmesi MPLab programına entegre olan PIC C 18 programıdır. Ücretli olan bu program yine Microchip firmasından temin edilebilir. İnternet'e bağlantı için gerekli programın hazırlanmasının zorluğu aşıkardır. Gerekli program yine Microchip firması tarafından 18 serisi ve üstü mikrodenetleyicilerde çalışacak

şekilde hazırlanmış ve ücretsiz olarak TCP/IP Stack adı altında sunulmuştur. Hazırlanacak donanıma uygun olarak düzenlenmesi gereken program üzerinde yapılacak değişiklikler için yine MPLab ve PIC C 18 programlarına ihtiyaç duyulmaktadır.

Gerekli donanım için 18 serisi mikrodenetleyiciler kullanılabilir. Bu serideki mikrodenetleyiciler Ethernet işlemcisi ve denetimi yapılacak sistem için gerekli giriş çıkış sayısına, işlemci hızına ve bellek miktarına sahiptir. Ethernet işlemcisi, Microchip firması tarafından üretilmiş ENC28J60 işlemcisi veya Realtek firmasının RTL8019AS işlemcisi olabilir. Her iki işlemcinin mikrodenetleyici tarafından denetimi için gerekli yazılım modülleri TCP/IP Stack içerisinde mevcuttur. Ethernet işlemcisi ağa RJ45 jackı ile bağlanır. İşlemci ile RJ45 arasında ağdaki sinyalleri gürültüden arındıracak manyetik filtre gurubu bulunmalıdır. Manyetik filtre gurubu bobinlerden oluşmaktadır ve hassas bir yapıya sahiptir. Gömülü web sunucu direk olarak bir bilgisayara bağlanacaksa bilgisayarın topraklamasının iyi yapılmış olmasına dikkat edilmelidir. Oluşabilecek kaçaklar bu manyetik gurubuna zarar vermektedir. Manyetik gurubunu da içeren RJ45 jackları Ethernet kartlarında kullanılmak üzere üretilmektedir. Yurt içinde satışı olmayan bu jack modüllerinin temini mümkündür ancak bağlantı şekillerine ve kullanılması gereken ek elektronik malzemenin toleranslarına dikkat etmek gerekmektedir. Projenin zahmet veren kısımlarından biri bu bağlantıdır. Toleranslara özen gösterilmemesiyle oluşabilecek sinyal kayıpları iletişimde sorunlar yaratacaktır.

Gömülü web sunucunun yazılımının seri kablo ile direk bilgisayar üzerinden güncellenmesi ve gözlenmesi mümkündür. Ethernet hattında oluşabilecek sorunlar nedeniyle gerekli olabilecek bu bağlantı için gömülü web sunucu üzerinde MAX232 entegresi bulunmalıdır. MAX232 seri iletişim için gerekli gerilim seviyesi dönüşümünü sağlamaktadır.

Gömülü web sunucu, İnternet üzerinden denetim için HTML tabanlı bir arayüze sahip olmalıdır. Bu arayüz web sitesi olarak anılacaktır. Web sitesinin aynı anda bağlanabilecek birden çok istemci tarafından görülebilmesi için gömülü web sunucunun belli miktarda hafızaya ve hıza ihtiyacı vardır. TCP/IP Stack'i işlemek için hızının ve hafızasının önemli kısmını kullanan mikrodenetleyicinin bu ihtiyaca

cevap verebilmesi için ek hafıza birimlerine ihtiyaç duyulmaktadır. Bu ihtiyacı karşılamak üzere gömülü web sunucu üzerinde seri EEPROM kullanılabilir.

EEPROM bellek miktarı ve işlemci yeterlilikleri gömülü web sunucunun limitlerini tanımlamaktadır. EEPROM, sunucuya yüklenecek web sitesinin büyüklüğünü, işlemci yeterlilikleri ise işlem hızının yanında kullanılacak giriş-çıkış sayısını, ek bağlanabilirlik özelliklerini (CAN, RS485, vb.) ve aynı anda gömülü web sunucu ile kurulabilecek HTTP bağlantı sayısını belirlemektedir.

Gömülü web sunucuya erişim, parola korumalı olmalıdır. İstenmeyen erişimler bu sayede sınırlandırılabilir. İnternet Servis Sağlayıcıların verdiği güvenliğin yanında kullanıcı adı ve parola ile gömülü web sunucuya erişim daha güvenli hale gelecektir. Şifreleme ile ilgili bilgilere “Programlama” bölümünde değinilecektir.

Bu çalışmada gömülü web sunucu olarak Modtronix firmasının SBC68EC modeli kartı kullanılmıştır.

3.2.1 SBC68EC Gömülü web sunucu

SBC68EC modeli kart, PIC18F6680, RTL8019AS, 24LC512 ve MAX202E entegrelerinden oluşmaktadır (Şekil 3.11). 10Mbps Ethernet ve RS232 arayüzleriyle haberleşebilmektedir. TCP/IP Stack ayarlarıyla birlikte HTTP Web Sunucu, Mail Client, Ethernet-RS232 ve Ethernet-RS485 arayüz dönüştürücüsü, Web Sunucu üzerinden denetim gibi uygulamaları desteklemektedir (Anonim L). Kartın pin diyagramı ve şematiği Ek-1 de sunulmuştur.



Şekil 3.11 SBC68EC Web Sunucu

Özellikleri

- 32 genel kullanımlı, programlanabilir ve görüntülenebilir G/Ç bulunmaktadır. Bunlardan 12 tanesi 10bit Analog Giriş ve 2 tanesi 10bit PWM çıkış olarak kullanılabilir.
- İnternet üzerinden güncelleme, denetim ve gözlemlemeye müsaittir.
- USART, CAN ve I2C arayüzleri mevcuttur.
- 64KByte FLASH, 3936 Byte SRAM ve 64KByte EEPROM içermektedir. Kart yazılımları hafızanın yarısı kadarını doldurmaktadır. Diğer kısım kullanıcı için ayrılmıştır.
- 7-35V arası besleme gerilimi kullanılabilir.

Kart üzerinde iki adet 20 pinlik bağlantı soketi bulunmaktadır. Bu soketlerin pin çıkışları aşağıdaki tabloda görülmektedir (Tablo 3.1).

Tablo 3.1 Gömülü Web Sunucu Pinleri

CON2 Bağlantı Soketi			CON1 Bağlantı Soketi		
İsim	Pin	Sinyal	İsim	Pin	Sinyal
TOP0	2	PIC pin RF0	TOP4	2	PIC pin RF4
TOP1	1	PIC pin RF1	TOP5	1	PIC pin RF5
TOP2	4	PIC pin RF2	TOP6	4	PIC pin RF6
TOP3	3	PIC pin RF3	TOP7	3	PIC pin RF7
SIG0	6	USART receive sinyali	GND	5	Şase
SIG1	5	USART transmit sinyali	+5V	7	Regüleli 0.5A 5V çıkış gerilimi
B0	13	PIC pin RB0	VIN	8	Regüle edilmemiş giriş gerilimi.
B1	14	PIC pin RB1	CLR#	6	PIC pin /MCLR
B2	11	PIC pin RB2	A0	10	PIC pin RA0
B3	12	PIC pin RB3	A1	9	PIC pin RA1
B4	9	PIC pin RB4	A2	12	PIC pin RA2
B5	10	PIC pin RB5	A3	11	PIC pin RA3
B6	7	PIC pin RB6 – ICP için de kullanılır.	A4	14	PIC pin RA4
B7	8	PIC pin RB7 – ICP için de kullanılır.	A5	13	PIC pin RA5
C4	17	PIC pin RC4 – I ² C için ayrılmıştır.	C0	16	PIC pin RC0
C5	18	PIC pin RC5	C1	15	PIC pin RC1
C6	15	PIC pin RC6	C2	18	PIC pin RC2
C7	16	PIC pin RC7	C3	17	PIC pin RC3 – I ² C için ayrılmıştır.
D6	19	RG2	D0	20	RG0
D7	20	RG3	D1	19	RG1

Tabloda giriş ve çıkış olarak kullanılabilen pinler görülmektedir. Bu pinlerden B6 ve B7, PIC programlayıcı cihaz yardımıyla programlama için ayrılmıştır. PIC üzerinde meydana gelebilecek hatalar veya bağlantı sorunları nedeniyle programlamada sorun yaşandığı durumlarda bu pinler ile kartın programlanması mümkündür. C3 ve C4, I²C bağlantısı için kullanılabilir. I²C bağlantısı ile haberleşebilen elektronik cihazlar bu pinler vasıtası ile sisteme eklenebilir. Dikkat edilmesi gereken husus, bağlanacak cihazların 400Kbit/s hızında iletişimi destekliyor olmasıdır. Yine I²C bağlantısı ile haberleşen display, karta

bağlanabilir. Aksi takdirde çıkış portlarından birini display için ayırmak gerekecektir. SIG0 ve SIG1 pinleri ise USART erişimi için ayrılmıştır. Bu pinler $\pm 15V$ olan RS232 gerilim seviyesine göre ayarlanmıştır. Özel olarak ayrılmış bu pinlere yapılacak hatalı girişler kartın kilitlemesine sebep olmaktadır.

Kart üzerinde 32 adet giriş ve çıkış olarak programlanabilecek pinler bulunduğundan bahsedilmiştir. Dijital olarak kullanılacak pinler 0-5V gerilim seviyesinde kullanılmaktadır. Girişler 3V olarak da kullanılabilir. Çıkışlar ise seri olarak bağlanacak diyotlar ile 3V seviyesine indirilebilir. 32 pinden 12'si analog giriş ve 2'si analog (PWM) çıkış olarak kullanılabilir. Her analog kanal 10 bit çözünürlüğe sahiptir.

Kart üzerinde dahili EEPROM ve dahili flash bellek bulunmaktadır. Stack bu alanın ilk 256 sekizlisini kullanmaktadır. Geriye kalan kısım kullanıcı tarafından kullanılabilir. Ancak tavsiye edilen son 512 sekizlik kısmın kullanılmasıdır. Dahili flash bellek ise 64 KByte'dır ve 2 KByte'lık kısmı bootloader yazılımı, 32 KByte'lık kısmı ise Stack tarafından kullanılmaktadır. Kalan 30 KByte'lık kısım kullanıcı uygulamaları için kullanılabilir. Ancak tavsiye edilen harici belleğin kullanılmasıdır. Detaylı bellek diyagramı Ek-6'da verilmiştir.

Hafızanın yeterli gelmediği durumlarda kullanılmak üzere kart üzerinde seri EEPROM yuvası bulunmaktadır. 24LC256 (32KByte) ve 24LC512 (64KByte) seri EEPROM'lar hafıza ihtiyacına göre kullanılabilir. EEPROM, I²C bağlantısı ile haberleşmektedir. Daha fazla kapasite için 24LC1024 (128KByte) entegresi kullanılmak istenirse kart üzerinde bazı jumper değişiklikleri yapılmalıdır. Her EEPROM türü için TCP/IP Stack üzerinde düzeltmeler yapılmalıdır. TCP/IP Stack üzerinde düzenlenmesi gereken kısımlar Ek-5'te sunulmuştur.

Gömülü web sunucu olarak ayarlanan karta aynı anda bağlanabilecek bilgisayar sayısı 10'dur. Bu sınır kart üzerinde kullanılan malzemelerin limitleri ile hesaplanmış ve TCP/IP Stack üzerindeki "projdefs.h" modülü içindeki MAX_SOCKETS parametresi ile sınırlandırılmıştır. Bu parametre, gömülü web sunucu ile kurulabilecek en fazla HTTP bağlantısını belirtmektedir. Varsayılan olarak tanımlanmış bu sınır değerinin arttırılması mümkündür. Ancak arttırılan bağlantı sayısı kartın kilitlemesine yol açmaktadır.

3.2.1.1 PIC 18F6680

PIC Serisi mikroişlemciler Microchip firması tarafından geliştirilmiştir. Üretim amacı çok fonksiyonlu lojik uygulamalarının hızlı ve ucuz bir mikroişlemci ile yazılım yoluyla karşılanmasıdır (Anonim H).

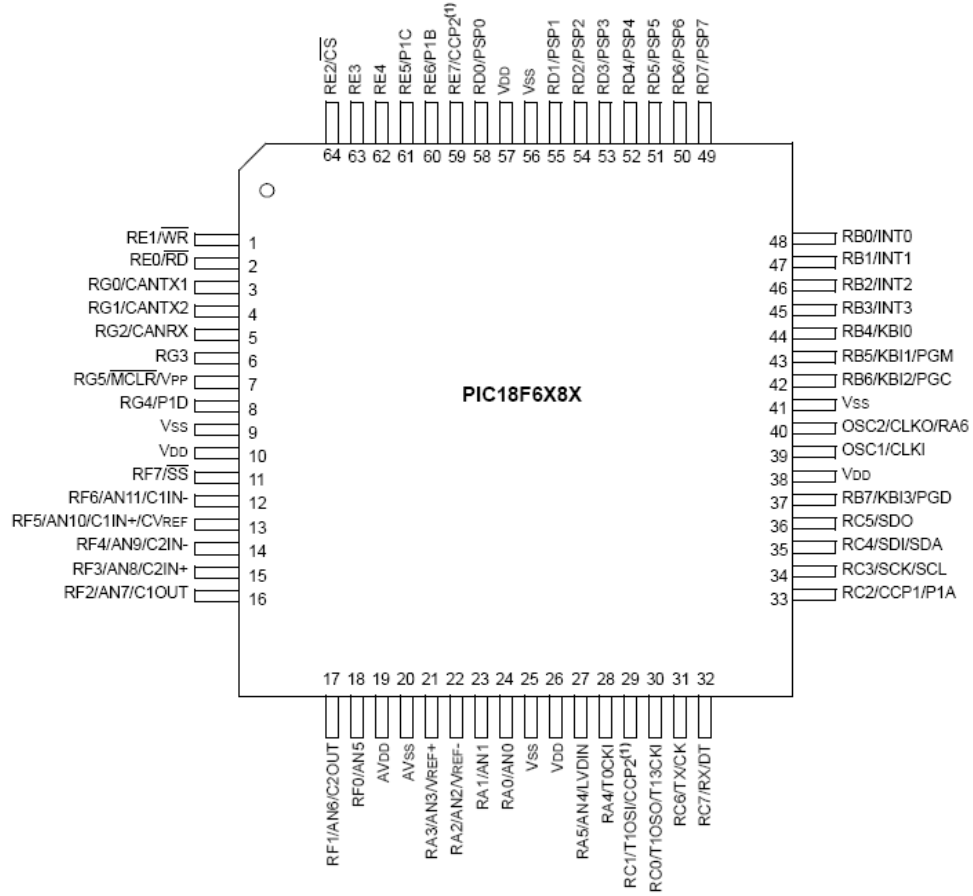
PIC'in kelime anlamı -PERIPHERAL INTERFACE CONTROLLER- Giriş çıkış işlemcisidir. İlk olarak 1994 yılında 16 bitlik ve 32 bitlik büyük işlemcilerin giriş ve çıkışlarındaki yükü azaltmak ve denetlemek amacıyla çok hızlı ve ucuz bir çözüme ihtiyaç duyulduğu için geliştirilmiştir.

1992 yılında 14 bitlik çekirdeğe sahip PIC16CXXX ailesi, daha fazla program alanının ve kesme işlemleri yanında A/D çeviriciler, 16 bit sayıcılar gibi çevre birimlerinin kullanımına olanak sağlamıştır. Bu ailedeki RISC komut seti de 12-bit çekirdektekilerle hemen hemen aynıydı ve 35 komuttan oluşuyordu. 1997'de çarpma yapabilen bir ALU'ya ve ileri arabirim yeteneklerine sahip 16-bit PIC17CXXX ailesi piyasaya sunulmuştur. Ardından 1999 yılında da genişletilmiş 16-bit çekirdekli PIC18CXXX ailesi sunulmuştur. Bu ailedeki işlemcilerde komut sayısı 77'dir ve bu yüksek-seviye dillerin derleyicilerinin ihtiyaçlarını daha fazla karşılar hale ulaşmıştır.

18F6680 mikrodenetleyicinin özellikleri Tablo 3.2'de gösterilmiştir.

Tablo 3.2 18F6680 Özellikleri

Program Hafıza Tipi	Flash
Program Hafızası (KB)	64
CPU Hızı (MIPS)	10
RAM Bytes	3,328
Data EEPROM (bytes)	1024
Dijital İletişim Çevrebirimi	1-A/E/USART, 1-MSSP(SPI/I2C)
Capture/Compare/PWM Çevrebirimi	1 CCP, 1 ECCP
Timer	1 x 8-bit, 3 x 16-bit
ADC	12 adet, 10-bit
Comparator (Karşılaştırıcı)	2
CAN	1 ECAN
Sıcaklık Limitleri (C)	-40 - 125
Çalışma Gerilimi Limitleri (V)	2 - 5.5
Pin Sayısı	64



Şekil 3.12 PIC18F6680 Pin Bağlantıları

3.2.1.2 RTL 8019AS

RTL8019AS, kendi başına çalışabilen, Ethernet II ile 10Base5, 10Base2 ve 10BaseT IEEE802.3 ağlarında çalışabilen ağ arayüz kartıdır (Şekil 3.13). Maksimum 10Mbps hızındaki veri alış verişini destekler. +5V ile çalışır. RTL8019AS, OSI referans modelinin birinci katmanı olan fiziksel katmanın görevini yapar. Manyetik filtre grubu ve RJ45 jack ile ağa bağlantıyı sağlar ve Ethernet protokolünü işler. Üst katmanlardan gelen paketler Ethernet katmanının veri kısmını oluşturur. RTL8019AS kendi içinde kayıtlı olan MAC adresiyle Ethernet çerçevesini ağa yollar. Ethernet ağında meydana gelebilecek çakışmanın tespiti ve çakışma sonrası yapılması gereken işlemler yine bu işlemcinin görevleridir (Anonim J).

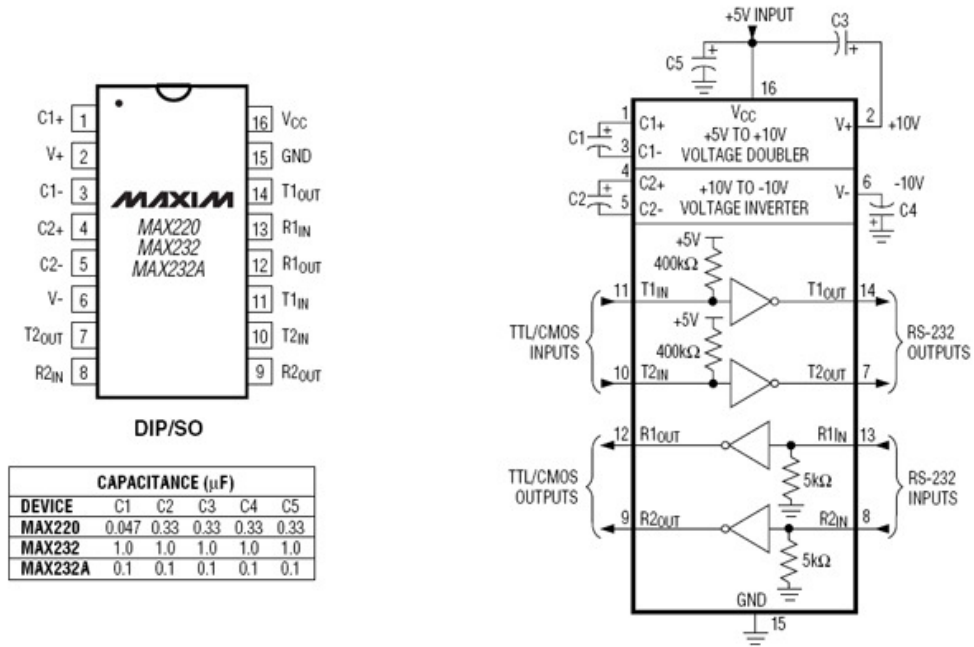


Şekil 3.13 RTL8019AS

3.2.1.3 MAX 232

Seri port RS-232 standartlarında ve asenkron olarak çalışır. Sinyal seviyelerinin ve sinyal gönderim şeklinin TTL devrelerine benzememesinden dolayı seri port çıkışından alınan bu verilerin mikrodenetleyicinin anlayabileceği TTL seviyelerine çevrilmesi için gömülü web sunucu kart üzerinde MAX232 sinyal dönüştürücüsü bulunmaktadır (Anonim F).

RS232’de -12 Volt “lojik 1” , +12 Volt “lojik 0” değerine karşılık gelir. Bu özellik daha uzun mesafeli iletimlerde, güçlendirmeye ihtiyaç duyulmamasını sağlar. RS232 ile birlikte çalışan devrelerde bu voltaj farkını dengeleyici ek yongalar kullanılır. Bunlardan en yaygın olanı MAXIM firmasının ürettiği MAX232 entegresidir. MAX232 gerilim seviye dönüştürücüsüdür. RS232, $\pm 12V$ kullandığından MAX232 bunu mikroişlemcinin çalıştığı 5V ya da 3,3V’a dönüştürerek uyumu sağlar. Bu entegrenin bacak bağlantıları ve içyapısı Şekil 3.14’de gösterilmiştir.



Şekil 3.14 MAX232 içyapısı ve bacak bağlantıları

3.2.1.4 24LC512 EEPROM

Mikrodenetleyici uygulamalarında dahili hafızaların yeterli olmadığı durumlarda harici seri hafızalara başvurulur. Amaç veri toplamak ve saklamak veya saklanmış verileri kullanmak olabilir. Veri erişim hızının yüksek olmadığı ve giriş/çıkış bacak sayısının önem kazandığı mikrodenetleyici uygulamalarında çevre donanım elemanlarının (EEPROM, RTC, sıcaklık ve nem algılayıcıları gibi) mikrodenetleyiciye bağlantı şekli önem kazanmış ve mümkün olan en az sayıda bağlantı olması arzulanmıştır. Bu şekilde data/clock (veri/saat pulsusu) veya 1 wire (tek telli) gibi bağlantı standartları geliştirilmiştir. Tabiidir ki bu bağlantıları destekleyen protokoller de geliştirilmiştir (I^2C , SPI, ...) (Anonim A).

RAM'lerin elektrik kesildiğinde bilgileri koruyamaması ROM ve PROM'ların yalnızca bir kez programlanabilmeleri bazı uygulamalar için sorun yaratmıştır. Bu sorunların üstesinden gelmek için teknoloji devreye girmiş ve EPROM'lar ortaya çıkmıştır. EPROM programlayıcı aygıt yardımı ile bir EPROM

defalarca programlanabilir, silinebilir. EPROM programlayıcı, EPROM'un üzerindeki kodlanmış programı mor ötesi ışınlar göndererek siler. Yonganın üzerindeki pencere, parlak güneş ışığı EPROM'u kolayca silebileceğinden, programlama işleminden sonra üzeri bir bantla kapatılır (Anonim B).

Çok yönlülükleri, kalıcı bellek özellikleri ve kolayca yeniden programlanabilirlikleri, EPROM'u kişisel bilgisayarlarda sıkça kullanılan bir konuma getirmiştir. EPROM'un bu bellek türünde bilgilerin yazılması için ve silinmesi için normal elektrik voltajı uygulanır. Diğer ROM türlerinde olduğu gibi içerdiği veriyi elektrik kesilse de saklar. Yine bütün ROM türlerinde olduğu gibi RAM bellek kadar hızlı değildir. Bu tür belleklerin bir kullanım sınırı vardır. Bu tür bellekler üzerine bilgiler yüzlerce ya da binlerce kere yazılabilir. EEPROM'un özel bir uygulaması Flash Memory'dir. Aralarındaki en önemli fark ise EEPROM'a bilgilerin byte byte yazılması; Flash'lara ise bilgilerin sabit bloklar halinde yazılmasıdır. Bu sabit bloklar 512 bytedan 256 KB'a kadar olan bir aralıkta değişir. Bu sabit bloklar halinde yazılma özelliği Flash Memory'yi EEPROM'a nazaran daha hızlı yapmıştır. EEPROM'larda olduğu gibi Flash Memory'nin de bir yaşam süresi vardır. Bu 100.000'den 300.000 kez yazmaya kadar değişebilir (Anonim K).

24LC512, 512K I2C CMOS Seri EEPROM'dur. Düşük güçlüdür ve 400kHz'e kadar çalışabilir. Bir milyon silme ve yazma ömrü vardır. 512bit (64Byte) hafızaya sahiptir. İki hatlı I2C haberleşmeyi destekler.

3.2.2 TCP/IP Stack

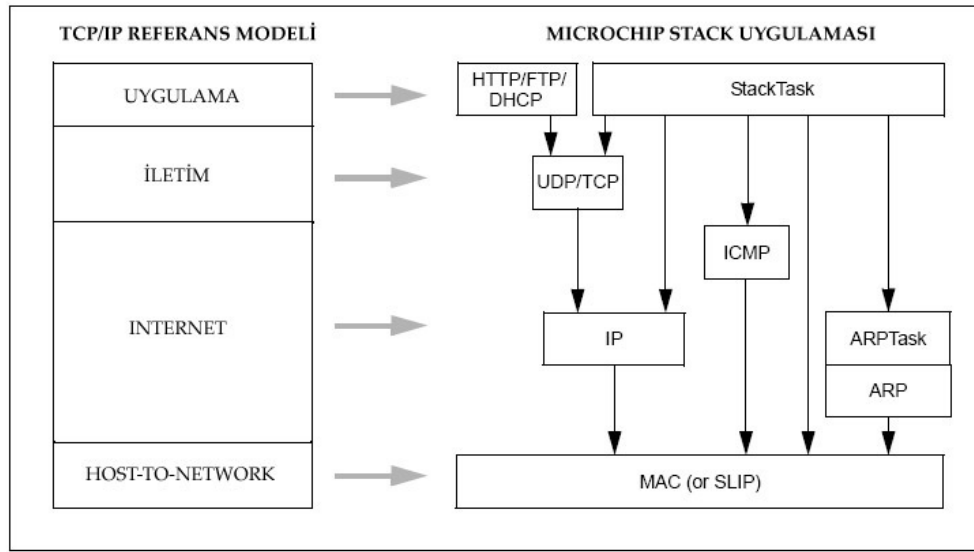
Bilgisayarların, birbirleri ile Ethernet kabloları ve İnternet üzerinden haberleşmeleri için gerekli bazı işlemleri yapması, bazı karışık protokolleri işlemesi gerektiğinden daha önceki bölümlerde bahsedilmişti. Bilgisayar mimarisindeki yazılımsal ve donanımsal olan değişik katmanların görevlerini bir mikrodenetleyiciye yaptırmak için kapsamlı bir yazılıma ihtiyaç duyulmaktadır. Oldukça karmaşık olan bu programa TCP/IP Stack denilmektedir ve Microchip geliştiricileri tarafından tasarlanıp ücretsiz olarak sunulmaktadır. Stack sayesinde genel mimari hakkında derinlemesine bilgi birikimine gerek olmadan yazılım geliştirmek mümkündür. Ancak Stack donanıma uygun şekilde düzenlenmelidir.

Stack içerisinde, TCP/IP tabanlı bazı temel uygulamalara ilişkin servisler tanımlanmıştır (HTTP sunucu, mail client, vs.). Stack, ihtiyaç duyduğu kaynaklar, çalışma hızı ve mikrodenetleyici üzerindeki hafıza ihtiyacı nedeniyle PIC18 sınıfı mikrodenetleyicilerde çalışacak şekilde tasarlanmıştır (Anonim M).

Stack, TCP/IP referans modeline uygun olarak çalışacak şekilde modüllerden oluşmaktadır (Şekil 3.15). Bu modüller aşağıdaki gibidir:

Announce, ARP, ARPTsk, Delay, DHCP, DNS, ENC28J60, FTP, Helpers, HTTP, I²C EEPROM, ICMP, IP, MPFS, RTL8019AS, SNMP, SPIEEPROM, StackTsk, TCP, Tick, UART, UDP.

Modüller ihtiyaca göre düzenlenebilir. Modüllerin tümü projemizde kullanılmamaktadır. Modüller, C kodlarıyla yazılmışlardır.



Şekil 3.15 TCP/IP Stack'in İçerdiği Modüller

TCP/IP Stack, kullanılacak donanıma uygun olarak düzenlenmelidir. C programlama diliyle yazılmış olan kodlar Microchip MPLab yazılımı ile düzenlenip derlenebilmektedir. Düzenlenmesi gereken tanımlamalar ve tanımlamaları içeren modül isimleri Ek-5'te sunulmuştur.

3.2.2.1 TCP/IP Stack tanımlamaları

TCP/IP Stack birçok modülden oluşmuştur. OSI referans modelindeki katmanlara ve alt görevlere ait bu modüller ve fonksiyonları yazılım işleyişinin daha rahat anlaşılması için bu bölümde açıklanacaktır.

3.2.2.1.1 Ortam erişim kontrolü katmanı (MAC)

MAC katmanının ve fiziksel katmanın görevlerini RTL8019AS Ethernet işlemcisi üstlenmektedir. İşlemciden gelen veriler yazılımın bu katmanında işlenir.

MACInit

Bu fonksiyon MAC katmanını başlatır. Tampon bellekleri aktifleştirir ve Ethernet işlemcisini bekleme konumuna resetler.

MACIsTxReady

Bu fonksiyon, MAC gönderim belleğinde boşluk olup olmadığını belirtir. Boşluk varsa bir mesaj gönderilebilir (transmit).

MACGetHeader

MAC alım (receive) belleğini kontrol eder. Eğer alınmış bir paket varsa kontrol eder ve sonucu gönderen tarafa bildirir. Gelen paket IP, ARP veya tanımlanamayan bir paket olabilir. Paket ilgili yere aktarılır.

MACGet

Başlık bilgisi onaylanmış bir paket varsa aktif alım veya gönderim belleğindeki bilginin sıradaki byte'ı işleme alınır.

MACGetArray

Aktif gönderim veya alım belleğindeki bir dizi bilgiyi (byte) işleme sokar.

MACDiscardRx

Aktif alım belleğindeki bilgiyi ıskartaya çıkarır ve belleği boş olarak işaretler. Alım işleminden sonra çağrılmalıdır.

MACPutHeader

Gelmiş olan son MAC başlık bilgisini aktif gönderim belleğine yükler.

MACPut

Gönderilmek istenen veri byte'ını aktif gönderim veya alım belleğine yükler.

MACPutArray

Gönderilmek istenen bir dizi veriyi aktif gönderim veya alım belleğine yazar.

MACFlush

Aktif gönderim belleğini gönderim için hazır olarak işaretler. Tüm işlemlerin ardından gönderim için hazır olduğunda bu fonksiyon çağrılır.

MACDiscardTx

Gönderimin ardından ilgili bellekteki bilgiyi iskarta eder ve belleği boş olarak işaretler.

MACSetRxBuffer

Aktif alım belleğinde alınacak bir sonraki veri kümesinin yerini işaretler.

MACSetTxBuffer

Gönderim belleğinde gönderilecek paketin başlangıç yerini gösterir ve bu bellek kısmını aktif hale getirir.

MACReserveTxBuffer

Gönderilecek paketin bir kopyasını saklar ve bu verinin kaydedildiği veri bölgesini kullanılamaz olarak işaretler. Verinin karşıdan alındığı onayı geldiği zaman bu veriyi iskartaya çıkarır ve alanı uygun olarak işaretler. TCP iletişimde oldukça kullanışlıdır.

MACGetFreeRxSize

Bu fonksiyon gelecek veri paketleri için uygun olan toplam alım belleği miktarını verir.

MACGetRxBuffer

Kullanımda olan (geçerli) alım belleğinin tanımlayıcısını (yerini) gösterir.

MACGetTxBuffer

Kullanımda olan gönderim belleğinin yerini gösterir.

3.2.2.1.2 Adres çözümleme protokolü (ARP ve ARPTask)

ARP katmanı, ARP ve ARPTask olmak üzere iki modülden oluşur. ARP modülü protokolün giriş basamaklarını sağlar. ARPTask ise girişin yanında ARP protokolünün tamamını içerir.

ARPIsTxReady

Bu makro MACIsTxReady fonksiyonunu çağırır. Bellek uygun ise ARP paketi yollanır.

ARPGet

Gelen ARP paketinin tamamını alır ve gerekli bilgiyi gönderir. ARP cevabı, isteği veya bilinmeyen ARP paketi alındı bilgileri bu fonksiyondan döner.

ARPPut

ARP cevabı veya isteği paketini MAC tampon belleğine gönderim için yükler.

3.2.2.1.2.1 ARPTask fonksiyonları*ARPInit*

ARPTask durum makinesi işletir ve ARP isteklerinin ve cevaplarının idaresi için hazır hale getirir.

ARPResolve

Uzak istemciye ARP isteği gönderir.

ARPIsResolved

Dahili cache belleği kontrol eder ve uyan adres bilgisini verir. Eğer gelen paketin IP adresi uyuyorsa paket üzerindeki MAC adresi MACAddr değişkenine kopyalanır.

3.2.2.1.3 İnternet protokolü (IP)

IP katmanı bu yazılımda pasif modadır. IP paketlerine cevap vermek yerine üst katmanların IP ihtiyaçlarını karşılar.

IPIsTxReady

Bu makro MACIsTxReady fonksiyonunun çağırır. Bellek uygun ise IP paketi yollanır.

IPSetTxBuffer

Bu makro daha üst katmanlara gönderim belleğinin başlangıç noktasını belirleme imkanı sunar.

IPPutHeader

Geçerli bir IP başlığı hazırlar ve bunu aktif gönderim belleğine yükler. Gönderilecek paket ICMP, TCP veya UDP olabilir.

IPPutArray

Aktif gönderim belleğine bir dizi byte'tan oluşan veri yazar. Bu makro MACPutArray fonksiyonunu çağırır.

IPGetHeader

Aktif bellekteki IP başlığını alır ve onaylar.

IPGetArray

Bu makro aktif alım veya gönderim belleğinden bir dizi veriyi alır.

3.2.2.1.4 İnternet denetim iletisi protokolü (ICMP)

ICMP katmanı bu yazılımda pasif modadır. ICMP paketlerine cevap vermek yerine üst katmanların ICMP ihtiyaçlarını karşılar.

ICMPisTxReady

Bu makro, en azından bir gönderim belleğinin boş olduğunu veya tümünün dolu olduğunu bildirir.

ICMPPut

Geçerli bir ICMP paketi oluşturur ve gönderir. Veri, veri uzunluğu, tanımlama bilgisi ve paket dizi numarasını oluşturur.

ICMPGet

Aktif alım belleğinden ICMP başlığını alır ve onaylar. Onayın ardından veri alınır işleme koyulur ve bellek boşaltılır.

3.2.2.1.5 Aktarım denetim protokolü (TCP)

Bu katman aktif bir katmandır. TCP paketlerini alır ve TCP durum makinesine göre uzak istemciye cevap verir. Ana uygulamayı ilişkilendirmeden birçok gerekli uygulamayı kendisi işler. Bu katman 2 den 253 e kadar TCP soketi sağlar. Bu sayı uygun hafıza ve kullanılan işlemciye göre sınırlanabilir. Bu sayıdaki soket aynı anda birçok uygulamanın bu katmanı kullanabilmesini sağlar. Özellikle

HTTP sunucu uygulamalarında kullanışlıdır. Aynı anda birden fazla kullanıcının işlemciye bağlanmasını ve veri alış verişini sağlar.

TCPInit

Bu fonksiyon TCP durum makinesini çalıştırır ve çoklu TCP bağlantısı için hazırlar.

TCPListen

Uygun soketlerden birisini verilen TCP portu dinlemesi için belirler.

TCPCConnect

Gelen bağlantı isteğine cevap vermek için uzak porta bağlantıyı sağlar.

TCPIsConnected

Verilen sokete bağlantının gerçekleşip gerçekleşmediğini belirler.

TCPIsDisconnect

Uzak istemciye bağlantı kesme isteğini yollar.

TCPIsPutReady

Soketin veri göndermeye hazır olma durumunu belirtir. Soket, istemciye bağlı ve istemcinin alım belleği boş ise hazırdır.

TCPPut

Verilmiş olan soket için bir veri byte'ını gönderim belleğine yükler.

TCPFlush

Gönderilecek bilgiliyi gönderime hazır olarak işaretler.

TCPIsGetReady

Belirtilmiş soket alım bilgisini içeriyorsa "true" değeri verir.

TCPGet

Alım belleğinden bir veri byte'ını alır.

TCPGetArray

Belirtilmiş soketin alım belleğinden bir dizi bilgiyi alır.

TCPDiscard

Alınan bilginin işlenmesinden sonra ilgili bellek birimini serbest bırakır.

TCPPProcess

Bu fonksiyon alınmış olan bir TCP paketini alarak karşılıklı bağlanmış soketler için TCP durum makinesini çalıştırır.

TCPTick

Tüm socketler için zaman aşımını kontrol eder. Belirli aralıklarla çağırılır.

3.2.2.1.6 Kullanıcı datagram protokolü (UDP)

Bu katman 254'e kadar UDP socketinin ve veri akışının kontrolünü sağlamaktadır. Dikkat edilmesi gereken bir nokta; her socket en fazla 19 byte'lık veri işleyebilir.

UDPInit

Bu fonksiyon UDP modülünü çoklu UDP bağlantısı için çalıştırır ve hazırlar.

UDPOpen

Sıradaki UDP socketini ilgili porttan gelebilecek veri transferine karşı hazırlar.

UDPClose

İlgili UDP socketini kapatır ve boş socket olduğu bilgisini gönderir.

UDPisPutReady

İlgili socketin gönderime hazır olma durumunu belirler. Bir socketin hazır olma durumu en azından bir MAC gönderim belleğinin boş olmasına bağlıdır.

UDPPut

Aktif socket için gönderim belleğine verilen veriyi yükler.

UDPFlush

Aktif sokete bağlı gönderim belleğini gönderime hazır olarak işaretler.

UDPisGetReady

İlgili socketin alınan bilgiyi içerme durumunu kontrol eder. Ayrıca ilgili socketi aktif socket olarak işaretler.

UDPGet

Aktif portun alım belleğinden bir byte'lık veriyi alır.

UDPDiscard

Aktif sokete bağlı alım belleğini boşaltır.

UDPProcess

Alınan paketi işleme sokar.

3.2.2.1.7 Dinamik istemci ayarlama protokolü (DHCP)

DHCP katmanı aktif bir katmandır ve ana uygulamadan bağımsız olarak çalışır. DHCP isteklerini otomatik olarak alır ve işler. IP ve gateway adresleri ile alt ağları konfigüre eder. Dikkat edilmesi gereken husus; kullanıcı yazılımı, adresler ayarlanana kadar ağ iletişimine müdahale etmemelidir. Bu çalışmada DHCP protokolü kullanılmamıştır. DHCP, bir sunucuya başka gömülü web kartları bağlanacağı veya gömülü web kartının bir ağa bağlanacağı durumlarda IP adreslerinin ayarlanması için kullanılmaktadır.

DHCPTask

Bu fonksiyon bu katmanın çekirdeğini oluşturur ve gerekli protokol olaylarını gerçekleştirir.

DHCPDisable

Bu makro DHCP modülünü kapatır.

DHCPAbort

Aktif DHCP görevini iptal eder ve uygulama süresince kapatır.

DHCPisBound

Bu makro DHCP modülünün IP konfigürasyonu içerme durumunu kontrol eder.

3.2.2.1.8 Dinamik HTTP sayfa oluşturulması

HTTP sunucu dinamik sayfalar oluşturulmasını sağlar. CGI uzantılı dosyalara cevap verir. Kullanıcı yazılımı içerisindeki %xx şeklindeki tanımlamalarda fonksiyon çağrılır ve tanımlamanın yerine işlemcinin ilgili pininin durumu aktarılır. Belirleyici karakter % işaretidir. “xx” değeri 00-99 arası değerler alır.

HTTPGetVar

HTTP sunucu, bir CGI sayfası içerisinde %xx şeklinde bir tanımlama ile karşılaştığında bu fonksiyon çağrılır. Alınan numaraya karşılık gelen pini işaretler.

HTTPExecCmd

GET metodu ile %xx şeklindeki bir tanımlama ile karşılaştığında bu fonksiyon çağrılır. Fonksiyon gerekli işlemi yürüterek gelen değişkenin karşılığı olan

veriyi CGI sayfasında belirtilen yere gönderir.

3.3 Programlama

Web Sunucu olarak kullanılabilen kart üzerinden gözlem ve denetim yapabilmek için HTTP protokolü kullanılmıştır. Gömülü web sunucu içerisindeki HTTP protokolü CGI uzantılı dosyalara cevap verecek şekilde tasarlanmıştır. HTM ve CGI uzantılarıyla kaydedilen web sayfaları uygun formata dönüştürülerek kart üzerindeki EEPROM'a yüklenir.

Sayfa içerisinde görüntülenecek pinler "%a02" şeklinde sayfaya yerleştirilir. "%" sembolü Stack içerisindeki bir fonksiyonu çağırır ve istenen pinin durumunu geriye gönderir. "a" port ismini belirtir. Kart üzerinde a-g arası portlar kullanılabilir. "02" ise pin numarasını vermektedir. 00-07 arası değerler portun durumuna göre 1 veya 0 bilgisi geri döndürecektir. 10-17 arası değerlerle durum "on" veya "off" şeklinde alınabilir. Değerler HEX olarak tanımlanmıştır. Analog portlarda ise 00-0A arası değerler 2 veya 3 dijital HEX değer, 10-1A; 0-255 arası desimal değer, 20-2A arası ise 5V referansa göre desimal değer döndürür.

Kart üzerindeki değişiklikler HTTP komutlarıyla yapılmaktadır. Bir pinin değerini değiştirmek için "http://10.1.0.1/?a2=1" satırı karta gönderilmelidir. Bu değer a portunun 2 numaralı çıkışını aktif edecektir. Bir pini giriş veya çıkış olarak ayarlamak için "ac0=0" değeri kullanılmalıdır. "c" değeri İngilizce configuration teriminden gelmektedir. "0", pini çıkış, "1" ise giriş yapmak için kullanılır. Kartın bu ayarları hafızasında tutması için ise "c" harfinin yanına "r" harfi eklenmelidir. Kullanılacak web sayfasına kullanıcı adı ve şifre eklenmek isteniyorsa uzantısına "x" harfi eklenmelidir (index.xcgi). Kullanıcı adı "la=kullanıcı_adi", ve şifre "lb=şifre" komutlarıyla verilebilir. Analog çıkışları aktif ya da pasif hale getirmek için "we" komutu kullanılır. "we= 1F" komutu 1-5 arası pinleri çıkış olarak ayarlayacaktır.

Modtronix firması tarafından sunulmuş olan kullanım kılavuzu ve örnek siteler yardımıyla hazırlanabilecek bir web sayfası ile kart üzerinde çevrimiçi birçok değişiklik yapmak mümkündür. Kart üzerinde yapılabilecek değişiklikler:

- Pinlerin giriş-çıkış olarak kullanımı
- Analog giriş-çıkış olarak kullanılacak pinlerin aktifleştirilmesi
- Analog ve dijital girişlerin anlık değerleri
- IP adresinin, alt ağ maskesinin, gateway adresinin ve MAC adresinin görüntülenmesi ve değiştirilmesi
- DHCP protokolünün kullanılıp kullanılmayacağı
- Seri iletişim için gerekli haberleşme parametreleri
- Sayfa içerisinde kullanıcı adı ve şifreli sayfaların kullanılıp kullanılmayacağı.

Web sayfaları JavaScript programlamayı desteklemekte ve resim ile ses dosyaları içerebilmektedir. Görsellik ve denetim pinlerinin daha etkin kullanımı için JavaScript kodları kullanılmıştır. Anlık pin değerlerindeki değişikliklerin ekranda görüntülenmesi için sayfanın yenilenmesi gerekmektedir. Kart içerisindeki TCP/IP Stack, Ajax programlama dilini desteklemektedir. Ajax ile yenilme gereği olmadan da anlık değişiklikleri takip etmek mümkündür.

Hazırlanan web sitesini kart içerisine yüklemek için, tüm web sayfaları ile ek dosyalar aynı klasör içerisine konularak binary imajı alınır ve FTP üzerinden karta gönderilir. İmaj oluşturma işlemi Microchip firması tarafından sağlanan MPFS yazılımı veya bootloader yazılımları sayesinde yapılabilir. SBC68EC kartının programlanması için firma tarafından sağlanan Modtronix Netloader yazılımı kullanılabilir. Bu yazılım ile web sayfalarının imajı alınır ve FTP ile İnternet üzerinden gönderimi yapılır. Java tabanlı programa, imajı alınacak dosyaların bulunduğu klasör gösterilir ve derlenmesi sağlanır. İmaj alma işlemi için kullanılacak bilgisayar, Java programındaki sorunlar nedeniyle, İngilizce işletim sistemi yüklü ve güncelleme yapılmamış olmalıdır. İmaj alınırken dikkat edilmesi gereken nokta, 64 byte'lık bir rezerve bloğun ayrılması gerektiğidir. Ana uygulama bazı konfigürasyon bilgilerini saklamak için EEPROM'u kullanabilir ve bunun için hafıza bloğunun başını kullanır. Microchip firması TCP/IP Stack yazılımı için imaj formatını 16 bit'lik adreslemeyi kullanacak şekilde belirlemiştir.

Hazırlanan web sayfalarının karta yüklenmesi için üç yöntem bulunmaktadır. Yükleme bootloader yazılımı ile, FTP üzerinden veya HyperTerminal gibi bir terminal programı ile yapılabilir. İletişim için kullanılan portlar ise UDP için 54211

(bootloader tarafından kullanılır), Netbios için TCP 135 ve UDP 139, FTP için ise 21 portudur. Ancak yükleme sırasında oluşabilecek hatalar nedeniyle kartta sorunlar meydana gelebilir ve bağlantı kopabilir. Bu durumda RS232 bağlantısı kullanılmalıdır. RS232 ile karta ait firmware ve bootloader yazılımı tekrar karta yüklenmelidir. Varsayılan RS232 ayarları; 57600 baudrate, 8 veri biti, parity biti yok ve 1 stop bitidir. RS232 portunda kullanılmasına engel olabilecek düzeyde bir hata oluştuysa PIC programlayıcı ile TCP/IP Stack tekrar karta yüklenmelidir.

3.4 Havuz Ekipmanları

Bu çalışmada havuz denetimi yapılmaktadır. Yüzme havuzlarının uzaktan denetiminin ticari boyutu değerlendirilmiş ve böyle bir gereksinim olduğu sonucuna varılarak bu çalışmaya başlanmıştır. Uygulamanın kapsamı genişletilerek değişik havuz uygulamalarında ve akvaryum sistemlerinde kullanılabilir.

Havuz ekipmanlarının kullanımı ve denetiminin sağladığı avantajlar aşağıda listelenmiştir.

- Sürekli pH ölçümü ile havuz suyunun zararlı olabilecek seviyeye ulaşması önlenir. pH, 7.2 seviyesinde olmalıdır. Gerekli durumda pH dengelemesi için suya asit ve baz eklenir.
- Suyu ozon ve klor verilerek suyun yosunlanması önlenir. Yosunlanma suyun boşaltılmasını ve havuz içinin temizlenmesini gerektirir.
- Su seviyesi istenen seviyede tutulur.
- Kum filtresinin uzaktan çalıştırılması ile havuz suyu temizliği yapılır.
- Kum filtresinin temizlenmesi için uygulanan ters yıkama ve durulama işlemleri ile filtrenin temiz kalması sağlanır.
- Havuz ışıkları istenildiği yerden açılabilir.
- Sıcaklık ölçümü ile havuz suyunun istenen sıcaklıkta tutulması için ısıtıcı çalıştırılabilir.
- Denetim ile havuz bakım maliyetleri düşürülür ve zaman kaybı önlenir.

Bu işlemler için havuzlarda; ısıtıcı, aydınlatma sistemleri, asit, baz ve klor dozajlama pompaları, doldurma pompası, kum filtresi, filtre üzerinde kullanılacak motorlu altı yollu vana ile sıcaklık, pH ve seviye algılayıcıları kullanılmalıdır.

3.5 Uygulama

Bu çalışmada havuzlarda kullanılan ekipmanların uzaktan denetimi yapılmaktadır. Uzaktan denetim, İnternet üzerinden yapılmakta ve İnternet'e bağlı bir bilgisayarın tarayıcısına gömülü web sunucunun IP adresinin girilmesi ile gerçekleşmektedir. IP adresinin girilmesi ile bilgisayar gömülü web sunucu ile haberleşmeye başlar ve HTML tabanlı bir arayüz görüntülenir. Bu arayüz ile görsel olarak sistemin denetimi sağlanır. Modüler bir yapıda olan gömülü web sunucu birçok değişik uygulamada web arayüzü olarak kullanılabilir. Tasarlanan denetim kartı ile kullanım alanı özelleştirilmiş olmaktadır.

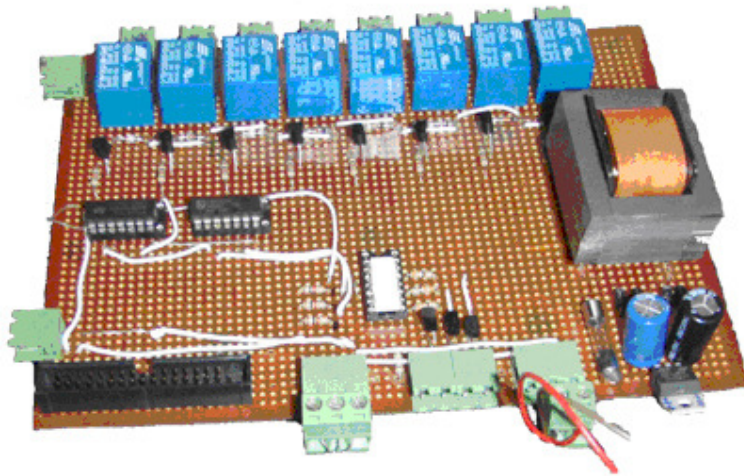
Bu çalışma için İnternet üzerinden denetim sistemleri incelenmiş ve gömülü sistemlerin kullanılmasına karar verilmiştir. Gömülü sistemlerin incelenmesiyle bu ölçüdeki bir denetim sistemi için yeterli kaynağa sahip mikrodenetleyici ve Ethernet işlemcisinden oluşmuş bir elektronik kartın İnternet ortamına çıkabileceği ve uzaktan erişime imkan sağlayacağı anlaşılmıştır. Gömülü web sunucu olarak kullanılacak elektronik kartın temininin ardından çalışma sisteminin anlaşılması gerekmektedir. Önceki bölümlerde bahsedilmiş olan sistem için web arayüzü hazırlanması gerekmektedir.

Bu bölümde havuz denetimi için gerekli denetim kartı, sunum materyali ve gömülü web sunucu yazılımından bahsedilecektir.

3.5.1 Denetim kartı ve sunum materyali

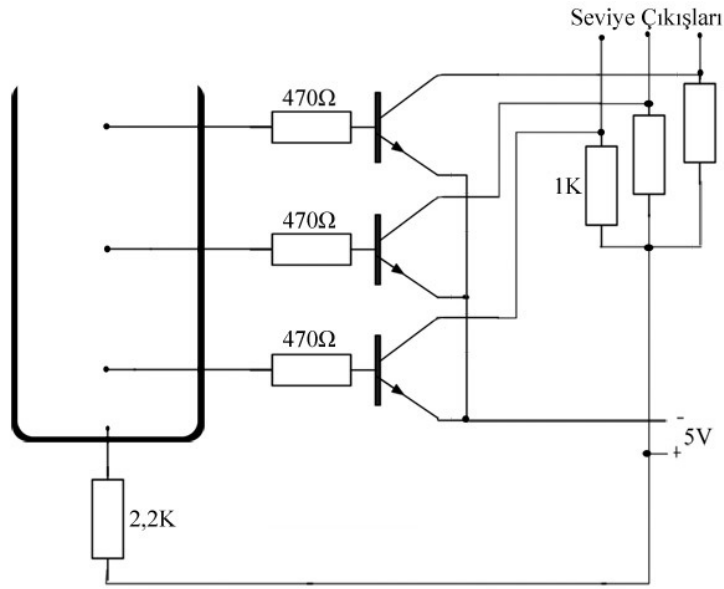
Mevcut havuz veya akvaryum ekipmanlarına bağlanabilecek bir denetim kartı ve gömülü web sunucu ile havuzlarda ihtiyaç duyulan ekipmanların denetimi ve gözlemlenmesi yapılabilmektedir. Havuzlarda ihtiyaç duyulabilecek, doldurma ve boşaltma, pH dengelemesi için kullanılacak asit, kostik ve klor pompaları, ozon pompası, havuz suyu ısıtıcısı, havuz iç ışıkları ile havuz suyu sıcaklığı, seviyesi ve

pH'inın gözlemlenmesi, kum filtresinin çalışması ve temizlenmesi için gerekli yazılım ve denetim kartı tasarlanmıştır (Şekil 3.16). Denetim kartı şeması Ek-2'de sunulmuştur. Gömülü web sunucunun çıkışları 5V'tur. Havuz ekipmanlarının ihtiyaç duyduğu 220V'luk gerilimin denetimi için denetim kartı üzerinde 5V-220V kontrollü röle kullanılmıştır. Röleler üzerinde oluşabilecek bir kısa devreden gömülü web sunucuyu korumak için röle girişlerine transistörlü sürme devreleri eklenmiştir. Transistör ve diyottan oluşan sürme devreleri gömülü web sunucudan gelecek sinyal ile 5V'luk çıkış üretmekte ve röleleri sürmektedir. Her röle gömülü web sunucunun bir çıkış pinine bağlıdır. Sıcaklık ölçümü LM35 algılayıcısı ile yapılmaktadır. 5V'luk besleme girişi ile çalışan, transistör benzeri bir yapıya sahip olan LM35'in çıkış bacağı gömülü web sunucunun analog girişlerinden birine bağlanmaktadır. LM35'in besleme gerilimi 5V'luk ikinci bir kaynaktan gelmektedir. Sıcaklık girişi, gömülü web sunucuya doğrudan gerilim sinyali göndereceğinden, sunucunun zarar görme ihtimali göz önünde bulundurularak böyle bir bağlantının daha uygun olduğuna karar verilmiştir. pH ölçümü için yine benzer bir bağlantı yapılmıştır. pH ölçümü için kullanılacak algılayıcıların ölçüm aralıkları ve çıkış gerilimleri farklı olabilmektedir. Çıkış gerilimini, kullanılacak gerilim seviyesine dönüştürmek için araya gerilim yükseltici devre eklenebilir. Ölçüm aralığı için ise web arayüzü üzerinde düzenlemeler yapılması gerekecektir.



Şekil 3.16 Denetim Kartı

Seviye ölçümü elektrot yöntemi ile yapılmaktadır. Bu yöntemde sıvıların iletkenliğinden faydalanılır. Şekil 3.17’te verilen devrede, depo içinde bulunan iletken sıvının (su, asit, boya vb.) seviyesi yükseldikçe deponun dibindeki direnç ile devreyi tamamlayan elektrotlardan akım akacaktır. Elektrotların, bazına bağlı olduğu transistörler ilettime geçecek ve kolektörlerinde bulunan çıkışlardan istenen seviyede (5V) bir çıkış gerilimi alınacaktır. Devremizde çıkışlara opto kuplörler bağlanmıştır; böylece gömülü web sunucuya gelecek girişler izole edilmiştir.

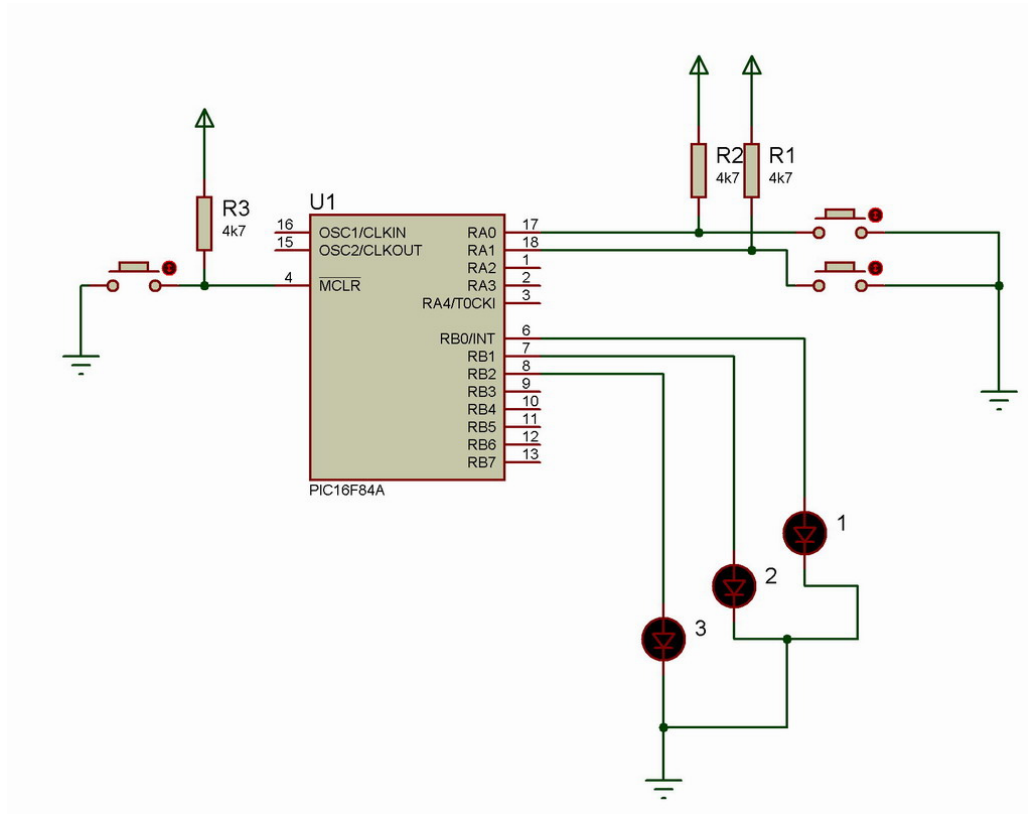


Şekil 3.17 Elektrot Yöntemiyle Seviye Ölçümü

Havuz içerisindeki suyun en üst ve en alt noktalara ulaştığı durumlarda doldurma ve boşaltma işlemlerinin durması gerekmektedir. Kart üzerine yerleştirilmiş “VE” ve “DEĞİL” kapı entegreleri ile pompalara gönderilen çalışma emirleri şarta bağlanmıştır. Bu şartlar ile doldurma emri havuz su seviyesinin tepe noktasında olmama durumuna bağlanmıştır. Boşaltma için ise su seviyesinin en altta olmama durumu ön koşuldur.

Altı yollu vana, kum filtresi üzerinde bulunmaktadır. Vana üzerinde filtre (filter), boşaltım (vaste), kapalı (closed), ters yıkama (back wash), çevrim

(recirculate) ve durulama (rinse) kademeleri bulunmaktadır. Filtre, boşaltım ve ters yıkama kademeleri için web sayfası üzerinde düğmeler tanımlanmıştır. Kapalı konumu varsayılan konumdur ve web sayfasından hiçbir talimat verilmediği durumdaki vana konumudur. Filtre ve ters yıkama konumlarından biri seçildiği takdirde hazırlanmış devre (Şekil 3.18) çalışmaktadır. Filtre konumu direk çalışmaktadır. Ters yıkama konumu seçildiğinde ise devre ters yıkama süresi olan 30sn.'lik süreyi başlatmaktadır. Öngörülen süre sonunda bu çıkış kapanmakta ve durulama otomatik olarak başlamaktadır. Durulama fazı için öngörülen süre yine 30sn.'dir. Bu süre sonunda filtre düğmesi hala basılı ise kum filtresi, filtre işlemine devam etmektedir. Aşağıdaki devre üzerinde LED olarak gösterilmiş kısımlar çıkış, düğme olarak gösterilmiş kısımlar ise devrenin girişleri yani gömülü web sunucu üzerinden gelen sinyallerdir.



Şekil 3.18 Altı Yollu Vana Kontrol Devresi

PIC temelli hazırlanan devrenin yazılımı aşağıdaki gibidir. İlk kısımda kullanılacak çıkışlar tanımlanmıştır. İkinci kısımda ise gelecek sinyallere göre çıkışların durumunu belirleyecek fonksiyon tanımlanmıştır. Fonksiyonda “b1” olarak adlandırılmış birinci giriş aktif olduğunda “led1” adıyla tanımlanmış B portunun 0 numaralı pini aktif olmaktadır. “b2” olarak tanımlanmış olan ikinci giriş enerjilendiğinde ise 0 numaralı pin sönmekte 1 numaralı aktif olmaktadır. Ardından 30 sn.’lik bekleme süresi çalışmaktadır. Ters yıkama fazı için tanımlanmış bu süre kullanılacak kum filtresinin ters yıkama süresine göre değiştirilebilir. Bu süre sonunda 1 numaralı çıkış pini kapanmakta 2 numaralı çıkış aktif olmaktadır. 30 sn.’lik ikinci bekleme süresinin ardından ise bu çıkış kapanmakta ve fonksiyon başa dönmektedir.

```

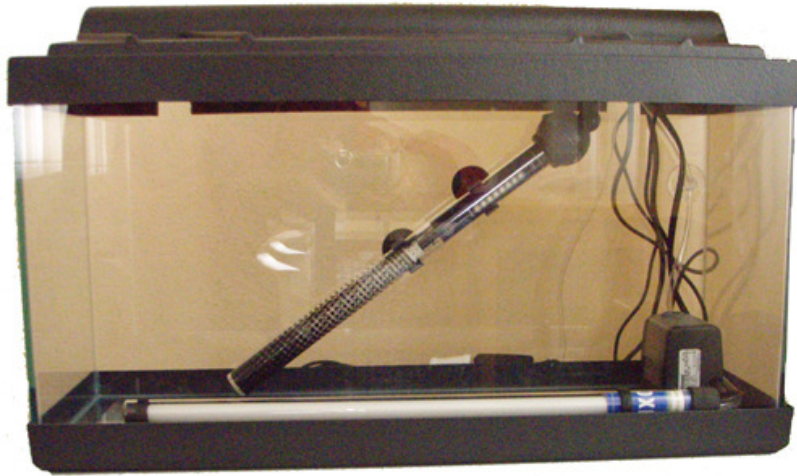
TRISA = %11111111
TRISB = %00000000
B1 VAR PORTA.0
B2 Var PORTA.1
LED1 VAR PORTB.0
LED2 VAR PORTB.1
LED3 VAR PORTB.2
low led1
low led2
low led3
Basla :
if b1 = 0 then
    high led1
else
    low led1
endif
if b2 = 0 then
    low led1
    high led2
    pause 30000
    low led2
    high led3
    pause 30000
    low led3
endif
goto basla

```

Gömülü web sunucu üzerindeki mikrodenetleyici içerisinde üç adet timer bulunmaktadır. Bu timer’lardan ikisi TCP/IP Stack tarafından kullanılmaktadır.

Timer1 olarak adlandırılan üçüncü timer kullanıcı tarafından kullanılabilir durumdadır. Bu timer'ın kullanılabilmesi için Stack üzerinde bazı düzeltmeler yapılarak işleyişinin programlanması gerekmektedir. Ancak timer'ın mikrodenetleyici için fazladan yük oluşturacağı unutulmamalıdır. Ayrıca kullanılacak farklı tip kum filtrelerinin farklı ters yıkama ve durulama süreleri için Stack'in tekrar programlanması sorunlara neden olabilir ve yorucu bir işlemdir. Kullanılacak harici bir mikrodenetleyicinin yeniden programlanması ve arıza durumunda değişimi daha kolaydır. Böyle ek bir devrenin kullanılmasının istenmediği durumda ters yıkama ve durulama evrelerinin manüel olarak web sitesi üzerinden idaresi mümkündür. Bunun için yapılması gereken gömülü web sunucu üzerinden bir çıkışın durulama evresi için ayrılması ve web sayfasına gerekli düğmenin eklenmesidir.

Havuz sisteminin görsel sunumu için bir akvaryum kullanılmıştır (Şekil 3.19). 15lt kapasiteli bu akvaryumun doldurulması ve boşaltılması için iki adet akvaryum su pompası kullanılmıştır. Isıtıcı ve ışık, su altında çalışacak şekilde yine akvaryum ekipmanları arasından seçilmiştir. Havuzlarda kullanılan ozon pompasının simülasyonu için bir hava pompası kullanılmıştır. Asit, kostik ve klor pompalama sistemleri için yine bir su pompası kullanılmıştır. Tüm bu ekipmanların denetimi için gerekli giriş ve çıkışlar denetim kartı üzerinden sağlanmıştır.



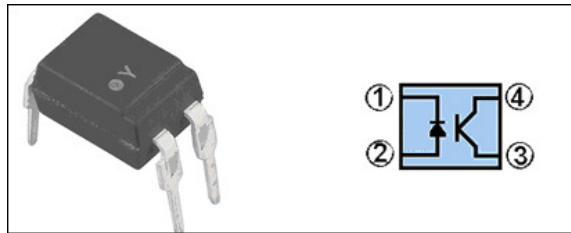
Şekil 3.19 Havuz Modeli

3.5.1.1 LM 35

National firması tarafından üretilen LM35 sıcaklık algılayıcısı, analog tipte olup santigrat derece başına 10 mV gerilim üretir. Yani, sıcaklığın her 1°C artışına karşılık, çıkış gerilimi 10mV artar. Örneğin, 20°C için çıkış gerilimi 200mV iken, 100°C için çıkış gerilimi 1V'dur. LM35 sıcaklık algılayıcısının pek çok tipi vardır. Her bir modelin sıcaklık ölçüm aralığı ve doğruluğu farklıdır. Örneğin, LM35DZ adlı algılayıcı, 0°C ile 100°C arasındaki sıcaklıkları ölçer ve son derece doğrusal bir karakteristiğe sahiptir (Anonim E).

3.5.1.2 Opto Kuplör

Opto Kuplörler bir infaruj LED ve foto alıcıdan oluşur. Aralarındaki iletişim ışık ile sağlanır. Işık kaynağı LED'dir. LED voltaj uygulandığında ışık verir ve denetim ucu ışığa hassas olan alıcı eleman iletime geçer. Transistör, Triyak, MOSFET, Tristör, TTL lojik, CMOS vb. farklı çeşitleri vardır. Bazılarında tetikleme uçları ekstra olarak vardır; özel durumlarda kullanmak için gerekebilir. Örnek olarak PC817'nin 4 bacağı vardır (Şekil 3.20). İki LED için diğer ikisi transistörün kollektör ve emiter uçları içindir. Baz ucu yoktur. 4N25 ise 6 bacaklıdır ve baz ucu vardır (Anonim G).

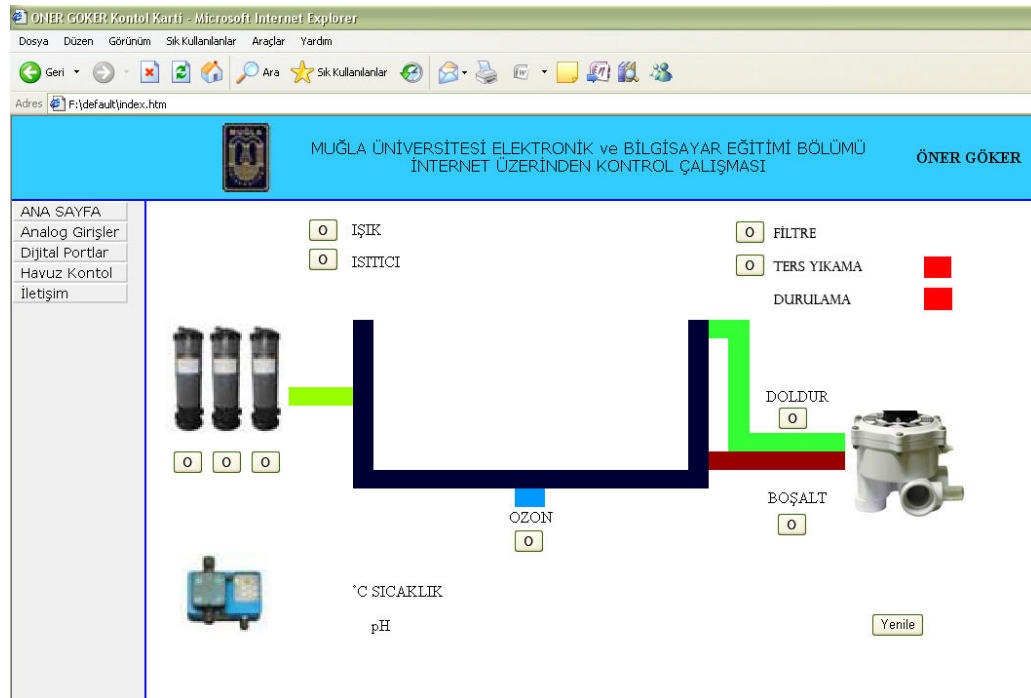


Şekil 3.20 Opto Kuplör

İki farklı devre arasında yalıtımı sağlamak için kullanılır. Denetim kartından gömülü İnternet sunucusuna gelecek bilgilerin (gerilimlerin) karta zarar vermesinin önlenmesi için arada tampon görevi yapacaktır.

3.5.2 Yazılım

Uygulamanın yazılımı, HTML ve JavaScript programlama dilleriyle yapılmıştır. Yazılımda Dreamweaver programından yararlanılmıştır. Kullanılan resim dosyalarının işlenmesinde ise Photoshop programı kullanılmıştır. Denetim kartının yönetimi için tasarlanmış web sayfasının kodları Ek-3'te sunulmuştur. Web sayfasının ekran görüntüsü Şekil 3.21'de görülmektedir.



Şekil 3.21 Web Sayfası Ekran Görüntüsü

Web sayfası üzerinde bulunan görsel öğeler; kısıtlı veri alanı ve bağlantı hızı nedeniyle düşük çözünürlüklüdürler. Havuz şekli, denetim kartından gelen sinyaller doğrultusunda seviyeyi gösterecek şekilde programlanmıştır. Web sayfası üzerinden sıcaklık ve pH değerleri analog olarak gözlenebilmektedir. Denetim kartı üzerindeki rölelerin çıkışlarını aktif etmek için sayfa üzerine düğmeler yerleştirilmiştir. Bu düğmelere basıldığında gömülü web sunucuya %xx şeklinde bir tanımlama gitmekte

ve ilgili pin aktif edilmektedir. Web sayfası üzerinden istenen pinlerin giriş veya çıkış olarak ayarlanması mümkündür. Yine tüm analog ve dijital olarak ayarlanmış pinlerin değerleri sayfa üzerinden görülebilmektedir. Ayrıca gömülü web sunucunun IP adresi İnternet üzerinden yine bu web sitesi vasıtasıyla değiştirilebilmektedir.

3.5.2.1 JavaScript kodları

Gömülü web sunucunun analog girişleri 0-255 arası değer alabilmektedir. Alınan bu veri aralığı uygun görsel formata dönüştürülmelidir. Aşağıdaki kodlar gerekli hesaplamayı yapmaktadır.

```
<script type="text/javascript">
var AN0 = %n10;
var AN1 = %n11;
var AN2 = %n12;
var AN3 = %n13;
var AN4 = %n14;
var AN5 = %n15;
var AN6 = %n16;
var AN7 = %n17;
var AN8 = %n18;
var AN9 = %n19;
var AN10 = %n1A;
var AN11 = %n1B;
function sicaklik(i)
{
var num = i/(5);
var result = num.toFixed(2);
document.write(result);
}
function pH(i)
{
var num = i/(17);
var result = num.toFixed(2);
document.write(result);
}
</script>
```

İlk kısımda gömülü web sunucu üzerindeki tüm analog girişler; ileride kullanılacak değişkenlere atanmıştır. İkinci kısımda ise hesaplamalar için çağrılacak fonksiyonlar tanımlanmıştır. Burada analog giriş değerlerinin uygun biçime

dönüştürülmesi için gerekli hesaplamalar görülmektedir. Sıcaklık için girişin değeri 5'e bölünmektedir. 0-5V arası değer veren LM35DZ algılayıcısı 0-100°C arası ölçüm yapmaktadır. Bu veriler doğrultusunda yapılan hesaplama ve ölçümler ile bu sonuca ulaşılmıştır. pH için ise girişin değeri 17'ye bölünmüştür. pH ölçüm aralığı 0-15 arasındır. Algılayıcıdan gelen değer en fazla 5V yani 255 değeri olduğundan $255/15=17$ değerine ulaşılmıştır.

Aşağıdaki kodlar ise web sayfası üzerine yerleştirilecek düğmelerin işlevlerini tanımlamak için yazılmıştır.

```
<script type="text/javascript">
function lnb() {
    document.write("<td class=bCel>&nbsp;");
}
function ln(io,val,name) {
    document.write("<td class=bCel align=center
style=\"padding:1px;\">");
    if(io==0) {
        document.write("<input style=\"width:38px;\"
type=submit
value="+val+" name=\""+name+"\">");
    }
    else {
        document.write(val);
    }
    document.write("</td>");
}
function tdh(s) {
    document.write("<td class=bCel align=center><b><div
style=\"width:30px;\">"+s+"</div></b></td>");
}
</script>
```

Web sayfası üzerinde, düğme yerleştirilmek istenen yerlere aşağıdaki gibi bir tanımlama yerleştirilmelidir.

```
<script type="text/javascript">
ln(%b50,%b00,"bx0");
</script>
```

Bu tanımlama ile yukarıdaki fonksiyon çağrılacaktır. Ayrıca fonksiyona üç değer gönderilmektedir. Bunlar; kullanılacak pinin giriş veya çıkış olma durumu, değeri ve adıdır. Fonksiyonda bu değerler işlenerek düğme yaratılır ve üzerinde pinin

değeri gösterilir. Fonksiyon içerisindeki “io” tanımlamasına pinin giriş veya çıkış olma durumu gönderilmektedir. Pinin durumu giriş ise düğme yaratılmayacak; pin değeri görüntülenecektir. Çıkış olarak tanımlandığında ise düğme oluşturulacak ve gömülü web sunucuya sinyal gönderilecektir. “val” tanımlaması pinin durumuna göre 1 veya 0 değeri alacaktır. Bu sayede düğmeye tekrar basıldığında eski durumunun tersi bir sinyal gönderilecektir. “name” tanımlamasına ise pinin aktif değeri atanmaktadır. Bu sayede pinin değeri düğme üzerinde yazılacak ve görüntülenecektir. Aynı fonksiyon birçok yerde çağrılabilir. Bu özellik, farklı kullanımlar için farklı sayfalar tasarlanabilmesini mümkün kılmaktadır. Sayfa üzerinde dijital ve analog pinlerin durumlarının görüntülenmesi için farklı sayfalar tasarlanmıştır.

Aşağıda belirtilmiş fonksiyon ise havuz su seviyesini web sayfası üzerinde görüntülemek için yazılmıştır. Tablolar içinde oluşturulmuş havuz şekli içerisinde üç satır bulunmaktadır. Her satıra farklı bir tanımlama numarası verilmiştir ve beyaz renk verilmiştir. Sayfa başında tanımlanmış değişkenler içerisine atanan pin değerleri ile bir karşılaştırma algoritması çalıştırılmış ve sonuca göre çerçevenin renginin beyaz veya mavi olması sağlanmıştır. Havuz içerisine yerleştirilmiş problemlerden sinyal geldiğinde satır arka planı mavi renk almaktadır.

```
<td colspan="13" rowspan="2" id="1"></td>
  <script language="javascript">
    var h = document.getElementById(1);
    h.style.background = AN11 < 200 ? '#FFFFFF' :
'#73A2EF' ;
  </script>
```

Altı yollu vananın denetiminde kullanılan elektronik devre, gömülü web sunucuya çıkış sinyali göndermektedir. Çıkış sinyalleri geldiğinde aşağıdaki kodlar işlem yapmakta ve çalışma adımlarını ekranda belirtilmiş çerçevelerin rengini değiştirerek göstermektedir.

Web sitesi üzerindeki düğmelere basılarak; gömülü web sunucuya ilgili pinin değeri, internet üzerinden gönderilir. Web sunucu kendi üzerindeki çıkışa istenen bu değeri aktarır. Ardından denetim kartına giden bu bilgi ile ilgili röle iletime gider. Kart üzerinden gelen veriler ise yine web sunucu üzerinden, istek yapılan yere gönderilir.

4. ARAŞTIRMA BULGULARI

Yapılan arařtırmalarda görölmüřtür ki İnternet üzerinden denetim, bir sisteme uzaktan eriřim istendiđinde kullanılabilir en etkili yöntem haline gelmiřtir. Tüm dünyayı bir ađ gibi sarmıř kablolama alt yapısı, farklı eriřim yöntemlerine (fiber optik, kablosuz bađlantı, vb.) elveriřli oluřu, güçlü firmalar tarafından güncelleniyor oluřu ve eriřtiđi hız ile bu hıza karřılık düşük bađlantı ücretli oluřu İnternet'i cazip kılmaktadır.

İnternet'e bađlantı için bir bilgisayarın veya pahalı denetim sistemlerinin (PLC) kullanılması yerine mikrodenetleyici tabanlı bir elektronik kartın kullanılması yeterlidir. Ancak kullanılacak mikrodenetleyicinin İnternet'e bađlantı için gerekli protokolleri iřleyebilmesi için belli özelliklere sahip olması gerekmektedir. Bu özellikler hız, kapasite ve yeterli giriř-çıkıř sayısıdır. 18 serisi mikrodenetleyicilerin bu özelliklere sahip olduđu görölmüřtür. 18 serisinde farklı kapasitelerde deđiřik mikrodenetleyiciler bulunmaktadır. Örneđin 18F26XX bir mikrodenetleyici ile tasarlanacak sistem en fazla üç kullanıcının aynı anda bađlanmasına imkan sađlarken daha üst mikrodenetleyicilerle bu sayı arttırılabilmektedir.

Gömülü sistemler ile denetim, havuz ve akvaryum denetiminde yeterli olmaktadır. Aktarılan veriler güvenli olarak iletilmekte ve sorun çıkmamaktadır. İnternet altyapısının da sađladıđı kesintisiz iletiřim ile sistem sorunsuzca çalışmaktadır. Ancak İnternet iletiřiminde ve İnternet servis sađlayıcılarında yařanabilecek sorunlar nedeniyle sistem ile bađlantının gerçekteřemeyeceđi durumu göz önünde bulundurulmalıdır. Ayrıca İnternet eriřiminin olmadıđı bölgeler de mevcuttur. Bu bölgelerde de sistem ile bađlantı gerçekteřtirilemeyecektir.

Web sayfası üzerinde anlık deđerlerin görüntülenmesi için sayfanın yenilenmesi gerekmektedir. Bu durumun önlenmesi için Ajax programlama dili kullanılabilir. Ajax ile yenilemeye gerek duyulmadan giriřlerin anlık deđerleri web sayfası üzerinde görüntülenebilecektir.

5. SONUÇ ve DEĞERLENDİRME

Bu tez çalışmasında mevcut havuz ekipmanlarının İnternet üzerinden denetimi ve gözlemi yapılmıştır. Havuzun dolun ve boşaltım pompaları, pH dengelemesi için kullanılan asit ve baz pompaları, temizliđi için klor ve ozon pompaları, ışık ve ısıtıcıları, pH, sıcaklık ve seviye ölçümleri ile kum filtresinin temizleme ve durulama işlemleri gerçekleştirilebilmiştir. Bu sistemler akvaryum ve ekipmanları ile simüle edilmiş ve başarılı sonuçlar elde edilmiştir. Mevcut denetim sistemi modüler yapıdadır ve küçük değişikliklerle benzer sistemlere uygulanabilir.

İnternet üzerinden denetim, günümüz denetim sistemleri gereksinimlerini önemli ölçüde karşılamaktadır. Ethernet ağı ile kapalı sistemlerde yeterli veri güvenliđi sağlanabilir ancak İnternet ağına meydana gelebilecek sorunlar güvenliđi tehlikeye atabilir. Ethernet ağının güvenliđi yüksek ve maliyetleri düşüktür. Ayrıca manyetik alanların çok yoğun olduđu bölgelerde fiber optik kabloların kullanımı mümkündür. Gerekli kablolama için iş gücü bulmak oldukça kolaydır. Ayrıca kablolanmanın mümkün olmadığı veya istenmediđi bölgelerde kablosuz iletişim olanaklarından yararlanılabilir. Gerekli modüller düşük fiyatlı ve yine bulunması kolay cihazlardır.

Çalışma, İnternet üzerinden denetim için denetlenecek ve denetleyecek sistemler olarak bilgisayarların veya PLC'lerin kullanıldığı denetim sistemleriyle kıyaslandığında düşük maliyetli olmuş ve benzer başarılı sonuçlar elde edilmiştir. Anlaşılmıştır ki yapılan çalışma, havuz firmalarının işlerini kolaylaştırmakta ve pazarlama için önemli bir avantaj sağlamaktadır. Ayrıca akvaryumu bir hobi olarak kullanan insanlara akvaryumlarına uzaktayken de müdahale etme şansı vermektedir.

KAYNAKLAR

Ahmed, I., Wong, H., Kapila, V., 2004. *Internet-Based Remote Control using a Microcontroller and an Embedded Ethernet*, American Control Conference Boston, Massachusetts June 30 -July 2,2004: 1329-1334

Berners-Lee, T., *RFC1866 - Hypertext Markup Language*, Network Working Group, 1995

Çölkesen, R., Örencik, B., 2003. *Bilgisayar Haberleşmesi ve Ağ Teknolojileri*, Papatya Yayıncılık Eğitim Bilgisayar Sis. San. Ve Tic A.Ş., İstanbul, 447p.

Doğan, B., *Web Üzerinden Sistem Kontrolü ve Uzaktan Erişimli Laboratuvar Uygulaması*, Yüksek Lisans, Marmara Üniversitesi, İstanbul, 2007

Droms, R., *RFC1541 - Dynamic Host Configuration Protocol*, Network Working Group, 1993

Eşme, E., *Uzaktan Kontrol Edilebilen Bir Kalp Cihazı Tasarımı*, Yüksek Lisans, Selçuk Üniversitesi, Konya, 2006

Fielding, R., *RFC2616 - Hypertext Transfer Protocol*, Network Working Group, 1999

Insam, E., 2003. *TCP/IP ve İnternet Üzerinden Kontrol*, Bileşim Yayınları, İstanbul, 287p.

Kaplan, Y., 2000. *Veri Haberleşmesi/Network Temelleri*, Papatya Yayıncılık Eğitim Bilgisayar Sis. San. Ve Tic A.Ş., İstanbul, 352p.

Kapucu, C., *Fotovoltaik Sistemler İçin İnternet Üzerinden Gerçek Zamanlı Bir İzleme Sistemi Tasarım ve Uygulaması*, Yüksek Lisans, Muğla Üniversitesi, Muğla, 2006

Ovalı, B., *İnternet Üzerinden Gezin Robot Denetimi*, Yüksek Lisans, Yıldız Teknik Üniversitesi, İstanbul, 2005

Özdağ, R., *Akıllı Ev Otomasyon Sistemi ve İnternet Üzerinden Uygulaması*, Yüksek Lisans, Yüzüncü Yıl Üniversitesi, Van, 2006

Plummer, D. C., *RFC826 - Ethernet Address Resolution Protocol*, Network Working Group, 1982

Postel, J., *RFC791 - Internet Protocol*, Defense Advanced Research Projects Agency Information Processing Techniques Office, 1981

Postel, J., *RFC792 - Internet Control Message Protocol*, Network Working Group, 1981

Postel, J., *RFC793 - Transmission Control Protocol*, Defense Advanced Research Projects Agency Information Processing Techniques Office, 1981

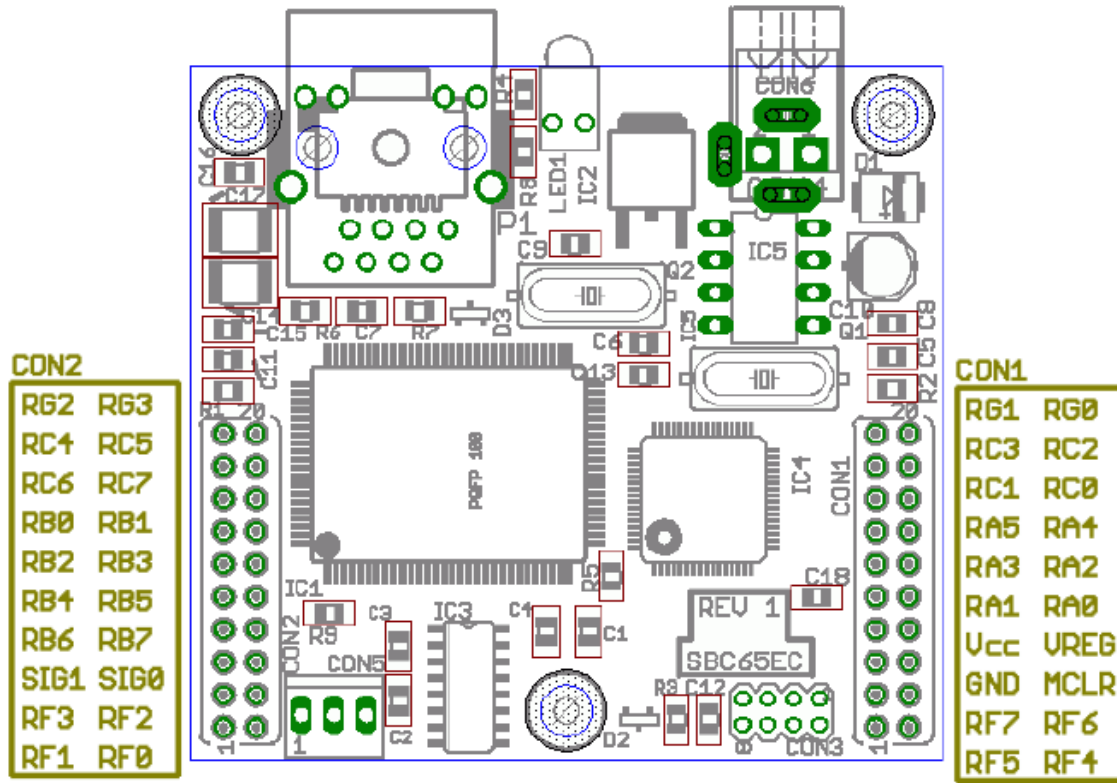
Postel, J., *RFC768 - User Datagram Protocol*, Defense Advanced Research Projects Agency Information Processing Techniques Office, 1980

Postel, J., *RFC959 - File Transfer Protocol*, Network Working Group, 1985

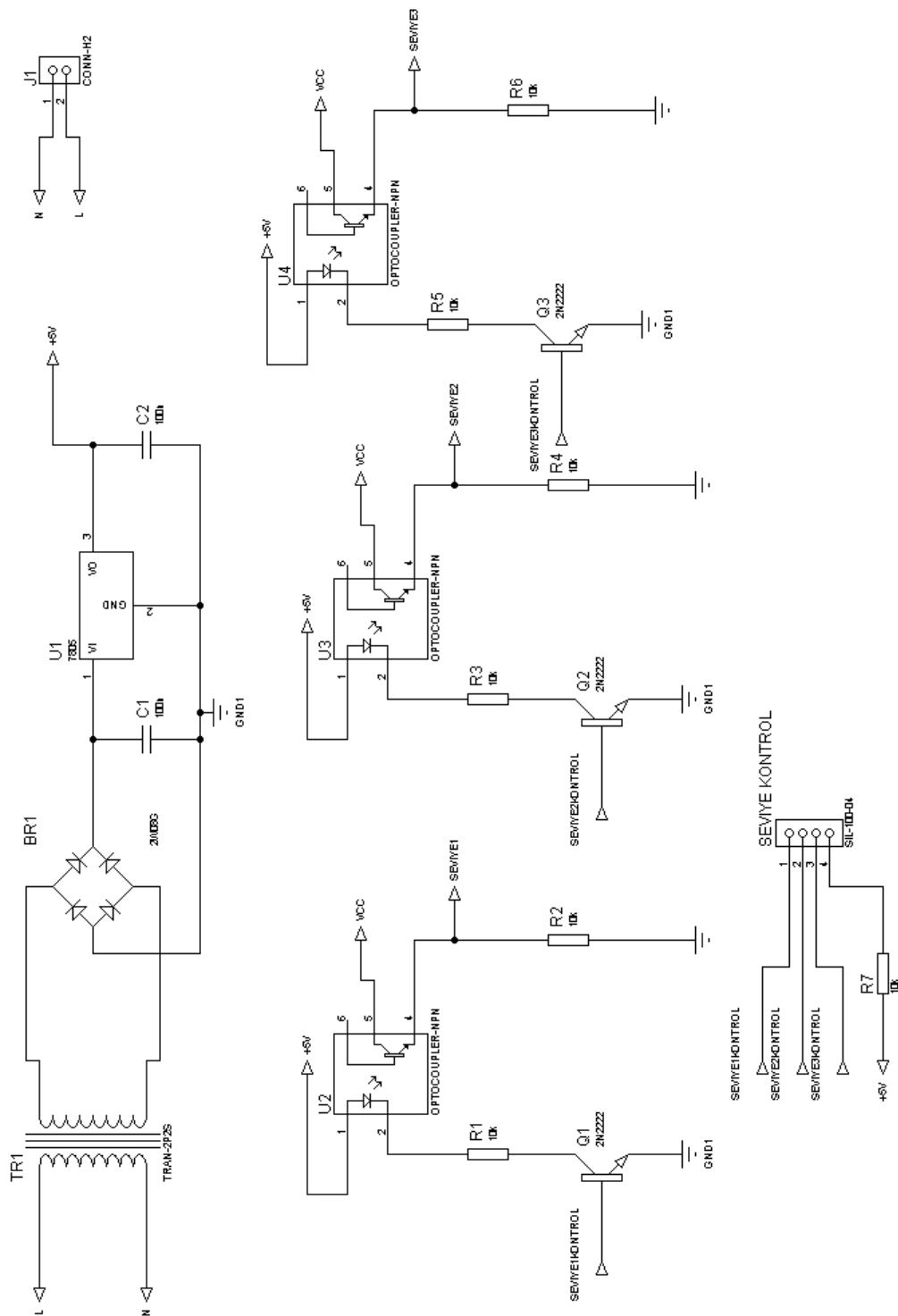
Anonim

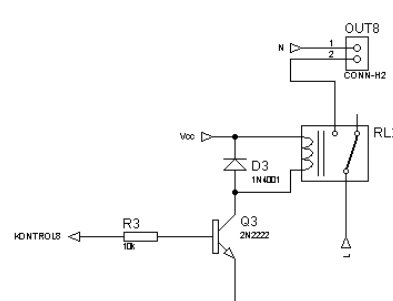
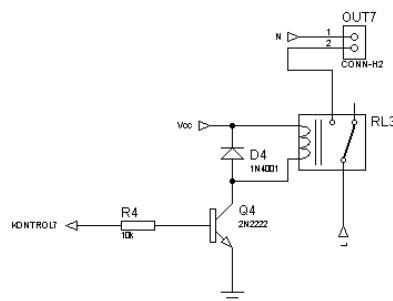
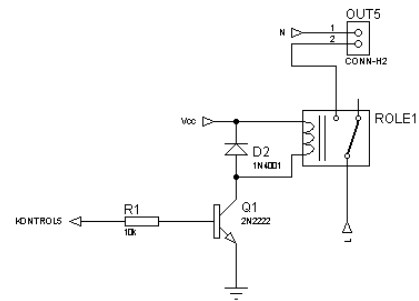
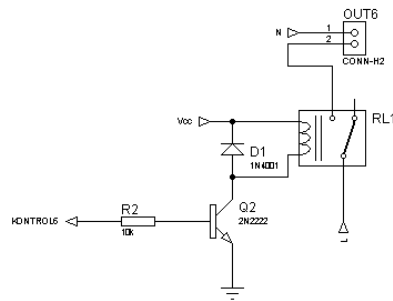
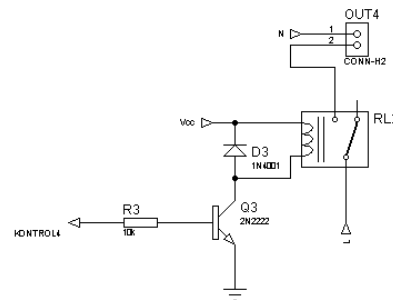
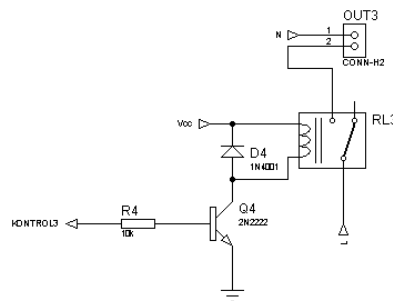
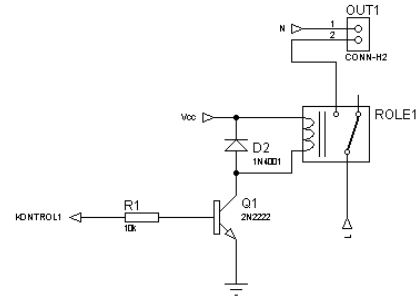
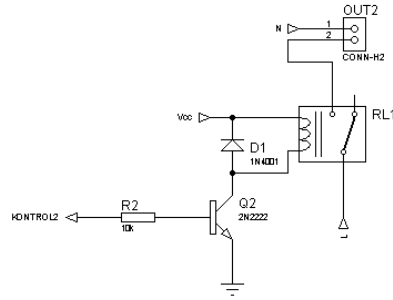
- a. 24CXX SERİ EEPROM, Antrak Gazetesi, http://www.antrak.org.tr/gazete/122003/ziya_erdemir.htm, Mart 2008
- b. 24lc512, Wulfden at Hawk's Mountain, http://wulfden.org/downloads/datasheets/24LC512_21754E.pdf, Mart 2008
- c. CGI Programlama, Ortak Bilgi, <http://www.ortakbilgi.com/12610/cgi-nedir.html>, Şubat 2008
- d. JavaScript, Bilgisayar Öğren, www.bilgisayarogren.com/java1.doc, Mayıs 2008
- e. Lm35, National Semiconductor, <http://www.national.com/mpf/LM/LM35.html>, Mart 2008
- f. MAX232, Maxim/Dallas Semiconductor, http://www.maxim-ic.com/quick_view2.cfm?qv_pk=1798, Mart 2008
- g. Opto Kuplör, 320Volt, <http://320volt.com/opto-kuplor-nedir-nasil-calisir-cesitleri>, Mart 2008
- h. PIC, Vikipedi, <http://tr.wikipedia.org/wiki/PIC>, Mart 2008
- i. PIC18f6680, Microchip Technology Inc., <http://ww1.microchip.com/downloads/en/DeviceDoc/30491c.pdf>, Mart 2008
- j. Realtek RTL8019AS, Studenckie Koło Naukowe MIPSAs, http://mipsa.p.lodz.pl/download/BazaWiedzy/ukl_komunikacji/RTL8019AS.pdf, Mart 2008
- k. ROM, RAM, EPROM, EEPROM, Forum Vadisi, <http://www.forumvadisi.com/elektronik-alani/150794-rom-ram-eprom-eprom-nedir.html>, Mart 2008
- l. SBC68EC, Modtronix Engineering, http://www.modtronix.com/product_info.php?products_id=196, Şubat 2008
- m. TCP/IP application note, Microchip Technology Inc., http://www.microchip.com/Stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=1824&appnote=en011993, Mart 2008

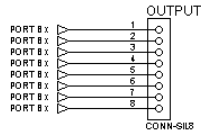
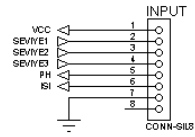
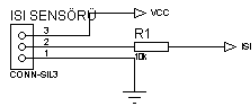
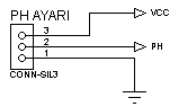
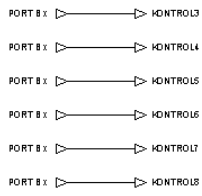
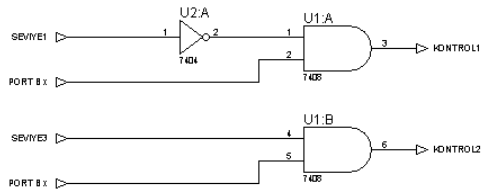
Ek 1-b: SBC68EC Bağlantı Pinleri ve Baskı Devresi



Ek 2: Havuz Kontrol Kartı Şematiği







Ek 3: Gömülü Web Sunucu Web Sayfası

Anasayfa

```

<html>
<head>
<title>ONER GOKER Kontrol Kartı</title>
<meta http-equiv="Content-Type" content="text/html;
charset=UTF-8">
<meta http-equiv="Pragma" content="no-cache">
<style type="text/css">
.top {border-bottom:2px double #0000FF; background-
color:#33CCFF; text-align:center; height:20px; padding:3px;}
.left {border-right:2px solid #0000FF; background-
color:#F0F0F0; padding:0px; vertical-align:top;}
.menuBtn {
padding-left:7px;
padding-right:7px;
border-top:2px solid #FFFFFF;
border-right:2px solid #CCCCCC;
border-bottom:2px solid #CCCCCC;
border-left:2px solid #FFFFFF;
white-space:nowrap;
font-size:14px;
}
A {color:#000000; text-decoration:none;}
body {height:100%; margin:0; padding:0; font-
family:Verdana,Arial,sans-serif;}
html {height:100%; overflow-y:hidden;}
.style1 {font-family: "Times New Roman", Times, serif}
.style2 {font-family: "Times New Roman", Times, serif; font-
weight: bold; }
</style>
</head>

<body>
<table id=mainTable height="100%" width="100%" cellpadding=0
cellspacing=0>
  <!-- Top -->
  <tr>
    <td class=top colspan=2>
      <table width="833" border="0" align="center"
bordercolor="#FFFFFF">
        <tr>
          <td width="50" height="75"></td>
          <td width="641" align="center"
valign="middle">MUĞLA ÜNİVERSİTESİ ELEKTRONİK ve BİLGİSAYAR
EĞİTİMİ BÖLÜMÜ İNTERNET ÜZERİNDEN KONTROL ÇALIŞMASI </td>
          <td width="120"><div align="center"
class="style2">ÖNER GÖKER </div></td>
        </tr>
      </table>
    </td>
  </tr>

```

```

</tr>

<tr>
<!-- Left -->
<td width="11%" class=left><table cellpadding=0
cellspacing=0>
<tr>
<td class=menuBtn><a href="intro.htm"
target=main>ANA SAYFA </a></td>
</tr>
<tr>
<td class=menuBtn><a href="aval.cgi"
target=main>Analog Girişler </a></td>
</tr>
<tr>
<td class=menuBtn><a href="ioval.cgi"
target=main>Dijital Portlar </a></td>
</tr>
<tr>
<td class=menuBtn><a href="expl.cgi"
target=main>Havuz Kontrol </a></td>
</tr>
<tr>
<td class=menuBtn><a href="contact.htm"
target=main>İletişim</a></td>
</tr>
</table></td>

<!-- Main -->
<!-- Option 1: Make height of td below 2000, and remove
"overflow-y: hidden;" from html style -->
<!-- Option 2: Make height of td below "100%", and add
"overflow-y: hidden;" to html style -->
<td width="89%" height="100%" valign=top>
<iframe width="100%" height="100%" name=main
src="intro.htm" scrolling="auto" frameborder=0></iframe>
</td>
</tr>
</table>

</body>
</html>

```

Anasayfa genel görünümünün verildiği sayfadır. Üç çerçeveden oluşmaktadır. Birinci çerçeve başlık, ikinci çerçeve bağlantı menüsü ve üçüncü çerçeve bağlantıların çağrıldığı penceredir.

Anasayfada giriş sayfası bulunmaktadır.

Havuz Kontrol Sayfası

```

<html>
<head>
<meta http-equiv="refresh" content="5">
<meta http-equiv="Content-Type" content="text/html;
charset=UTF-8" />
<title>ÖNER GÖKER</title>
<script src="lib01.js"></script>
<script type="text/javascript">
function lnb() {
    document.write("<td class=bCel>&nbsp;</td>");
}
function ln(io,val,name) {
    document.write("<td class=bCel align=center
style=\"padding:1px;\">");
    if(io==0) {
        document.write("<input style=\"width:38px;\"
type=submit value="+val+" name=\""+name+"\">");
    }
    else {
        document.write(val);
    }
    document.write("</td>");
}
function tdh(s) {
    document.write("<td class=bCel align=center><b><div
style=\"width:30px;\">"+s+"</div></b></td>");
}
</script>
<script type="text/javascript">

var AN0 = %n10;
var AN1 = %n11;
var AN2 = %n12;
var AN3 = %n13;
var AN4 = %n14;
var AN5 = %n15;
var AN6 = %n16;
var AN7 = %n17;
var AN8 = %n18;
var AN9 = %n19;
var AN10 = %n1A;
var AN11 = %n1B;

function sicaklik(i)
{
var num = i/(5/4);
var result = num.toFixed(2);
document.write(result);
}

function pH(i)
{
var num = i/(255/17);

```

```

var result = num.toFixed(2);
document.write(result);
}

</script>
<style type="text/css">
<!--
.style1 {color: #FFFFFF}
-->
</style>
</head>

<body>
<form method=GET action=expl.CGI>
<table width="768" height="424" border="0">
  <tr>
    <td width="1" height="26">&nbsp;</td>
    <td width="122">&nbsp;</td>
    <td width="55">
      <script type="text/javascript">
        ln(%b50,%b00,"bx0");
      </script> </td>
    <td width="202"><strong>IŞIK</strong></td>
    <td width="114"><strong>FİLTRE</strong></td>
    <td width="70"><script type="text/javascript">
      ln(%g50,%g00,"gx0");
    </script></td>
    <td width="55">&nbsp;</td>
    <td width="56">&nbsp;</td>
    <td width="80">&nbsp;</td>
    <td width="1">&nbsp;</td>
  </tr>
  <tr>
    <td height="36">&nbsp;</td>
    <td>&nbsp;</td>
    <td>
      <script type="text/javascript">
        ln(%b51,%b01,"bx1");
      </script> </td>
    <td><strong>ISITICI</strong></td>
    <td><strong>TERS YIKAMA</strong></td>
    <td><script type="text/javascript">
      ln(%g51,%g01,"gx1");
    </script></td>
    <td colspan="2" td id="4"><div
align="center">YIKAMA</div></td>
    <script language="javascript">
      var h = document.getElementById(4);
      h.style.background = AN6 < 200 ?
'#FFFFFF' : '#73A2EF' ;
    </script>
    <td colspan="1" td id="5">DURULAMA</td>
    <script language="javascript">
      var h = document.getElementById(5);

```

```

                                h.style.background = AN7 < 200 ?
'#FFFFFF' : '#73A2EF' ;
                                </script>
                                <td>&nbsp;&nbsp;&nbsp;</td>
</tr>
<tr>
                                <td height="21">&nbsp;&nbsp;&nbsp;</td>
                                <td>&nbsp;&nbsp;&nbsp;</td>
                                <td colspan="6" rowspan="4"><table width="571" border="0"
cellpadding="0" cellspacing="0">
                                <tr>
                                <td width="18">&nbsp;&nbsp;&nbsp;</td>
                                <td width="18">&nbsp;&nbsp;&nbsp;</td>
                                <td width="18">&nbsp;&nbsp;&nbsp;</td>
                                <td width="18">&nbsp;&nbsp;&nbsp;</td>
                                <td width="18">&nbsp;&nbsp;&nbsp;</td>
                                <td width="18">&nbsp;&nbsp;&nbsp;</td>
                                <td width="18">&nbsp;&nbsp;&nbsp;</td>
                                <td width="18">&nbsp;&nbsp;&nbsp;</td>
                                <td width="18">&nbsp;&nbsp;&nbsp;</td>
                                <td width="18">&nbsp;&nbsp;&nbsp;</td>
                                <td width="34">&nbsp;&nbsp;&nbsp;</td>
                                <td width="26">&nbsp;&nbsp;&nbsp;</td>
                                <td width="36">&nbsp;&nbsp;&nbsp;</td>
                                <td width="18">&nbsp;&nbsp;&nbsp;</td>
                                <td width="18">&nbsp;&nbsp;&nbsp;</td>
                                <td width="18">&nbsp;&nbsp;&nbsp;</td>
                                <td width="18">&nbsp;&nbsp;&nbsp;</td>
                                <td width="18">&nbsp;&nbsp;&nbsp;</td>
                                <td width="18">&nbsp;&nbsp;&nbsp;</td>
                                <td width="17">&nbsp;&nbsp;&nbsp;</td>
                                <td width="19">&nbsp;&nbsp;&nbsp;</td>
                                <td width="83">&nbsp;&nbsp;&nbsp;</td>
                                </tr>
                                <tr>
                                <td>&nbsp;&nbsp;&nbsp;</td>
                                <td>&nbsp;&nbsp;&nbsp;</td>
                                <td>&nbsp;&nbsp;&nbsp;</td>
                                <td rowspan="7" bgcolor="#000033">&nbsp;&nbsp;&nbsp;</td>
                                <td colspan="13" rowspan="2" td id="3"></td>
                                <script language="javascript">
                                var h = document.getElementById(3);
                                h.style.background = AN9 < 200 ?
'#FFFFFF' : '#73A2EF' ;
                                </script>
                                <td rowspan="7" bgcolor="#000033">&nbsp;&nbsp;&nbsp;</td>
                                <td colspan="2" bgcolor="#33FF33">&nbsp;&nbsp;&nbsp;</td>
                                <td>&nbsp;&nbsp;&nbsp;</td>
                                </tr>
                                <tr>
                                <td height="31">&nbsp;&nbsp;&nbsp;</td>
                                <td>&nbsp;&nbsp;&nbsp;</td>
                                <td>&nbsp;&nbsp;&nbsp;</td>
                                <td>&nbsp;&nbsp;&nbsp;</td>
                                <td rowspan="5" bgcolor="#33FF33">&nbsp;&nbsp;&nbsp;</td>
                                <td>&nbsp;&nbsp;&nbsp;</td>

```

```

</tr>
<tr>
  <td>&nbsp;</td>
  <td>&nbsp;</td>
  <td>&nbsp;</td>
  <td colspan="13" rowspan="3" td id="2"></td>
    <script language="javascript">
      var h = document.getElementById(2);

      h.style.background = AN10 < 200 ?
'#FFFFFF' : '#73A2EF' ;
    </script>
  <td>&nbsp;</td>
  <td>&nbsp;</td>
</tr>
<tr>
  <td colspan="3" bgcolor="#99FF00">&nbsp;</td>
  <td>&nbsp;</td>
  <td><div
align="center"><strong>DOLDUR</strong></div></td>
</tr>
<tr>
  <td height="28">&nbsp;</td>
  <td>&nbsp;</td>
  <td>&nbsp;</td>
  <td>&nbsp;</td>
  <td><div align="center"><script
type="text/javascript">
ln(%b54,%b04,"bx4");
</script></div></td>
</tr>
<tr>
  <td>&nbsp;</td>
  <td>&nbsp;</td>
  <td>&nbsp;</td>
  <td colspan="13" rowspan="2" td id="1"></td>
    <script language="javascript">
      var h = document.getElementById(1);
      h.style.background = AN11 < 200 ?
'#FFFFFF' : '#73A2EF' ;
    </script>
  <td>&nbsp;</td>
  <td bgcolor="#33FF33"><div align="center"></div></td>
</tr>
<tr>
  <td>&nbsp;</td>
  <td>&nbsp;</td>
  <td>&nbsp;</td>
  <td colspan="3" bgcolor="#990000">&nbsp;</td>
</tr>
<tr>
  <td>&nbsp;</td>
  <td>&nbsp;</td>
  <td>&nbsp;</td>
  <td colspan="15" bgcolor="#000033">&nbsp;</td>

```

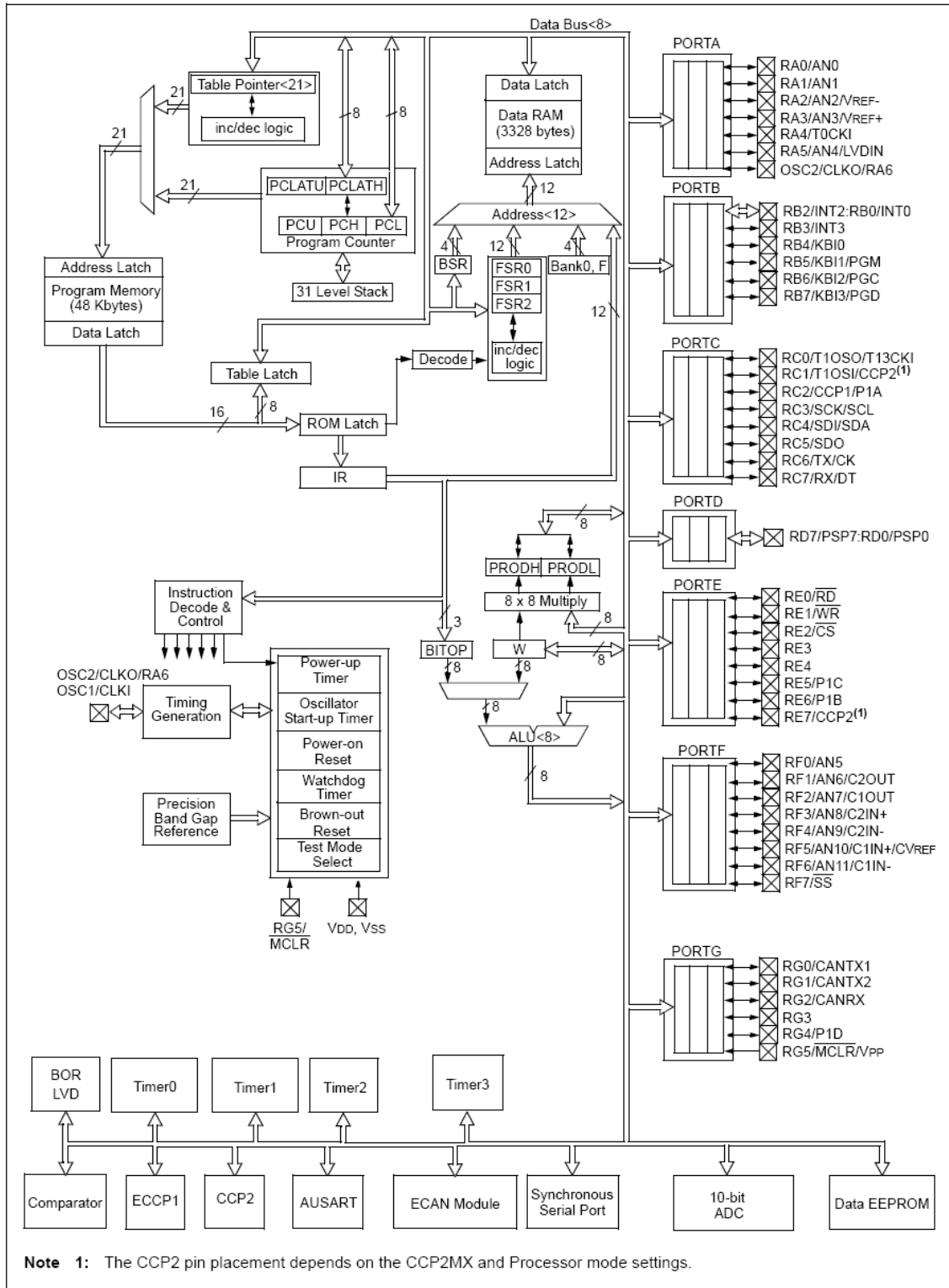


```

        </tr>
        <tr>
            <td colspan="3"><div
align="center"><strong>OZON</strong></div></td>
        </tr>
    </table></td>
    <td>&nbsp;</td>
    <td>&nbsp;</td>
</tr>
<tr>
    <td height="120" rowspan="3">&nbsp;</td>
    <td height="114"><div align="right"></a></div></td>
    <td>&nbsp;</td>
    <td rowspan="3">&nbsp;</td>
</tr>
<tr>
    <td rowspan="2" align="center" valign="top"><table
border="0">
        <tr>
            <td><script type="text/javascript">
ln(%b52,%b02,"bx2");
</script></td>
            <td><script type="text/javascript">
ln(%b52,%b02,"bx2");
</script></td>
            <td><script type="text/javascript">
ln(%b52,%b02,"bx2");
</script></td>
        </tr>
    </table>
    <p align="center">&nbsp;</p>
    <p>&nbsp;</p></td>
    <td valign="top"></td>
</tr>
<tr>
    <td valign="top">&nbsp;</td>
</tr>
<tr>
    <td height="21">&nbsp;</td>
    <td rowspan="3"><div align="center"></div></td>
    <td colspan="2">&nbsp;</td>
    <td colspan="4">&nbsp;</td>
    <td>&nbsp;</td>
    <td>&nbsp;</td>
</tr>
<tr>
    <td height="30" rowspan="2">&nbsp;</td>
    <td height="41">
    <script type="text/javascript">sicaklik(AN8);</script>
    </td>
    <td><strong>&#730;C SICAKLIK </strong></td>

```


Ek 4: PIC 18F4620 Blok Diyagramı



Ek 5: TCP/IP Stack Konfigürasyonu

TANIMLAMA	DEĞER	KULLANILDIĞI MODÜL	AÇIKLAMA
CLOCK_FREQ (compiler.h)	Oscillator Frequency (Hz)	Tick.c	Sistemin çalışma frekansını ayarlamak için kullanılır.
TICKS_PER_SECONDS	10 - 255	Tick.c	Bir saniyeyi hesaplamak için kullanılır.
TICK_PRESCALE_VALUE	2, 4, 8, 16, 32, 64, 128, 256	Tick.c	Clock sayıcı değerini belirlemek için kullanılır.
MPFS_USE_PGRM	N/A	MP File System (MPFS.c)	Program hafızası MPFS hafıza bloğu olarak kullanılacaksa programa eklenir.
MPFS_USE_EEPROM	N/A	MPFS.c	MPFS hafıza bloğu harici EEPROM'da tutulacaksa programa eklenir.
MPFS_RESERVE_BLOCK	0 - 255	MPFS.c	MPFS hafıza bloğu başlamadan önce hafızada ayrılacak rezerv kısmını büyüklüğünü tanımlar.
EEPROM_CONTROL	External Data EEPROM Control Code	MPFS.c	Harici EEPROM'u adreslemek için kullanılır.
STACK_USE_ICMP	N/A	StackTsk.c	ICMP kullanılacaksa yazılıma eklenir.
STACK_USE_SLIP	N/A	SLIP.c	SLIP kullanılacaksa yazılıma eklenir.
STACK_USE_IP_GLEANING	N/A	StackTsk.c	IP GLEANING kullanılacaksa yazılıma eklenir.
STACK_USE_DHCP	N/A	DHCP.c, StackTsk.c	DHCP kullanılacaksa yazılıma eklenir.
STACK_USE_FTP_SERVER	N/A	FTP.c	FTP Server kullanılacaksa yazılıma eklenir.
STACK_USE_TCP	N/A	TCP.c, StackTsk.c	TCP kullanılacaksa yazılıma eklenir. TCP kullanan bir modül yazılımda mevcutsa otomatik olarak aktifleştirilir.
STACK_USE_UDP	N/A	UDP.c, StackTsk.c	UDP kullanılacaksa yazılıma eklenir. UDP kullanan bir modül yazılımda mevcutsa otomatik olarak aktifleştirilir.
STACK_CLIENT_MODE	N/A	ARP.c, TCP.c	Client ile ilgili kodlar aktifleştirilir.
TCP_NO_WAIT_FOR_ACK	N/A	TCP.c	TCP bir sonraki paket gönderilmeden önce ACK

			bekleyecektir.
MY_DEFAULT_IP_ADDR_BYTE? MY_DEFAULT_MASK_BYTE? MY_DEFAULT_GATE_BYTE? MY_DEFAULT_MAC_BYTE?	0 - 255	User Application	IP, MAC, Gateway ve Subnet Mask değerleri tanımlanır.
MY_IP_BYTE? MY_MASK_BYTE? MY_GATE_BYTE? MY_MAC_BYTE?	0 - 255	MAC.c, ARP.c, DHCP.c, User Application	Geçerli IP, MAC, Gateway ve Subnet Mask değerleri uygulama tarafından kaydedilir. DHCP kullanılıyorsa değerler oraya yansıtılır.
MAX_SOCKETS	1 - 253	TCP.c	Seçilen TCP uygulaması için desteklenen toplam socket sayısı tanımlanır.
MAX_UDP_SOCKETS	1 - 254	UDP.c	Seçilen UDP uygulaması için desteklenen toplam socket sayısı tanımlanır.
MAC_TX_BUFFER_SIZE	201 - 1500	TCP.c, MAC.c	Transmit buffer ların büyüklüğünü tanımlar.
MAX_TX_BUFFER_COUNT	1 - 255	MAC.c	Transmit buffer ın toplam sayısını tanımlar
MAX_HTTP_CONNECTIONS	1 - 255	HTTP.c	Aynı anda izin verilen maksimum HTTP bağlantı sayısını tanımlar.
MPFS_WRITE_PAGE_SIZE (MPFS.h)	1 - 255	MPFS.c	MPFS hafızasına kaydedilebilecek sayfa sayısı tanımlanır.
FTP_USER_NAME_LEN (FTP.h)	1 - 31	FTP.c	FTP kullanıcı adının maksimum uzunluğu belirtilir.
MAX_HTTP_ARGS (HTTP.c)	1 - 31	HTTP.c	Kullanılabilecek HTML form alanlarının sayısı belirtilir.
MAX_HTML_CMD_LEN (HTTP.c)	1 - 128	HTTP.c	HTML formların maksimum URL tanımları yapılır.

ÖZGEÇMİŞ

Öner GÖKER 1982 yılında Erzincan'da doğdu. İlk ve orta öğrenimini İstanbul'da bulunan devlet okullarında tamamladı. 2000 yılında Haydarpaşa Anadolu Meslek Lisesi Otomatik Kumanda Bölümü'nden mezun oldu. 2005 yılında Marmara Üniversitesi Elektronik ve Haberleşme Öğretmenliği'nden mezun oldu. 2007 yılında Anadolu Üniversitesi Açık Öğretim Fakültesi İktisat Bölümü'nden mezun oldu.

2005 yılında Altek Alarko Doğalgaz Çevrim santralinde otomasyon teknikeri olarak çalıştı. 2006 yılında Marmaris Çetibeli Ali Fikret Bilgin İlköğretim Okulu'nda bilgisayar öğretmenliği görevine başladı. Halen bu görevine devam etmektedir.