

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

**PIC MİKROKONTROLÖRLERİ İLE KAMPUS
OTOMASYONU**

YÜKSEK LİSANS TEZİ

Elk.Elektrn.Müh. Ahmet GÜLOĞLU

Enstitü Anabilim Dalı : ELEKTRİK-ELEKTRONİK MÜH.
Enstitü Bilim Dalı : ELEKTRONİK
Tez Danışmanı : Yrd. Doç. Dr. Gürsel DÜZENLİ

Mayıs 2007

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

**PIC MİKROKONTROLÖRLERİ İLE KAMPUS
OTOMASYONU**

YÜKSEK LİSANS TEZİ

Elk.Elektrn.Müh. Ahmet GÜLOĞLU

Enstitü Anabilim Dalı : ELEKTRİK-ELEKTRONİK MÜH.

Enstitü Bilim Dalı : ELEKTRONİK

Bu tez 21 / 06 /2006 tarihinde aşağıdaki jüri tarafından Oybirliği ile kabul edilmiştir.

**Yrd. Doç. Dr. Gürsel DÜZENLİ
Jüri Başkanı**

**Doç. Dr. Sadettin AKSOY
Üye**

**Prof. Dr. Hüseyin EKİZ
Üye**

TEŐEKKÜR

En baŐta bu tezi hazırlamamda bana destek olan deđerli hocam Yrd. Doç. Dr. Gürsel DÜZENLİ'ye, zor zamanlarımda bana manevi destek veren aileme, yardımlarından dolayı çalıŐma arkadaŐım Engin ÖZAYYILDIZ'a, imkânlarını kullanmama izin verdiđi ve gerekli toleransı sağladıđı için İstanbul Elektronik Anahtar'a teŐekkürlerimi bir borç bilirim.

İÇİNDEKİLER

TEŞEKKÜR.....	ii
İÇİNDEKİLER	iii
SİMGELER VE KISALTMALAR LİSTESİ.....	vii
ŞEKİLLER LİSTESİ	viii
TABLOLAR LİSTESİ.....	x
ÖZET.....	xi
SUMMARY.....	xii
BÖLÜM 1.	
GİRİŞ.....	1
BÖLÜM 2.	
ETHERNET.....	3
2.1. Ethernetin Tarihi.....	3
2.2. Ethernet veya IEEE 802.3.....	5
2.3. Ethernetin Çalışma Şekli.....	5
2.3.1. Verinin aktarımı: Paketler (Frames).....	5
2.3.1.1. MAC adresi.....	6
2.3.1.2. Multicast ve Broadcast adresleri.....	8
2.3.1.3. CRC hata denetimi.....	8
2.3.1.4. Kabloyu kim kullanacak (CSMA/CD).....	9
BÖLÜM 3.	
ENC28J60 VE ETHERNET YAPISI.....	12
3.1. Paket Formatı.....	12
3.2. Preamble/Start-Of-Frame Delimiter.....	13

3.3. Hedef Adres (Destination Address).....	13
3.4. Kaynak Adres (Source Address).....	13
3.5. Tip/Uzunluk (Type/Length).....	14
3.6. Data.....	14
3.7.Padding.....	14
3.8.CRC.....	15
BÖLÜM 4.	
ENC28J60 HAKKINDA BİLGİ.....	16
4.1. ENC28J60 Blok Yapısı.....	17
4.2. ENC28J60 Temel Uygulama Devresi.....	18
BÖLÜM 5.	
CONTROLLER AREA NETWORK (CAN).....	19
5.1. Veri İletimi.....	19
5.2. Bit Zamanlaması.....	21
5.3. Katmanlar (Layers).....	21
5.4. Paketler.....	22
5.4.1. Data paketi.....	23
5.4.1.1. Temel paket formatı.....	23
5.4.1.2. Genişletilmiş paket formatı.....	24
5.4.2. Uzak paketi (Remote Frame).....	25
5.4.3. Hata paketi (Error Frame).....	26
5.4.4. Aşırı yükleme paketi (Overload Frame).....	26
BÖLÜM 6.	
ETHERNET – CAN KARŞILAŞTIRMASI.....	28
6.1. CAN Bus.....	28
6.2. Ethernet – CAN Karşılaştırması.....	29
BÖLÜM 7.	
SİSTEM AKIŞ DİYAGRAMLARI.....	36
7.1. Sunucu Modülü Akış Diyagramları.....	36

7.1.1. Sunucu main program akış diyagramı.....	37
7.1.2. Sunucu menü kontrol akış diyagramı.....	38
7.1.3. Sunucu “PAGE_CLIENT” yapısı.....	39
7.1.4. Sunucu RS232 seri port kayıt kontrol.....	40
7.1.5. Sunucu cihaz durumlarını isteme yapısı akış diyagramı.....	41
7.1.5. Sunucu clientların durumlarını isteme yapısı akış diyagramı.....	42
7.1.6. Sunucu TCP komut gönderme yapısı akış diyagramı.....	43
7.2. İstemci Modülü Akış Diyagramları.....	44
7.2.1. İstemci main program akış diyagramı.....	44
7.2.2. İstemci TCP iletişimi akış diyagramı.....	45
7.2.3. İstemci RTC yapısı ile cihaz kontrolü akış diyagramı.....	46
7.2.3.1. Başlangıç.....	46
7.2.3.2. Ana program.....	47
7.2.3.3. Program ayarı.....	48
7.2.3.4. RTC saat kontrolü.....	49
7.2.3.5. Program kontrol.....	50
7.2.3.6. Periyodik LAN açma/kapatma kontrolü.....	51
7.2.3.7. Monitor kontrol.....	51
7.2.3.8. PC seçim kontrol.....	52
7.2.3.9. Turnike manüel kontrol.....	53

BÖLÜM 8.

SİSTEM ÇALIŞMA YAPISI.....	54
8.1. Sunucu Devresi Kullanımı.....	55
8.1.1. Tanım.....	55
8.1.2. Özellikleri.....	55
8.1.3. Sunucuda kullanılan grafik lcd yapısı.....	56
8.1.4. Sunucu modülü devre şeması.....	58
8.1.5. Buton ve led işlevleri.....	59
8.1.6. Grafik LCD ekranı.....	60
8.1.6.1. Ana kontrol ekranı.....	60
8.1.6.2. İstemci bilgi ekranı.....	61
8.1.7. Seri port istemci bilgileri kayıt kontrol.....	61

8.2. İstemci Devresi Kullanımı.....	63
8.2.1. Tanım.....	63
8.2.2. Özellikleri.....	64
8.2.3. İstemci modülü devre şeması.....	65
8.2.4. İstemci ekranı.....	66
8.2.5. Buton ve led işlevleri.....	67
8.2.6. Periyodik LAN kontrolü.....	67
BÖLÜM 9.	
SONUÇLAR	68
BÖLÜM 10.	
TARTIŞMA VE ÖNERİLER.....	71
KAYNAKLAR.....	73
ÖZGEÇMİŞ.....	74

SİMGELER VE KISALTMALAR LİSTESİ

MCU:	Mikrodenetleyici (Microcontroller Unit)
TCP:	Transfer Control Protocol
IP:	Internet Protocol
RTC:	Real Time Clock
LAN:	Local Area Network
MAC:	Media Access Control
CRC:	Cyclic Redundancy Check
DA:	Destination Address
SA:	Source Address

ŞEKİLLER LİSTESİ

Şekil 2.1.	Dr. Robert M. Metcalfe tarafından çizilen ilk Ethernet yapısı.....	4
Şekil 2.2.	Ethernet üzerinde veri yolu.....	7
Şekil 3.1.	Ethernet paket formatı.....	12
Şekil 4.1.	MCU - ENC28J60 temel bağlantı diyagramı.....	16
Şekil 4.2.	ENC28J60 blok diyagramı.....	17
Şekil 4.3.	ENC28J60 temel uygulama devresi.....	18
Şekil 5.1.	CAN bit zamanlaması.....	21
Şekil 5.2.	Temel paket formatı.....	24
Şekil 5.3.	Genişletilmiş paket Formatı.....	25
Şekil 5.4.	Uzak paketi yapısı.....	25
Şekil 5.5.	Hata paketi yapısı.....	26
Şekil 6.1.	Engelleme olasılığı - CAN üzerinde talep edilen yük.....	31
Şekil 6.2.	Ortalama gecikme - CAN üzerinde talep edilen yük.....	32
Şekil 6.3.	Ethernet ve CAN için mesaj gecikmelerinin CDF'leri.....	33
Şekil 6.4.	Ortalama Ethernet ve CAN gecikmelerinin karşılaştırılması.....	34
Şekil 6.5.	Ethernet ve CAN gecikmelerinin en kötü durum için karşılaştırılması.....	35
Şekil 8.1.	Kampus otomasyon sistemi blok yapısı.....	54
Şekil 8.2.	Sunucu modülü dış görünümü.....	55
Şekil 8.3.	KS0108 GLCD pin çıkışları.....	56
Şekil 8.4.	Sunucu modülü devre şeması.....	58
Şekil 8.5.	ENC28J60 Ethernet arabirimi devresi	59
Şekil 8.6.	Ana kontrol ekranı	60
Şekil 8.7.	İstemci bilgi ekranı	60
Şekil 8.8.	İstemci modülü dış görünümü.....	63
Şekil 8.9.	İstemci modülü devre şeması	65

Şekil 8.10. ENC28J60 Ethernet arabirimi devresi.....	66
Şekil 8.11. İstemci ekranı.....	66

TABLULAR LİSTESİ

Tablo 5.1.	Dominant/Recessive ve lojik AND için doğruluk tablosu.....	20
Tablo 5.2.	Temel paket formatı.....	23
Tablo 5.3.	Genişletilmiş paket formatı.....	24
Tablo 5.3.	Genişletilmiş paket formatı (Devam).....	25

ÖZET

Anahtar kelimeler: PIC, Ethernet, Internet

Bu çalışmada, internet tabanlı bir mikroişlemcili kontrol sistemi tasarlanmıştır. Çalışma, sistemin alt yapısının daha iyi anlaşılabilmesi için teorik bilgilerle desteklenmiştir.

Teorik bölümde CAN Bus ile Ethernet yapısı karşılaştırılarak mikroişlemcili sistemler için uygulama alanına göre avantajları incelenmiştir. Böylece tasarladığımız mikroişlemcili kontrol sistemi için Ethernet'in neden uygun olduğu gösterilmiş ve CAN Bus'ın hangi tip mikroişlemcili sistemler için daha uygun olduğunun anlaşılması sağlanmıştır.

Bugüne kadar gerçekleştirilen internet tabanlı kontrol sistemlerinde genel olarak bilgisayar tabanlı yapılar kullanılmış, gerekli donanım kontrolleri de bilgisayarın dışarı açılan seri port ya da paralel port gibi pencerelerinden yapılmıştır. Bu sistemlerde bilgisayar donanımının yanı sıra harici mikroişlemci tabanlı donanımlar da kullanılmıştır. Tüm bunların, özellikle bilgisayar kullanımının maliyeti, bir kontrol sistemi için gereğinden fazla olmuştur. Bu çalışmada bilgisayara gerek duymayan, Ethernet üzerinden haberleşen mikroişlemcili bir kontrol sistemi tasarlanmış ve gerçekleştirilmiştir.

Tasarlanan sistem, Ethernet üzerinden Sunucu/İstemci yapısı ile çalışan mikroişlemcili bir kontrol sistemi olarak tasarlanmıştır. İletişim TCP/IP protokolü üzerinden yapılarak gerekli bilgilerin kayıpsız bir şekilde karşılıklı iletilmesi sağlanmıştır. İstemci modül bu çalışmada Turnike, Monitör, PC, LAN gibi 4 farklı cihazı kontrol etmek üzere tasarlanmıştır. Tüm cihazların durumları sunucu üzerinden sorgulanabilmektedir. Sorgulamanın yanı sıra, sunucu üzerinden gönderilen komutlarla, istemci üzerinde çalışan Turnike, PC ve Monitor cihazlarının durumları değiştirilebilmektedir. Sistem 40 taneye kadar istemciyi kontrol edebilmektedir. Sistem iletişimi TCP/IP protokolü üzerinden gerçekleştirilerek, sistemin internet üzerinden kontrol yapılabilmesi sağlanmıştır.

CAMPUS AUTOMATION USING PIC MICROCONTROLLERS

SUMMARY

Key Words: PIC, Ethernet, Internet

In this project, a control system, which can communicate through internet, has been designed. The project is supported theoretically so that the basics of the system can be understood better.

In the theoretical part, the advantages of the bus structures are examined for microcontroller based systems by comparing Ethernet and CAN Bus structures. Thus, it is shown why Ethernet is the best for our microcontroller based system and it is explained which microcontroller based systems are better for using CAN Bus.

Generally, computer based structures are used in communication of internet based control systems. Serial port or parallel port is used as a window between control systems and computers. As known, a computer based control system costs much more than needed for a control system. In this project, a microcontroller based control system, which do not need a computer and communicates through Ethernet, has been developed.

The developed system is a microcontroller based control system which works through Ethernet in a Server/Client structure. Using Server module, querying can be done to get information of devices from a Client and it is possible to send a command to change the status of a device on a Client. In this project, a Client can control 4 different devices: Turnstile, Monitor, PC, LAN. It can be done to query the information for all devices over Server. This system supports 40 Clients to control. It is possible to control the system through internet because communication is made by TCP/IP in this development.

BÖLÜM 1. GİRİŞ

Tüm dünya üzerinde yaygın olarak kullanılan bir iletişim aracı olması, dolayısıyla her yerden istenilen bir hedefe ulaşım sağlaması ve uygun maliyeti, kontrol sistemlerinde internet kullanımının önemli bir noktaya gelmesini sağlamıştır.

Bugüne kadar gerçekleştirilen internet tabanlı kontrol sistemlerinde genel olarak bilgisayar tabanlı yapılar kullanılmış, gerekli donanım kontrolleri de bilgisayarın dışarı açılan seri port ya da paralel port gibi pencerelerinden yapılmıştır. Bu sistemlerde bilgisayar donanımının yanı sıra harici MCU tabanlı donanımlar da kullanılmak zorunlu olabilmektedir. Tüm bunların, özellikle bilgisayar kullanımının maliyeti, bir kontrol sistemi için gereğinden fazla olmaktadır.

Bu çalışmada bilgisayarsız olarak, Ethernet üzerinden bir sistem kontrolü sağlanmıştır. İletişimde veri kaybını engellemek için protokol olarak TCP/IP protokolü seçilmiştir.

Sistemde hem server(sunucu) hem de client(istemci) modülünde işlemci olarak PIC18F4620 ve Ethernet controller olarak ENC28J60 kullanılmıştır. 18F4620, büyük program hafızası, bacak sayısının sisteme uygunluğu ve 18F serisinin performansı ile sistemde kullanılmaya uygun görülmüştür.

Programlama yapılırken tüm yapı içerisinde “state machine” yapısına bağlı kalınmaya özen gösterilmiştir. Bu sayede kod içerisinde hiçbir zaman hiçbir işlemin bitmesinin beklenmesine gerek kalmadan, akışın öteki işleme geçirilmesi ile tüm yapının multitask(çok görevli) bir yapıya benzetilmesi sağlanmıştır. Programın bu yapısı sayesinde MCU'nun çalışmakta olduğu status led üzerinden gözlenebilmektedir.

Sistem genel mantık olarak İstemci \leftrightarrow Sunucu mantığı üzerine kurulmuştur. Bir sunucu 40 istemciye kadar modülü kontrol edebilir. Sunucu üzerine istemcilerin adları ve IP adresleri seri port üzerinden kaydedilebilmektedir. Yine gerekli değişiklikler seri port üzerinden yapılabilir. Bu kayıt 18F4620'nin dahili EEPROM'una yapılmaktadır.

Sunucu modül istemciler üzerinde bağlı cihazların durumlarını gözlemek ve gerektiğinde istemci üzerindeki cihazların durumunu TCP/IP üzerinden değiştirmek için kullanılır. Bir istemci kendisine bağlı 4 farklı cihazı kontrol edebilir ve gerekli bilgileri sunucuya TCP/IP protokolü üzerinden gönderebilir. İstemci modül içinde programda tasarlanan özel bir RTC yapısı ile saat gösterimi de bulunmaktadır. Bu saat yapısı sayesinde istemci modül kendi içinde periyodik kontroller de yapabilmektedir.

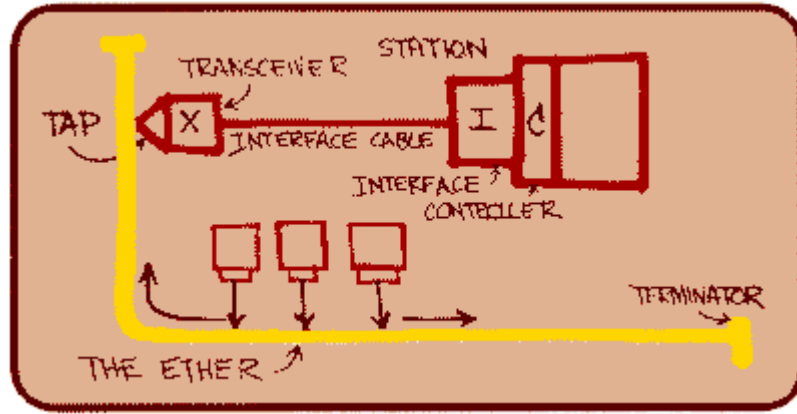
BÖLÜM 2. ETHERNET

Günümüzde bir çok LAN teknolojisinden söz edilse de, Ethernet açık ara farkla en yaygın LAN teknolojisidir. Ethernet ilk ortaya çıkışından itibaren teknolojisi ve üretim haklarıyla herkese açıktır. Kullandığı teknolojinin üretimi kolaydır ve ucuza mal edilebilir. Aynı zamanda güvenilir olduğu ve kullanıcıların ihtiyaçlarını karşıladığı için en yaygın yerel ağ teknolojisi haline gelmiştir. En yaygın teknoloji olması ethernetin üreticiler için büyük bir pazar haline gelmesine ve sürekli geliştirilmesine yol açmaktadır.

2.1. Ethernetin Tarihi

Ethernet Xerox firmasının Palo Alto araştırma merkezinde 1970'li yıllarda Dr. Robert M. Metcalfe tarafından geliştirildi. Metcalfe "geleceğin ofisi" projesi üzerinde çalışıyordu ve elinin altında dünyanın ilk workstation bilgisayarlarından biri olan Xerox Alto bilgisayarlar bulunuyordu.

1972 yılının sonlarında, Metcalfe ve Xerox'ta çalışan iş arkadaşları Xerox Alto'ları birbirine bağlamak için deneysel olarak Ethernet'i geliştirdiler. Böylece Alto bilgisayarlar diğer sunucular ve lazer yazıcılar birbiriyle haberleşebiliyordu. İlk Ethernetin çalışma hızı Alto'larla uyumlu olması için Alto'nun çalışma hızı ile aynı tutulmuş ve sonuçta ağ 2.94 Mega Bit/Saniye hızında çalışmıştır. İlk ethernet tek parça bir koaksiyel kablo kullanıyordu.



Şekil 2.1. Dr. Robert M. Metcalfe tarafından çizilen ilk Ethernet yapısı

Bu diyagram, Dr. Robert M. Metcalfe tarafından 1976 yılının haziran ayında National Computer Conferance'da ethernetin doğuşu sırasında çizildi.

Ethernetin doğuşundan beri bu diyagramdaki temellere dayanan kullanım süregeldi.

Metcalfe önce "Alto Aloha Network" olan sistemin ismini 1973 yılında "Ethernet" olarak değiştirdi. Böylece sistemin sadece Alto bilgisayarlarda değil tüm bilgisayarlarda çalışabileceğini vurgulamak istiyordu. Ethernet kelimesi bir zamanlar tüm uzayı doldurduğuna ve elektromanyetik sinyallerin aktarımını sağladığına inanılan "ether" den geliyordu. Metcalfe'nin sisteminde de veri bitleri tüm sistemlere ulaştığı için sonuçta "Ethernet" doğmuş oldu.

1979 yılına kadar sadece Xerox içinde kullanılan Ethernet'in resmi duyurusu 1980 yılında yapıldı. Xerox, DEC(Digital Equipment Corporation) ve Intel firmaları ile beraber, sonradan "DIX Standart" olarak anılan Ethernet standardını yayınladı. DIX standardı koaksiyel kablo üzerinden 10 MBs hızında çalışan Ethernet'i tanımlamıştır. Böylece Ethernet, firma içi deneysel bir çalışmadan herkese açık gerçek bir ürün haline gelmiş oldu.

2.2. Ethernet veya IEEE 802.3

DIX standardından sonra Ethernet, Institute of Electrical and Electronics Engineers (IEEE)'in 802 kodlu komisyonu tarafından geliştirilmeye devam etti. IEEE 1985 yılında "IEEE 802.3 Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications" şeklinde bir isimle yeni Ethernet standardını yayınladı. İzleyen dönemde IEEE standardı International Organization for Standardization (ISO) tarafından yürütülmeye devam etti. ISO günümüzde bilgisayar ağları ile ilgili tüm standartları yürüten kuruluştur.

1985 yılından itibaren üretilen tüm ürünler IEEE 802.3 standardına göre üretilmektedir. Aslında bu ürünleri "IEEE 802.3 CSMA/CD" standardını kullanan ürünler olarak tanımlamak daha doğrudur. Ama dünya çapında hala genel olarak "Ethernet" kelimesi tüm bu ürünler ve dahil oldukları teknolojiyi tanımlamak için kullanılmaktadır.

Ethernet tek bir ağ teknolojisi olmaktan çok, aynı bus topolojisini, frame yapısını ve network access(ağ erişimi) metodunu kullanan ağ teknolojileri ailesini tanımlar.

2.3. Ethernetin Çalışma Şekli

Ethernet'i geliştiren ekip üç ana problemi çözmek zorundaydı:

1. Kablo üzerinden veri nasıl gönderilecek
2. Gönderen ve alıcı bilgisayarlar nasıl tespit edilecek
3. Belli bir anda kabloyu kimin kullanılacağına nasıl karar verilecek

2.3.1. Verinin aktarımı: Paketler(Frames)

Tüm bilgisayar ağları ağ üzerinden aktarılacak veriyi sabit boyutta küçük paketler halinde iletirler. Bu yöntemin iki önemli faydası vardır. Birincisi büyük bir dosya transferi yapan bir bilgisayar ağın tamamını uzun bir süre meşgul durumda tutmamış olur. Bir sistem veriyi paketler halinde yollarken, her paketi göndermeden önce kablonun kullanımda olup olmadığını kontrol ettikten sonra paketi yollar. Paket

karşıya ulaştığında, kablo tekrar ağdaki tüm makineler için boş duruma gelmiş olur. Az önceki makine ikinci paketi yollamadan önce tekrar kabloyu kontrol etmek zorundadır. Bu arada diğer bir sistem kendi paketini yollayabilir. Paketler küçük yapıda olduğu için saniyeler içinde yüzlercesi değişik bilgisayarlar tarafından yollanıp-alınabilir. Bilgisayarları kullanan insan için durum, ağda sanki herkes aynı anda veri alışverişi yapıyormuş gibidir. Veri paketler halinde gönderilmeseydi, bir kullanıcı 50 MB bir dosyayı başka bir bilgisayara yollarken belki 3–5 dakika boyunca diğer hiçbir sistem ağı kullanamayacaktı.

Paketli yapının ikinci faydası ise şudur: 50 MB'lık dosyanın bir biti bile aktarım esnasında bozulursa, bu tüm dosyanın en baştan tekrar gönderilmesi anlamına gelir. Oysa veri paketlere bölünüp yollandığında, sadece bozuk giden paketin tekrar yollanması kâfidir.

Ethernet veri paketinin yapısı sabittir. Her paket şu dört bilgiyi içerir:

1. Alıcının MAC adresi
2. Gönderenin MAC adresi
3. Gönderilecek veri'nin kendisi
4. CRC kodu

2.3.1.1. MAC adresi

Ethernet ağına dahil her cihaz ya da Ethernet arayüzüne sahip her cihaz "node" olarak adlandırılır. Bilgisayarlara Ethernet kartı takınca bir node haline gelirler, ancak Ethernet girişi olan başka cihazlar da olabildiği için(router'lar mesela) genel kavram node'dur.

Ethernet ağında sistemler birbirinden sahip oldukları MAC adresi ile ayırt edilirler. Her node veya basitçe her Ethernet kartı dünyada eşi olmayan bir adrese sahiptir. Bu adres 48 bitlik bir sayıdır. Örnek olarak bir ağ kartının MAC adresi şöyledir:

100100000110101001010010100011001101100000011

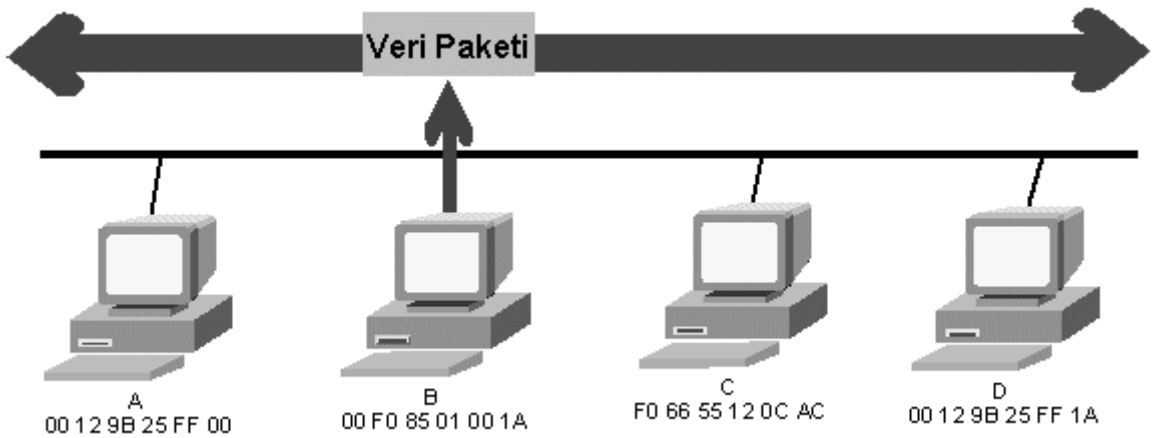
İkili sistemdeki bu sayıyı söylemek ve yazmak zor olduğu için bu sayı 16'lı sayı sisteminde yazılır: 12 0D 4A 51 9B 03

Her üretici firmanın IEEE'den aldığı 24bitlik bir üretici kodu vardır. Bu kod Organizationally Unique Identifier(OIU) olarak adlandırılır ve her üretici için farklıdır. Üretici firma, 24 bitlik OIU numarasından geri kalan 24 biti ise kartın seri numarası(Device ID-başka bir karta daha verilmeyecek-) olmak üzere MAC adresi olarak belirleyip, ağ kartının üzerinde programlanabilir bir entegreye bu numarayı yazar. Böylece bu kartın dünyada eşi olmayan bir MAC adresi olur.

Bunun sayesinde sizin almış olduğunuz her ağ kartı üreticisi, üretim tarihi, markası-modeli ne olursa olsun farklı bir MAC adresine sahip olacaktır. Ethernet sisteminde node'ları birbirinden ayırmak için bu MAC adresleri kullanılır.

MAC adreslerinin kullanımı:

MAC adresleri sayesinde sistemler ağ üzerinden kendilerine ulaşan veri paketinin kendilerine gelip gelmediğini anlarlar. Ethernet ağında, bir bilgisayar bir veri paketi yolladığında, bu paket ağdaki tüm sistemlere ulaşır. Her makine paketin ilk bölümü olan alıcı MAC adresini okur ve kendi MAC adresiyle kontrol eder. Eğer gelen paket kendine aitse işler, değilse göz ardı eder.



Şekil 2.2. Ethernet üzerinde veri yolu

Ethernet'in bu özelliđi ciddi bir güvenlik açığına yol açabilir. Packet Sniffer olarak adlandırılan programlar, bilgisayara gelen veri paketlerini MAC adresi ne olursa olsun alıp kullanmaya izin verirler. Bu tip programlar iyi niyetle kullanıldığında problemlerin çözümüne yarayabileceđi gibi, yerel ağınızda meraklı bir kullanıcının sizin aslında başka bir makineye göndermekte olduğunuz her dosyayı izlemesine neden olabilir.

2.3.1.2. Multicast ve Broadcast adresleri

Bir grup sistemin aynı veriyi alması isteniyorsa, bu gruba dahil olması istenen sistemlerde Ethernet arayüzü (bilgisayardaki ağ kartı mesela) belli bir multicast adresine yollanmış veriyi kendi MAC'ine gelen bir veriyi alır gibi alması için ayarlanabilir. Yani multicast adresler belli bir grup cihazın aynı veriyi almasını sağlar.

Broadcast adresi ise 48 biti de 1 olan özel bir adrestir. Bu adrese yollanmış bir veri paketini alan her ağ kartı bu paketi kabul eder ve işleme koyar. Bazen tüm bilgisayarlara gitmesi gereken bir mesaj göndermek gerekebilir. Bu durumda mesajı içeren veri paketleri broadcast adresine yollanır böylece ağa dahil tüm cihazlar bu mesajı alır.

2.3.1.3. CRC hata denetimi

Cyclic Redundancy Check veri paketlerinin elektrik sinyali olarak kablodan geçerken bozulmaları durumunda bu bozulmanın yani veri paketinin karşıya yolda deđişmiş olarak ulaştığının tespitine yarar.

Gönderen taraf, veri paketine konacak veriyi matematiksel bir işlemde geçirir. İşlemin sonucu CRC kodudur. Veri ve CRC kodu karşı tarafa yollanır. Alıcı paketi açar, veriyi okur, aynı matematiksel işlem veriye uygulanır. Sonuç eđer veri yolda bozulmadan, yani bir bit'i bile deđişmeden gelmişse, CRC kodu ile aynı olmalıdır. Aksi halde alıcı gönderen makineye ilgili paketi tekrar yollamasını söyler.

Bu noktaya kadar en başta karşımıza çıkan iki problem yani verinin nasıl gönderileceği(paketler halinde) ve ağa dahil sistemlerin nasıl birbirinden nasıl ayırt edileceği(MAC adresi ile) anlatıldı. Bu noktadan sonra baksa bir önemli konu, kimin veri yollamak için kabloyu nasıl kullanacağıdır.

2.3.1.4. Kabloyu kim kullanacak (CSMA/CD)

Ethernet'te trafik akışını kontrol etmek için “Carrier Sense”, “Multiple Access” ve “Collision Detection” teknikleri kullanır. Carrier Sense, Multiple Access, Collision Detection veya kısaca CSMA/CD'ye göre, Ethernet kartı veri gönderimine başlamadan önce kablonun kullanımda olup olmadığını kontrol eder. Eğer o anda diğer bir sistem kablodan veri aktarıyorsa buna Carrier denir. Kabloda aktarım olup olmadığını tespit Carrier Sense'dir.

Kablo boşta olduğunda her Ethernet arayüzüne sahip cihaz eşit hakka sahiptir ve veri aktarımına başlayabilir. Buna Multiple Access denir. Bir Ethernet ağında bilgisayar üzerinde çalışan işletim sistemi veya kullanıcısı önemli değildir. Bir DOS makinesi Ethernet'in kabloyu kullanma şansı açısından W2000 server ile aynıdır.

Bazı durumlarda iki sistem kablonun boş olduğunu tespit ederek aynı anda veri aktarımına başlayabilir. Bu durumda iki tarafın yolladığı veri çakışır(Collision). Ethernet kartları çakışmayı hemen tespit ederler(Collision Detection).

Collision durumları:

Collision veya çakışma kelimesi insanda olumsuz bir etki uyandırır da, bir ethernet ağında çakışmaların oluşması gayet normaldir.

Eğer birden fazla Ethernet kartı aynı anda veri iletimine geçerse çakışma oluşur. Sistemler kendi yolladıklarıyla kablodan geleni karşılaştırarak bunu hemen tespit ederler. Bunun akabinde her iki taraf da özel bir algoritma ile belirlenen rasgele bir süre boyunca beklerler.

Çakışmaların oluşması ethernetin doğasında olan bir şeydir ve her Ethernet ağında çakışma olması kaçınılmazdır.

Eğer ağ limitlerin dışında kullanılmıyorsa(ağa dahil sistem sayısı, kullanılan kablo uzunlukları, veri aktarım yükü vs.) çakışmalar saniyenin milyonda biri gibi sürelerde giderilir. Yani çakışmanın ardından birkaç mikrosaniye bekleyen sistem veriyi yollamaya tekrar başlar.

Eğer ağ çok yoğun kullanılıyorsa, aynı frame/veri paketi gönderilirken birden fazla çakışma olabilir. Bu durumda sistemler rasgele belirlenen bekleme süresini uzatmaya başlarlar. Burada süre rasgele belirleniyorsa nasıl daha uzun veya kısa olabilir diye bir soru akla gelebilir. Sürenin rasgele olması her iki tarafında aynı süre bekleyip, sonra da yine aynı anda aktarım yapmalarının önüne geçmek için rasgeledir. Örneğin her iki tarafta birden ona kadar bir sayı tutar ve o kadar milisaniye bekler. Ancak süre belirlenirken, aynı paketin gönderiminde üst üste çakışma oluyorsa(ağda yoğun trafik varsa) süre 1-10 arası değil belki 50-100 arasında seçilir.

Ethernetin bu yapısı ağdaki trafik yoğunluğu arttıkça kendisini duruma uydurmasını sağlar. Ethernet aynı veri paketini 16 denemeden sonra hala gönderemediyse bu paketi iptal eder. Bu ancak çok uzun bir süre çok aşırı yoğunluk yaşanması durumunda veya kabloda meydana gelen bir arıza nedeniyle olabilir.

Bu noktada ethernetin diğer ağ teknolojilerinde de olduğu gibi veri aktarımını %100 garanti etmediğini görüyoruz. Bu açık üst katman protokollerinin sağladığı veri kontrolü ile telafi edilir. Bir paket yolda kaybolursa veya 16 denemede de yollanamayıp iptal edilirse, alıcı taraftaki üst katman protokol (TCP/IP kullanılıyorsa; TCP) gönderen taraftaki TCP'ye gelen veride bir eksiklik olduğunu bildirecek ve tekrar yollanmasını isteyecektir.

Ethernetin kullandığı CSMA/CD tekniğinin basit yapısı Ethernet ağ kartlarının ve diğer ekipmanların rakip teknolojilere (Token Ring) göre daha ucuza üretilmesini sağlar. Böylece Ethernet ağları çok daha ucuza mal olur.

Ethernet hızı:

CSMA/CD tekniđi nedeniyle Ethernet veri aktarımı yapabileceđi belli bir süreyi çakışmalarla uğraşırken harcar. 90 kullanıcılı bir Ethernet ađının olduđu firmada pazartesi sabahı 9:00'da herkes aynı anda oturup makinelerini açıp, şifrelerini girip, gün boyunca kullandıkları programa girmeye çalıştığı anda ađda çok büyük miktarda çakışma oluşur. Kullanıcılar açısından sanki herkes aynı anda ađı kullanıyor gibidir ama aslında CSMA/CD çalışmaktadır.

Her Ethernet ađı belli bir süreyi çakışmalarla ve broadcast mesajlarıyla harcar. Dolayısıyla hiçbir Ethernet söylendiđi gibi 10Mbs veya 100Mbs'de çalışmaz. Daha doğrusu sizin birim zamanda aktardığımız veri miktarı bu değerlere hiçbir zaman ulaşamaz çünkü ađdaki bu veri aktarım kapasitesinin bir bölümü collision ve broadcast mesajları ile harcanmaktadır.

Ethernetin kullandığı basit iletişim yapısı nedeniyle performans kaybı kaçınılmazdır. Ancak bu basit yapı ucuz üretim maliyetleri anlamına gelir. Sonuç itibariyle getirisi-götürüsü karşılaştırıldığında Ethernet yine de en uygun çözüm durumundadır.

BÖLÜM 3. ENC28J60 VE ETHERNET YAPISI

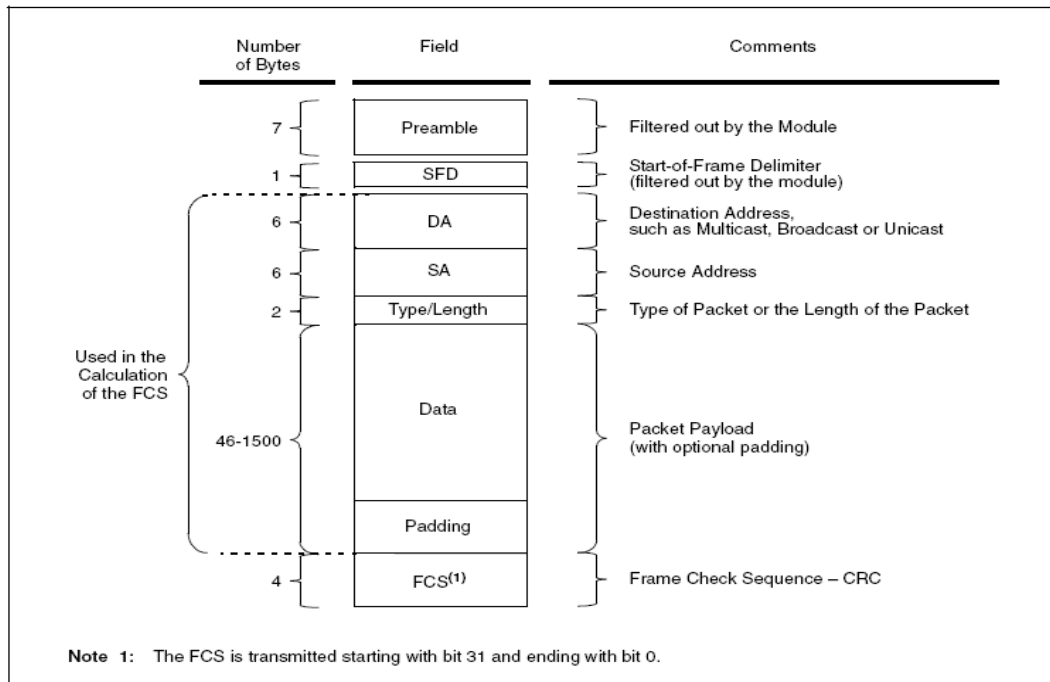
3.1. Paket Formatı

Normal IEEE 802.3 tabanlı Ethernet paketleri 64 ve 1518 byte uzunluğundadır.

Paketler 5 ya da 6 farklı alandan oluşur:

- Hedef MAC adresi (DA)
- Kaynak MAC adresi (SA)
- Alan tipi/uzunluğu (Type/Length)
- Data
- Seçimsel bir destek alanı (padding)
- CRC (Cyclic Redundancy Check)

Ek olarak, 7 byte başlangıç alanı (preamble field) ve paket başlangıcı limitleyicisi (start-of-frame delimiter) Ethernet paketinin başına eklenir. Böylece, bükülmüş çiftli kablodaki trafik Şekil 3.1.'deki gibi görünür.



Şekil 3.1. Ethernet Paket Formatı

3.2. Preamble/Start-Of-Frame Delimiter

ENC28J60 kullanarak bilgi gönderilip alınırken, preamble ve start of frame delimiter byteleri otomatik olarak oluşturulacak ya da paketlerden çıkarılacaktır. İşlemcinin bu işle ilgilenmesine gerek yoktur. Normal olarak, işlemcinin padding ve CRC işlemleri ile de ilgilenmesine gerek yoktur. ENC28J60 bu işlemleri otomatik olarak yapabilir. Fakat padding ve CRC alanları paketler alındığında receive tamponuna yazılacaktır. Dolayısıyla, işlemci istenirse bunları değerlendirebilir.

3.3. Hedef Adres (Destination Address)

Hedef adresi 6 byte uzunluğundadır ve paket gönderilecek cihazın MAC adresini içerir. Eğer MAC adresinin ilk byteının en düşük değerlikli biti "1" ise adres bir multicast hedeftir. Örnek olarak, 01-00-00-00-F0-00 ve 33-45-67-89-AB-CD birer multicast adrestir, 00-00-00-00-F0-00 ve 32-45-67-89-AB-CD değildir.

Multicast hedef içeren paketler Ethernet noktalarındaki belirli bir gruba ulaşması için tasarlanmıştır. Eğer hedef adresin tüm bitleri "1" ise (FF-FF-FF-FF-FF-FF), paket bir broadcast paketidir ve ağdaki herkese gönderilir. Eğer MAC adresinin ilk byteının en düşük değerlikli biti "0" ise, adres unicast adrestir ve sadece bir adres noktası tarafından işleme alınır.

ENC28J60 veri alımı sırasında alıcı filtrelerini çalıştırır ve multicast, broadcast ve/veya unicast adreslerinin kabul ya da ret işlemlerini kendisi yapar. Fakat veri paketi gönderirken hedef adresinin gönderim tamponuna yazılmasından işlemci sorumludur.

3.4. Kaynak Adres (Source Address)

Kaynak adresi 6 byte uzunluğundadır ve Ethernet paketini oluşturan cihazın MAC adresini içerir. ENC28J60 kullanıcıları kullandıkları her kontrolör için kendi unique(eşi olmayan) MAC adreslerini oluşturmalıdırlar. MAC adresi 2 parça içerir.

İlk 3 byte'ı Organizationally Unique Identifier (OUI)'dır. OIUlar IEEE tarafından dağıtılır. Son 3 byte OIU'yu satın alan firmanın isteğine bağlı olarak değiştirilebilir.

Paket gönderim işlemi yapılırken, tahsis edilmiş olan kaynak MAC adresi gönderim tamponuna işlemci tarafından yazılmalıdır. ENC28J60, unicast adres filtresi tarafından kullanılan MAADR registerının içeriğini otomatik olarak gönderim işlemine dahil etmez.

3.5. Tip/Uzunluk (Type/Length)

Tip/Uzunluk alanı 2 byte uzunluğundadır ve paket içeriğinin hangi protokole ait olduğunu belirtir. Duruma göre, eğer bu alanda 05DCh(1500) bilgisi ya da daha küçük bir bilgi var ise bu alan bir uzunluk alanı olarak dikkate alınır ve data alanındaki bilginin padding içermeyen uzunluğunu belirtir. Kendine özel ağ tasarımı yapmak isteyen kullanıcılar belki bu alanı uzunluk alanı olarak seçecektir, fakat paket gönderiminde Internet Protocol (IP) ya da Address Resolution Protocol (ARP) gibi protokolleri kullanmak isteyenler bu alanı uygun olan tip belirleyici ile doldurmalıdırlar.

3.6. Data

Data alanı 0 ile 1500 byte arasında değişebilir. Daha uzun data paketleri Ethernet standartlarını ihlal eder ve Ethernet noktaları tarafından iptal edilir. Fakat ENC28J60 daha uzun paketleri gönderebilme ve alabilme özelliğine sahiptir. Bunun için Huge Frame Enable biti "1" yapılmalıdır (MACON3.HFRMEN = 1).

3.7. Padding

Padding alanı değişken uzunlukta bir alandır ve küçük veri kullanıldığında IEEE 802.3 gereksinimlerine uyum sağlamak için kullanılır. Ethernet paketinin hedef, kaynak, tip, data ve padding bilgileri 60 bytetan küçük olmamalıdır. 4 byte CRC alanını eklendiğinde bu uzunluk 64 bytetan küçük olmamalıdır. Eğer data alanı 46 bytetan daha az ise padding alanı gereklidir.

Paket gönderimi yaparken, ENC28J60, eğer MACON3.PADCFG<2:0> bitleri ayarlanmışsa otomatik olarak sıfır padding oluşturur. Diğer türlü, gönderim yapılmadan önce işlemci gerekli padding bilgilerini pakete eklemelidir. ENC28J60 düşük boyutlu paketlerin iletimini engellemez bunu işlemci yapmalıdır.

Paket alımı yaparken, ENC28J60 18 bytetan küçük paketleri otomatik olarak ret edecektir. Bu kadar küçük bir paketin minimum gereksinimler olan hedef ve kaynak adreslerini, tip bilgisini ve FCS checksumını içermediği varsayılır. 18 bytetan büyük tüm paketler standart filtreleme kriterlerine uyabilir ve normal trafik olarak algılanabilir. IEEE 802.3 gereksinimlerine uymak için alınan bütün paketler 64 bytetan küçük ise uygulama içinde ret edilmelidir.

3.8. CRC

CRC alanı 4 byte uzunluğundadır ve hedef, kaynak adresleri, tip, data ve padding alanlarından hesaplanır.

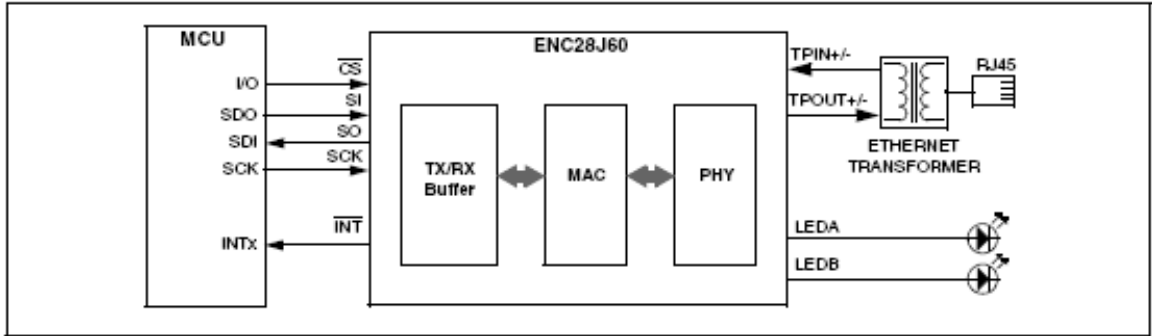
Paket alımı yapılırken, ENC28J60 her gelen paketin CRC'sini kontrol edecektir. Eğer ERXFCON.CRCEN biti "1" ise, yanlış CRC'li paketler otomatik olarak iptal edilecektir. Eğer CRCEN biti "0" ise ve paket diğer bütün alım kriterlerine uyuyorsa, paket alım tamponuna yazılır ve işlemci CRC'nin geçerli olup olmadığını alım durum vektörünü kontrol ederek anlayabilir.

Paket gönderimi yapılırken, ENC28J60 eğer MACON3.PADCFG<2:0> bitleri ayarlı ise, otomatik olarak geçerli bir CRC oluşturur ve gönderir. Diğer türlü, işlemci geçerli CRC'yi hesaplamalı ve gönderim tamponun eklemelidir. CRC'nin hesaplaması işleminin ENC28J60'a bırakılması tavsiye edilir.

BÖLÜM 4. ENC28J60 HAKKINDA BİLGİ

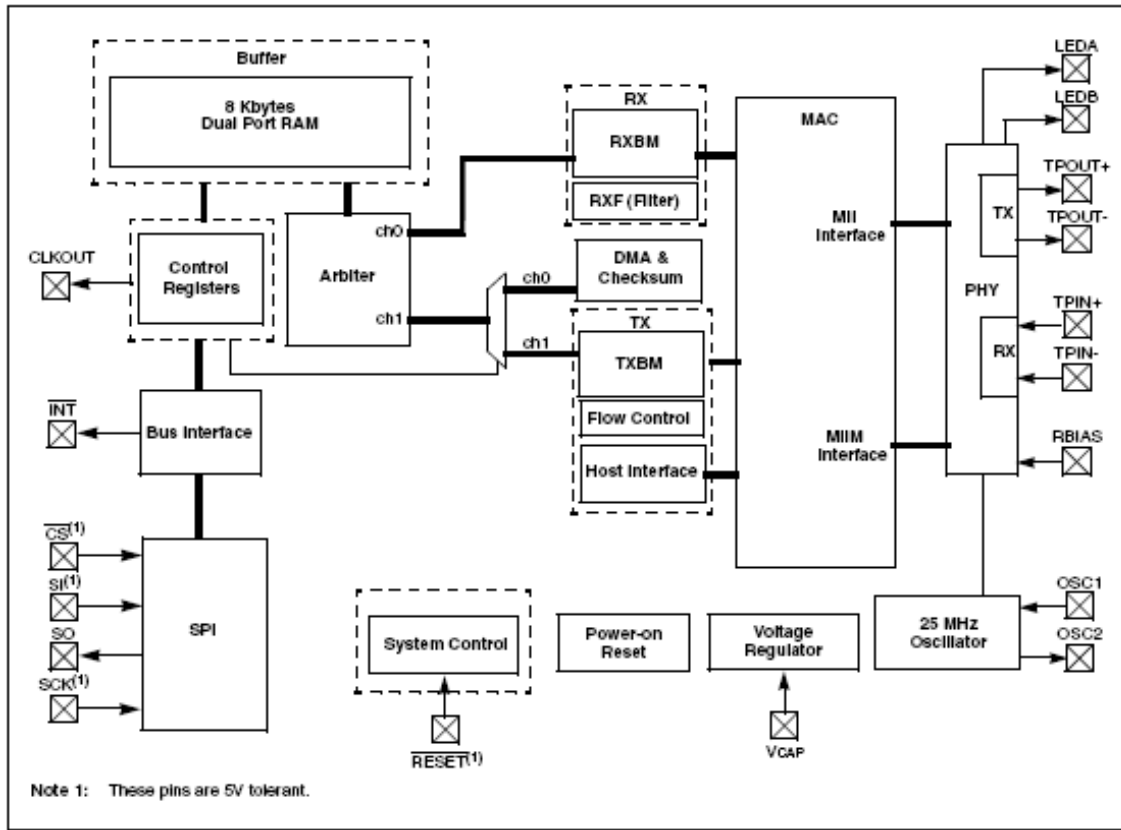
ENC28J60, endüstriyel standart olan SPI arabirimine sahip bir Ethernet kontrolörüdür. SPI arabirimine sahip herhangi bir işlemci için Ethernet ağı arabirimi olarak tasarlanmıştır. ENC28J60 tüm IEEE 802.3 standartlarına uyar. Gelen paketler için filtreleme yapısı içerir. Bunun yanında, hızlı data işlemek için DMA modülü ve donanımsal checksum hesaplayıcı içerir. İşlemci (MCU) ile iletişimi bir kesme pini ve 20MHz'e kadar hızı destekleyen SPI arabirimi ile gerçekleştirir. ENC28J60 üzerinde 2 tane pin, LED olarak link ve ağ aktivitesini göstermek üzere tanımlanmıştır.

ENC28J60'ın temel blok diyagramı Şekil 4.2.'de gösterilmiştir. Fikir vermesi amacıyla temel bir uygulama devresi Şekil 4.1. ve 4.3.'de verilmiştir.



Şekil 4.1. MCU - ENC28J60 Temel Bağlantı Diyagramı

4.1. ENC28J60 Blok Yapısı



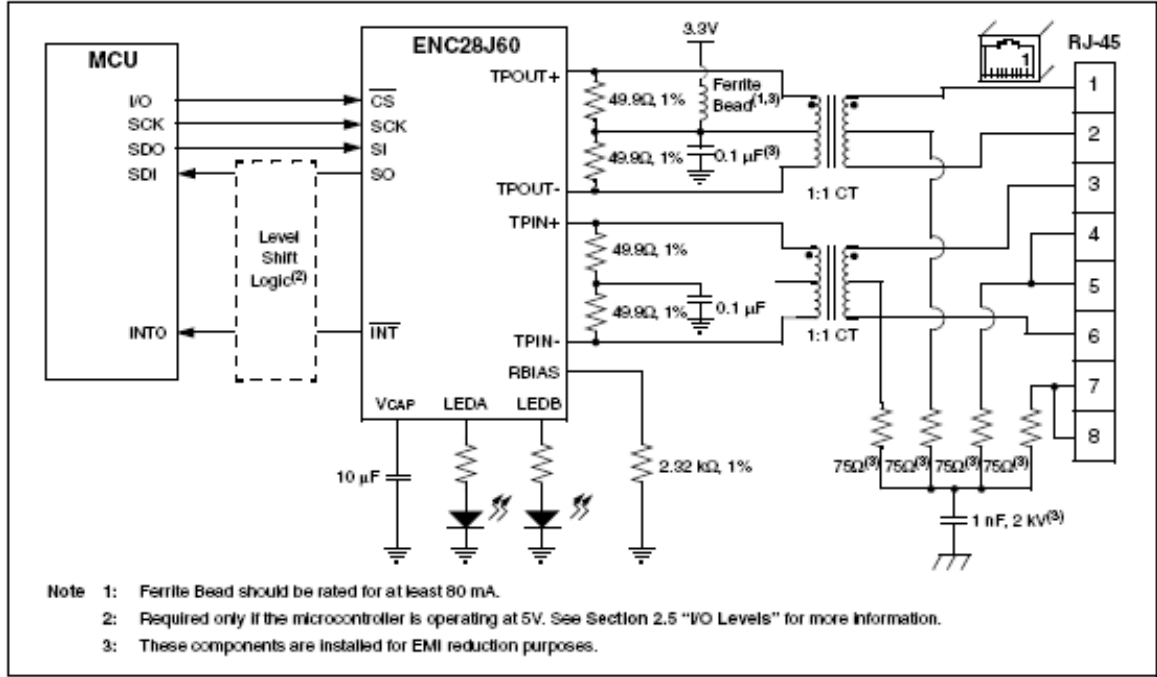
Şekil 4.2. ENC28J60 Blok Diyagramı

ENC28J60, yedi ana işlevsel bloktan oluşur:

1. İşlemci ile ENC28J60 arasında iletişim kanalı olarak çalışan SPI arabirimi.
2. ENC28J60'ı kontrol etmeye ve izlemeye yarayan kontrol registerları.
3. Alınan ve gönderilen data paketleri için çiftli RAM tamponu.
4. İstekler DMA, gönderim ve alım bloklarından geldiğinde RAM ulaşım yetkisini kontrol eden bir hakem.
5. SPI arabiriminden gelen dataları ve komutları yorumlayan veri arabirimi.
6. IEEE 802.3 uyumlu MAC yapısı ile çalışan MAC (Medium Access Control) modülü.
7. Bükülmüş çiftli kablodaki analog verileri işleyen PHY (Physical Layer) modülü.

Bunların yanında ENC28J60, oscillator, dahili voltaj düzenleyicisi, 5V toleranslı giriş/çıkış ve sistem kontrol yapısını sağlamak için seviye çeviricileri gibi diğer destekleyici blokları da içerir.

4.2. ENC28J60 Temel Uygulama Devresi



Şekil 4.3. ENC28J60 Temel Uygulama Devresi

Şekil 4.3.'ten görüldüğü gibi ENC28J60 ile Ethernet iletişimi sağlamak için iki tane darbe transformatörü ve birkaç pasif eleman yeterlidir.

BÖLÜM 5. CONTROLLER AREA NETWORK (CAN)

Controller Area Network (CAN) bir broadcast, farksal veri yolu standardıdır. İlk olarak Robert Bosch GmbH tarafından 1980'lerde, elektronik kontrol ünitelerini(ECU) haberleştirmek için geliştirilmiştir. CAN özellikle elektromagnetik alan gürültüsü yayan çevresel donanımlarla çalışacak şekilde tasarlanmıştır ve RS-485 gibi farksal dengelenmiş hatlarda çalışabilir. CAN, eğer çiftli bükülmüş kablo kullanılırsa, gürültüye iki kat daha fazla dayanabilir. Başlangıçta otomotiv sektörü için üretilmesine rağmen, bugünlerde birçok gürültüye maruz kalabilecek embedded sistemlerde kullanılmaktadır. CAN veri yoluyla gönderilen mesajlar küçüktür (maksimum 8 byte) fakat CRC-15 ile korunur.

40m'nin altında uzunluğu olan ağlarda hız 1Mbit/s'ye kadar çıkabilir. Daha uzun ağlar için hız düşürülebilir(örn. 125kbit/s @ 500m).

CAN veri yolu standardı ISO11898-1'dir. Bu standart ana olarak, (Logical Link Control (LLC) ve Media Access Control (MAC) alt katmanından oluşan)veri yolu katmanından ve OSI referans modelinin physical katmanı hakkında bazı bakış açılarından bahseder. Diğer tüm protokol katmanları ağ tasarımcısının tercihine bırakılmıştır.

5.1. Veri İletimi

CAN otomatik "hakemli" iletim sistemine sahiptir. Yüksek öncelikte gönderilen bir CAN mesajı hakemliği sonucunda galip olacaktır ve düşük öncelikli mesajı gönderen nokta bunu anlayıp geri çekilecek ve bekleyecektir.

Bu "dominant (baskın)" ve "recessive(bastırılmış)" yapıya sahip binary bir model üzerinden CAN veri gönderimi ile elde edilir. Burada dominant lojik "0", recessive

lojik “1”dir. Eğer bir noktadan dominant bir bit gönderilirse ve diğer nokta recessive bit gönderirse, dominant bit veri yolunu kazanır(Lojik AND).

Tablo 5.1. Dominant/Recessive ve Lojik AND için doğruluk tablosu

İki nokta arasındaki veri yolu durumu			Lojik AND		
	dominant	recessive		0	1
dominant	dominant	dominant	0	0	0
recessive	dominant	recessive	1	0	1

Dolayısıyla, eğer recessive bir bit gönderiyorsanız ve başka biri dominant bit gönderirse, dominant bir bit görürsünüz ve bir kaçışma olduğunu anlarsınız. Bu sistemin çalışma şekli şöyledir: Yollar arasında bir gerilim oluşturularak bir dominant bir bit hatta sokulur (recessive bit hat içinde bastırılınca kadar). Eğer herhangi biri gerilim değişimi yaparsa, herkes bunu görür, bu sebeple dominanttır. Böylece yüksek öncelikli mesajlarda gecikme olmaz ve düşük öncelikli mesajları gönderen nokta otomatik olarak, dominant mesaj bittikten sonra 6 bit clock’u yeniden gönderir.

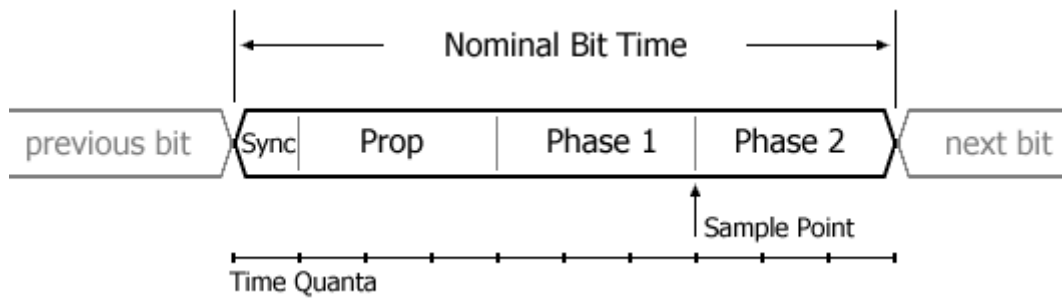
Farksal veri yolu ile kullanıldığında, bir Carrier Sense Multiple Access/Bitwise Arbitration (CSMA/BA) planı sıklıkla kullanılır. Eğer iki ya da daha fazla cihaz aynı anda iletme başlarsa, birinin iletimine izin veren öncelik temelli bir hakemlik planı uygulanır. Bu öncelikli hakemlik sistemine sahip CAN yapısı, CAN’i gerçek zamanlı önceliklendirilmiş sistemler için çok kullanışlı yapar.

Hakemlik müddetince, her gönderici nokta veri yolu durumunu gözler ve alınan bit ile gönderilen biti karşılaştırır. Eğer bir recessive bit gönderildiğinde bir dominant bit alınırsa, iletimi yapan nokta gönderimi keser. Hakemlik, identifier(tanımlayıcı) alanının gönderimi süresince yapılır. Aynı zamanda gönderime başlayan bütün noktalar dominant bit ile bir ID gönderir. ID’leri daha büyük bir numara(düşük öncelikli) olur olmaz, recessive karakter gönderirler ve dominant karakter görürler

böylece geri çekilirler. ID gönderiminin sonunda, geri çekilen hariç bütün noktalar ve en yüksek öncelikli mesaj engellenmemiş olur.

5.2. Bit Zamanlaması

CAN ağındaki her nokta kendi clock'una sahiptir ve veri iletimi sırasında clock gönderilmez. Senkronizasyon, paket içindeki bitlerin segmentlere bölünmesiyle yapılır: Synchronisation, Propagation, Phase 1 ve Phase 2. Her segmentin uzunluğu ayarlanabilir.



Şekil 5.1. CAN Bit Zamanlaması

5.3. Katmanlar (Layers)

Temel mantık olarak, CAN protokolünün yapısı şu bölümlerle açıklanabilir:

- 1) Uygulama Katmanı (Application Layer)
- 2) Nesne Katmanı (Object Layer)
 - a) Mesaj Filtreleme
 - b) Mesaj ve Durum Kontrolü
- 3) İletim Katmanı

Transfer katmanı CAN protokolünün temelini temsil eder. Nesne katmanına alınan mesajları gösterir ve nesne katmanından gönderilen mesajları kabul eder. Transfer katmanı, bit zamanlama, senkronizasyon, mesaj paketleme, hakemlik, onaylama, hata algılama, sinyal işleme ve hata engellemeden sorumludur. Şunları gerçekleştirir:

- a) Hata Engelleme
- b) Hata Algılama

- c) Mesaj Doğrulama
 - d) Onaylama
 - e) Hakemlik
 - f) Mesaj Paketleme
 - g) İletim Hızı ve Zamanlaması
 - h) Bilgi Gönderimi
- 4) Fiziksel Katman

Fiziksel katman gerçekte sinyallerin nasıl gönderildiğini tanımlar. Görevleri şunlardır:

- a) Sinyal Seviyesi ve Bit Gösterimi
- b) İletim Ortamı

5.4. Paketler

Bir CAN ağı iki farklı mesaj formatı ile çalışacak şekilde ayarlanabilir: standart ya da temel paket formatı (CAN 2.0 A) ve genişletilmiş paket formatı (CAN 2.0 B). İki format arasındaki tek fark, “CAN temel paket” formatı 11bit uzunluğunda tanımlayıcı (identifier) destekler, “CAN genişletilmiş paket” formatı 29bit uzunluğunda tanımlayıcı destekler. Genişletilmiş paket, 11bit tanımlayıcı (temel tanımlayıcı) ve 18bit uzantıdan (tanımlayıcı uzantısı) oluşur. CAN temel paket formatı ile CAN genişletilmiş paket formatı arasındaki ayırım IDE biti ile yapılır. IDE biti 11bit’lik temel paket içinde dominant olarak gönderilir. 29bit’lik paket içinde recessive olarak gönderilir. Genişletilmiş paket formatını destekleyen CAN denetleyicileri, aynı zamanda CAN temel paket formatında da iletişim yapabilir. Tüm paketler kesin olarak paket başlangıcını gösteren bir paket başlatıcı (start-of-frame: SOF) biti ile başlar.

CAN dört paket tipine sahiptir:

1. Data Frame: İletim için bir noktaya ait data taşıyan paket
2. Remote Frame: Belirli bir tanımlayıcı (identifier) için iletişim talep eden paket
3. Error Frame: Hata bulan herhangi bir nokta tarafından gönderilen paket
4. Overload Frame: Data ve/veya uzak paketler arasına gecikme soğan paket

5.4.1. Data paketi

Data paketi gerçek data iletimi için tek pakettir. İki mesaj formatı vardır:

1. Temel Paket Formatı: 11bit tanımlayıcı
2. Genişletilmiş Paket Formatı: 29bit tanımlayıcı

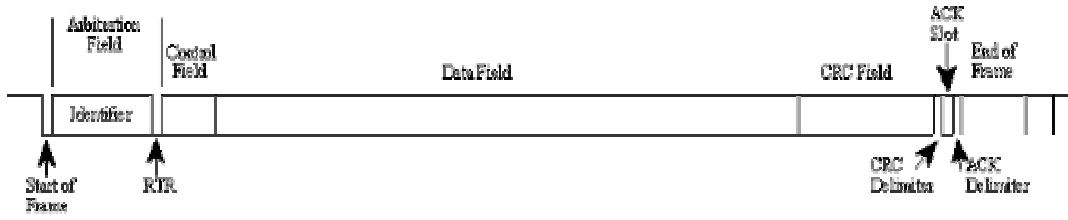
CAN standardı için bir tasarım temel paket formatını kabul etmelidir ve genişletilmiş paket formatını kabul edebilir düzeyde olmalıdır. Genişletilmiş paket formatına toleransı olması şarttır.

5.4.1.1. Temel paket formatı

Paket formatı yapısı aşağıdaki gibidir:

Tablo 5.2. Temel Paket Formatı

Alan Adı	Uzunluk(bit)	Amacı
Start-of-frame	1	Paket başlangıcını gösterir
Identifier (Tanımlayıcı)	11	Veri için eşsiz(unique) bir tanımlayıcı
Remote transmission request (RTR)	1	Dominant olmalıdır (0) Seçimsel
Identifier extension bit (IDE)	1	Dominant olmalıdır (0) Seçimsel
Reserved bit (r0)	1	Ayrılmış bit (Dominant yapılmalıdır(0), fakat dominant ya da recessive olarak alınır)
Data length code (DLC)	4	Verinin byte sayısı (0-8 bytes)
Data field	0-8 byte	Gönderilecek veri (uzunluğu DLC alanında belirlenir)
CRC	15	Cyclic redundancy check
CRC delimiter	1	Recessive olmalı (1)
ACK slot	1	Gönderici recessive gönderir (1) ve herhangi bir alıcı dominant görebilir (0)
ACK delimiter	1	Recessive olmalıdır (1)
End-of-frame (EOF)	7	Recessive olmalıdır (1)



Şekil 5.2. Temel Paket Formatı

Tanımlayıcıdaki tek kısıtlama, ilk 7 bitin hepsi birden recessive olamaz. (örn. 111111xxxx yanlışdır)

5.4.1.2. Genişletilmiş paket formatı

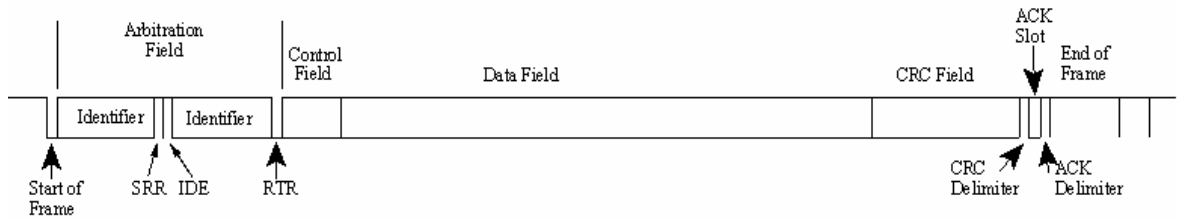
Paket formatı yapısı aşağıdaki gibidir:

Tablo 5.3. Genişletilmiş Paket Formatı

Alan Adı	Uzunluk (bit)	Amacı
Start-of-frame	1	Paket başlangıcını gösterir
Identifier A (Tanımlayıcı A)	11	Veri için eşsiz(unique) bir tanımlayıcının ilk parçası
Substitute remote request (SRR)	1	Recessive olmalıdır (1) Seçimsel
Identifier extension bit (IDE)	1	Recessive olmalıdır (1) Seçimsel
Identifier B (Tanımlayıcı B)	18	Veri için eşsiz(unique) bir tanımlayıcının ikinci parçası
Remote transmission request (RTR)	1	Dominant olmalıdır (0)
Reserved bits (r0, r1)	2	Ayrılmış bitler (Dominant yapılmalıdır(0), fakat dominant ya da recessive olarak alınır)
Data length code (DLC)	4	Verinin byte sayısı (0-8 byte)
Data field	0-8 byte	Gönderilecek veri (uzunluğu DLC alanında belirlenir)
CRC	15	Cyclic redundancy check
CRC delimiter	1	Recessive olmalıdır (1)

Tablo 5.3. Genişletilmiş Paket Formatı (Devam)

Alan Adı	Uzunluk (bit)	Amacı
ACK delimiter	1	Recessive olmalıdır (1)
End-of-frame (EOF)	7	Recessive olmalıdır (1)



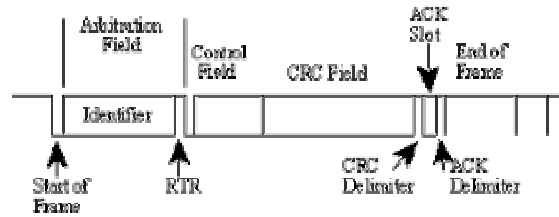
Şekil 5.3. Genişletilmiş Paket Formatı

A ve B tanımlayıcı alanları 29 bit bir tanımlayıcıdan oluşur.

5.4.2. Uzak paketi (Remote Frame)

Uzak paketi, data paketine benzer. Fakat şunlar farklıdır:

- RTR bir recessive (1) yapılır.
- Data uzunluğu, data paketinden alınan byte sayısını içerir.



Şekil 5.4. Uzak Paketi Yapısı

5.4.3. Hata paketi (Error Frame)

Hata paketi iki farklı alandan oluşur.

İlk alan, farklı istasyonların katkıda bulunduğu hata bayraklarının (Error Flags) süper pozisyonlarından oluşur. Takip eden ikinci alan hata sınırlayıcıdır(Error Delimiter).

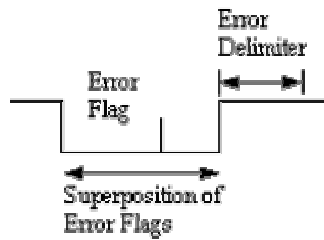
İki tip hata bayrağı vardır.

1. Aktif Hata Bayrağı:

“Hata Aktif” hata durumunda bulunan ağ üzerindeki hata algılayan bir nokta tarafından gönderilir.

2. Pasif Hata Bayrağı:

“Hata Pasif” hata durumunda bulunan ağ üzerindeki aktif hata paketi algılayan nokta tarafından gönderilir.



Şekil 5.5. Hata Paketi Yapısı

5.4.4. Aşırı yükleme paketi (Overload Frame)

Aşırı yükleme paketi iki bit alanından oluşur, aşırı yükleme bayrağı (Overload Flag) ve aşırı yükleme sınırlayıcısı (Overload Delimiter). Aşırı yükleme bayrağının gönderilmesine sebep olabilecek iki çeşit aşırı yükleme durumu vardır:

1. Bir sonraki veri ya da uzak paketi için gecikme gerektiren, alıcının iç durumu.
2. Görev sırasında dominant bit tespiti.

İlk maddedeki sebepten dolayı başlatılacak aşırı yükleme paketi, sadece beklenen görevin ilk bit zamanında başlatılabilir. Oysa ikinci maddedeki sebepten dolayı oluşan aşırı yükleme durumu , dominant bit algılandıktan bir bit sonra başlar. Aşırı yükleme bayrağı 6 dominant bit içerir. Aşırı yükleme bayrağının yapısı, görev boşluğunun sabit yapısını yok eder. Bunun bir sonucu olarak, aşırı yükleme tespit eden de dahil tüm diğer istasyonlar kendi parçalarında aşırı yükleme bayrağı iletmeye başlarlar. Aşırı yükleme sınırlayıcı sekiz recessive bitten oluşur. Aşırı yükleme sınırlayıcısı hata sınırlayıcısı ile aynı yapıdadır.

BÖLÜM 6. ETHERNET – CAN KARŞILAŞTIRMASI

6.1. CAN Bus

Controller Area Network (CAN), çok yüksek güvenlik seviyesine sahip verimli bir seri iletişim protokolüdür. Uygulama alanı, yüksek hızlı ağlardan düşük maliyetli çok katmanlı iletim sistemlerine kadar uzanır. Otomotiv elektroniğinde, motor kontrol üniteleri, sensörler, anti patinaj sistemleri birbirine 1Mbit/s hızlara kadar ulaşabilen CAN veri yolu ile bağlanmışlardır. Veri yolu ne zaman boş olursa, herhangi bir modül mesaj göndermek üzere iletişime geçebilir. Eğer 2 ya da daha fazla modül aynı anda veri göndermeye başlarsa, bu çakışma IDENTIFIER kullanılarak bit bit işlenip bir hakem tarafından çözümlenir. Bu hakem mekanizması bilgi ya da zaman kaybı olmayacağını garanti eder. Hakemlik işlemi süresince, her gönderici gönderilen bitin seviyesini veri yolunda gözlenen seviye ile karşılaştırır. Eğer bu seviyeler eşitse, modül göndermeye devam eder. Bir “recessive(bastırılmış)” seviye gönderildiğinde “dominant(baskın)” bir seviye gözleniyorsa modül bir tane daha bit göndermeden geri çekilmelidir.

Bit bit hakemlik işleminde, bir bit göndermek için gerekli olan zaman, tüm kablo uzunluğunu doldurmak için yeteri kadar uzun olmalıdır ve tüm istasyonlar tutarlı bir geri dönüş değerine sahip olmalıdır. 1Mbit/s’de, ISO11898 standardına göre CAN veri yolu 40m’yi geçemez. 1Mbit/s’nin daha üstünde bir veri hızına ulaşmak için birçok endüstriyel veya otomotiv uygulamalarında kablo uzunluğunun düşürülmesi gerekir. Böylece, minimum kullanılabilen kablo uzunluğu kısıtlamaları, maksimum CAN hızı için temel limitlerde olması gerekir.

Bit bit hakemlik sistemi mesajlar için gecikme zamanlarını garanti edebilmesi için gereklidir. Böyle garantiler gerçek zamanlı sistemler için zorunludur. Eğer CAN veri yolundaki tüm mesajlar periyodik ise, CAN veri yoluna rate monotonic (RM)

planlama uygulanabilir. RM planlama kullanarak, belirli bir IDENTIFIER' a ait mesaj gecikmesi için en yüksek limit hesaplanabilir. RM planlama, herhangi bir planlama algoritması yapılabilir bir plan üretebiliyorsa, RM planlamasının da bunu yapabileceğini garanti eden optimum bir sabit-öncelik planlama algoritmasıdır. Ancak sabit planlamada RM algoritmasını kullanabilmek için, bütün mesajlar bir periyot içinde ayarlanmış olmalıdırlar. Mesaj ulaşım zamanları, bu periyottan maksimum kuyruk zamanından fazla sapamaz. Bu yüzden, sabit planlama uygulanabilir, fakat mesaj ulaşım zamanlarının hiçbir zaman planlama için varsayılan değerden daha kısa olmadığı bir yolda, dönemsel olmayan mesajların tampona alınmasına dikkat edildiği müddetçe yapılabilir.

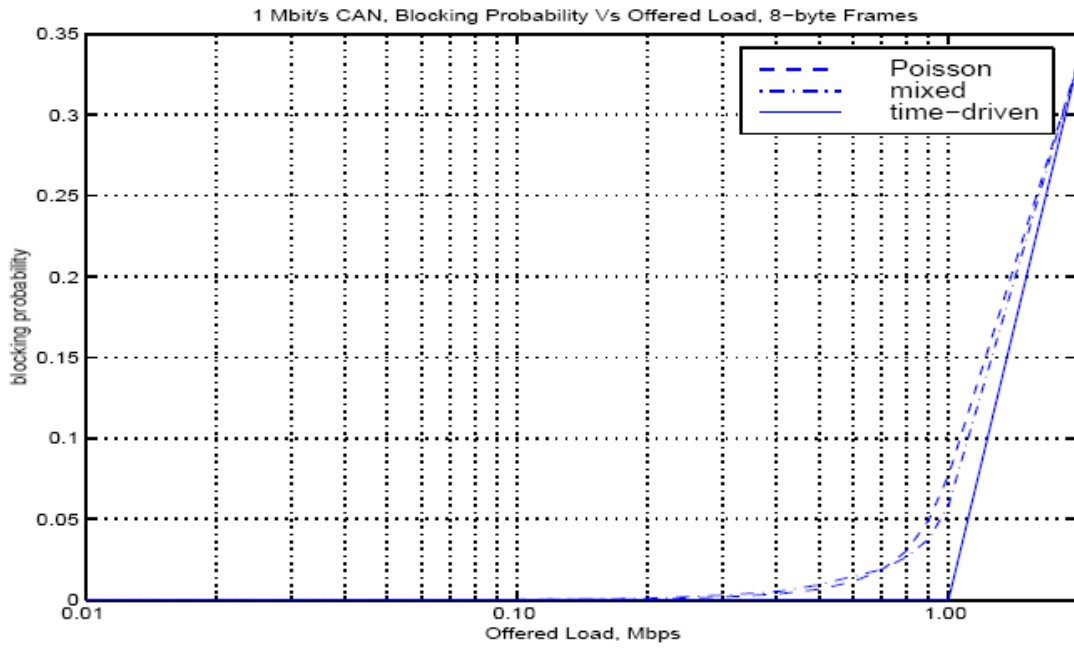
Sonuç olarak, CAN, zaman-sürümlü mesajların, Ethernet'te meydana gelebilen tarzda çakışmaların çözümlenmesi ile geçirilen zaman kaybına sebep olmadan mesajların yönetilmesi için tasarlanmıştır. Can veri yolunda sabit ve optimum planlama yapmak mümkündür. Fakat CAN tasarımları, Ethernet'ten yavaş olup, yaklaşık olarak 1Mbit/s den daha hızlı çalışamaz.

6.2. Ethernet – CAN Karşılaştırması

Bu bölümde standart 10base-T Ethernet ile Controller Area Network(CAN) için gecikme zamanlarını karşılaştıracamız. İki ağ da farklı bit hızlarında çalışırlar, fakat CAN yapısının özellikle embedded sistemler için tasarlandığını göz önüne alırsak, embedded sistemler için bir performans analizi yapılabilir. CAN'in bozulmayan bit bit hakemlik sistemi, eş zamanlı seri hale getirilmiş gönderimlerin veri yolu genişliği israf edilmeden yapılabilmesini sağlar. CAN sistemi tasarımcıları, zaman-sürümlü mesajların uygun ve çakışmasız olmasını bekler. Gerçek zamanlı sistemlerde CAN, RM planlama yapılarak bir üst limitin mesajlar içine teorik olarak yerleştirilebilmesi özelliği ile kesin olarak en kullanışlı protokoldür.

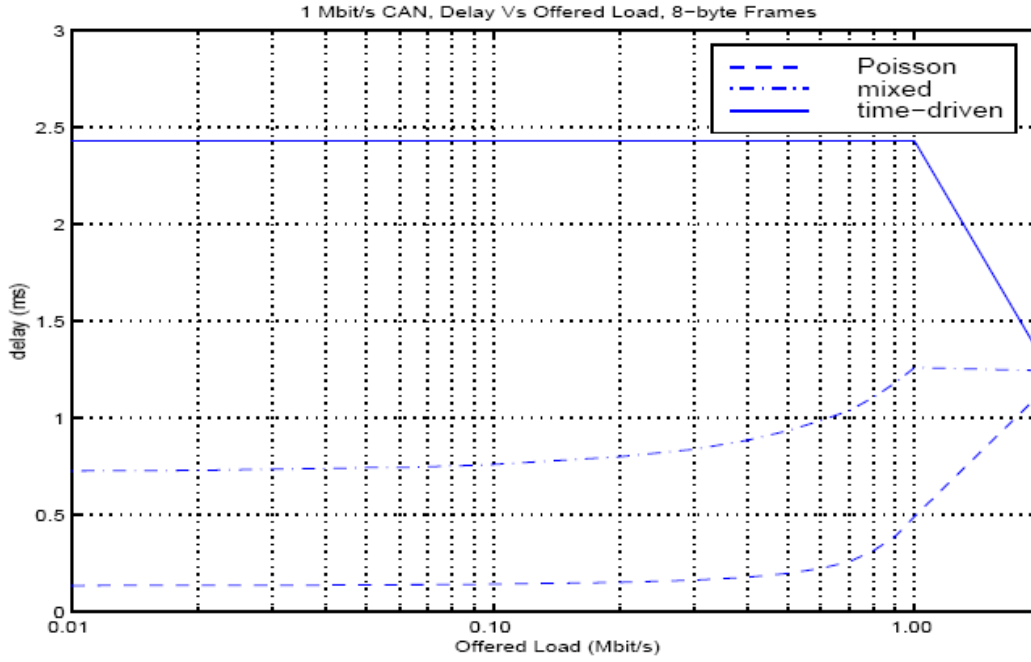
Fakat karşılaştırmalarda Ethernet'in daha iyi bir mesaj gecikmesi ortalaması olduğunu göreceğiz. Ethernet üzerinden mesajlar CAN'e oranla %99 oranında daha iyi bir gecikme ile iletilir.

Düşük bit hızına rağmen, CAN veri yolu hakemliği eş zamanlı iletim için verimli bir kontrol sağlar. Şekil 6.1.'de üç ayrı çalışma şekli için talep edilen (offered load) yüke göre engelleme olasılığı (blocking probability) gösterilmiştir. Engelleme olasılığı burada, bir mesajın hat dolu iken mesaj çıkışına ulaşma olasılığı olarak tanımlanmıştır. Böyle mesajlar, hatta girerken "engelle"dir ve hattan atılır. CAN mesajları için, her CAN IDENTIFIER'a ait bir çıkış slotu vardır. Bu yüzden, eğer belirli bir IDENTIFIER için önceki mesaj düğümün çıkış slotundan temizlenmemişse, mevcut mesaj engellenir. Her CAN mesajında verinin bir byte'ının gönderilmesi toplam mesaj uzunluğunu 8 byte üzerinde yapar. Zaman-sürümlü veri iletiminin engelleme olasılığı "0"da tutulur ve sadece talep edilen yük ağ kapasitesi olan 1Mbit/s'yi geçtiği zaman yükselir. Bu CAN'nin serileştirilmiş zaman-sürümlü mesajlarda ne kadar verimli olduğunu göstermektedir. Tüm iletimler belirlendiği takdirde hiçbir mesaj engellenmez. Fakat poisson ve karışık veri yapıları için, ağ 1Mbit/s'de tam yüklendiğinde engelleme olasılığı değeri sırasıyla %6 ve %8 oranlarına ulaşır. Poisson tipi mesajlar için, mesaj ulaşım zamanlarında üssel dağılım görülmesi, mesajlar için küçük ulaşım zamanına sahip bir hakemlik ile mesaj oluşturulabilmesi anlamına gelir. Eğer bir mesaj ağdan temizlenmiş olan önceki mesajdan önce iletilirse, mesaj engellenecektir. Bu Poisson yapısı için %6 oranında engelleme oranındadır. Karışık(mixed) yapı için, daha az poisson mesajları ortaya çıkar, fakat bu mesajların yapının diğer yarısını oluşturan periyodik mesajların serileştirilmesinden dolayı daha çok geciktirilmesi muhtemeldir. Bundan dolayı, poisson mesajları hiç periyodik mesaj oluşmaması durumunda bile engellenmesi daha muhtemeldir. Dolayısıyla, CAN zaman-sürümlü trafiği iyi kontrol ettiği müddetçe bile, mesajların küçük bir kısmı poisson kaynakları oluştuğunda kaybedilecektir.



Şekil 6.1. Engelleme olasılığı - CAN üzerinde talep edilen yük

Şekil 6.2. ortalama gecikme – CAN üzerinde talep edilen yük grafiğini göstermektedir. Zaman-sürümlü yapı için mesaj gecikmeleri tamamiyle belirlidir ve talep edilen yük miktarına göre değişmez. Hattı periyodik olarak mesajlarla doldurmak ile sonra onları hat boşalınca kadar tek tek göndermek arasında bir benzetme yapılabilir. Hattı doldurma zamanlamaları arasındaki fark talep edilen yükü belirler. Toplam talep edilen yük 1Mbit/s'yi geçmediği müddetçe, hattı temizlemek için gerekli zaman, hattı doldurma zamanlama aralığına bakılmaksızın aynı olacaktır. Karışık ve poisson yapıları için talep edilen yük ağ kapasitesi olan 1Mbit/s'ye yaklaştıkça gecikme zamanı artar. Gecikme zamanının artması aslında Şekil 6.1.'de görülen engelleme olasılığının artmasındandır. Ağdaki geciken mesajlar gelen mesajların engellenme sayısının artmasına sebep olabilir.

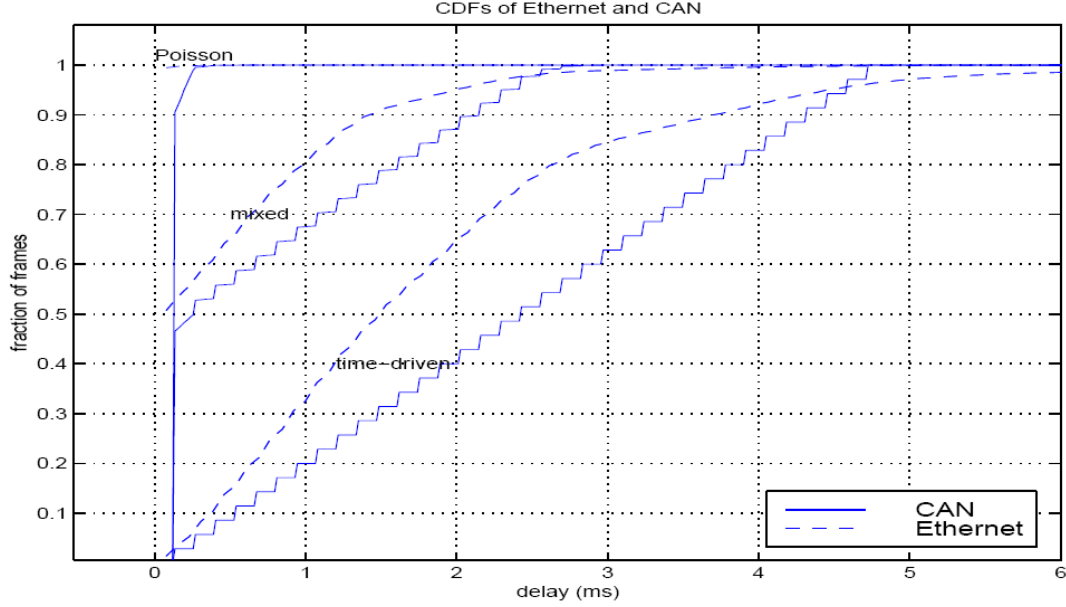


Şekil 6.2. Ortalama gecikme - CAN üzerinde talep edilen yük

Şekil 6.3.'de poisson, karışık ve zaman-sürümlü yapılar için, CAN'ın CDF'leri (Cumulative Distribution Functions) ve Ethernet karşılaştırması görülmektedir. Zaman-sürümlü ve karışık yapılardaki merdiven adımı görünümü mesajların eş zamanlı iletimi ve temizlenmesinden kaynaklanmaktadır. Zaman-sürümlü yapıda, verilen herhangi bir mesaj belirlidir ve aynı gecikmeye sahiptir, çünkü genel hattaki pozisyonu her zaman aynıdır. Her mesaj unique(essiz) bir identifier(tanımlayıcı)'a sahiptir ve böylece mesajlar her zaman aynı komutta gönderilir. Bundan dolayı, zaman-sürümlü yapı için CDF her bir mesaj için bir tane olmak üzere 35 ayrı gecikmeye sahiptir. Karışık CAN yapısı için, 17 ayrı gecikme oluşur (her bir zaman sürümlü kaynak için bir tane). Aynı zamanda Karışık CAN yapısı için CDF, poisson olan kaynağın yarısı ile tutarlı olarak, iletilen tüm mesajların yarısının neredeyse gecikmesiz olduğunu gösterir.

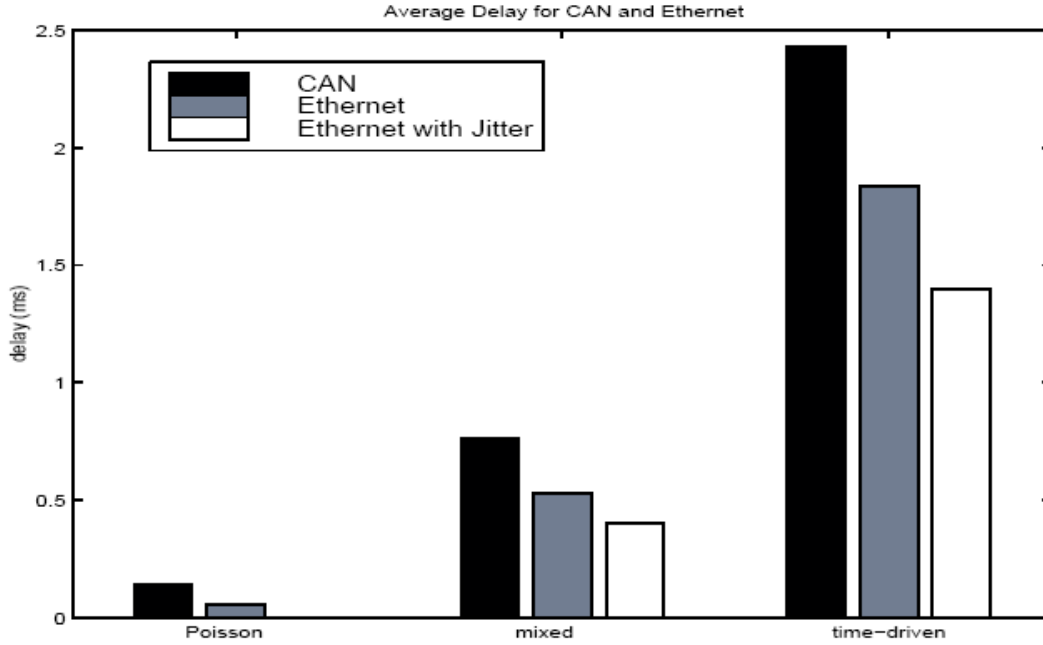
Ethernet CDF'lerinin CAN CDF'lerine göre avantajları ve dezavantajları vardır. Poisson yapısı için, Ethernet CDF'leri bir adım fonksiyonudur. Ağ hızlarındaki büyük fark yüzünden, çakışma olasılığı düşük olduğunda Ethernet net olarak soft real-time olarak CAN performansı gösterir. Şaşırtıcı bir şekilde, Ethernet CDF'leri

ortalama his olarak, zaman-sürümlü mesajlar (karışık ve zaman-sürümlü yapıda) oluştuğunda bile CAN CDF'lerinden daha iyidir.



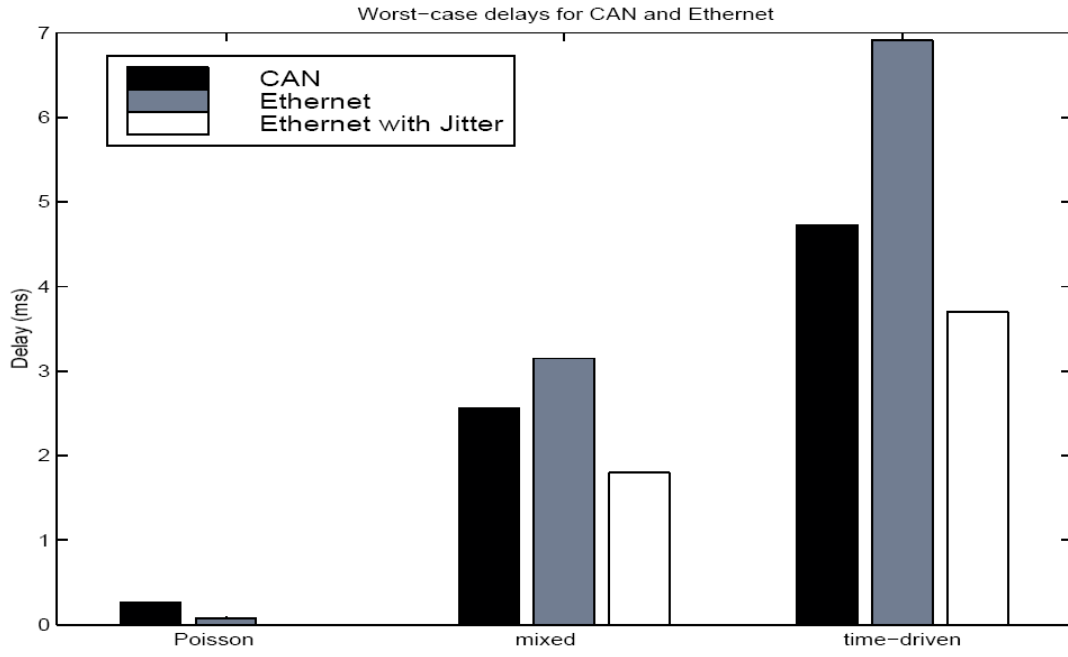
Şekil 6.3. Ethernet ve CAN için mesaj gecikmelerinin CDF'leri

Şekil 6.4.'te CAN ve Ethernet için poisson, karışık ve zaman-sürümlü yapılarda ortalama gecikmeler karşılaştırılmaktadır. Her yapı için, Ethernet daha düşük gecikmelere sahiptir. Bu zaman-sürümlü Ethernet mesajlarının veri yolu üzerinde yer tutmadan birçok kez çarpışmasına rağmen, Ethernet ve CAN hızlarındaki büyüklük farkı verimsizliği kontrol etmek için ortalama yeterlidir. Ayrıca Şekil 6.4.'ten, Ethernet üzerinde ark olduğu durumdaki ortalama gecikmenin, CAN ve Ethernet'teki(arksız) ortalama gecikmeden daha az olduğu görülmektedir.



Şekil 6.4. Ortalama Ethernet ve CAN gecikmelerinin karşılaştırılması

Şekil 6.5.'te CAN, Ethernet ve arklı Ethernet için en kötü durumlar karşılaştırılmaktadır. Çünkü Ethernet'in binary üssel durumu çok uzun gecikmelere sebep olabilir. Büyük miktarda çakışma olduğunda, %99 oranında mesaj gecikmesinin ortalama Ethernet gecikmesinden önemli ölçüde fazla olması sürpriz değildir. Zaman-sürümlü mesajlar oluşturulduğunda, Ethernet üzerinde birçok çakışmanın olacağı kesindir. Böylece, karışık ve zaman-sürümlü yapılar için gecikmelerin %99'u ortalama gecikmeden daha kötüdür. Fakat en kötü durum için CAN ve Ethernet mesajlarındaki fark karışık yapı için 0.5ms'den daha azdır ve zaman sürümlü yapı için 2ms civarındadır. Bu, Ethernet MAC tamamen senkronize iş yüküne bağlı olduğunda bile, onun neredeyse zaman periyodundaki tüm mesajların serileştirilmesini, CAN gecikmesinden sadece çok az fazla bir gecikmeyle yönettiğini gösterir. Ark oluştuğunda, Ethernet'in en kötü gecikmesi CAN gecikmesinden(karışık ve zaman sürümlü yapılar için) gerçekten daha kısadır.



Şekil 6.5. Ethernet ve CAN gecikmelerinin en kötü durum için karşılaştırılması

BÖLÜM 7. SİSTEM AKIŞ DİYAGRAMLARI

Bu bölümde tüm sistemin akış diyagramları server(sunucu) ve client(istemci) olarak iki ayrı başlık altında verilmiştir.

7.1. Sunucu Modülü Akış Diyagramları

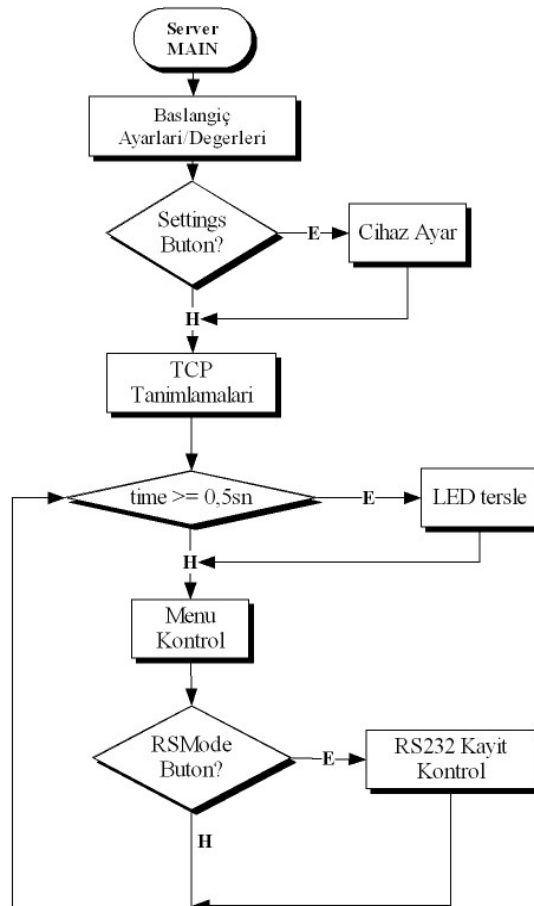
Sunucu modülünün tüm yapısı bu başlık altında 7 parçaya ayrılarak incelenmiştir.

Bunlar:

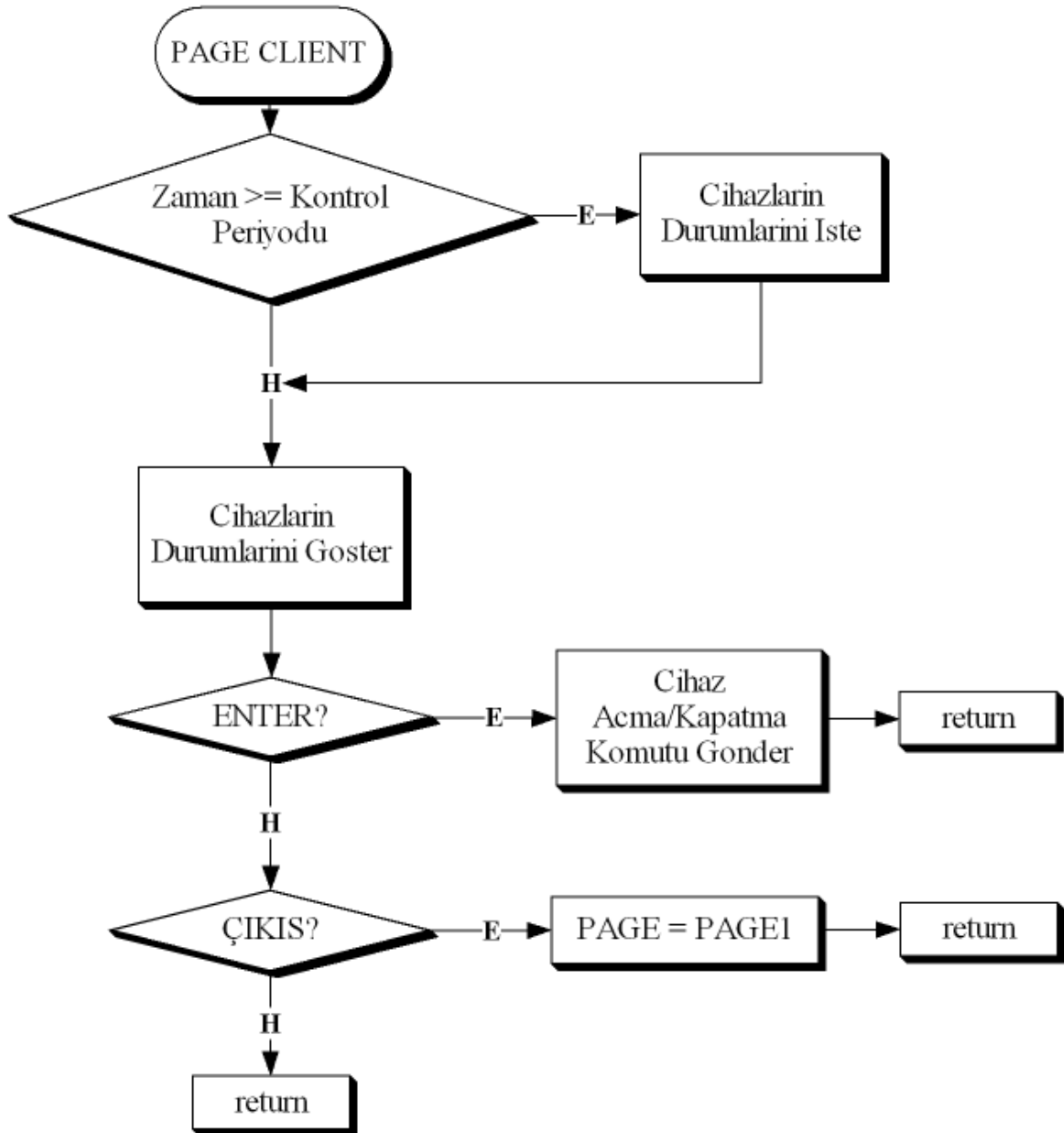
- Sunucu “Main Program”
Sunucu modülünün ana çalışma yapısını oluşturan bölümüdür. Bu bölümde gerekli başlangıç tanımlamaları, sistemin aktif olduğunu gösteren “status led” kontrolü, menü kontrolü ve rs232 seri port kayıt kontrolü yapılır.
- Sunucu “Menü Kontrol”
Kullanıcı için oluşturulan, sunucu cihazın kontrolünü sağlayan bir kullanıcı arabirimidir. Bu kısımda bulunan maksimum kayıt miktarı kontrol edilir ve kayıta bulunan istemcilere ait bilgiler ekranda gösterilir. İstemcilerin “açık/kapalı” durumları ve bir istemci üzerindeki cihazlara ait “açık/kapalı” durumları bu kısım üzerinden görülür.
- Sunucu “PAGE CLIENT” yapısı
Menüden seçilen belirli bir istemciye ait cihazların durumlarının gözlenildiği ve istendiğinde istemci üzerindeki cihazlara komut gönderilen kısımdır.

- Sunucu "RS232 Kayıt Kontrol"
Sunucu modülünün seri portu üzerinden istemcilerin bilgilerine ait ekleme, değişim, silme ve bilgi okuma gibi işlemlerin yapıldığı kısımdır.
- Sunucu "Cihaz Durumlarını İsteme" yapısı
Belirli bir istemciye ait cihazların durumlarının TCP protokolü ile istendiği kısımdır.
- Sunucu "Clientların Durumlarını İsteme" yapısı
Sunucu EEPROM'unda kayıtlı istemcilere ait IP'lerin kontrolü yapılarak istemcilerin "açık/kapalı" durumlarının belirlendiği kısımdır.
- Sunucu "TCP Komut Gönderme" yapısı
Sunucu üzerinden istemciye TCP protokolü ile komut gönderim işleminin yapıldığı kısımdır.

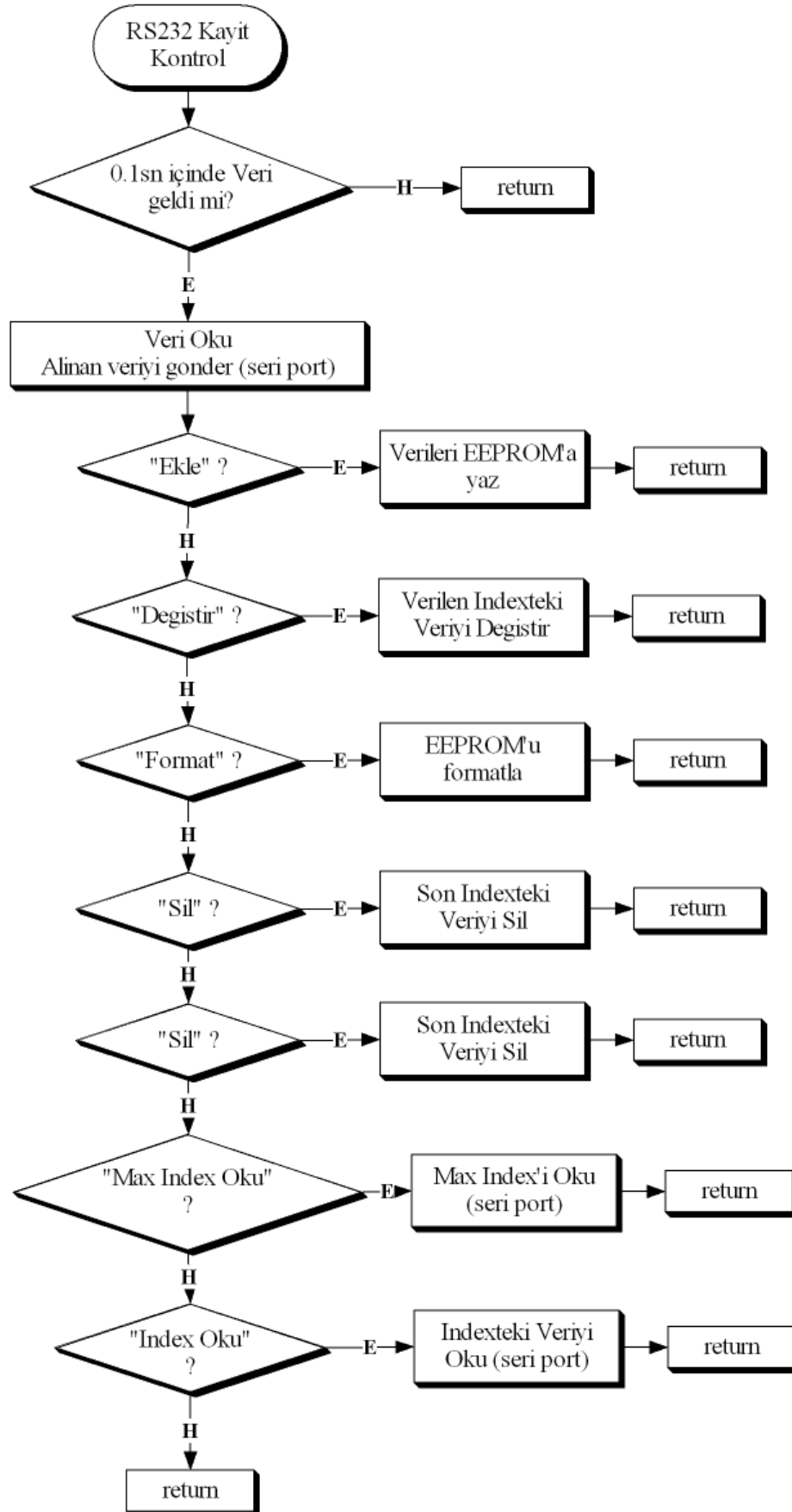
7.1.1. Sunucu main program akış diyagramı

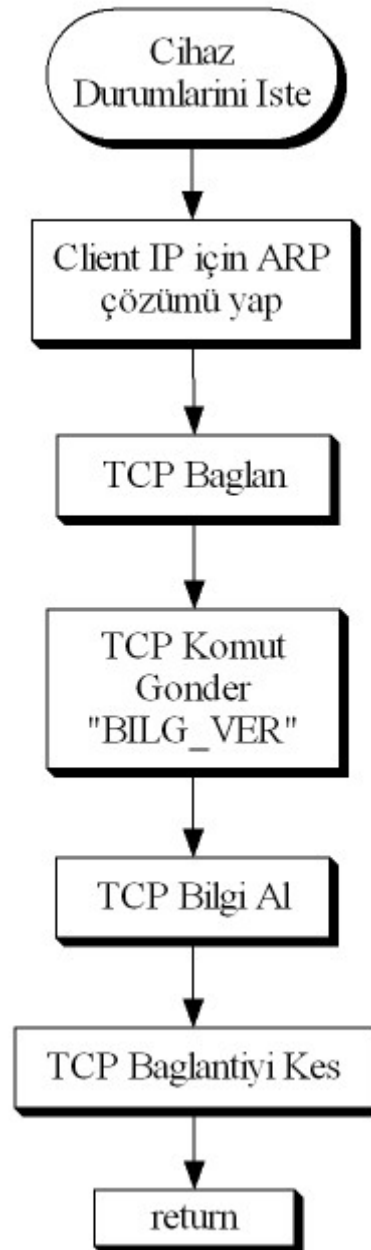


7.1.3. Sunucu "PAGE CLIENT" yapısı

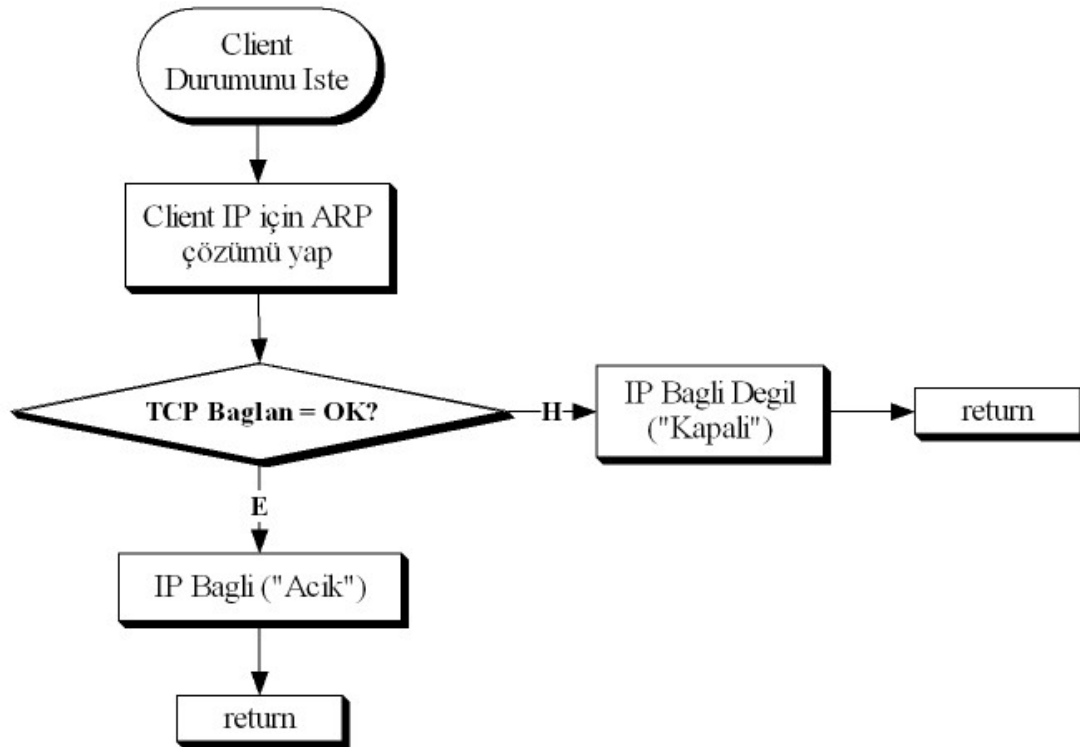


7.1.4. Sunucu RS232 seri port kayıt kontrol

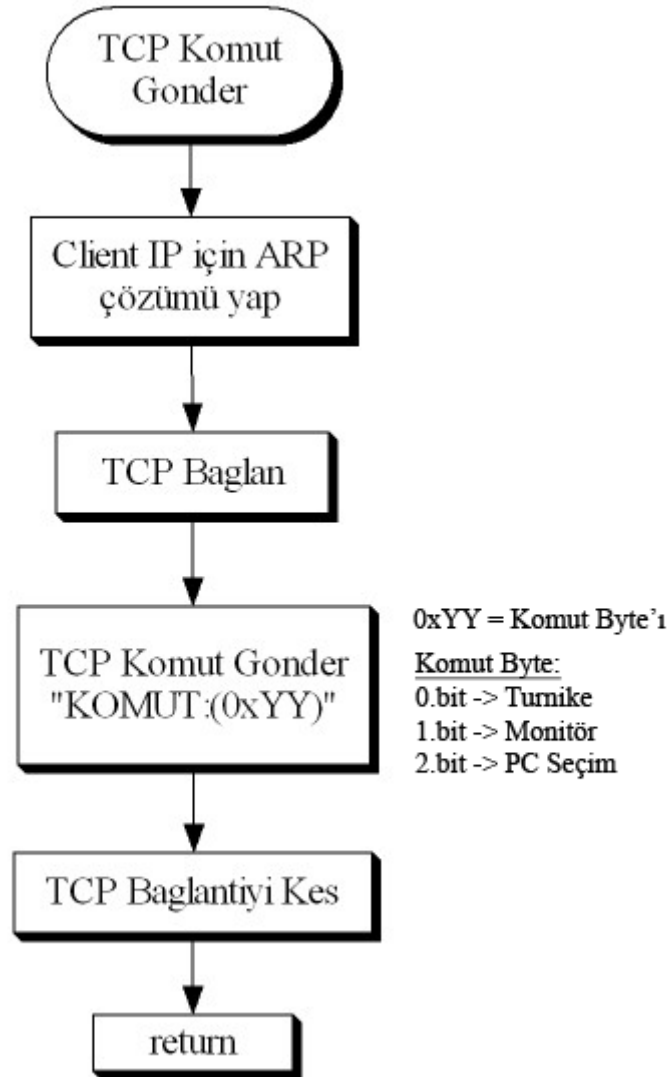


7.1.5. Sunucu cihaz durumlarını isteme yapısı akış diyagramı

7.1.6. Sunucu clientların durumlarını isteme yapısı akış diyagramı



7.1.6. Sunucu TCP komut gönderme yapısı akış diyagramı

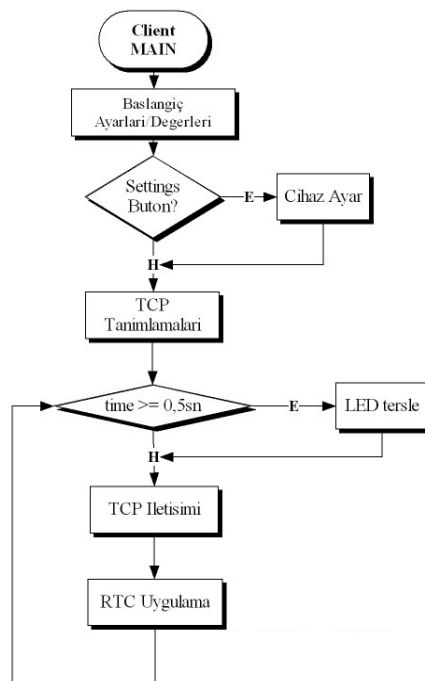


7.2. İstemci Modülü Akış Diyagramları

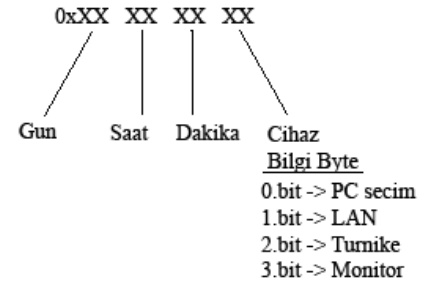
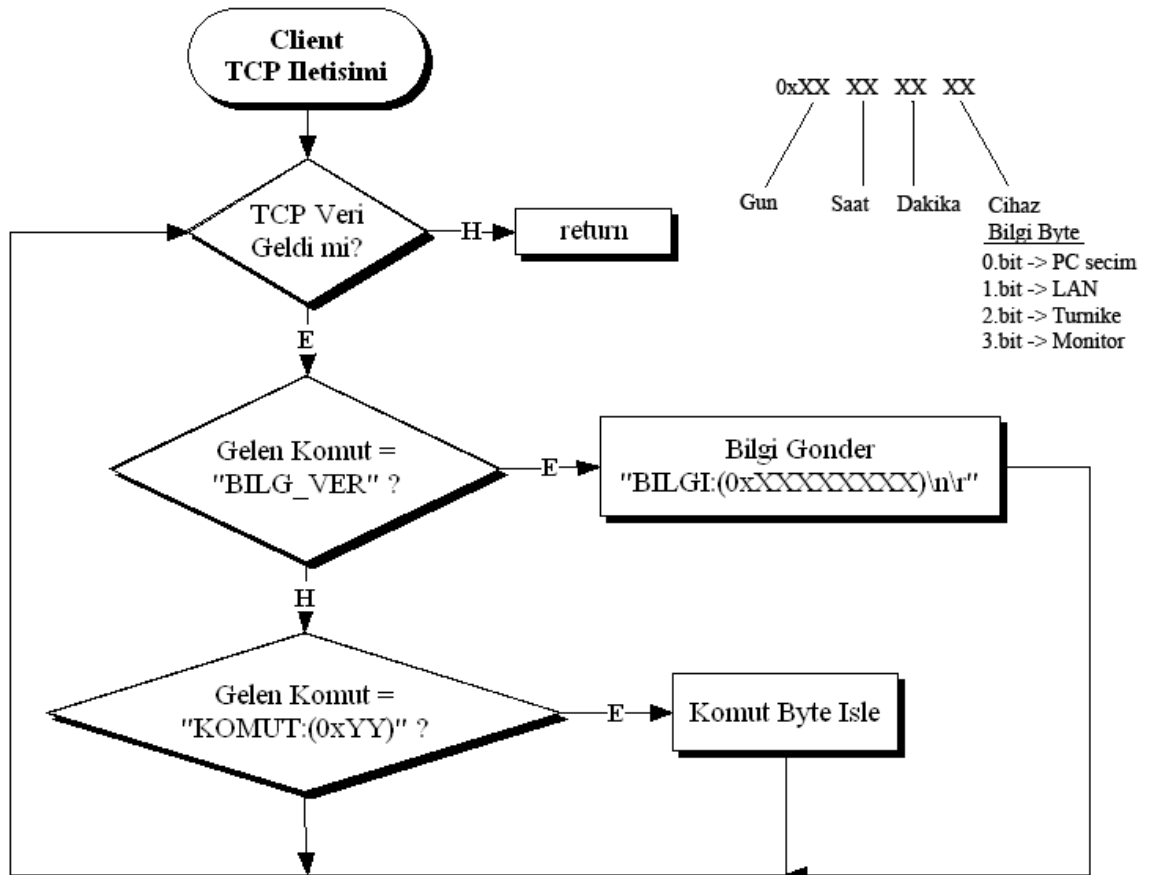
İstemci modülünün tüm yapısı bu başlık altında 3 ana parçaya ayrılarak incelenmiştir. Bunlar:

- İstemci “Main Program”
İstemci modülünün ana çalışma yapısını oluşturan bölümüdür. Bu bölümde gerekli başlangıç tanımlamaları, sistemin aktif olduğunu gösteren “status led” kontrolü, TCP iletişimi ve RTC yapısı ile cihaz kontrolü yapılır.
- İstemci “TCP İletişimi”
Sunucunun haberleşmesini sağlayan TCP protokolü komutlarını içerir.
- İstemci RTC yapısı ile cihaz kontrolü
Cihazların bir RTC yapısı ile kontrolünün yapıldığı bölümdür. Bu bölümde RTC saat kontrolü, Program kontrol, Periyodik LAN açma/kapatma kontrolü, PC seçim kontrolü, Seri alt program kontrolü, Manüel turnike kontrolü, Monitör kontrolü gibi kontroller yapılır.

7.2.1. İstemci main program akış diyagramı



7.2.2. İstemci TCP iletişimi akış diyagramı

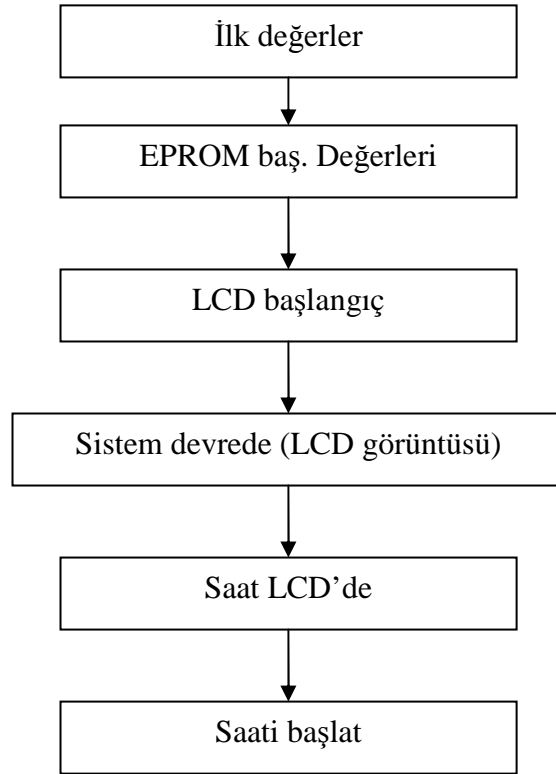


0xYY = Komut Byte'ı

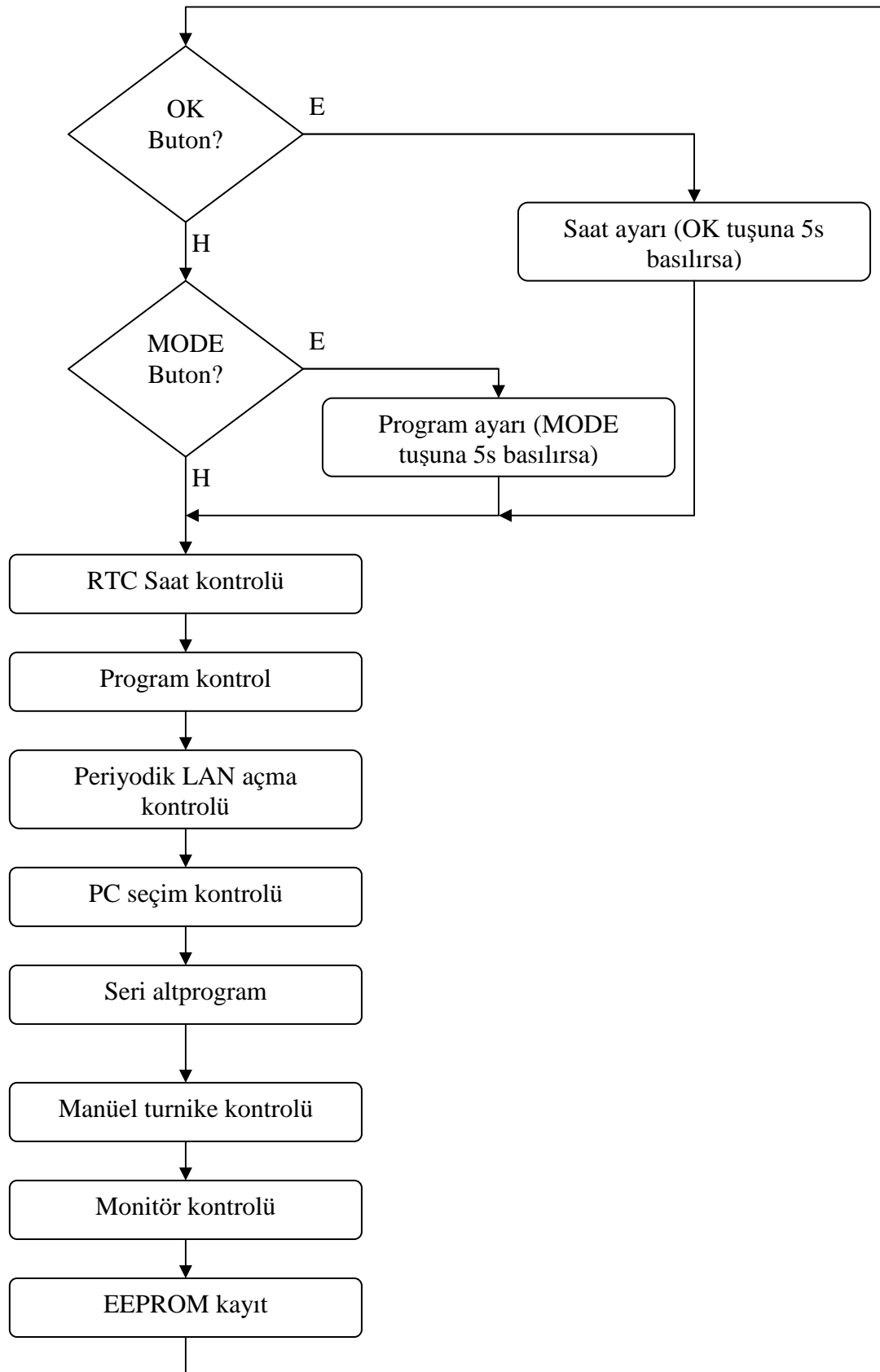
Komut Byte:
 0.bit -> Turnike
 1.bit -> Monitör
 2.bit -> PC Seçim

7.2.3. İstemci RTC yapısı ile cihaz kontrolü akış diyagramı

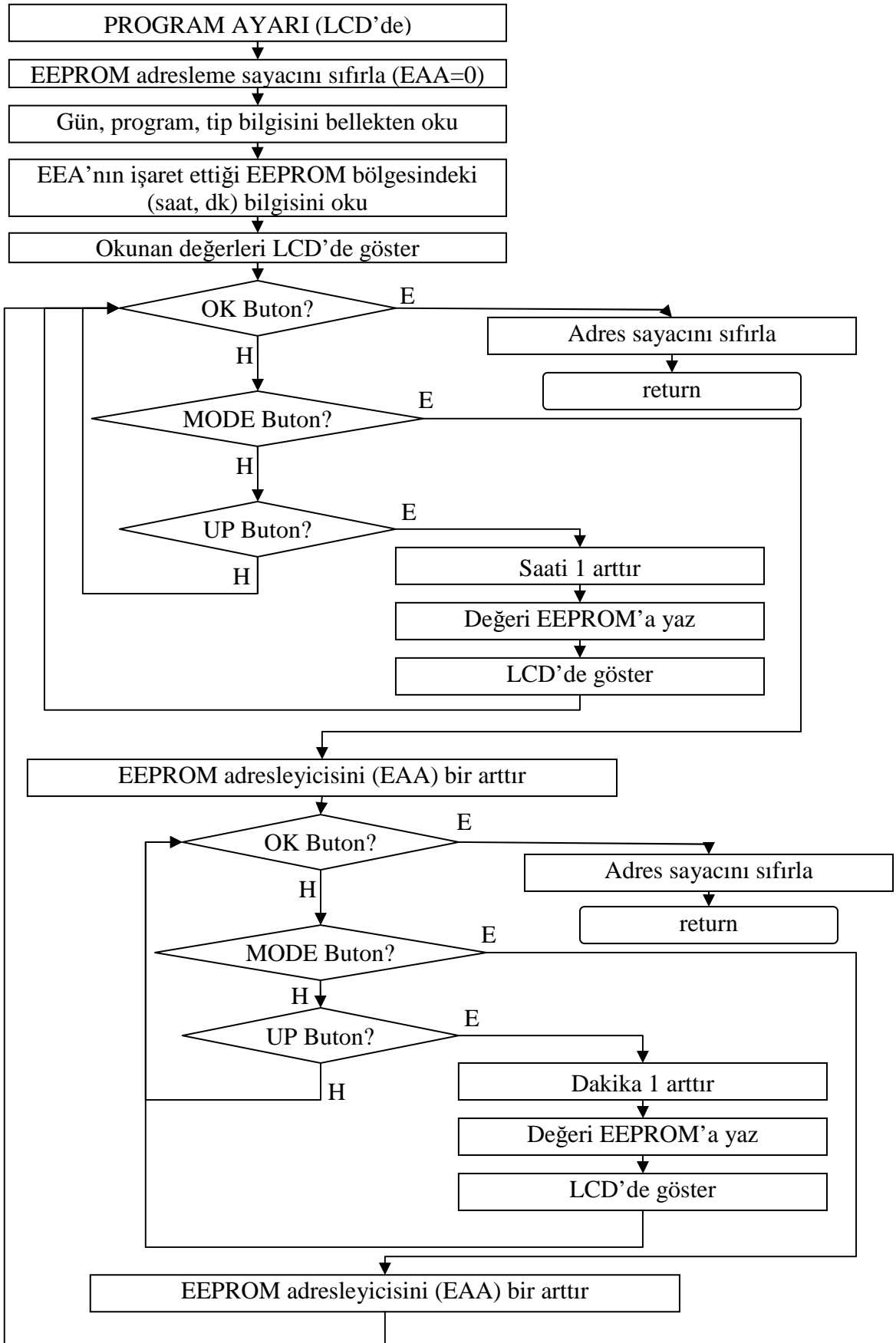
7.2.3.1. Başlangıç



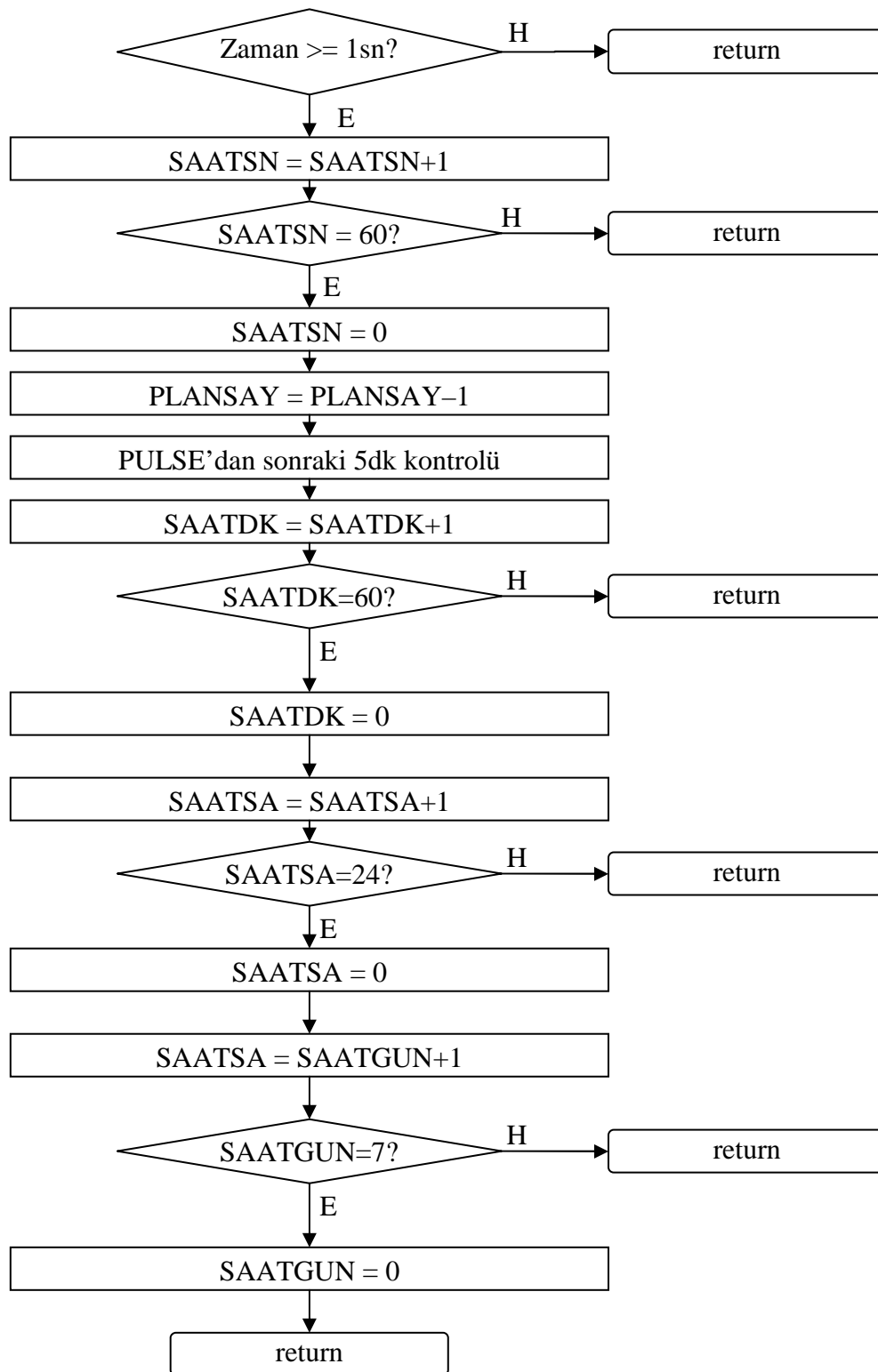
7.2.3.2. Ana program



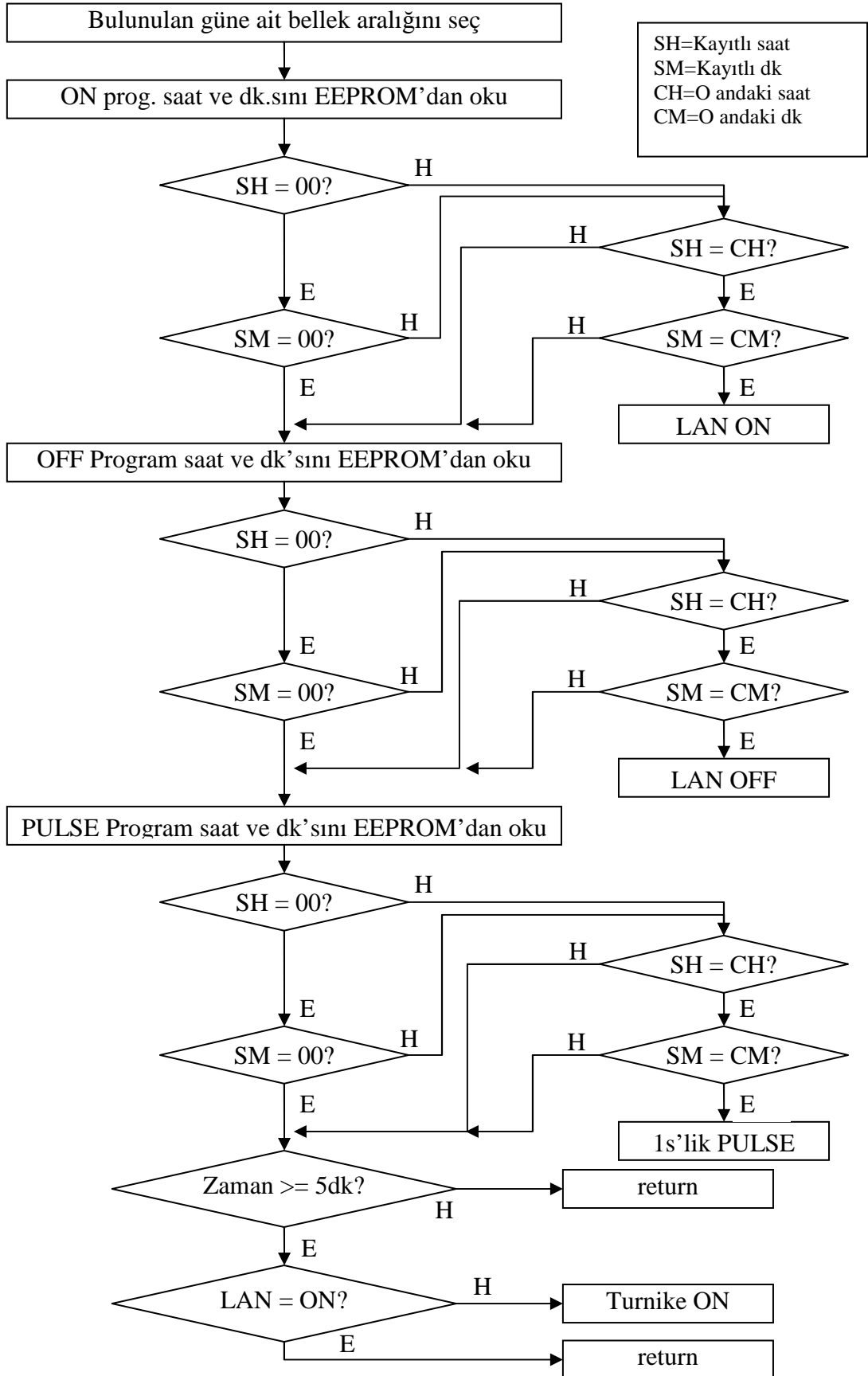
7.2.3.3. Program ayarı



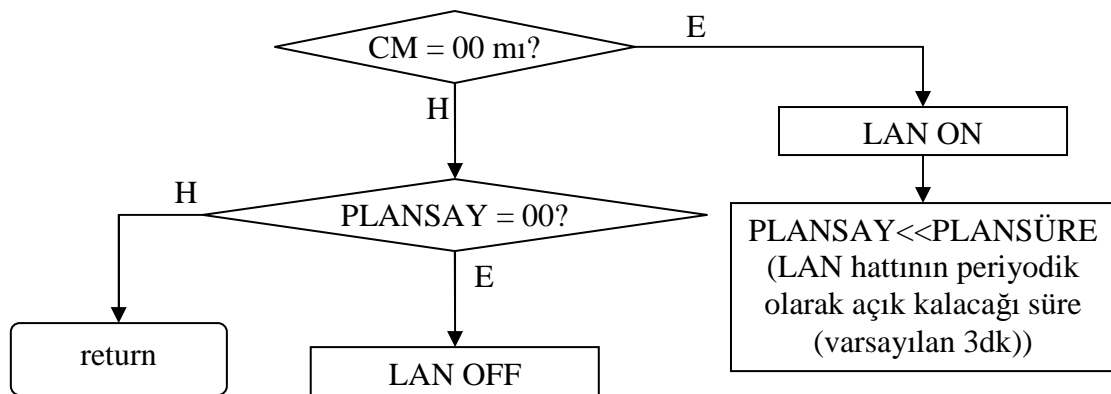
7.2.3.4. RTC saat kontrolü



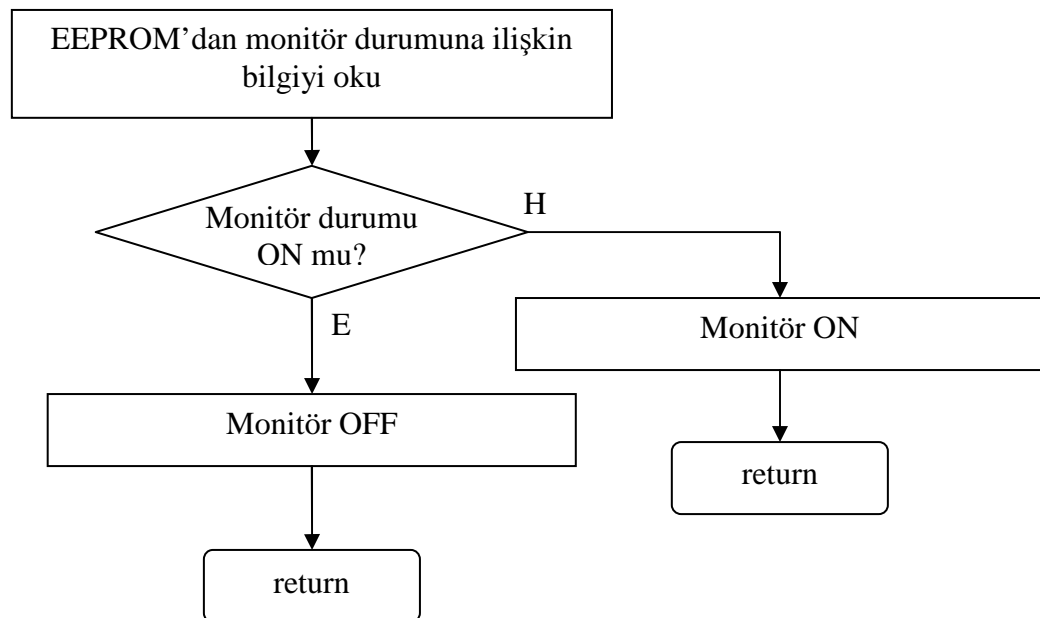
7.2.3.5. Program kontrol



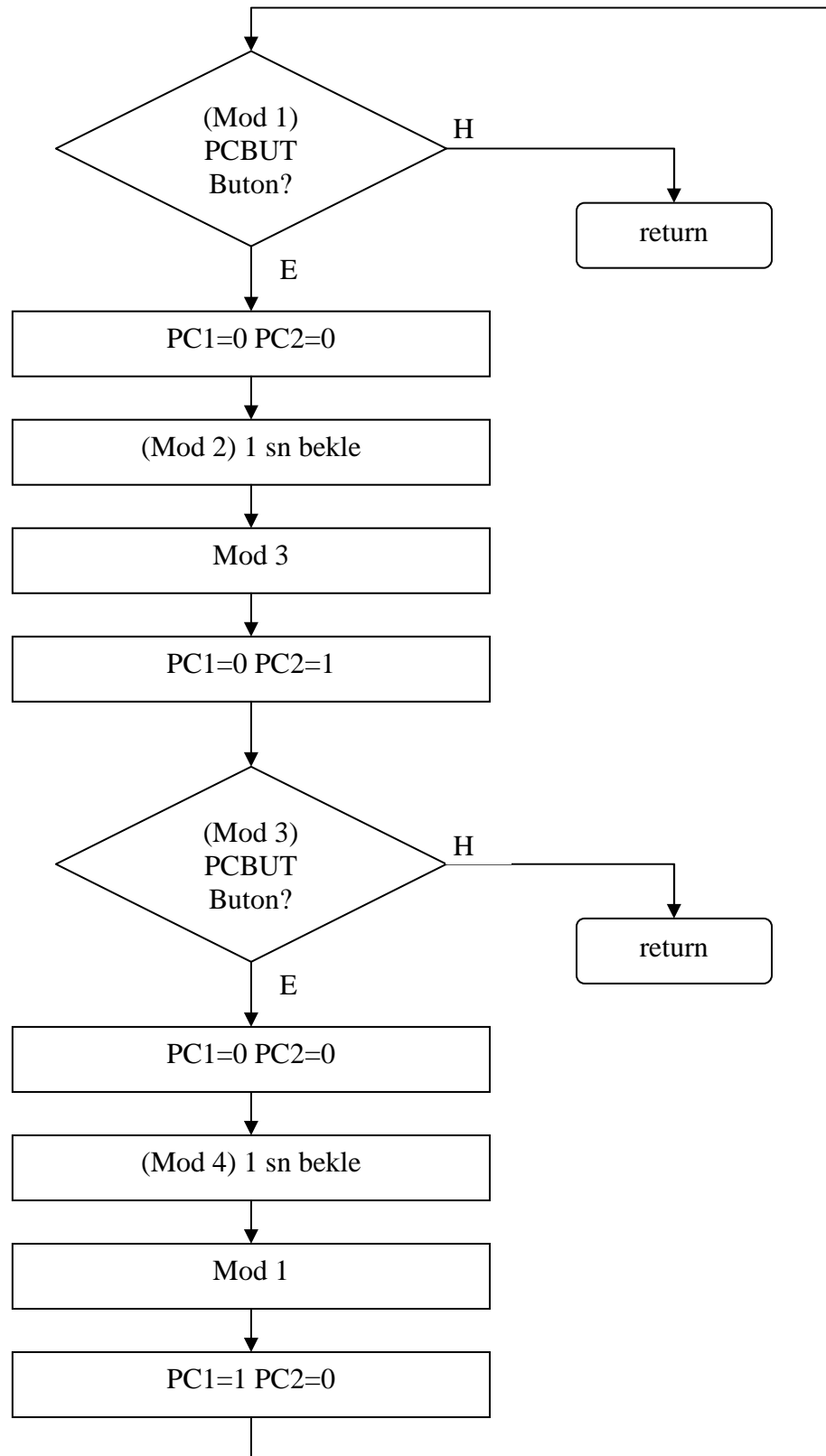
7.2.3.6. Periyodik LAN açma/kapatma kontrolü



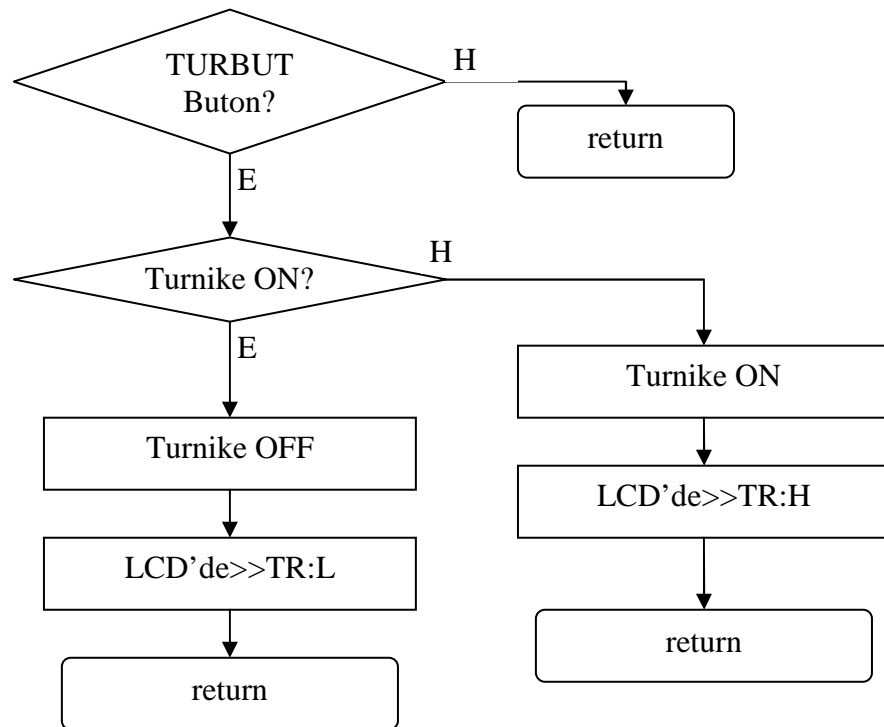
7.2.3.7. Monitör kontrol



7.2.3.8. PC seçim kontrol

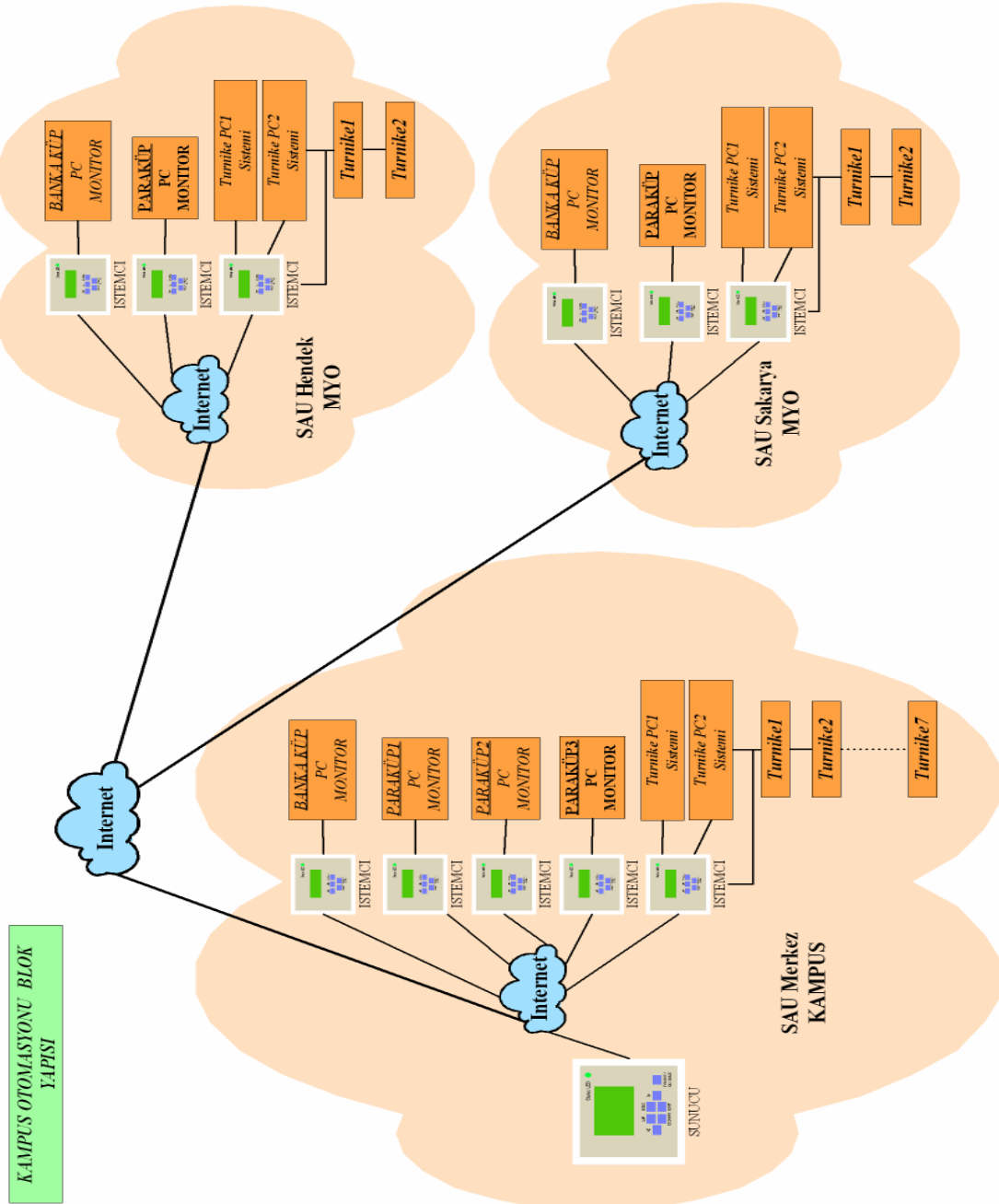


7.2.3.9. Turnike manüel kontrol



BÖLÜM 8. SİSTEM ÇALIŞMA YAPISI

Şekil 8.1.'de, tasarlanan otomasyon sisteminin blok yapısı gösterilmiştir.



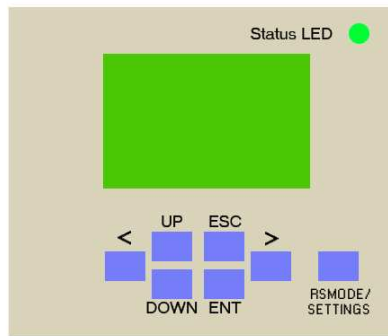
Şekil 8.1. Kampüs otomasyon sistemi blok yapısı

Blok yapısından da görüleceği gibi sistemimiz BankaKüp, ParaKüp ve Turnike sistemlerine bağlı olan PC, monitör ve Turnike gibi cihazları kontrol etmektedir.

8.1. Sunucu Devresi Kullanımı

8.1.1. Tanım

Server(Sunucu) devresi tüm bilgilerin kontrol edildiği yerdir. Bu sebeple ana kontrol ünitesi de diyebiliriz. Görevi, Client(istemci) devrelerden gerekli bilgileri toplamak gerektiğinde istemci devrelere komut göndermektir.



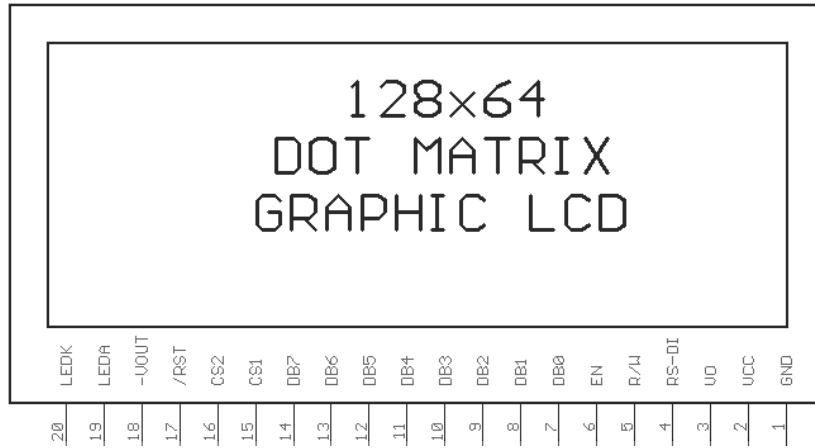
Şekil 8.2. Sunucu modülü dış görünümü

8.1.2. Özellikleri

- 40 taneye kadar istemci kontrol edebilir.
- İstemci bilgileri “İstemci Adı” ve “IP Adresi” şeklinde EEPROM’a kaydedilir.
- Her istemciye ait “Açık/Kapalı” durumu, “Turnike”, “Monitör”, “PC” ve “LAN” durumları “Açık/Kapalı” şeklinde server devresinden kontrol edilebilir.
- Gerektiğinde server devresinden istemci devresinin kontrol ettiği cihazlar açılıp kapatılabilir.
- 128x64 pixel Grafik LCD’si kullanımı kolaylaştırır.
- Seri porttan istemci bilgileri eklenebilir/değiştirilebilir.

8.1.3. Sunucuda kullanılan grafik lcd yapısı

Sunucu modülünde KS0108 kontrolörlü bir grafik lcd kullanılmıştır. Pin çıkışları Şekil 8.3.'de gösterilmiştir.



Şekil 8.3. KS0108 GLCD pin çıkışları

LEDA(19) ve LEDK(20) pinleri aydınlatma için kullanılmaktadır. LEDA Vcc'ye, LEDK da GND' ye bağlanmalıdır. Bu iki pinin herhangi birinde 5-10ohm'luk bir direnç bağlanmalıdır. Bu bağlantı direkt olarak yapılabileceği gibi LEDK ucuna BC547 ya da benzeri bir transistor bağlanıp ışık ayarı PWM sinyali ile işlemciden yapılabilir.

-Vout(18) pini, Vo pini için gerekli olan negatif gerilimi üretir. Bu pin bir potansiyometre ile Vo pinine bağlanmalıdır.

CS1(15) ve CS2(16) pinleri LCD'nin kontrolörünü seçmek için kullanılır. Bu LCD'nin sağ ve sol tarafını kontrol eden iki adet kontrolör vardır. LCD'ye bilgi gönderilirken ilk önce bu pinlerden LCD'nin hangi tarafına bilgi gönderileceği seçilmelidir.

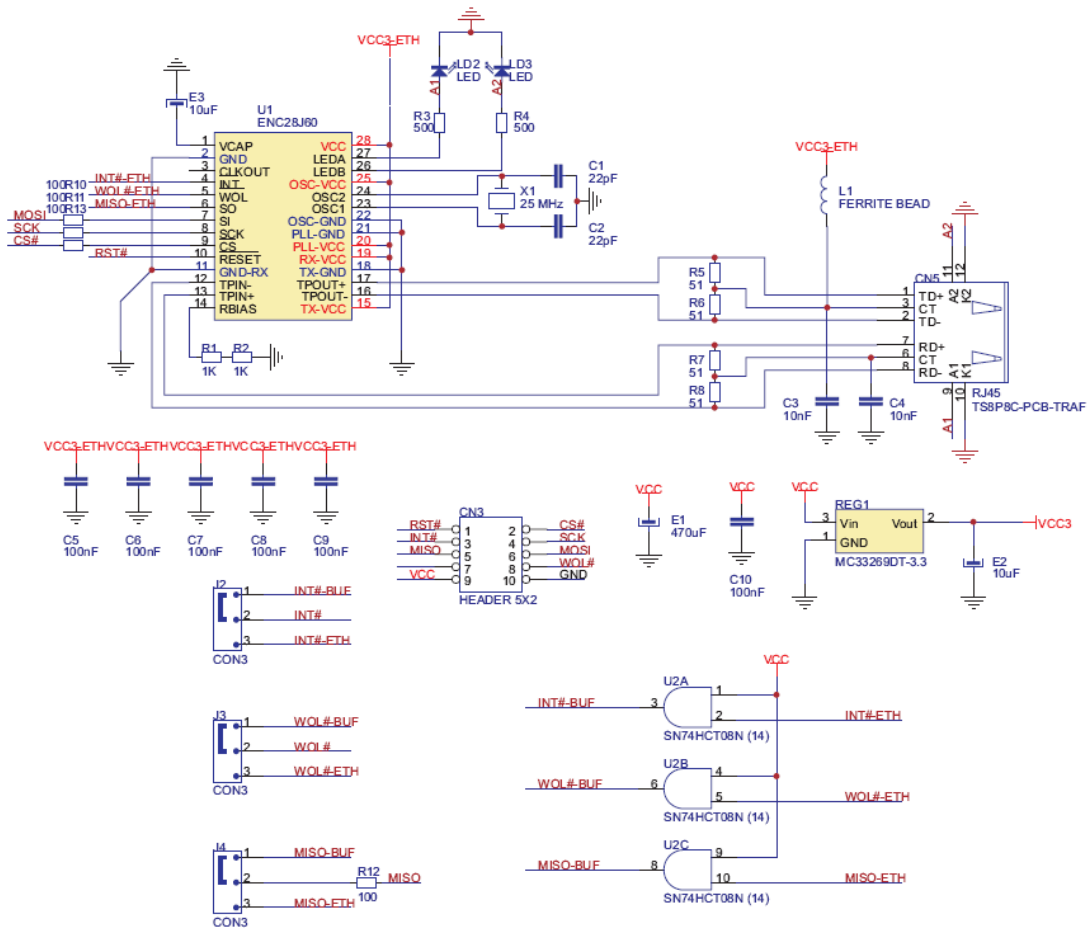
DB0-DB7(7-14) pinleri veri pinleridir. LCD'ye data veya komut gönderilirken bu pinler üzerinden paralel olarak gönderilir.

EN(6) pini LCD'ye komut ya da veri gönderilirken verinin hazır olduğunu LCD'nin kontrolörüne bildirmek için kullanılır. Bu pin "0" a düşürüldüğünde data pinlerindeki bilgi RS-DI pininin durumuna göre data veya komut olarak LCD kontrolörü tarafından algılanır.

R/W(5), LCD'ye bilgi gönderme/okuma işlemini seçmek için kullanılır.

RS-DI(4), LCD'ye gönderilen bilginin ekrana yazılacak bir bilgi mi yoksa komut mu olduğunu belirtir.

Vo(3), LCD için kontrast ayarını yapmak için kullanılır. Bu pin için negatif gerilim gereklidir. Bunun için -Vout pini ile beraber kullanılmalıdır.



Şekil 8.5. ENC28J60 Ethernet arabirimi devresi

8.1.5. Buton ve led işlevleri

AŞAĞI(DOWN): Menü içerisinde imleci aşağı kaydırır. Sayfa sonunda ise sonraki sayfaya geçer (Eğer sonraki sayfa varsa).

YUKARI(UP): Menü içerisinde imleci yukarı kaydırır. Sayfa başında ise önceki sayfaya geçer (Eğer önceki sayfa varsa).

GİRİŞ(ENTER): Menü içerisinde imlecin işaret ettiği istemciyi seçer.

ÇIKIŞ(EXIT): Menü içerisinde bir istemci seçili durumda ise yani istemci bilgi ekranında ise bu menüden çıkılır ve ana kontrol ekranına dönlür.

SAYFA İLERİ(PAGE RIGHT): Eğer varsa sonraki sayfaya geçirir.

SAYFA GERİ(PAGE LEFT): Eğer varsa önceki sayfaya geçirir.

SETTING/RSMODE: Bu butonun iki ayrı vazifesi vardır.

Birincisi, cihaz kapalı iken bu butona basılarak cihaz açılırsa cihaz ayar moduna girer. Bu modda cihazın ip adresi, gateway adresi, subnet mask... gibi bilgileri değiştirilebilir.

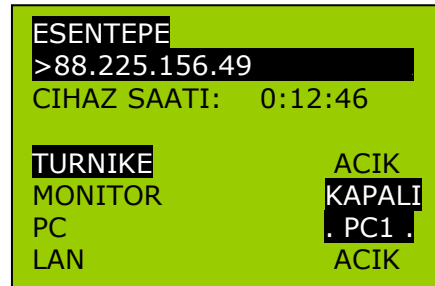
İkincisi, cihaz çalışır durumda iken bu butona basılı tutulursa cihaza seri porttan istemci bilgileri sorgulanabilir, eklenebilir, değiştirilebilir, silinebilir.

Status Led: MCU'nun çalıştığını gösterir. 1sn'lik periyot ile yanıp söner.

8.1.6. Grafik LCD ekranı



Şekil 8.6. Ana kontrol ekranı



Şekil 8.7. İstemci bilgi ekranı

8.1.6.1. Ana kontrol ekranı

Ana kontrol ekranı kayıtlı istemcilerin açık/kapalı durumunun kontrol edildiği ekrandır. Bu ekranda kayıtlı istemcilerin ip adreslerinin bağlantı durumları kontrol edilir. Bu kontrol yaklaşık 2sn'lik tekrarlar ile yapılır.

8.1.6.2. İstemci bilgi ekranı

İstemci bilgi ekranı seçilen istemciye ait isim, ip, cihaz saati ve bağlı cihazların açık/kapalı durumlarını gösteren ekrandır. Bu ekranda Turnike, Monitör ve LAN bağlantıları için açık/kapalı durumu PC içinde PC1/PC2 seçim durumu gösterilir. Bu ekranda Turnike, Monitör ve PC bağlantılarının durumları değiştirilebilir. Bunu yapmak için istenilen bağlantının üzerine imleç getirildikten sonra enter tuşuna basılır. Cihaz açık ise kapanır kapalı ise açılır. Örneğin MONITOR kapalı ise açmak için bir kere aşağı basılır. Bu durumda MONITOR yazısı siyah üzerine açık renk ile yazılı hale gelir. Bu durumda iken enter tuşuna basılması ile komut gönderilir. Komut istemci devrede işleme konduktan sonra bir sonraki kontrolde MONITOR bilgisi kapalı iken açık duruma geçtiği görünecektir. İstemci bilgileri yaklaşık 2sn'ik tekrarlar ile kontrol edilir.

Bu ekranda görünen cihaz saati şu şekilde yorumlanmalıdır:

00:12:46 → 00 = Gün bilgisi, 12 = Saat bilgisi, 46 = Dakika bilgisi

8.1.7. Seri port istemci bilgileri kayıt kontrol

Bu moda cihaz çalışır iken SETTING/RSMODE butonuna basılı tutularak girilir. Komutlar sadece bu butona basılı tutulduğu sürece işleme alınır. Yani eğer bu butona basılı tutulmadan seri porttan komut gönderilirse MCU komutu görmezden gelecektir.

Bu modda istemci bilgileri eklenebilir, değiştirilebilir, silinebilir, EEPROM'a format atılabilir(tüm bilgiler silinir), Max index bilgisi okunabilir, bir indexe ait istemci bilgisi istenebilir.

İstemci bilgisi, "İstemci adı" ve "IP" şeklinde kayıt edilir.

Not: (Hex bilgiler parantez içinde "0x" ile beraber yazılacaktır)

EKLE Komutu: EKLE>AKYAZI>88.164.247.36>(0x0A)

Format: EKLE>"İstemci Adı">"IP Adresi">(0x0A)

Bu komut EEPROM'da kayıtlı bulunan bölgenin sonuna AKYAZI istemcisini ve IP adresini ekleyecektir.

Bu komutla beraber verilen parametreler yanlış olursa "YD(0x0A)" bilgisi seri porttan geri döner ve kayıt yapılmaz. Örneğin max istemci adı bilgisi 11 karakter olmalıdır. Bu sayının aşılması durumunda "YD(0x0A)" bilgisi döner. Bunun gibi IP formatının yanlış bir şekilde girilmesi durumunda da bu hata döner. (Örn: 125.121.2265... gibi) Bilgiler doğru bir şekilde kayıt edilmişse "OK(0x0A)" bilgisi geri döner.

DEĞİŞTİR Komutu: DGST>KAMPUS>88.164.234.13>2(0x0A)

Format: DGST>"İstemci Adı">"IP Adresi">"Index"(0x0A)

Bu komut, verilen indexteki istemci adını ver IP'sini değiştirir. Bilgiler doğru bir şekilde değiştirilmişse "OK(0x0A)" bilgisi geri döner.

İstemci adı çok uzun ya da IP bilgisi formatı yanlış ise "YD(0x0A)" bilgisi döner. Verilen index kayıtlı değil ise "IY(0x0A)" bilgisi döner.

FORMAT Komutu: FRMT>(0x0A)

Bu komut EEPROM'a biçimlendirme yapar ve tüm bilgileri siler. İşlem tamamlandıktan sonra "OK(0x0A)" bilgisi geri döner.

SİLME Komutu: SILX>(0x0A)

Bu komut kayıtlı bulunan son indexteki istemci bilgilerini siler. İşlem tamamlandıktan sonra "OK(0x0A)" bilgisi geri döner.

Max Index OKU Komutu: OKUI>(0x0A)

Bu komut kayıtlı bulunan istemci sayısını geri döner. Bu komuta cevap “5(0x0A)” şeklinde döner. “5” burada kayıtlı bulunan istemci sayısıdır (yani max index).

Index’e ait Bilgi OKU Komutu: OKUD>3(0x0A)

Format: OKUD>”Index”(0x0A)

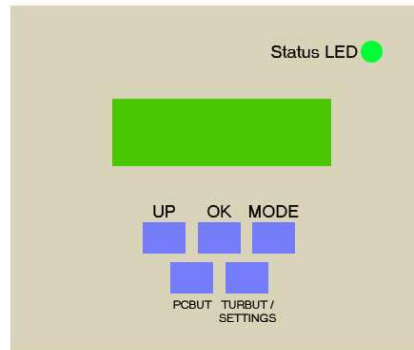
Bu komut parametre olarak girilen indexe ait istemci bilgilerini geri döner. Bu komuta cevap “İstemci adı”>“IP adresi”(0x0A) şeklinde döner. (Örn: ESENTEPE>88.225.156.49(0x0A))

Girilen index bilgisinin yanlış olması (hafızada olmaması) durumunda “TY(0x0A)” bilgisi geri döner.

8.2. İstemci Devresi Kullanımı

8.2.1. Tanım

Client(İstemci) devresi server devresinden gelen bilgileri işleme koyan devredir. Görevi, kontrol edilen cihazların durumlarını server(sunucu) devreye göndermek, sunucu devreden gelen komutları işleme almak, 4 farklı bağlantı noktasını (cihaz) kontrol etmektir.

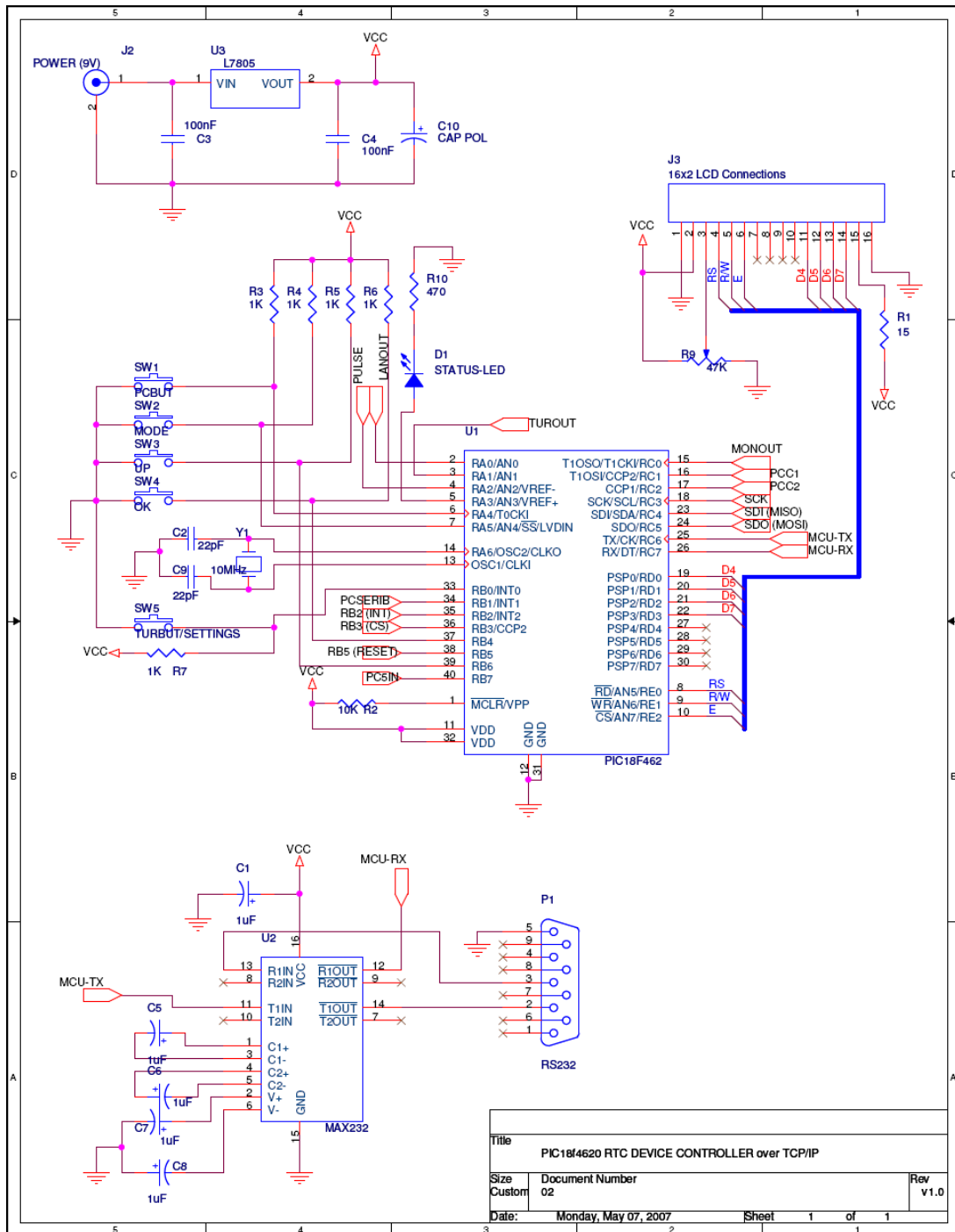


Şekil 8.8. İstemci modülü dış görünümü

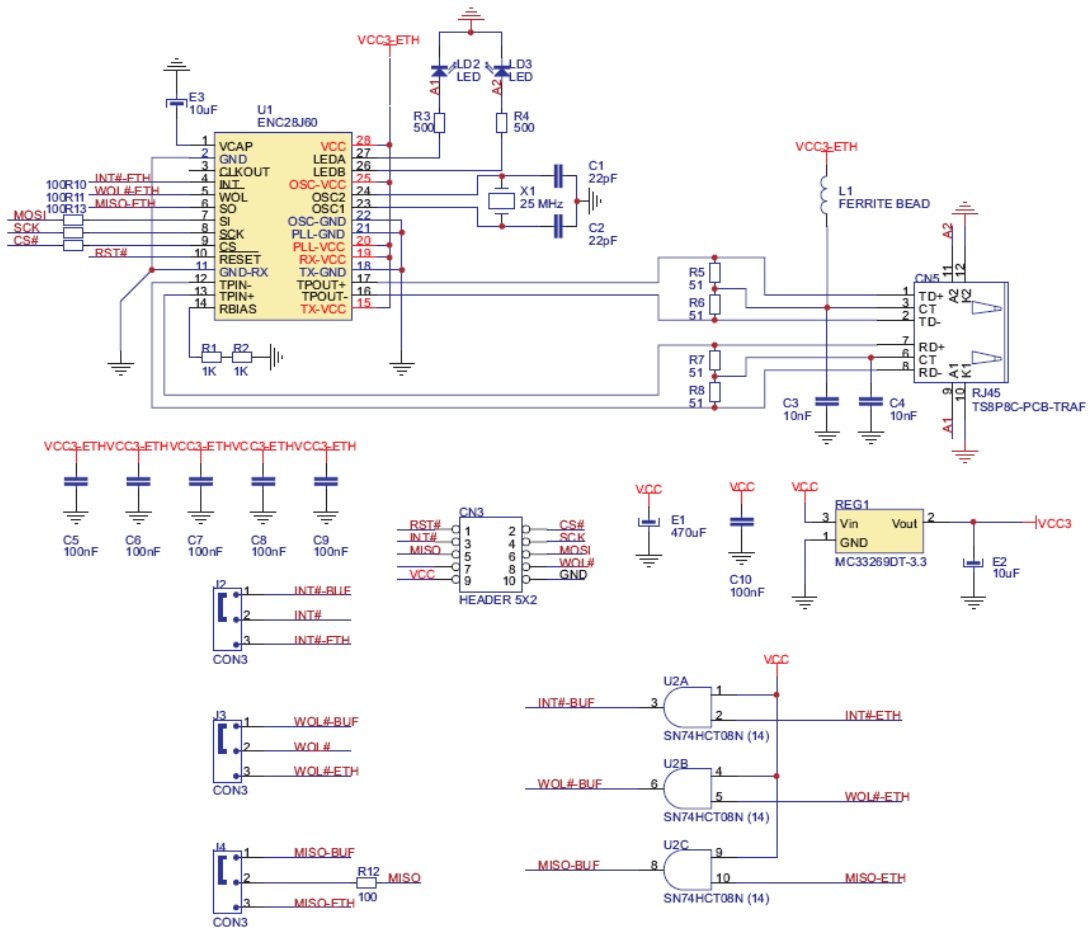
8.2.2. Özellikleri

- Sunucudan gelen komutları işler (Cihaz açma/kapama)
- Sunucunun yaptığı periyodik sorgulamalara cevap verir. (Cihaz durumlarını gönderir)
- Tuş takımından cihazların durumu değiştirilebilir.
- Cihazların durumunu 16x2 karakter LCD'si üzerinden gösterir.
- RTC(Real Time Clock) ile saat gösterir
- Program ve Saat ayarı yapılabilir.
- Periyodik LAN kontrolü yapar.
- 7 gün için her güne 3 ayrı LAN açma/kapatma ve pulse zamanı programlanabilir.

8.2.3. İstemci modülü devre şeması



Şekil 8.9. İstemci modülü devre şeması



Şekil 8.10. ENC28J60 Ethernet arabirimi devresi

8.2.4. İstemci ekranı

GN:00KP	TR:L
02:45:16	PC:1

Şekil 8.11. İstemci ekranı

Bu ekranda “GN:00” ile gösterilen bilgi gün bilgisi, “KP” ile gösterilen bilgi monitörün durumu (Açık: AC / Kapalı: KP), “TR:L” ile gösterilen bilgi Turnikenin durumu, “02:45:16” olarak gösterilen bilgi sırasıyla saat:dakika: saniye bilgisi, “PC:1” ile gösterilen bilgi de hangi PC’nin seçili olduğunu gösteren bilgidir.

8.2.5. Buton ve led işlevleri

PCBUT: PC seçimini değiştirir. PC1 ise PC2, PC2 ise PC1 i seçer.

UP: Saat ayarı ya da program ayarı yaparken aktif olan bölümü 1 arttırır.

MODE: Bu butona basılı tutulursa program ayarlama ekranına geçilir. Diğer bir özelliği program ya da saat ayarı yaparken gün-saat-dakika seçenekleri arasında geçişi sağlar.

OK: Bu butona basılı tutulursa saat ayarı ekranı açılır. Diğer bir özelliği program ya da saat ayarı yaptıktan sonra işlemi tamamlar ve ana programa dönüşü sağlar.

TURBUT/SETTING: Bu butonun iki ayrı vazifesi vardır.

Birincisi, cihaz kapalı iken bu butona basılarak cihaz açılırsa cihaz ayar moduna girer. Bu modda cihazın ip adresi, gateway adresi, subnet mask... gibi bilgileri değiştirilebilir.

İkincisi, cihaz çalışır durumda iken bu butona basılırsa Turnike çıkışı açık ise kapanır, kapalı ise açılır. Bu durum ekrandan da izlenebilir.

Status Led: MCU'nun çalıştığını gösterir. 1sn'lik periyot ile yanıp söner.

8.2.6. Periyodik LAN kontrolü

Bu özelliği ile istemci devresi her saat başı programda belirlenen süre kadar LAN çıkışını açık tutar. Diğer zamanlarda LAN çıkışı kapalıdır.

BÖLÜM 9. SONUÇLAR

Çoğunlukla tasarlanan internet tabanlı kontrol sistemlerinde uzaktaki sistemlerin kontrolü, bilgisayar üzerinden bilgisayarın dışarı açılan seri port ya da paralel port gibi kapıları kullanılarak yapılmaktadır. Bu çalışma ile bilgisayar olmadan internet üzerinden uzaktaki bir sistemin kontrolünün sağlanabildiği somut bir çalışma ile gösterilmiş oldu.

Tasarlanacak bir kontrol sistemi için bilgisayar kullanılmadan bir sistemin uzaktan kontrol edilebilmesi, tasarlanan sistemin maliyetini büyük ölçüde düşürmüştür.

Sistemde TCP protokolü kullanılması, sistem üzerinde akan verinin iletimini garanti etmiştir. Böylece veri güvenliğinden endişe edilmemesi sağlanmıştır. Bu da veri güvenliğinin önemli olduğu kontrol uygulamalarında bu sistemin rahatlıkla kullanılabileceğini göstermektedir.

Teorik olarak yapılan incelemelerde CAN Bus'ın veri iletiminin eş zamanlı olduğu görülmüştür. Buna karşılık Ethernet üzerinden veri iletiminde mesafeye göre değişen zaman farkları oluşmaktadır.

CAN Bus, veri iletimi güvenliği yüksek olmasına ve modüller arası eş zamanlı veri iletimi yapmasına rağmen, sistemimiz için uygun görülmemiştir. Çünkü CAN Bus veri yolu hızı mesafeye bağlıdır ve bizim sistemimiz için gerekli olan mesafe için CAN Bus kesinlikle kullanılamaz. Ancak gürültüye maruz kalabilecek, modülleri arasında uzun mesafe olmayan mikroişlemcili sistemler için kullanılabilir. Özellikle otomotiv sektöründeki uygulamalarda modüller arası iletişimde eş zamanlı iletişim önemli olduğundan ve araç içinde uzun mesafeler söz konusu olmadığından CAN Bus daha verimlidir.

Ethernet ile CAN Bus arasında teorik olarak yapılan karşılaştırmalarda Ethernet'in CAN Bus'a göre sistemimize daha uygun olduğu anlaşılmıştır. İki ağ yapısını düşündüğümüzde Ethernet, CAN Bus'a göre daha uzun mesafeler desteklemekte, daha yüksek hızlara çıkabilmekte ve bant genişliği sayesinde aynı anda daha çok cihaz bağlantısı sağlayabilmektedir.

CAN Bus'ın veri güvenliği yüksek ve mesafeye bağımlı olarak değişen hız yapısına karşı, Ethernet'in mesafe limiti olmayan ve dünya üzerinde standart olan TCP, UDP gibi protokolleri desteklemesi Ethernet'i sistemimiz için ideal yapmıştır. Ethernet'in bu özelliği, sistemimizin tüm dünyada internet üzerinden kullanılabilir olmasının sebebidir.

ENC28J60'ın SPI protokolü üzerinden kontrol edilebilmesi, devre üzerinde kullanılan pasif elemanlarının azlığı, az pin sayısı sayesinde diğer Ethernet arabirim entegrelerine göre pcb üzerinde daha az yer kaplaması ve bu kadar basit yapısına karşı performansı ile ENC28J60, sistemimiz için en uygun Ethernet arabirim entegresi olmuştur.

Kampus otomasyon sistemi için kurulan otomasyon ağında sistem TCP protokolü ile tamamen internet üzerinden çalıştığından herhangi bir yerdeki herhangi bir istemciye bağlı cihazlar sunucu tarafından gözlenebilir ve kontrol edilebilir olmuştur. Tasarlanan sistemde bir sunucu maksimum 40 istemciyi kontrol etmektedir. Bu sebeple, 40 adetten fazla istemciyi kontrol etmek için daha fazla sunucu kullanılabilir.

Sunucu üzerinde kayıtlı olan istemci adları ve IP adresleri seri port üzerinden değiştirilebilir olduğundan, sistem bir kere kurulduktan sonra eğer istemci adreslerinde değişiklik yapılmak istenirse, sistem üzerindeki sunucu işlemcisinin tekrar programlanmasına gerek kalmamıştır.

İstemci modüllerin üzerinde yazılımsal bir RTC(Real Time Clock) yapısının olması bu modüller üzerindeki cihaz durumlarının programlanabilir olmasını sağlamıştır.

İstemci modül tasarımında Turnike, PC, Monitor ve LAN gibi çevresel donanımların hepsinin birden tek modül üzerinden kontrol edilebilir olması gereksiz modül kalabalığını gidermiştir. Böylece bir sunucu daha çok cihazı kontrol edebilir duruma gelmiştir.

BÖLÜM 10. TARTIŞMA VE ÖNERİLER

Çalışmada gerçekleştirilen kontrol sistemi için sistemde bir güvenlik algoritması kullanılmasına gerek görülmemiştir. Ancak veri gizliliğinin önemli olduğu uygulamalarda sistem için özel bir güvenlik algoritması tasarlanabilir ya da AES, 3DES gibi yüksek seviyeli bir güvenlik algoritması kullanılabilir.

Bu çalışmada maksimum 40 istemci için kontrol yapılabilmektedir. İstemci adresleri EEPROM hafızası üzerinde tutulmaktadır ve 40 istemci adresi EEPROM hafızasının tamamını kaplamamaktadır. EEPROM hafızası tamamen kullanıldığında her sunucu 64 adet istemciyi kontrol edebilir duruma getirilebilir.

Sistemde veri güvenliği öncelikli olduğundan TCP protokolü kullanılmıştır. Ancak TCP protokolünün yapısından dolayı veri akış hızı yavaş olmaktadır. Bizim sistemimiz için bu problem olmamaktadır. Ancak yüksek hız gerektiren uygulamalarda UDP protokolü kullanılabilir. Bu durumda da iletilen verinin doğru iletilmişliğinin garantisi olmayacaktır. Çünkü UDP protokolü ile yapılan iletişimde veri gönderildikten sonra karşı taraftan bir onay alınmaz. Yani iletişimde bir kontrol yoktur. Oysaki TCP protokolünde checksum kontrolü yapılır. Checksum veri paketi içindeki bilgilerden hesaplanır. Checksum hatalı olduğunda karşıdan NACK cevabı gelir ve veri tekrar gönderilir. Bu hata kontrolü veri iletimi durumunu garanti etmektedir.

Sistem bilgisayarların otomasyon ağına katılmasıyla genişletilebilir. Bu durumda gerekli ayarlar yapıldığında sistem aynı zamanda bilgisayar üzerinden de kontrol edilebilir duruma gelecektir. Ancak gerekli bütün kontroller sunucu modülü üzerinden yapılabildiğinden ve sistem maliyetini yükseltmemek için buna gerek

görülmemiştir. Yine de ileride gerekli görülürse sistemimiz bilgisayarlar ile desteklenebilir. Bu durumda bilgisayar üzerinden kontrol edilebilecek istemci sayısı limiti olmayacaktır. Sisteme bilgisayarlar entegre edilirken, aynı istemciyi birden fazla sunucu modül ya da bilgisayarın kontrol etmemesi sağlanmalıdır.

Bir sunucu 40 istemciyi kontrol edebilmektedir. Tasarlanan otomasyon sisteminde yapı tek sunuculu olarak düşünülmüştür. Ancak gerektiğinde sunucu sayısı artırılarak ağ yapısı büyütülebilir. Bu durumda kontrol edilebilen cihaz sayısı artacaktır.

İleriye yönelik çalışmalarda, sisteme değişik kontrol yapıları eklenebilir. Bunlar arasında GSM, SMS, GPRS, DTMF gibi uzaktan kontrol yapılmasını sağlayacak sistemler düşünülebilir. Bu sayede mobile ve sabit telefonlar ile sistemin kontrolü sağlanmış olacaktır. Ancak sistem maliyetini arttıracak, SMS ücreti, data ücreti, telefon arama ücreti gibi ek ücretler sisteme eklenecektir.

KAYNAKLAR

- [1] HENDREY,G.R., KOOPMAN, P., Standard Ethernet as an Embedded Communication Network, Project Report For the Degree Of Master Of Science, *Carnegie Mellon University, Department of Electrical and Computer Engineering*, Nisan 1999, s24-29.
- [2] <http://www.microchip.com>, “Ethernet Overview”, “ENC28J60 Data Sheet (DS39662b)”, s31-32
- [3] http://en.wikipedia.org/wiki/Controller_Area_Network, “Controller Area Network”
- [4] <http://www.bilgiportal.com/v1/idx/18/322/makale/Ethernet-ile-İlgili-Temel-Bilgiler.html>, “Ethernet ile İlgili Temel Bilgiler”
- [5] <http://www.kvaser.com/index.htm>, “The CAN Protocol Tour”

ÖZGEÇMİŞ

Ahmet Gülođlu, 18.08.1982 de İstanbul' da doğdu. İlk, orta ve lise eğitimini K.Çekmece'de tamamladı. 1999 yılında K.Çekmece Lisesinden mezun oldu. 1999 yılında Sakarya Üniversitesi, Elektrik-Elektronik Mühendisliği Bölümüne girdi ve 2003 yılında mezun oldu. Şu anda 2005 yılında girdiđi İstanbul Elektronik Anahtar firmasında ARGE mühendisi olarak görev yapmaktadır.