

**MİKRODENETLEYİCİLİ ASANSÖR DENETİMİNDE SERİ HABERLEŞME
KULLANAN BİR MODELİN GERÇEKLEŞTİRİLMESİ**

Abdullah ORMAN

**YÜKSEK LİSANS TEZİ
(ELEKTRONİK VE BİLGİSAYAR EĞİTİMİ)**

**GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

Temmuz 2005

ANKARA

Abdullah ORMAN tarafından hazırlanan MİKRODENETLEYİCİLİ ASANSÖR DENETİMİNDE SERİ HABERLEŞME KULLANANAN BİR MODELİN GERÇEKLEŞTİRİLMESİ adlı bu tezin Yüksek Lisans tezi olarak uygun olduğunu onaylarım.

Yrd.Doç.Dr. O.Ayhan ERDEM
Tez Yöneticisi

Bu çalışma, jürimiz tarafından Elektronik ve Bilgisayar Eğitimi Anabilim Dalında Yüksek Lisans tezi olarak kabul edilmiştir.

Başkan :Prof.Dr. Ömer Faruk BAY

Üye :Yrd.Doç.Dr. M.Timur AYDEMİR

Üye : Yrd.Doç.Dr. O. Ayhan ERDEM

Bu tez, Gazi Üniversitesi Fen Bilimleri Enstitüsü tez yazım kurallarına uygundur.

**MİKRODENETLEYİCİLİ ASANSÖR DENETİMİNDE SERİ HABERLEŞME
KULLANAN BİR MODELİN GERÇEKLEŞTİRİLMESİ
(Yüksek Lisans Tezi)**

Abdullah ORMAN

GAZİ ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

Temmuz 2005

ÖZET

Bu çalışmada, seri iletim hatlı mikrodenetleyici denetimli asansör tasarımı ve prototipi gerçekleştirildi. Mevcut asansör haberleşme sistemlerinde kullanılan kablolama şekillerine alternatif olarak seri iletişim prortokolünün kullanılması sağlayan bir tasarım gerçekleştirildi. Tüm asansör sistemlerinde kullanılabilmesi amacıyla farklı standartlarda olan asansörler içinde farklı programlar yazıldı. Bu programlar PIC C de yazılmıştır. Tasarımı gerçekleştirilen devre, laboratuvar ve gerçek ortamda test edildi. Tasarlanan devrede çok küçük değişiklikler yapılarak herhangi iki sistem arasında seri iletişim yapılması mümkündür.

**Bilim Kodu : 705
Anahtar Kelimeler : Seri İletim, RS 232, Asansör, Mikrodenetleyici
Sayfa Adedi : 78
Tez Yöneticisi : Yrd.Doç.Dr. Ayhan ERDEM**

**IMPLEMENTATION OF A MICROCONTROLLED ELEVATOR MODEL
UTILIZING SERIAL DATA TRANSMISSION**

(M.Sc. Thesis)

Abdullah ORMAN

GAZI UNIVERSITY

INSTITUTE OF SCIENCE AND TECHNOLOGY

July 2005

ABSTRACT

In this thesis a microcontrolled elevator model utilizing serial data transmission study has been implemented. The implemented design provides an alternative for the complex wiring systems required for communication in the existing elevators. Different programs have been generated for the elevators with different standards so that this system can be used in all elevator systems. The codes have been written in C language for PIC programmers. The designed circuit has been tested in the laboratory and in the real environment. It is possible to use this system for the communication of any two systems with slight modifications.

Science Code : 705
Key Words :Serial transmission, Rs232, Elevator, Microcontroller
Page Number : 78
Adviser : Assist.Prof.Dr. O. Ayhan ERDEM

TEŞEKKÜR

Çalışmalarım boyunca değerli yardım ve katkılarıyla beni yönlendiren Hocam Sn. Yrd.Doç.Dr. O. Ayhan ERDEM'e, Gazi Üniv. M.M.F Bilgisayar Mühendisliği Bölüm Başkan Yardımcısı Sn. Yrd.Doç.Dr.M. Ali AKCAYOL'a, teknik desteğini esirgemeyen Karsel Asansörleri çalışanlarına, Sn. Mak.Müh Recep KARHAN'a, manevi destekleriyle beni hiçbir zaman yalnız bırakmayan çok değerli eşim Rukiye ORMAN'a ve aileme teşekkürü bir borç bilirim.

İÇİNDEKİLER

	Sayfa
ÖZET	iii
ABSTRACT	iv
TEŞEKKÜR.....	v
İÇİNDEKİLER	vi
ÇİZELGELERİN LİSTESİ.....	viii
ŞEKİLLERİN LİSTESİ.....	ix
RESİMLERİN LİSTESİ.....	x
SİMGELER ve KISALTMALAR	xi
1. GİRİŞ	1
2. ASANSÖR DENETLEYİCİ SİSTEMLERİ.....	9
2.1. Klasik Sistemde Asansör Elektrik Tesisatı	13
2.2. Seri Veri İletimi.....	14
2.3. Veri İletim Hatları	14
2.4. Senkron – Asenkron Veri İletimi.....	15
2.4.1. Senkron ile asenkron veri iletiminin Karşılaştırılması	16
2.5. RS-232'nin Seri İletişimin Temel Kavramları.....	16
3. GERÇEKLEŞTİRİLEN SERİ İLETİM HATLI ASANSÖR DENETLEYİCİ SİSTEMİ	19
3.1. Besleme Katı	24
3.2 .RS232 Seviye Dönüştürme Katı	24
3.3. Kontrol ve Gösterge Katı	27
3.4. Girişler Katı	28

3.5. Mikrodenetleyici Katı.....	30
3.6. Tasarımı Gerçekleştirilen Asansör Prototipi.....	37
4. SONUÇ VE ÖNERİLER.....	40
KAYNAKLAR	42
EKLER	44
Ek-1. Kontrol Kartı Programı	45
Ek-2. Kabin Kartı Programı	52
Ek-3. 74LS164'ün bağlantı şeması ve fonksiyon tablosu.....	61
Ek-4. Prototipe Ait Resimler.....	62
Ek-5. Baskı Devre Şemaları.....	64
ÖZGEÇMİŞ.....	67

ÇİZELGELERİN LİSTESİ

Çizelge	Sayfa
Çizelge 2.1. Kat Kablo İlişkisi	13

ŞEKİLLERİN LİSTESİ

Şekil	Sayfa
Şekil 2.1. Asansör Motoru ve Asansör Boşluğu	10
Şekil 2.2. Kuyu içi görüntüsü.....	12
Şekil 2.3. Seri veri iletimi.....	14
Şekil 2.4. DB25S konnektör	17
Şekil 2.5. Kayıt Dosyası Haritası.....	17
Şekil 2.6. RS-232 şeması ve bacak isimleri	18
Şekil 3.1. Birebir bağlantı ve seri bağlantı	19
Şekil 3.2. Asansör kontrol devresi blok diyagramı.....	21
Şekil 3.3. Asansör Elektrik Tesisatı Blok Diyagramı.....	23
Şekil 3.4. Besleme devresi.....	24
Şekil 3.5. RS232 seviye dönüştürme devresi.....	25
Şekil 3.6. ASCII A karakterinin TTL/CMOS ve RS232 grafikleri.....	26
Şekil 3.7. Kontrol ve gösterge katı	27
Şekil 3.8. Kabin kartındaki mikrodenetleyicinin devre şeması.....	31
Şekil 3.9. Kontrol kartındaki mikrodenetleyicinin devre şeması	32
Şekil 3.10. Kabin programı akış şeması.....	33
Şekil 3.11 Kontrol programı akış şeması	34
Şekil 3.12 Asansör Prototipi.....	39

RESİMLERİN LİSTESİ

Resim	Sayfa
Resim 3.1.a Birebir bağlantı sitemi	20
Resim 3.1.b Seri bağlantı sistemi.....	20
Resim 3.2. Kabin kartının görüntüsü.....	35
Resim 3.3 Kontrol kartının görüntüsü.....	36
Resim 3.4. Adım motor kontrol devresi	37
Resim 3.5. Asansör kumanda kartı	38

SİMGELER ve KISALTMALAR

Bu çalışmada kullanılmış bazı simgeler ve kısaltmalar, açıklamaları ile birlikte aşağıda sunulmuştur.

Simgeler	Açıklama
C	Kondansatör (F)
D	Diyot
f	Frekans (Hz)
µsn	mikro saniye
µA	mikro amper
mA	mili amper
MHZ	mega hertz
Kısaltmalar	Açıklama
GND	Toprak (Ground)
LCD Display	Sıvı Kristal Gösterge (Liquid Crystal Display)
LSB	En Az Değerli Bit (Least Significant Bit)
MSB	En Değerli Bit (Most Significant Bit)
EEPROM	Silenebilir ve programlanabilir hafıza
RAM	Rastgele erişimli hafıza
MCLR	Master clear (Silme)
COM1	PC'nin 1. seri portu
I / O	Giriş / Çıkış

1. GİRİŞ

Asansörler , teknolojinin, insanlığın hizmetine sunduğu bir hizmet aracıdır ve şu anda asansörler günlük hayatta insanların işini kolaylaştıran temel araçlardan bir tanesi hale gelmiştir(1).

İnşaatlarda yük kaldırmak için mekanik araçlar kullanılması Roma dönemine dayanır. MÖ I. yüzyılda yaşayan Romalı mimar Vitruvius yazmış olduğu kitabında, insan, hayvan ya da su gücüyle çalışan makaralar ve bocurgatların hareket ettirdiği kaldırma platformları tarif etmiştir. Kaldırma makinelerine ait ilk örneklerden bir tanesi de Asurlular zamanından kalma bir kabartma taş üzerindeki resimde görülen, ağır bir taş heykelin taşınmasına ait modeldir. İlk ahşap proje ise Alman George Agricola'nın 1556 yılında basılı 12 ciltlik "Der Metallica" adlı kitabı ve modelidir. Rönesans devri bilim adamı Leonardo Da Vinci'nin(1452-1519) "Storia Della Tecnica" adlı kitabında kaldırma makinalarına ait projeleri asansör modellerinin temellerini oluşturur. Bu tür aygıtlar 1800'lü yıllarda İngiltere de buhar gücüyle çalıştırılmaya başlandı. İlk defa 19. yy. başlarında hidrolik bir asansör geliştirildi. Bu asansör, asansör boşluğunun altında toprağa gömülmüş bir silindirin dalma pistonuna bağlı olan bir platformdan oluşmaktaydı. Toprağa gömülü silindirin derinliği, asansör boşluğunun boyuna eşitti. Silindirdeki akışkana basınç uygulamak için de buharlı bir pompadan yararlanılıyordu. Sonraları asansörün hızını artırmak ve dalma pistonunun uzunluğunu azaltmak amacıyla makara sistemi kullanılmaya başlandı. Tüm bu araçlarda asansörün ağırlığını dengelemek için karşı ağırlıklar kullanılıyordu, dolayısıyla da taşıyıcıya konan yükü kaldıracak bir kuvvet uygulamak yeterliydi(1).

1850'lerin ortalarına değin bu çalışmalar daha çok yük asansörlerine yöneltildi. O dönemde genellikle kenevirden yapılan halatların yeterince güvenilir olmayışı nedeniyle asansör yolcu taşımada kullanılmıyordu. ABD'li Elisha Grawes Otis 1853'te geliştirdiği bir güvenlik aygıtıyla, yolcu taşıyan asansörlerin yapılmasını olanaklı kıldı. New York'taki Crystal Palace

sergisi'nde tanıtılan bu aygıtın askı halatlarındaki gerginliğin boşalması durumunda kabinin üzerinde hareket ettiği rayları sıkıca kavrayan mengeneleri vardı. İlk yolcu asansörü 1857'de New York'ta, Haughwout Büyük Mağazası'nda hizmete girdi. Buhar gücüyle çalışan ve bir dakikadan daha kısa bir sürede beşinci kata çıkabilen bu asansör önemli bir başarıydı(1).

Bunu izleyen 30 yıl içinde buhar gücüyle çalışan daha başarılı asansörler yapıldıysa da, 1880'lerin ortalarına değin önemli bir teknolojik bir gelişme olmadı. Bu tarihten sonra asansörlerde elektrik motoru kullanılmaya başlandı ve 1889'da da ticari amaçlı ilk elektrikli yolcu asansörü yapıldı. New York kentinde Demarest Binası'nda kurulan bu elektrikli asansör, binanın zemin katına yerleştirilen elektrik motorunun, halat sarılı bir makarayı çevirmesiyle hareket ediyordu. Elektrik kullanılmaya başlaması beraberinde yeni gelişmelerin ortaya çıkmasına yol açtı. 1894'de düğmeli kumanda sistemi geliştirildi, 1895'te de İngiltere de, taşıma kuvvetini asansör boşluğunun tepesine yerleştirilen bir makaraya aktaran yeni bir kaldırma aygıtı gerçekleştirildi. Bu sistemde, asansörün ve karşı ağırlığın ağırlıkları asansörün hareket etmesine yetiyordu. Halat sarma esasına dayalı olarak çalışan makara mekanizmasının yetersizliklerini aşan bu türden tahrik makinelerinin kullanılmaya başlanmasıyla daha yükseğe çıkan ve daha hızlı hareket eden asansörlerin yapılması olanaklı duruma geldi. 1904'te tahrik makinesinin doğrudan elektrik motorunun armatürüne bağlanmasını sağlayan dişlisiz bir aktarma yönteminin geliştirilmesiyle de hız sınırlamasından kurtulma olanağı doğdu (1).

Güvenlik, hız ve yükseklik sorunlarının çözülmesinden sonra asansörün kullanım kolaylığını arttırma ve maliyetini düşürme sorunları gündeme geldi. 1915'te otomatik kat ayarı adı verilen bir buluş kullanıma girdi. Asansörü kullanan kişi, bir katın kapısına yaklaştığında elle kumandayı kesiyor, her katta bulunan otomatik aygıtlar kumandayı devralarak asansörü tam kat hizasında durduruyordu. Kapılarda elektrik ile çalışır duruma getirildi.

Binaların yükseklikleri arttıkça, asansörlerin hızları da arttı. 1931'de New York kentindeki Empire State Binası'nın üst katlara çalışan asansörlerinde dakikada 365 metrelik, 1975'te Chicago'da John Hancock Merkezinin asansörlerinde dakikada 594 metrelik hızlara ulaşıldı(1).

Ekonomik olması nedeniyle hastanelerde ve çok katlı yapılarda yaygın olarak kullanılan otomatik çalışma sistemi, toplamalı komuta mekanizmasını geliştirilmesi ile daha da kolay kullanımlı hale getirdi. Bu mekanizma, bir yada birkaç asansörün, komutları yerine getirirken aşağıdan yukarıya yada yukarıdan aşağıya doğru bir sıra gözetmesini sağlıyordu. Bütün asansörlerde ortak olan temel bir güvenlik ögesi de, asansörün hareket edebilmesi için önce asansör boşluğu kapısının kapatılmasını gerektiren kilit sistemiydi. 1950'ye gelindiğinde, asansör operatörlerini ve motor çalıştırmalarını gereksiz kılan otomatik sistemler hizmete girmişti.

Yüksek yapılarda asansör donanımının kapladığı hacmi en aza indirmek amacıyla atılan ilk adım, çift katlı asansör yapımı oldu. İlk kez 1932'de denenen bu tür asansörlerde üst üste monte edilmiş iki kabin bulunuyordu ve böylece asansörün her duruşunda iki kata birden servis yapılıyordu. Bu tekniğin kullanımı giderek yaygınlaştı ve 1971'de ABD' de Chicago'daki Time-Life Binası'nda otomatik çift katlı asansörler çalışmaya başladı.

Modern asansörlerin alışılmış türlerinin yanı sıra gemilerde, barajlarda ve roket rampaları gibi çok özel yapılarda kullanılan değişik amaçlı türleri de vardır. Çok yüksek inşaat işlerinde, yukarıya ağır yük çıkaran ve aşağıya hızla inen asansörler bulunmaktadır. Tümü elektrikle çalışan bu asansörlerin kimisi halat, makara ve karşı ağırlık sistemiyle, kimisi halat saran kasnak mekanizmasıyla(birçok alçak kuleli yük asansörlerinde hala kullanılmaktadır) kimisi de elektrohidrolik bir sistemle hareket etmektedir. Çok sayıda halat kullanımıyla (üç yada daha çok) hem kaynağın çekme yüzeyi genişletilmekte hem de güvenlik arttırılmaktadır. Halat kopması çok seyrek rastlanan bir olaydır.

Tahrik motorunda düşük hızlar için genellikle alternatif akım, yüksek hızlar için doğru akım kullanılır. Doğru akım motorlarında kullanıldığı durumlar da hız denetimi, doğru akım üreticinin alan gücünün değiştirilmesi ve üreticinin armatürü ile tahrik motorunun armatürü arasındaki doğrudan bağlantının sağlanması yoluyla gerçekleştirilir. Yüksek hızlı asansörlerde dişlisiz sistemler kullanılır ve genellikle halatlar tahrik makarasının kasnağına iki kez sarılır. Tahrik kasnaklı çekme esası uyarınca çalışan asansörler için yükseklik sınırı yoksa da, yaklaşık 33 metreyi aşan yüksekliklerde dengeleme halatlarına ihtiyaç vardır. Bu halatların bir ucu asansör kabinin altına, öbür ucu da karşı ağırlığın altına bağlanır. Kabin yükseldikçe bu dengeleme halatının ağırlığı kabine biner, kabin alçaldıkça da halatın ağırlığı karşı ağırlığa kayar. Böylece tahrik makinesine binen yük hemen hemen sabit kalır(2).

Kısa yüksekliklerde çalışan yolcu asansörleri ile ağır yük asansörlerinde hidrolik silindirler ve dalma pistonlar kullanılır. Piston, silindir içindeki basınçlı yağın etkisiyle platformu yukarıya doğru iter. Kabini yükseltmek için gereken basıncı, yüksek devirli bir elektrikli pompa sağlar. Kabinin indirilmesi ise, silindirdeki yağı yedek bir depoya gönderen elektrikli vanaların çalıştırılmasıyla gerçekleştirilir. Bazı az rastlanan uygulamalarda özel tipte hidrolik silindir ve dalma piston sistemlerinden yararlanır. Uçak gemilerinde ağır yükleri kısa yüksekliklere kaldırmak için, 1900'lerin başlarında yaygınlaşan halatlı yada dişli tip hidrolik asansörler kullanılır. Bunlarda piston ve silindirin uçlarında kasnaklar bulunur. Pistona basınç uygulandığında kasnaklar arası uzaklık arttığından, bu kasnaklara sarılı halatlar asansörü yukarıya çeker.

Askı halatlarıyla yükseltilecek asansörlerde, zemin güvenlik fren aygıtının bulunması gerekir. Paraşüt olarak adlandırılan bu aygıt, devreye sokulduğu anda çelik kılavuz raylarına kelepçelenip kabini frenleyerek kısa sürede durdurur. Genellikle kabin zemininin altına takılan paraşüt, bir halat aracılığıyla harekete geçirilen hız regülatörü tarafından devreye sokulur.

Asansör normalin üstünde bir hızla alçalacak olursa, bu halat paraşüt komutasını çekerek çalıştırır. Paraşüt ilk aşamada motoru durdurur ve hız yeteri kadar azalmazsa da güvenlik frenini uygular.

Modern asansörlerin çoğu otomatiktir ve bunları tek tek ya da gruplar halinde çalıştırmak üzere çeşitli komuta sistemleri kullanılır. En eski otomatik komuta sistemi, asansörün yalnızca yolcusu tarafından kullanılmasına olanak tanıyan tek düğmeli otomatlardan oluşur. Bu sistem günümüzde de küçük toplu konutlarda ve yük asansörlerinde kullanılmaktadır.

Tek asansörlü binalarda toplamalı komuta sisteminin kullanımı da yaygınlaşmıştır. Bu sistemde kabin önce bir yöndeki komutların tümüne uyar ve daha sonra yön değiştirerek öbür yöndeki komutları yerine getirir. Toplamalı komuta çok katlı konutlarda, hastanelerde ve küçük büro binalarında kullanılmaktadır. İki taşıyıcılı asansörlerde, toplamalı komuta sistemi her iki kabinin birlikte çalışmasını ve çağrılarını paylaşmasını sağlar. İki ya da daha çok asansörün birden denetimini sağlayan grup toplamalı komuta sistemi asansörlerin çalışma düzenini belirli bir işlem aralığına göre zamanlar. Hastanelerde, büyük mağazalarda ve bürolardaki gibi yoğun bir trafik ve iki yada daha çok asansör bulunuyorsa, grup toplamalı komuta sistemleri kullanılır

Asansör boşluğu ve asansör kabini için ayrı ayrı kapıların kullanılması, modern asansör sistemlerinin temel özelliklerindedir. Her iki kapıda genellikle aynı biçimde çalışır. Kapıları, asansörün üzerinde bulunan bir elektrik motoru açıp kapatır. Kapının hızı, kapanma sırasında arada kalabilecek yolcuların zarar görmeyeceği biçimde ayarlanır. Eğer kapanma sırasında kapı bir nesneye çarpacak olursa, elektrikli bir algılayıcı kapının yeniden açılmasını sağlar. Kapının ters yöne hareketi için fotoelektrik algılayıcılar ve elektronik yakınlık aygıtları da kullanılmaktadır. İç kapılar kapanmadıkları sürece asansörün hareketine izin vermeyecek biçimde tasarımlanır (1).

Yük asansörlerinde, düşey doğrultuda açılıp kapanan iki parçalı sürme kapılar yaygındır. Bu kapılar alt ve üst kanatlardan oluşur. Kanatlar arasındaki mekanik bağıntı, alt kanat kabinin zemine kadar inerken, üst kanadın tavanına kadar yükselmesini sağlar. Bu asansörlerde de genellikle koruyucu bir iç kapı gereklidir(1).

Bazı ülkelerde, az sayıda insanın yararlandığı yerlerde, en çok da özel konutlarda, asansör ile yapı dışındaki bir santral arasında telefon bağlantısı bulunması yasa gereğidir. Çoğu yapılardaki asansörlerde, mekanik arıza durumunda kullanılmak üzere karşılıklı haberleşme sistemleri, alarm düğmeleri acil ışıklandırma ve yedek güç kaynakları bulunur(1).

Modern yük asansörlerinde otomatik yükleme boşaltma araçları vardır. Bir çağırma düğmesi otomatik taşıma işlemi başlatır; asansör gelir, yük kabinin içine çekilir, kabin istenen kata çıkar ve yük boşaltılır.

ABD’de New York kentindeki Dünya Ticaret Merkezinin ikiz yapılarında asansörlerin kapladığı alanı en aza indirmek için yeni bir teknik kullanılmıştır. 110 katlı bu iki yapının her biri üç bölüme ayrılmıştır. Yapını ilk 41 katından oluşan birinci bölüm için, her birinde 6 asansör bulunan 4 grup çalışmaktadır; benzer biçimde orta ve üst bölümlerde de, altışar asansörlü dörder grup vardır. 44. ve 78. katlarda birer “gökyüzü lobisi” bulunmakta ve bunların zemin katla ilişkisi 11 tane yüksek hızlı mekik asansörle sağlanmaktadır (1).

Son yıllarda yapıların dışına yerleştirilen saydam duvarlı asansörler ve gözlem kuleleri ilgi görmektedir. Bunların ilki Paris’teki, Eiffel Kulesi’nde (1889) kurulmuş olan asansördür. Bu gibi gözlem asansörlerinin dışarıyı seyretmeye olanak veren cam kabinleri vardır (1).

Günümüzde asansörler beş amaç için kullanılmaktadır. Bunlar; konut asansörleri, konut dışı asansörler (oteller, iş hanları, resmi binalar, fabrika ve

depo gibi yerlerde kullanılan), sađlık tesisi asansörleri (sedye, tekerlekli sandalye veya tıbbi malzeme taşıma amaçlı) yük asansörleri, servis asansörleridir.

Çađımızın en hızlı gelişen bilim dalı olan elektronik sahasında en önde gelen çalışmalar yarı iletken teknolojisindedir. Bu teknolojinin en gelişmiş ürünü olan mikroişlemciler 80'li yıllardan sonra iki ana dala ayrılarak üretilmektedir. Evrim ağacının bir kolunu kişisel bilgisayarların (PC) önemli bir parçası sayılan Pentium veya Power-PC mikroişlemci yongaları oluşturmaktadır. Bu alandaki gelişme tamamen ham verinin daha hızlı işlenmesini amaçlamaktadır. Diğer dalı ise üretim hacmi mikroişlemci üretim hacminin 30 katını bulan mikrodenetleyiciler teşkil etmektedir. Bu daldaki gelişmenin amacı tek bir yonga üzerine bir çok özelliğin sığdırılması ve bu sayede tek bir yonga kullanarak denetim amaçlı giriş/çıkış uygulamalarının yapılabilmesidir (4).

Bu tez asansör kabini ile ana kumanda panosu arasındaki iletişimi mikrodenetleyici denetimi ile seri iletişim protokolü kullanarak gerçekleştirmek amaçlanmıştır. Bu tez temel olarak 4 bölümden oluşmuştur.

Çalışmanın ikinci bölümünde asansör sistemleri, klasik asansör denetleme sistemleri, iletişim sistemleri ve asansör haberleşme sistemleri açıklanmıştır. Klasik asansör sistemlerindeki sıkıntılar üzerinde durulmuş ve çözüm önerileri getirilmiştir.

Çalışmanın üçüncü bölümünde tasarımı gerçekleştirilen devrenin blok diyagramı ve devreyi oluşturan tüm katlar açıklanmış, PIC C ortamında yazılan programa ait bilgiler verilmiştir. Tüm kartların yerleştirildiği asansör prototipi anlatılmıştır.

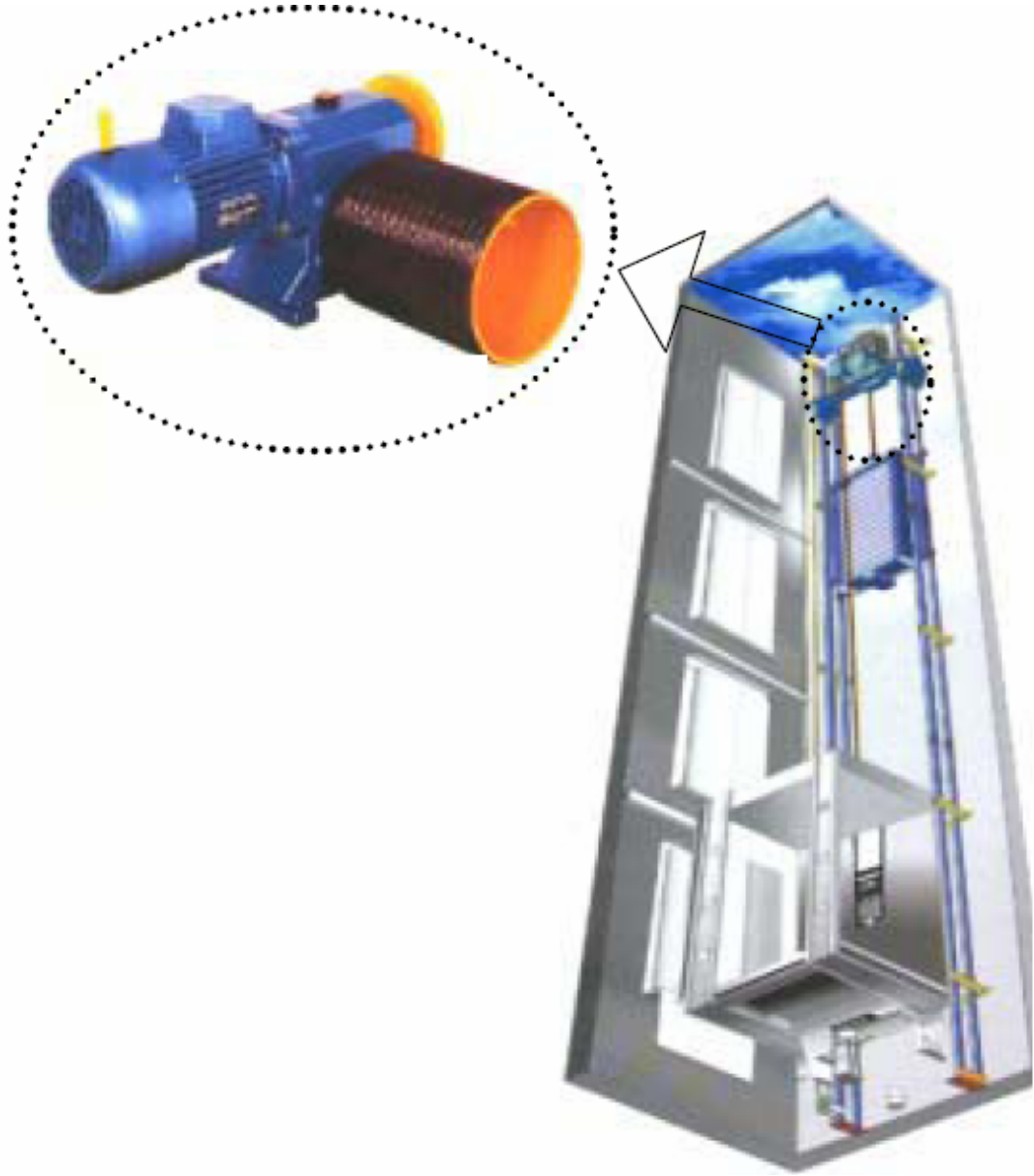
Dördüncü bölümde sonuçlar ve değerlendirmeler kısmı olup, yapılan çalışmanın performansı değerlendirilmiş ve çalışmanın geliştirilmesi için öneriler sıralanmıştır.

Ekler kısmında yapılan devrenin baskı devre görüntüleri, yazılan bilgisayar programının kaynak kodları, kullanılan mikrodenetleyicilerin programları ve devrede kullanılan bazı elemanlara ait teknik bilgiler verilmiştir.

2. ASANSÖR DENETLEYİCİ SİSTEMLERİ

Genellikle günümüzdeki asansörler makine dairesindeki tahrik motorunun kumanda panosundan aldığı komut vasıtasıyla harekete geçip, kabini istenilen yöne hareket ettirmesi yoluyla çalışır. Kabin, karşı ağırlıkla birlikte çalışır. Aralarındaki çelik halatın tahrik kasnağı ile sürtünmesinden kaynaklanan bir hareket aracılığıyla yukarı aşağı hareket ederler. Kabin ve karşı ağırlık yaklaşık eşit ağırlıktadırlar. Bu sebeple enerjiden tasarruf edilmiş olmaktadır. Bu tür tahrik makinelerine sürtünme tahrikli asansörler denir. Ayrıca halat veya zincirin tahrik kasnağı üzerine sarılması ile çalışan asansörlerde mevcuttur. Bu şekilde çalışan asansörlere de tamburlu asansörler denir. Asansörler taşıma kapasitesine göre türlere ayrılır. Konutlarda kullanılan asansörler; 100, 180, 240, 320, 400, 630, 1000 kg olmak 7 türe ayrılır. Konut dışı yerlerde kullanılan asansörler; 630, 800, 1000, 1200, 1600 kg olmak üzere 5 türe ayrılır. Sağlık tesislerinde kullanılan asansörler; 1600, 2000, 2500 kg olmak üzere 3 türe ayrılır. Asansörler hızlarına göre 0,63 m/s, 1 m/s, 1,6 m/s, 2,5 m/s ve 2,5 m/s'nin üzerindeki hızlar olmak üzere 5 çeşittir. Bunlara bağlı olarak asansörlerin kabin, makine dairesi ve kuyu boşluğu ebatları da değişiklik göstermektedir. Yine bunlara ek olarak asansörler hızlarına göre değişik güçte motor ve fren sistemi kullanılmaktadır. Regülatör düzeneği asansör halatlarının kopması durumunda devreye girip kabini frenleyen bir emniyet tertibatıdır (1).

Şekil 2.1'de modern bir asansörün bina içindeki kurulumu gösterilmiştir. Tipik bir asansör sisteminde asansör boşluğunda düşey raylar döşenir ve kabin bu raylara oturtulur. Raylı bir sistem üzerinde bulunan kabin, çelik halatlarla bina tepesindeki motora bağlanır. Kabinin kolayca hareket etmesini sağlayan ise motora bağlı bulunan ağırlıklardır. Motoru kontrol eden birim mikrodenetleyici veya PLC denetimli kumanda kartıdır. Kumanda kartı bilgileri yorumlayarak sıralama yapar. Asansör kabini bu sıralamaya göre ilgili katlarda durdurulur(2).

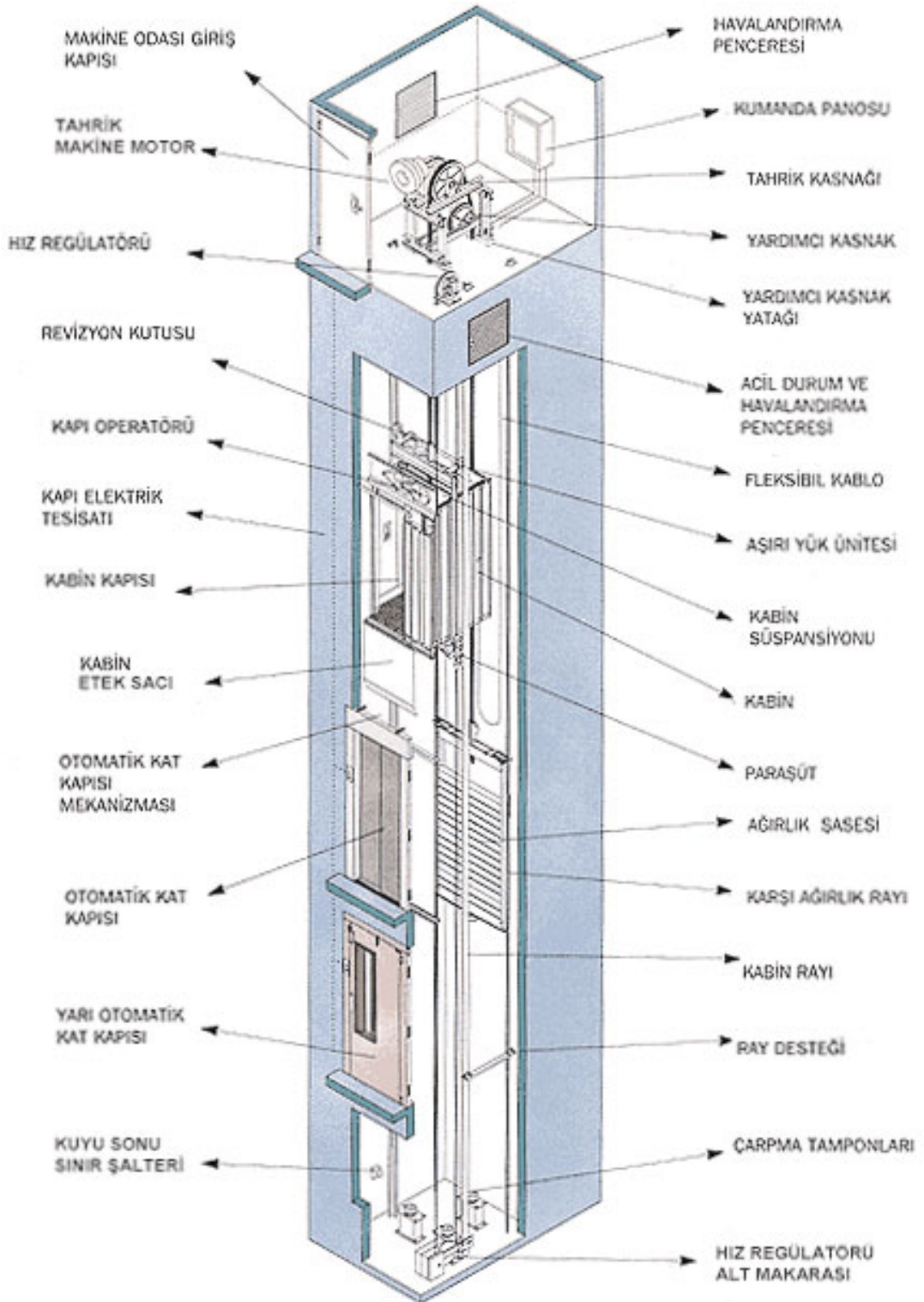


Şekil 2.1. Asansör Motoru ve Asansör Boşluğu

Kabin hız ayarı ise motorun hızının değiştirilmesiyle sağlanır. Motor hızı ise şebeke frekansı ile doğru orantılı ve kutup sayısı ile ters orantılıdır. Hız ayarları tek kademeli hız, iki kademeli hız veya frekansla ayarlanarak çok kademeli hız olarak yapılabilir. Fakat kademe arttıkça maliyet artacaktır(2).

Asansör kabin ile kontrol kartı arasındaki veri iletişimi iletim kablosu ile sağlanmaktadır. Ayrıca, kabine enerji de kablo ile sağlanmaktadır.

Kullanılmakta olan bu kablolar özel yapım kablolardır. Bu kabloların esneme katsayısı ve kuvvete dayanabilme oranları yüksektir. Fakat bu özellikler sisteme yüksek maliyet olarak geri dönmektedir. Ayrıca iletim kablolarının içindeki iletkenlerin kırılma ihtimalleri, ömürlerinin kısalığı ve kabloların belirli zaman aralıklarında yenilenmesi de dikkate alınırsa, verilerin kablolar ile iletimi performans, verim ve maliyet açısından istenmeyen durumlar oluşturmaktadır. Halen yürürlükte olan asansör yönetmeliğine(95/16/AT) göre bu kablolar 3 yılda bir değiştirilmelidir. Şekil 2.2'de asansörün kuyu içi görüntüsü görülmektedir (3).



Şekil 2.2. Kuyu içi görüntüsü

2.1. Klasik Sistemde Asansör Elektrik Tesisatı

Klasik sistemde, asansör elektrik tesisatı asansör kabini ile asansör boşluğunun tepesinde bulunan ana kumanda kutusu arasında kablolar vardır. Bu kablolar asansör boşluğunda sallanır. Boyları binanın katsayısı ile orantılıdır örneğin 5 katlı bir binada yaklaşık 15m dir. Kabloların sayısı 23 tane temel veri kablosu olmak üzere her kat için ek olarak 1 tanedir. Bu temel veriler durdurma, iç kapı, kilit, halat gevşeme kontağı, tampon, hız düzenekleri, imdat butonu, güvenlik sınır anahtarları, yük butonu ve revizyon anahtarlarıdır. İstenilirse bu sayı güvenlik önlemlerine göre veya maliyete göre değiştirilebilir. Çizelge 2.1'de katsayısına göre değişen kablo miktarları verilmiştir(2).

Çizelge 2.1. Kat kablo ilişkisi

Kat Sayısı	Kablo Sayısı
5	$23+5=28$
10	$23+10=33$
15	$23+15=38$

Asansör sisteminde görev yapan her algılama ünitesi (kasetler, aşırı yük, termistör) veya eylemci ünite (fren, kapı) en az iki veya daha fazla sayıda kablo ile kumanda tablosuna bağlıdır. İki ünite arasındaki bağlantıda kullanılması gereken kablo sayısı o iki ünite arasındaki bilgi akışının karmaşıklığına paralel olarak artar. Kablo sayısının artması öncelikle asansörün montaj safhasındaki zamanı uzatır. Asansörün kullanıma girmesinden sonra da hem elektrik arızalarının olasılığını artırır hem de servis elemanlarının arızayı bulmak için daha fazla zaman harcamalarına neden olur (3).

Yukarıda bir asansör örneği ile açıklanan elektrikselsel bağlantı sistemi birebir bağlantı sistemidir. Bu sistemde her fonksiyon veya bilgi iletişimi için bir kablo

bağlantısı gereklidir. Örneğin 30 katlı bir asansör kabininden sadece çağrılar tabloya iletmek için yaklaşık 30 kablo gereklidir. Birebir bağlantı sistemi özellikle mesafeler uzadığı zaman ciddi kablo maliyetlerine sebep olur.

2.2. Seri Veri İletimi

Seri veri iletimi, bir veri içindeki bitlerin, aynı hat üzerinden art arda gönderilmesidir. Bitlerden ilk önce gönderilen EKB (En Küçük Değerlikli Bit), en son gönderilen EBB (En Büyük Değerlikli Bit)'dir. Her bit belli bir zaman aralığında gönderilir. Eğer bit aralığı 1 ms ise kelime içindeki her bit için voltaj basamağı 1 ms görünecektir. Dolayısı ile 7 bitlik bir ascı kodu 7 ms'de gönderilir (5)(14).

Verici ve alıcı eşzamanlı olarak çalışabildikleri gibi eşzamansız olarak da çalışabilirler. Seri veri iletiminde verinin başlangıç noktasını belirtmek için "başlama" biti, veri aktarma işlemi sırasında oluşabilecek bozulmaları ortaya çıkarabilmek için veri bitlerinin hemen ardından "eşlik" biti ve verinin bitiş noktasını belirtmek için "bitiş" biti kullanılır. Şekil 2.3'de seri veri iletiminde veri aktarım sırası görülmektedir (5).



Şekil 2.3. Seri veri iletimi

2.3. Veri İletim Hatları

Hatlar sayısal bilginin iletildiği ortamlardır. Bu ortamlarda veri iletilirken band genişliğinin dar olması ve hattın kapasitesinin sınırlı olması gibi sorunlarla

karşılaşılır. Genelde haberleşme hatları Simplex, Half-duplex ve Full-duplex olarak üç tipte incelenir (6).

Simplex Hatlar: Bir verici ve bir alıcının olduğu sistemlerdir. Verici yayın yaparken alıcı sadece dinleyebilir. Vericiye mesaj gönderilemez. Örneğin; TV yayınlarını alıcı izlerken, aynı hat üzerinden veri gönderemez.

Half-Duplex Hatlar: Hattı ilk olarak verici kullanır. Cevap istendiği zaman, karşı taraf yani alıcı hattı kullanır. Örnek olarak; telsiz konuşmaları.

Full -Duplex Hatlar: Bir yandan hatta veriler gönderilirken bir yandan da hattan cevap niteliği taşıyan veriler alınabilir. Örnek olarak; telefon görüşmelerinde, aynı hat hem alıcı hem de verici tarafından kullanılır.

2.4. Senkron - Asenkron Veri İletimi

Bilgisayarların ve terminallerin haberleşmesi için iletimden önce birbirlerini uyarmaları gerekir. Verici bir bit göndermeden önce alıcıyı uyarmazsa alıcı gelen bit serisi için gerekli zamanı ayıramaz. Bu nedenle zamanlama çok önemlidir. Sinyal alıcı tarafından yanlış zamanda incelenirse gelen bilgi yanlış yorumlanabilir. Bilginin uygun zamanda incelenmesine senkronizasyon (eşzamanlı) adı verilir. Senkronizasyonu sağlamak için saat(clock) sinyali kullanılır (7).

Saat sinyalinin iki görevi vardır. Bunlar;

Veri tamamen ulaşmadan önce alıcıyı iletme senkron hale getirmesi

Alıcıyı giren veri bitleriyle senkron halde tutmasıdır.

Seri veri iletimi saat sinyalinin iletilmesine göre ikiye ayrılır. Bunlar:

Senkron veri iletimi: Senkron veri iletiminde veri bloğunun başındaki ve sonundaki bildiri karakterleri dışında alıcı ve verici arasında saat sinyali

taşıyan ayrı bir hat vardır. Bu hattan uygulanan saat sinyali alıcı ve vericinin aynı bit zamanında çalışmasını sağlar. Senkron iletimde iletim sonunda "iletim sonu" sinyali gönderilir.

Asenkron veri iletimi: Asenkron iletme çoğu zaman başla-bitir iletimi de denir. Asenkron iletimde veri ile saat sinyali iletilmez, çünkü senkronizasyon için başla bitir bitleri kullanılır. Başla-bitir bitleri veri bloğunun başı ve sonunda kullanılır.

2.4.1. Senkron ile asenkron iletimin karşılaştırılması

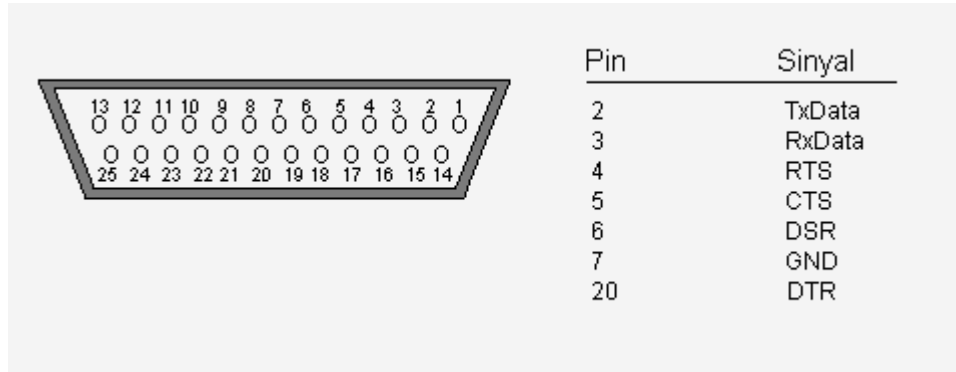
- Asenkron veri iletiminde her bloğun başında ve sonunda başla bitir bitleri vardır. Dolayısı ile iletim sınırlı bir zaman aralığında yapılıyorsa senkron veri iletimi daha hızlıdır.
- Senkron iletimde alıcı ve vericiyi senkron halde tutmak için ayrı bir hat gerekmektedir. Bu yüzden uzun mesafelerde senkron iletim dezavantaj oluşturur (8).

2.5. RS-232 İletişiminin Temel Kavramları

RS-232 bilgisayar dışındaki cihazların bilgisayar ile haberleşmelerinde en çok kullanılan iletişim standartlarından birisidir. RS-232 temel olarak bir seri iletişim birimidir. Seri iletişim biriminde karakterler bir hat üzerinden bit bit yollar. Seri iletişimin paralel iletişime göre en önemli üstünlüğü bağlantı kolaylığıdır. Bilgisayardan cihaza karakter göndermek için bir hat, cihazdan gelen karakterleri okumak üzere bir hat ve bir toprak hattı olmak üzere toplam üç hat kullanılarak iletişim gerçekleştirilebilir. Bu standardın önemli dezavantajı, haberleşme hızı arttıkça bilgi kaybına yol açmamak için kablo uzunluğunun da kısılması gerekliliğidir. Standard RS-232 19200 haberleşme hızında en fazla 20m kablo uzunluğuna izin vermektedir. Yeni seri iletişim standartlarından RS-422, RS-449 çok daha yüksek haberleşme hızlarında

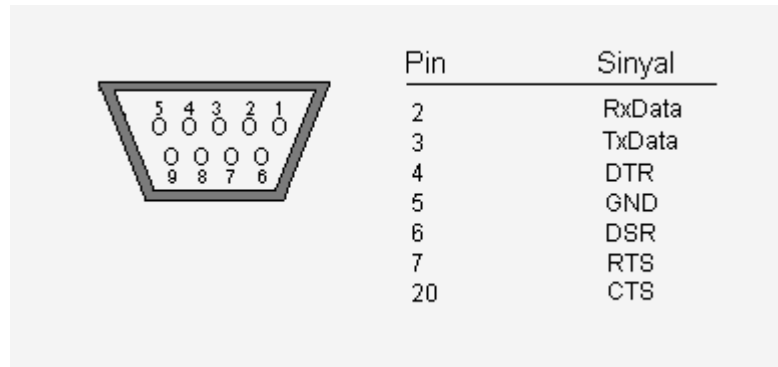
çok daha uzun kabloları desteklemektedir. Örneğin RS-422 1600 m uzunluğunda bir kablo üzerinden bir megabit/saniye haberleşme hızlarını desteklemektedir (9).

Şekil 2.4.'de 25 pin DB25S konnektöre ait bacak isimleri verilmiştir.



Şekil 2.4. DB25S konnektör

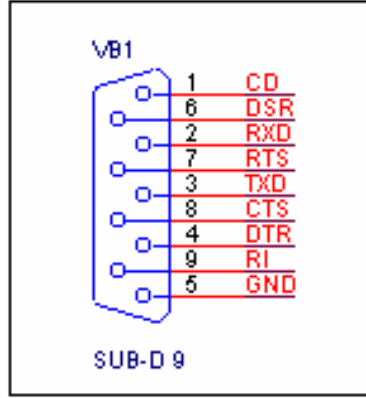
Şekil 2.5.'de 9 pin DB9S konnektöre ait bacak isimleri verilmiştir.



Şekil 2.5. DB9S konnektör

RS-232C çıkışında hangi sinyallerin konnektörde hangi ayaklara getirildiği aşağıda gösterilmektedir. "Data Set Ready" ve "Data Terminal Ready" sinyalleri kendi aralarında kısa devre edilerek sistemin veri iletimine hazır

olması sağlanır. Şekil 2.6'de RS-232'nin şeması ve bacak isimleri görülmektedir.



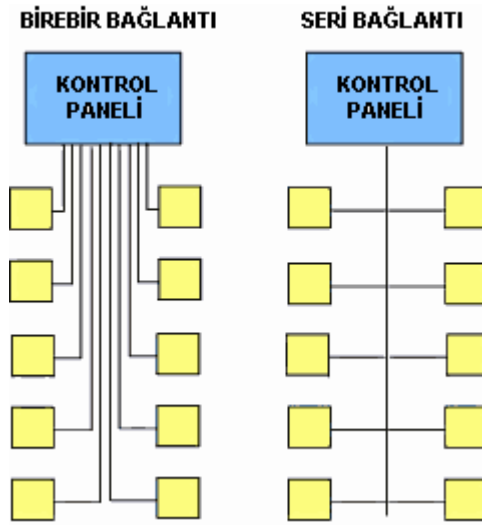
Şekil 2.6. RS-232 şeması ve bacak isimleri

Pin isimlerinin açıklaması:

1. CD (Carrier Detect) : Taşıyıcı Bulma (kullanılmıyor)
2. RXD (Received Data) : Alınan Veri
3. TXD (Transmit Data) :Gönderilen Veri
4. DTR (Data Terminal Ready) : Veri Terminali Hazır
5. GND (System Ground) : Sistem Şasesi
6. DSR (Data Set Ready) : Veri Kurma Hazır
7. RTS (Request to Send) : Gönderme İstemi (kullanılmıyor)
8. CTS (Clear to Send) : Göndermeleri Temizle (kullanılmıyor)
9. RI (Ring Indicator) : Çevirme Göstergesi (kullanılmıyor)

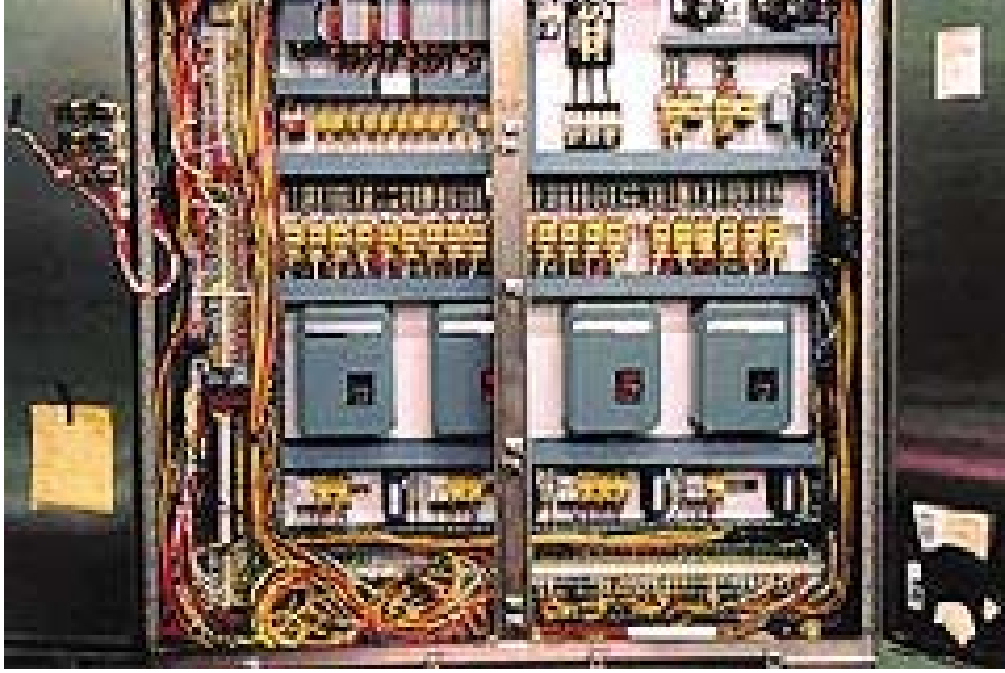
3. GERÇEKLEŞTİRİLEN SERİ İLETİM HATLI ASANSÖR DENETLEYİCİ SİSTEMİ

Seri haberleşme sistemleri daha önce açıklanan olumsuzlukların giderilmesi amacı ile geliştirilmiştir. Her ünite 2 - 4 kablo ile diğer ünitelere bağlıdır. Fonksiyonların sayısı veya karmaşıklığı kablo sayısını arttırmaz. Örneğin kabin ile kumanda panosu arasındaki tüm bilgi alışverişi iki kablo üzerinden yapılabilir. Kablo sayısının azalması montaj ve tamir işçiliklerini ve kablo maliyetini biçimde azaltır. Ancak bunun yanında donanım maliyetini arttırır. Bunun sebebi ise sistemde kullanılan tüm ünitelerin akıllı olması gerekmektedir. Şekil 3.1'de soldaki tabloda birebir bağlantı sağdakinde ise seri bağlantı sistemi kullanılmıştır(10).



Şekil 3.1. Birebir bağlantı ve seri bağlantı

Resim 3.1 de bir asansör kumanda panosundan katlara ve asansör kabinine yapılan kablo bağlantısı örnekleri görülmektedir. Soldaki panoda klasik birebir bağlantı sağdaki panoda seri bağlantı görülmektedir.



Resim 3.1. Birebir bağlantı ve seri bağlantı

Asansörde seri haberleşmenin kullanılabilceği noktalar şunlardır:

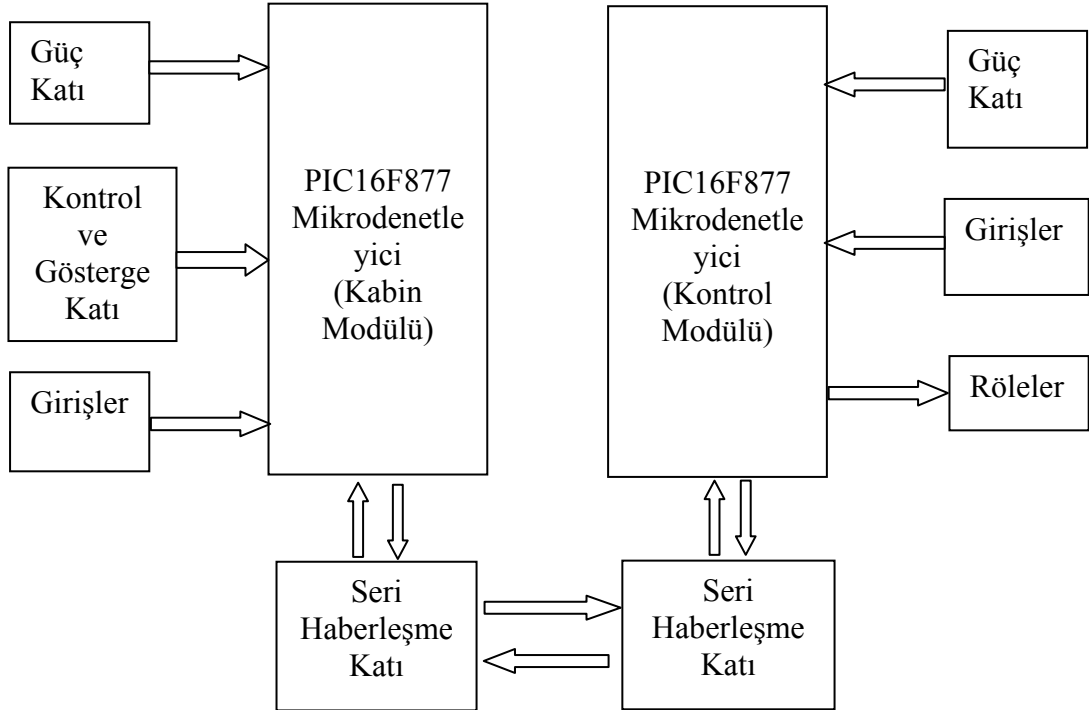
- Kabin ile kumanda tablosu arasında haberleşme
- Kat kasetleri ile kumanda tablosu arasında haberleşme
- Bilgisayar ile kumanda tablosu arasında haberleşme
- Modem ile kumanda tablosu arasında haberleşme
- Hız kontrol cihazı vb ünitelerle haberleşme
- Mutlak kod çözücü, aşırı yük sistemi gibi algılama sistemleri ile haberleşme

Seri haberleşme sistemleri elektroniğin gelişmesine paralel bir gelişme göstermişlerdir. En çok kullanılan seri haberleşme standartları RS-232 ve RS-485 standartlarıdır(11,14).

Tasarımı gerçekleştirilen devre temel olarak iki ayrı modülden (karttan) meydana gelmektedir. Bunlardan birincisi kabin kartı diğeri ise kontrol kartıdır. Bu iki farklı modül arasındaki haberleşme ise RS232C seri

haberleşme modülü üzerinden gerçekleştirilmektedir. Bu modüller ise farklı karttan oluşmaktadır. Kabin modülü; güç katı, kontrol ve gösterge katı, girişler katı ve seri haberleşme katından oluşmaktadır. Kontrol modülü ise; güç katı, seri haberleşme katı ve rölelerin bulunduğu çıkış katından oluşmaktadır.

Buradaki amaç iki modül arasındaki bilgi iletiminin üç kablo üzerinden yapılarak kablo maliyetinin en aza indirilmesidir. Şekil 3.2'de tasarımı gerçekleştirilen mikrodenetleyici kontrollü asansör denetleyici sistemini oluşturan katlar blok şema halinde gösterilmiştir.



Şekil 3.2. Asansör kontrol devresi blok diyagramı

Devrenin çalışması şu şekildedir; iki ayrı modül arasında seri haberleşme kullanılarak bilgi alış verişi yapılır. Burada girişlerden ve yedi parçalı göstergeden alınan bilgiler kontrol modülüne iletilerek ilgili çıkışı sürer. Seri hatta herhangi bir sebeple veri iletişimi kesilirse kontrol kartı asansör motorunu durdurur ve tüm programlar yeniden başlar. Yine seri iletişimde

herhangi bir sebepten dolayı bilgilerin gitmemesi veya eksik gitmesini önlemek için saniyede 10 kez bilgiler tampondan gönderilir.

Şekil 3.3'de tasarımı gerçekleştirilen kontrol ve kabin kartının asansör kumanda kartına bağlantı şeması verilmiştir. Şekilde giriş ve çıkışlar alfabetik olarak verilmiştir. Bu giriş ve çıkışların açıklaması aşağıda verilmiştir.

A (Kat butonu bilgisi): Kabin içindeki kat buton panosundan gelen isteklerdir. 16 duraklı bir sistem için 16 adettir

B (Seri iletim hattı): iki kart arasındaki iletişimi sağlayan iletim hattıdır. Toplam 3 hattır. Bunlar: toprak (Gnd), veri alımı (Rx) ve veri gönderme (Tx) uçlarıdır.

C (Kat bilgisi): Kabinden gelen isteğin kumanda panosuna iletilmesini sağlayan hattır. 16 duraklı bir asansörde 16 adet kablodur.

D (Kayıt aldı çıkışı): Asansör kabininden bir kat butonuna basıldığı anda kabin kartı isteği kontrol kartı üzerinden kumanda panosuna iletir. İsteği alan kumanda panosu eğer bir sorun yok ise kontrol kartı üzerinden asansör kabini üzerindeki kabin buton lambasını yakar.

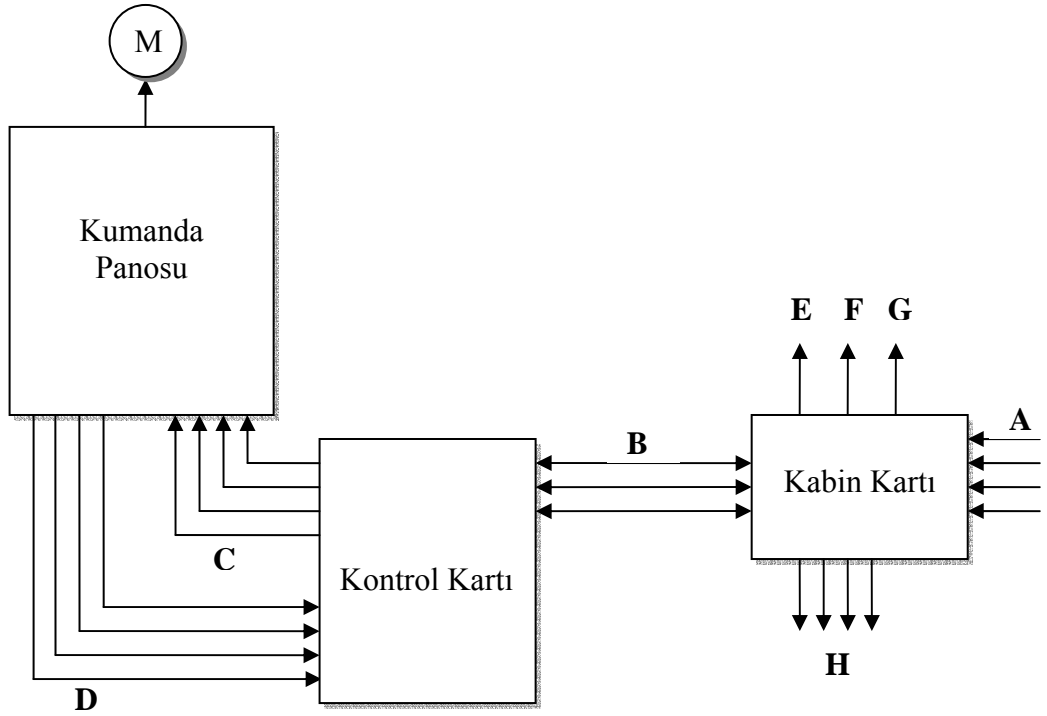
E (Kabin içi aydınlatma): Kabin içi aydınlatılmasında kullanılan çıkış bilgisidir.

F (Kapı açma): Diğer bir adı da LİR-POMP'dur Kabin kata geldiğinde kapı kilidi kontağının açılmasını sağlayan çıkış bilgisidir.

G (Aşırı yük lambası): Kabin içindeki aşırı yük lambasının veya ikazının çalışmasını sağlayan çıkış bilgisidir.

H (Kabin buton lambası): Kabin içindeki butonlardan birine basıldığında eğer herhangi bir sorun yok ise kumanda panosundan kabine gönderilen ve kat butonun etrafındaki veya yanındaki lambanın yanmasını sağlayan çıkış bilgisidir.

M(Asansör Motoru): Asansör kabinini hareket ettiren alternatif gerilim motorudur.



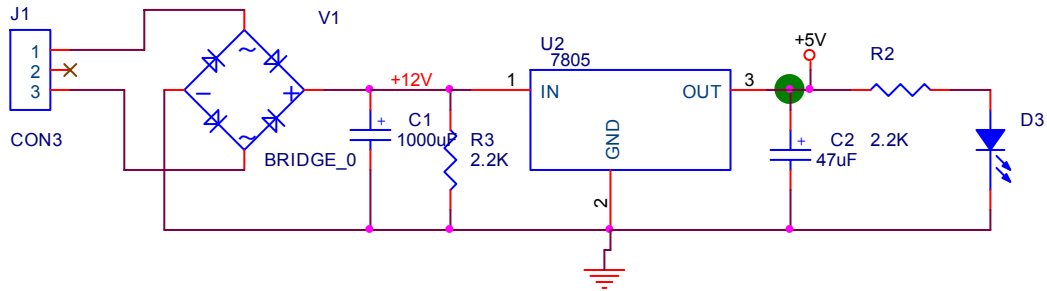
Şekil 3.3. Asansör Elektrik Tesisatı Blok Diyagramı

Asansör sistemlerinde birden fazla standart kullanıldığından yazılan programlarda her bir standarda uygun değişiklikler yapılarak birden fazla program yazılmıştır. Kullanılan standarda uygun program kartlara yüklenerek sistemin daha iyi çalışması sağlanmıştır.

Kartların tasarımı ORCAD 9.0 baskı devre hazırlama programı kullanılarak yapıldı. Kartlar çift yüzlü ve delik içi kaplama olacak şekilde tasarlandı. Kartların devre şemaları ve baskı devre şekilleri ekte verilmiştir.

3.1. Besleme Katı

Yapılan devrenin besleme gerilimi Şekil 3.4'deki besleme devresinde görülmektedir. Besleme katında 7805 pozitif gerilim regülatörü kullanılmıştır. 7805 7,5V ve 30V arasındaki giriş gerilimlerinde 5V çıkış veren bir regülatördür.



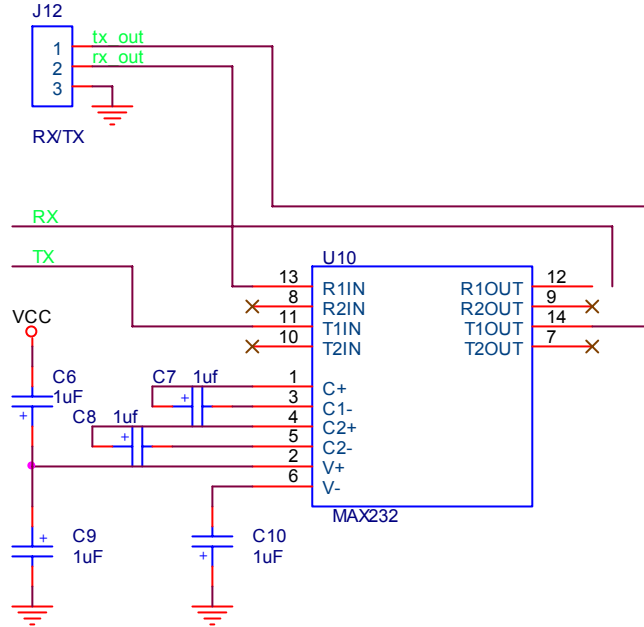
Şekil 3.4. Besleme devresi

Kabin kartı ve kontrol kartı şaseleri ortak olması ve kabin kartının mevcut asansörün beslemesinden bağımsız olmasından dolayı çeşitli voltaj standartları olan eski tip ve yeni tip tüm asansörlerde tasarımı gerçekleştirilen devre kullanılabilir.

3.2. RS232 Seviye Dönüştürme Katı

RS232 seviye dönüştürme devresi Şekil 4.3'de gösterilmiştir. Bu devrenin amacı devre ve bilgisayar arasında yapılan haberleşmede gerilim uygunlaştırması sağlamaktır. TTL/CMOS gerilim seviyeleri ve RS232 gerilim seviyeleri birbirinden farklıdır.

Mikrodenetleyicilerle bilgisayarların iletişiminde temel olarak TxD, RxD ve GND hatları kullanılır. Pek çok mikrodenetleyicide, bilgisayarla haberleşmek üzere TxD ve RxD hatları bulunur. Fakat mikrodenetleyicilerde bu hatların sinyalleri 0 volt ve 5 voltur. Bilgisayarla haberleşmede gereken RS-232 sinyalleri ise +12 volt ve -12 voltur. Bu yüzden mikrodenetleyici ve bilgisayar arasındaki iletişimde gerilim dönüşümünü sağlamak üzere bir gerilim dönüştürücü kullanılır. Bu tür gerilim dönüştürücülerin en çok kullanılanı ise MAX232 entegresidir Şekil 3.5 de RS232 seviye dönüştürme katına ait devre şeması görülmektedir.(12).

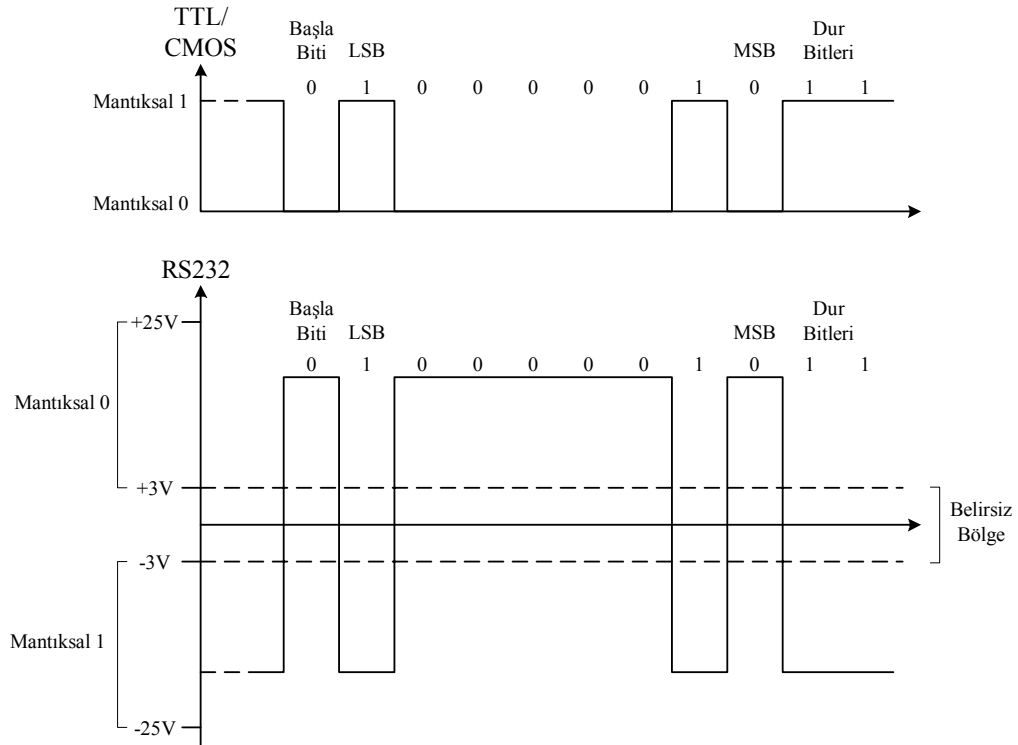


Şekil 3.5. RS232 seviye dönüştürme devresi

Genelde mikrodenetleyici tarafında kullanılacak RS232 konnektörü dişi yapılıdır. Dolayısı ile mikrodenetleyici ile bilgisayarı bağlayacak ara kablonun bir ucuna 9 pin erkek konnektör, bir ucuna da 9 pin dişi konnektör bağlanır. Konnektörlerin 5 nolu uçları (GND) direkt olarak birbirine bağlanır. 2 (TxD) ve 3 (RxD) uçları ise birbirine çapraz bağlanır. Böylece mikrodenetleyicinin GÖNDER (TxD) ucu bilgisayarın AL (RxD) ucuna ve bilgisayarın YOLLA ucu mikrodenetleyicinin AL ucuna bağlanmış olur(13)(14).

RS-232 seri iletişim standardı elektriksel özelliklerinde “1” biti -3 volttan -25 volta kadar bir elektrik sinyali olarak tanımlanmıştır. “0” sinyali ise +3 volttan +25 volta kadar elektrik sinyali olarak tanımlanmıştır. -3 volt ile +3 volt arasındaki herhangi bir sinyal ise belirsiz bir sinyaldir(13).

Şekil 3.6’da ASCII A karakterinin TTL/CMOS ve RS232 sinyal şekilleri gösterilmiştir. Görüldüğü gibi aradaki tek fark gerilim seviyeleridir. Bir başla biti ve iki dur biti olduğu görülmektedir. Kullanılan RS232 haberleşmesine göre bunların yapısı değişebilir. Şekil 3.6’daki sinyallerde 8 bit haberleşme gösterilmiştir. Eğer 7 bit haberleşme kullanılırsa 8. bit eşlik biti olarak kullanılabilir (13).

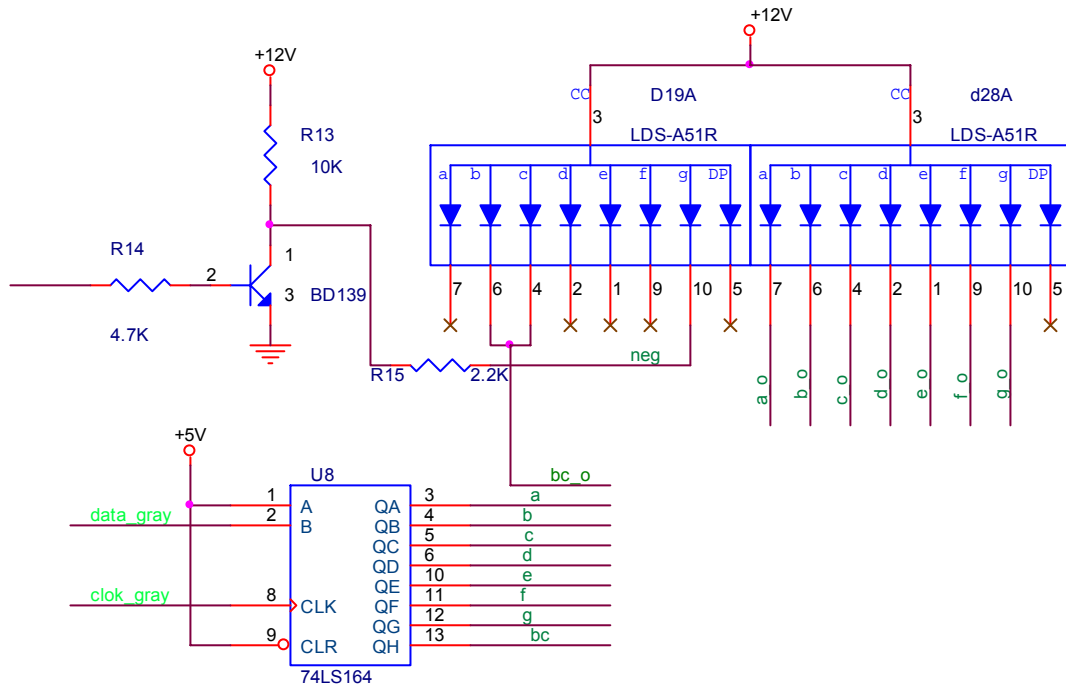


Şekil 3.6. ASCII A karakterinin TTL/CMOS ve RS232 grafikleri

3.3. Kontrol ve Gösterge Katı

Butona basıldığında kat bilgilerini gösteren kattır. İki tane yedi parçalı ortak anot display kullanılmıştır. Göstergeleri sürerken seri giriş-paraalel çıkışlı 74LS164 shift register kullanılmıştır. Bunun amacı mikrodenetleyicide port kullanımını azaltmak ve 8 bitlik bir veriyi mikrodenetleyicinin tek bir portunu kullanarak göstergeleri sürmektir (16).

Şekil 3.7'de kontrol ve gösterge katı devre şeması görülmektedir. 74LS164'ün bağlantı şeması ve fonksiyon tablosu ekte verilmiştir.



Şekil 3.7. Kontrol ve gösterge katı

3.4 Girişler Katı

Bu katta tüm asansörlerde olabilecek girişleri mikrodenetleyici portlarına uygun hale getiren devreler bulunmaktadır. Aşağıda asansör kumanda kartı girişleri listelenmiş ve açıklanmıştır.

Kat butonu bilgisi : Asansör kabininden yapılan çağrılar ana kumanda kartına ileten girişlerdir. Tasarımı gerçekleştirilen devre en çok 16 duraklı bir asansör sistemi için uygun olan kat giriş bilgilerini alacak şekilde yapılmıştır. Mikrodenetleyicinin giriş portlarının yetersiz olması sebebiyle bu bilgiler önce 74AS150 multiplekser kullanılarak 16 giriş 4 bitlik veriye dönüştürülerek mikrodenetleyicinin giriş portlarına verilmiştir. Böylece daha az port kullanılmıştır.

Tasarım aşamasında ve ilk kart üretiminde kullanılan 74AS150 multiplekseri daha sonraki aşamada üretimden kaldırılması sebebiyle 74AS150 16x4 multiplekser yerine iki adet 74HC251 8x3 multiplekser entegrasyonu özel bir bağlantı şekli ile bağlanarak sistemde kullanılmıştır.

Gray kod bilgisi : Ana kumanda kartına asansör kabininin o an hangi katta olduğunu bildiren 4 bitten oluşan bir veri gönderilir. Bu verinin gray kod sisteminde olmasının sebebi gray kod sisteminde bilginin değerinin bir atmasında sadece verideki bir bitin değeri değişmesidir. Bu değişiklikte herhangi bir hata olursa yani bir bit değil de birden fazla bitin değişmesi durumunda asansör sistemi veri akışında bir hata olduğunu algılayarak kabini durdurur.

Aşağı yön sınır kesici : Asansör kuyusunun altına yerleştirilen bir kontaklıdır. Eğer kabin bir şekilde bu kontakla ulaşırsa kontak mekanik olarak açılır ve ana kumanda panosuna bir sinyal iletir. Bu sinyali alan kumanda kartı asansörü yavaşlatır. Motor kabini durduramazsa mekanik fren sistemi

kabini durdurur. Bu sayede kabinin aşağı yönde yere çakılması engellenmiş olur.

Yukarı yön sınır kesici : Asansör kuyusunun en tepesine yerleştirilen bir kontaklıdır. Eğer kabin bir şekilde bu kontakla ulaşırse kontak mekanik olarak açılır ve ana kumanda panosuna bir sinyal iletir. Bu sinyali alan kumanda kartı derhal asansörü yavaşlatır. Motor kabini durduramazsa mekanik fren sistemi kabini durdurur. Bu sayede kabinin yukarı yönde çarpması engellenmiş olur.

Hassas durdurucu :Asansör rayları üzerinde kat seviyelerinin altında ve üstünde bulunan mıknatıs vardır. Kabinde bulunan algılayıcı bu manyetik algılayıcılardan geçtiği anda kabin kumanda katına katın durma seviyesine geldiğini gösteren bir sinyal gönderir ve ana kumanda kartı da motoru durdurarak kabinin tam kat hizasında durmasını sağlar.

Aşırı yük sinyal girişi : Asansör kabininin ağırlığını kontrol den ağırlık algılayıcısından gelen sinyaldir Aşırı yük sinyali alan kumanda kartı asansörün hareket etmesine müsaade etmez ve kabindeki aşırı yük sinyal lambasını yakar.

Revizyon (bakımda) anahtarı girişi : Asansör bakımda olduğu anda teknisyenin asansörü kullanım dışına çıkarmasını sağlayan anahtardan gelen sinyaldir.

Revizyon aşağı yön buton girişi : Asansör bakımda iken teknisyenin kabini aşağı hareket ettirmesini sağlayan butona bağlı bir giriştir. Teknisyen butona bastığı sürece kabin aşağı yönde hareket edecektir. Aşağı yön sınır kesici sinyali olduğu sürece hareket mümkündür.

Revizyon yukarı yön buton girişi : Asansör bakımda iken teknisyenin kabini yukarı yönde hareket ettirmesini sağlayan gir butona bağlı bir giriştir.

Teknisyen butona bastığı sürece kabin yukarı yönde hareket edecektir. Yukarı yön sınır kesici sinyali olduğu sürece hareket mümkündür.

Dur düğmesi: Kabin içine yerleştirilen ve kullanıcının asansör kabinini durdurabilmesi amacıyla yerleştirilmiş bir butondur. Yeni tip otomatik kapılı asansörlerde bulunmamaktadır.

Otomatik kapı girişi: Yeni tip asansörlerde bulunan ve kabin kapılarını açmak ve kapatmak amacıyla yerleştirilen girişlerdir. 4 adettir.

Tüm girişler opto kuplor kullanılarak optik olarak yalıtılarak oluşabilecek ani voltaj yükselmelerinden korunmuştur.

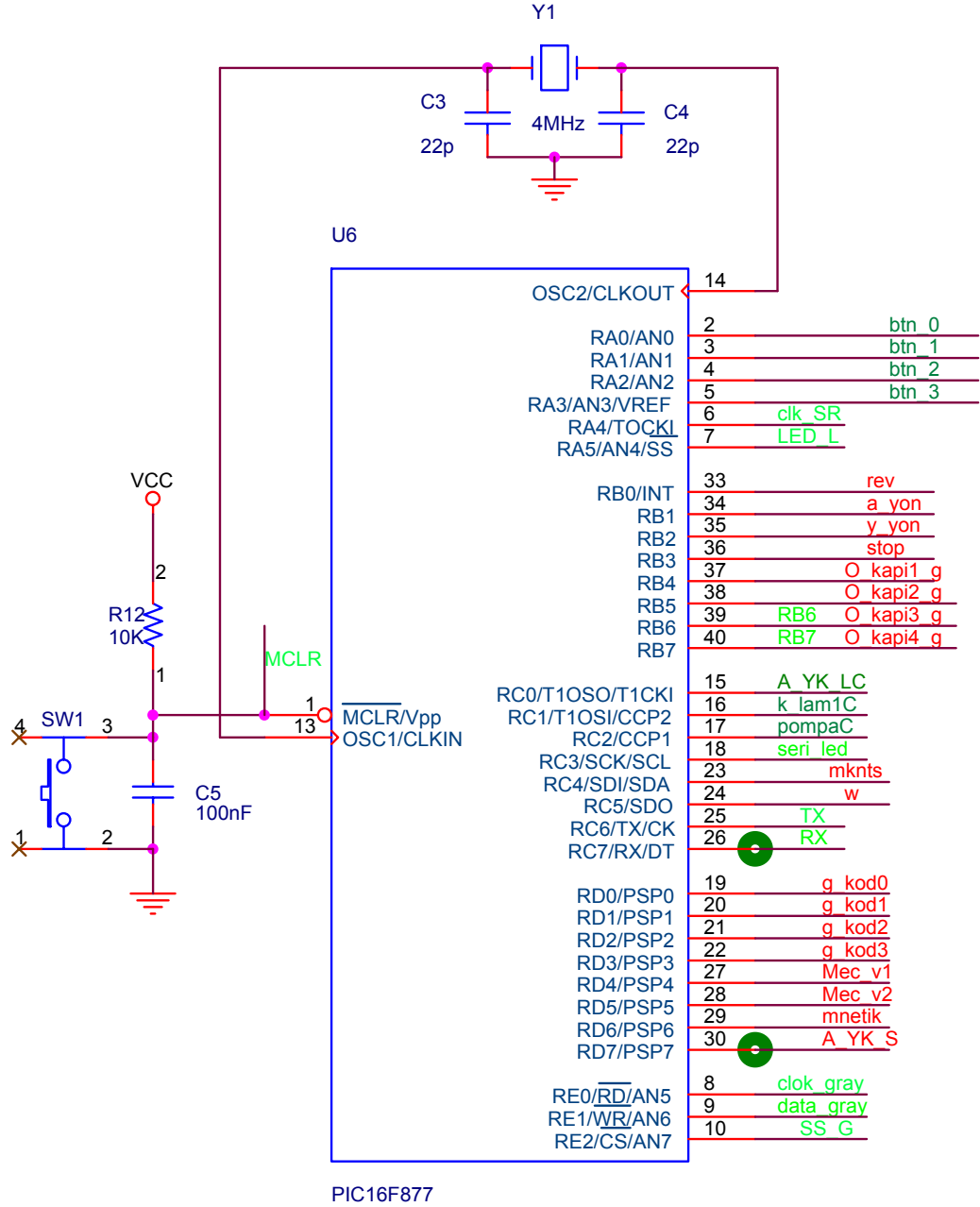
3.5. Mikrodenetleyici Katı

Kabin kartındaki mikrodenetleyici dış dünyadan aldığı bilgileri kontrol kartındaki mikrodenetleyiciye seri haberleşme yöntemiyle iletir. Kabin kartından Kontrol kartına aktarılan bilgiler bölüm 3.4 de verilmiştir.

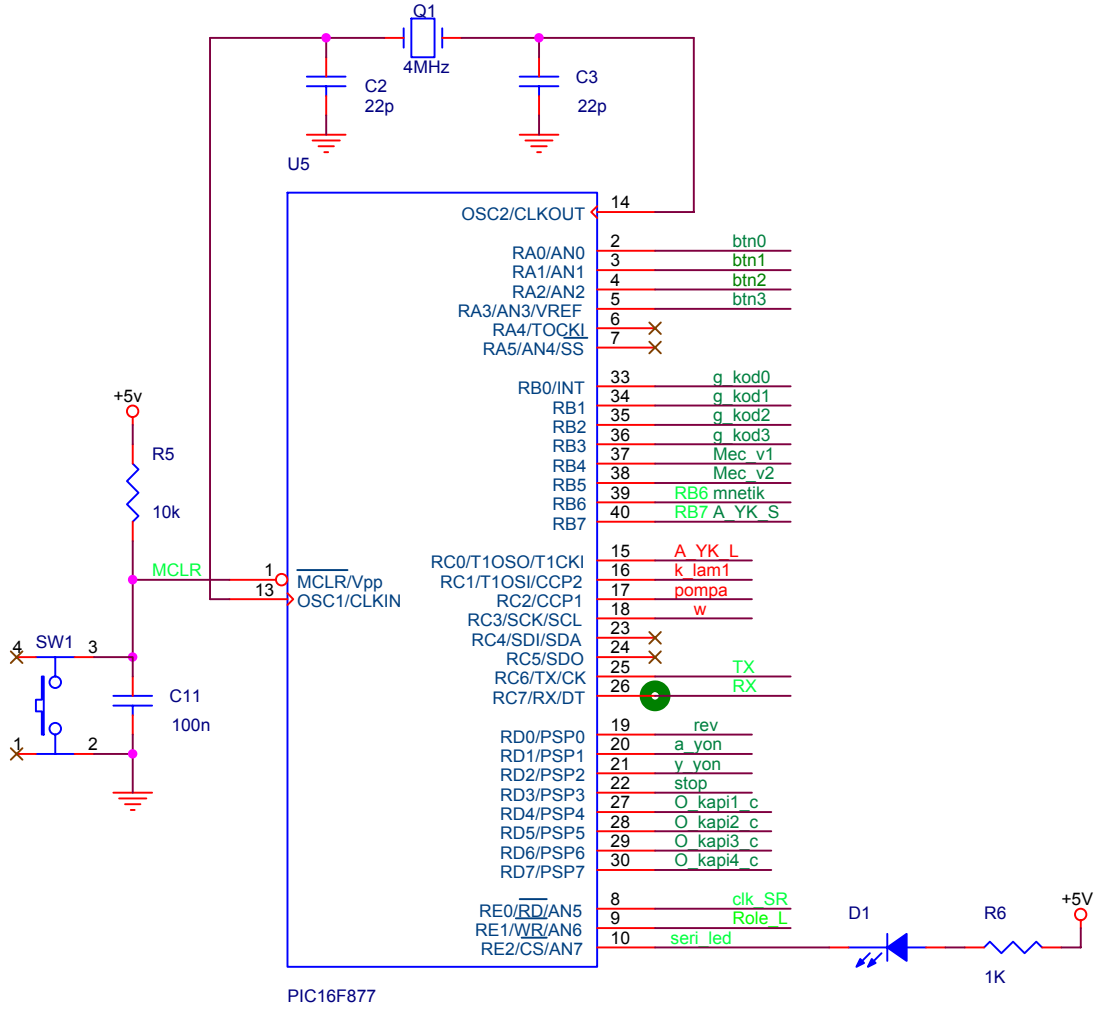
Devrede kat bilgilerini görüntülemek için display kullanılmıştır. SW1 butonunun görevi ise mikrodenetleyiciye reset atmaktır. PIC16F877 mikrodenetleyicinin osilatör devresinde 4MHz kristal kullanılmıştır. PIC16F877 4MHz'lik sinyali 4'e bölerek kullanmaktadır. Yani PIC16F877 1MHz'de çalışmaktadır. Herbir döngü 1µS sürmektedir. Şekil 3.8'de kabin kartındaki mikrodenetleyicinin devre şeması gösterilmiştir (17).

Şekil 3.9'da kontrol kartındaki mikrodenetleyicinin devre şeması gösterilmiştir. Kontrol kartındaki mikrodenetleyici kabin kartından aldığı bilgilere göre ilgili çevre birimlerini yani röleleri çalıştırmaktadır. Bütün girişlerde opto izolatör(kuplör), çıkışlarda ise röle kullanılarak devrenin yüksek gerilimden zarar görmesi önlenmiştir. D1 ledinin görevi ise iki mikrodenetleyici arasında seri iletişimin olup olmadığını kontrol etmektedir.

Eğer seri iletişim varsa led yanıp sönmektedir, seri iletişim kesildiğinde ise led sönmektedir. Kartlardaki multiplekserin görevi mikrodenetleyicideki portların sayısını artırarak daha çok bilgi girişini sağlamaktır.

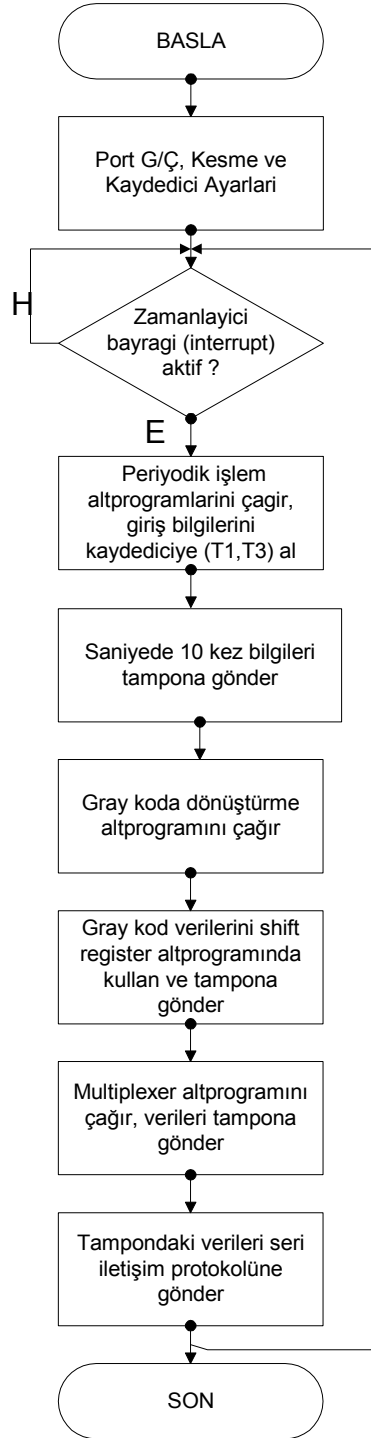


Şekil 3.8. Kabin kartındaki mikrodenetleyicinin devre şeması

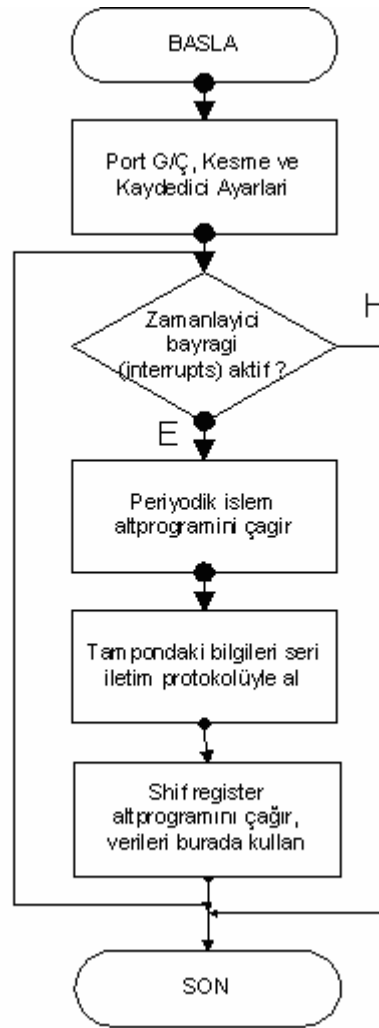


Şekil 3.9. Kontrol kartındaki mikrodenetleyicinin devre şeması

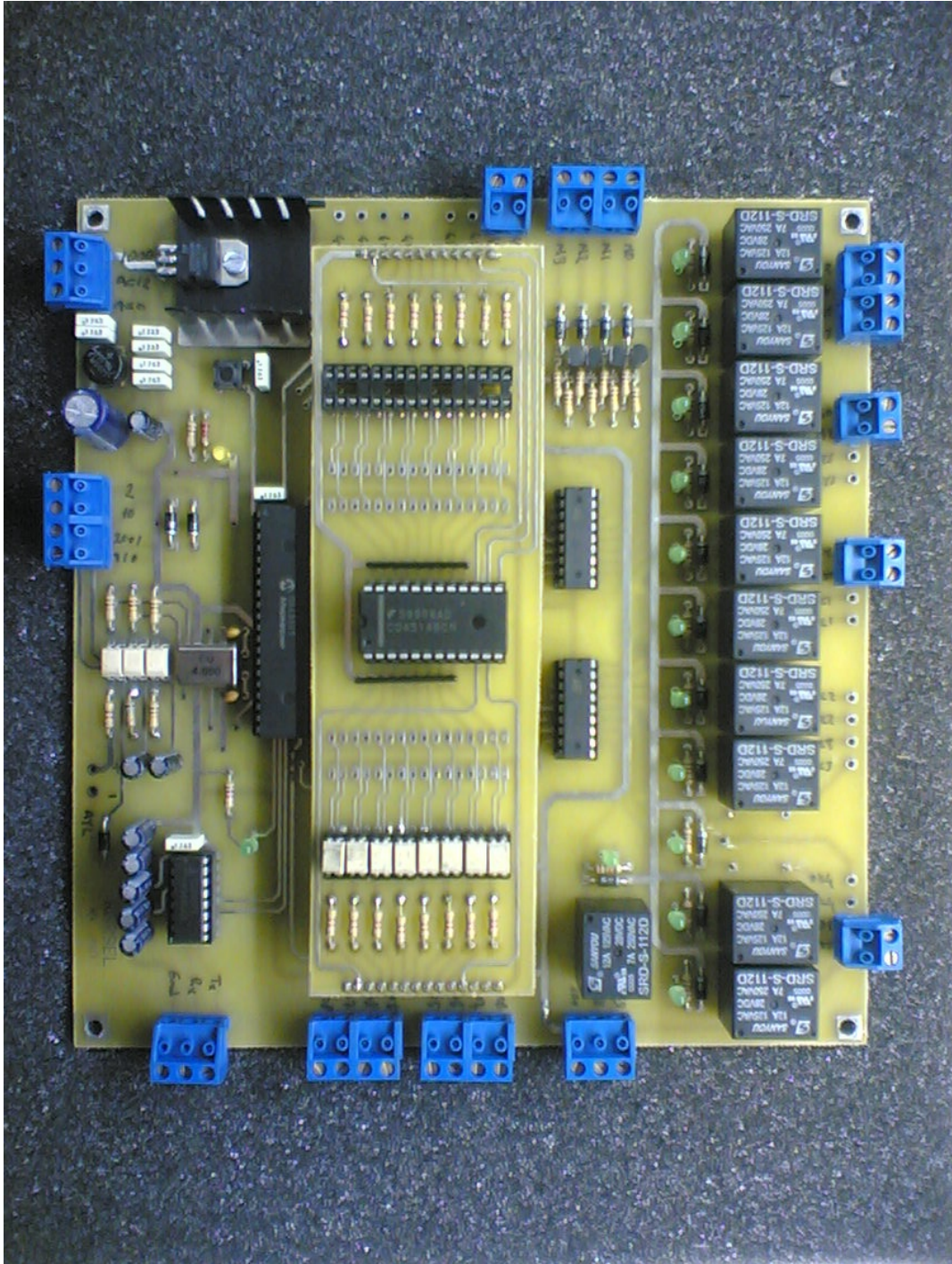
Mikrodenetleyici katının programına ilişkin kabin kartına programının akış şeması Şekil 3.10 ve kontrol kartına ait programın akış şeması Şekil 3.11'de verilmiştir. Devreye ait PIC16F877 programı Ek-1'de verilmiştir.



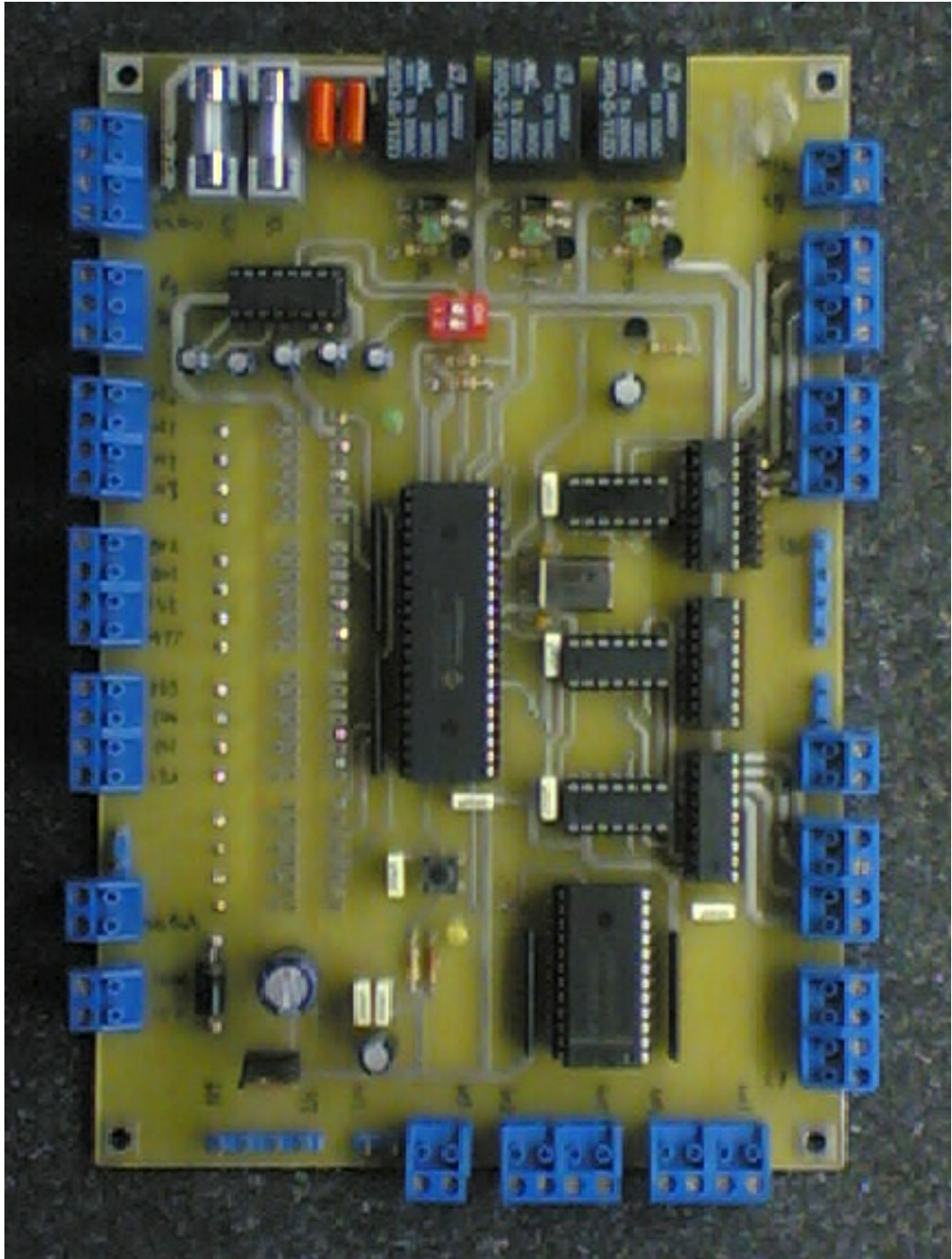
Şekil 3.10. Kabin programı akış şeması



Şekil 3.11. Kontrol programı akış şeması



Resim 3.2. Kabin kartının görüntüsü



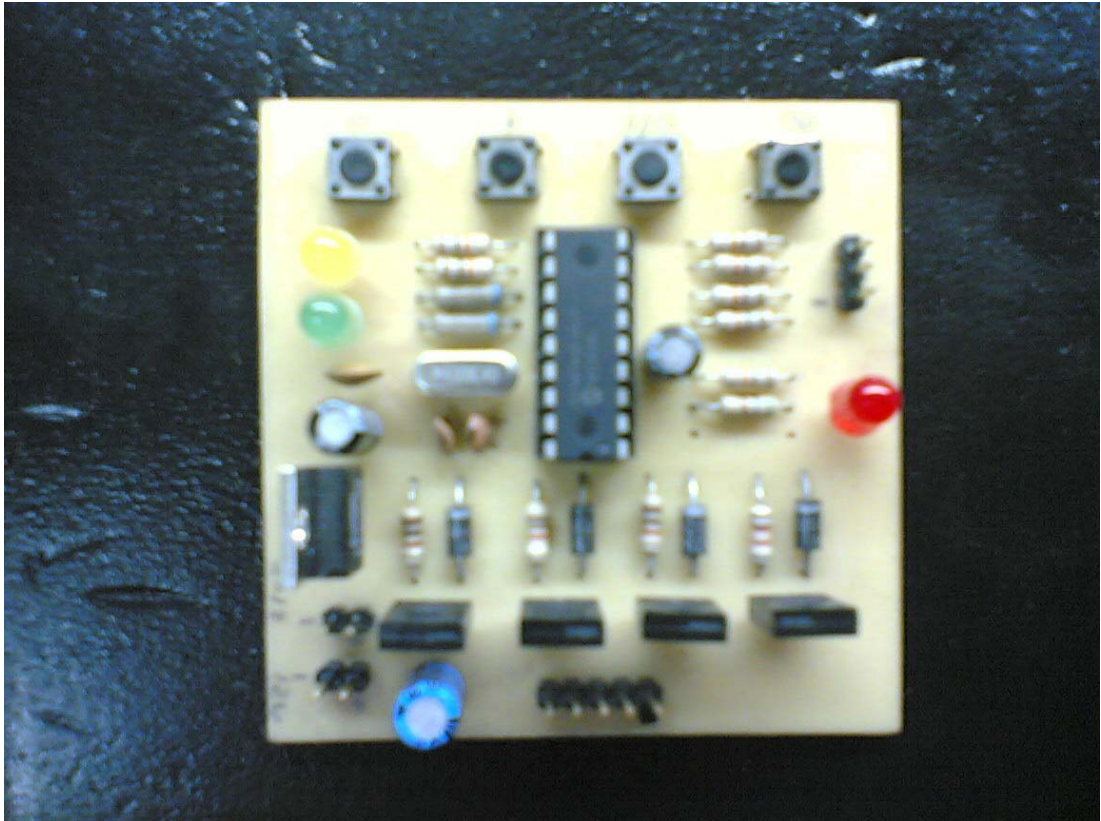
Resim 3.3. Kontrol kartının görüntüsü

Tasarımı gerçekleştirilen devre iki adet üretilmiştir. Üretilen kartlardan biri laboratuvar ortamında 3 ay süreyle çalıştırılmıştır. Herhangi bir sorunla karşılanmamıştır. Üretimi yapılan ikinci kart takımı gerçek bir asansör üzerine

Mayıs 2003 tarihinde montaj edilmiştir. Necatibey Cad. Bilecik Apartmanı No:27 Çankaya Ankara adresindeki 9 duraklı asansörde takılan sistem halen çalışmaktadır(Temmuz 2005).

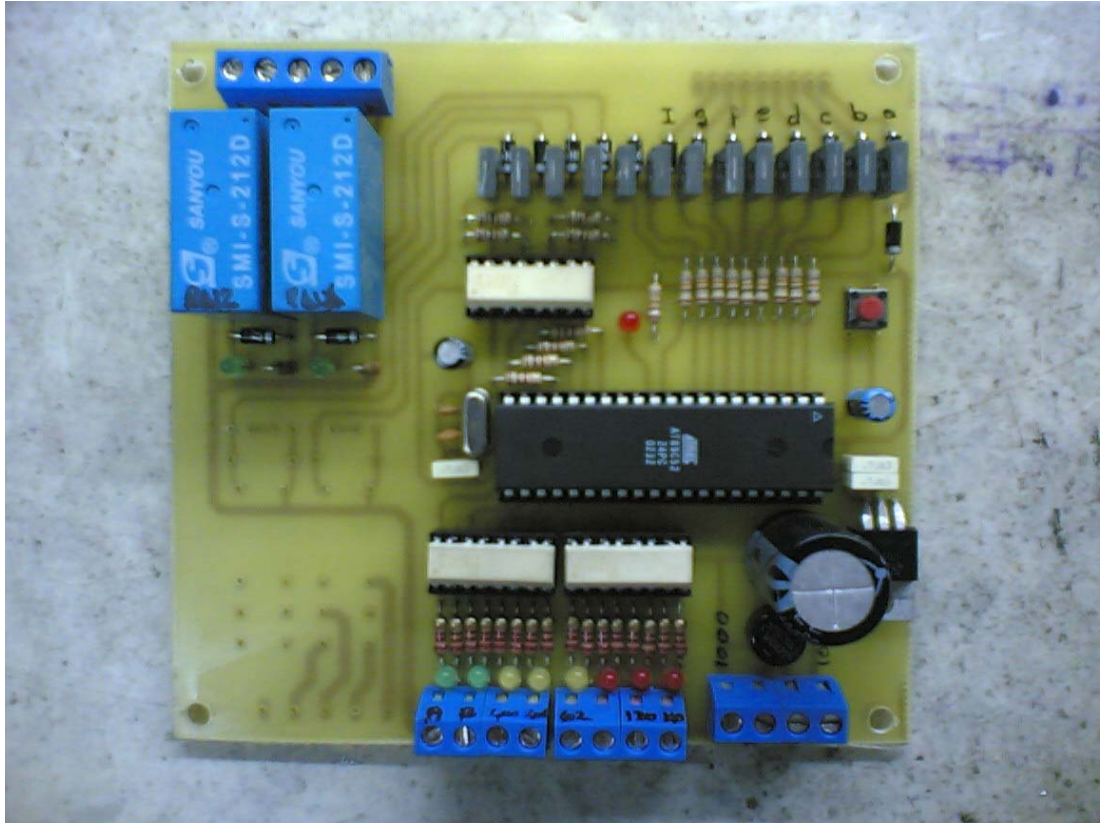
3.6 Tasarımı Gerçekleştirilen Asansör Prototibi

Gerçekleştirilen devrenin çalıştırılması ve test edilmesi amacıyla bir asansör prototipi yapılmıştır. Prototip 3 katlı bir bina için gerçekleştirilmiştir. Asansör motoru yerine unipolar bir step motor kullanılmıştır. Bu step motoru sürmesi ve hızını ayarlaması için 16F84 mikrodenetleyicili bir kumanda kartı tasarlanmıştır. Bu kart kumanda kartının çıkışına bağlanmıştır. Kumanda kartı olarak gerçek bir dört duraklı asansör kumanda kartı kullanılmıştır. Resim 3.4'de prototibe ait step motor kontrol devresi görülmektedir.

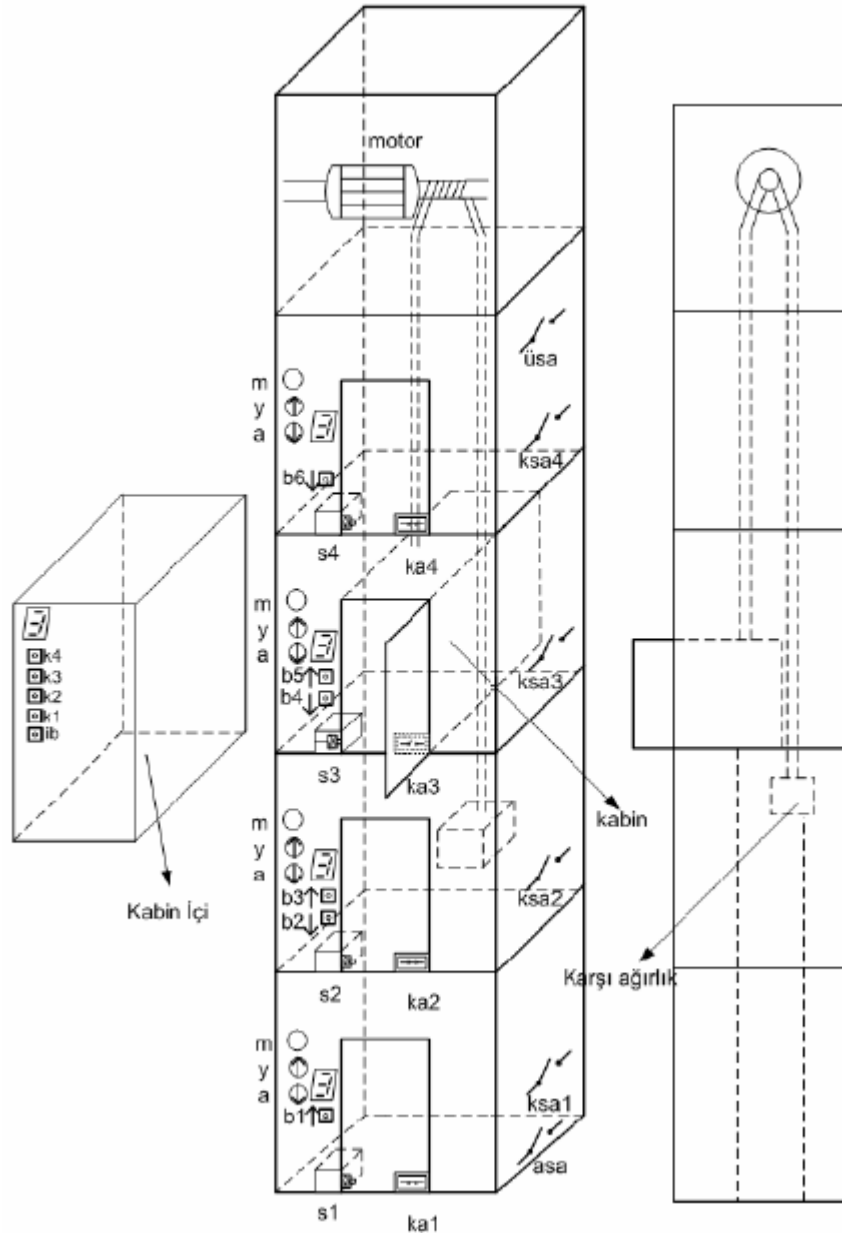


Resim 3.4. Step Motor Kontrol Devresi

Resim 3.5 de Prototipte kullanılan asansör kumanda kartı görülmektedir. Bu devreye ait baskı devre şeması ekte verilmiştir.



Resim 3.5. Asansör Kumanda kartı



Şekil 3.12. Asansör Prototibi

4. SONUÇ VE ÖNERİLER

Bu tez çalışmasında seri iletim hatlı mikrodenetleyici denetimli asansör tasarımı ve prototipi gerçekleştirildi. Mevcut asansör sistemlerinde kabin ile ana kumanda panosu arasındaki iletişim birebir kablo bağlantısı ile yapılmakta iken tasarımı gerçekleştirilen devrede RS-232C seri iletişim protokolü kullanılmıştır. Bu sayede kurulum ve bakım zamanından tasarruf edilmiştir ve maliyet azaltılmıştır. Böylece asansörlerin kurulumu ve bakımı daha basit bir hale getirilmiştir.

Program geliştirmek için PIC assembly diline göre program geliştirme ve hata bulma bakımından daha üstün olan PIC C dili kullanılmıştır. Tasarımda PIC mikrodenetleyicileri kullanılması tasarımın son derece esnek bir yapıya sahip olmasını sağlamıştır.

Mevcut asansör haberleşme sistemlerinde kullanılan kablolama şekillerine alternatif olarak seri iletişim protokolünün kullanılmasını sağlayan bir tasarım gerçekleştirilmiştir.

PIC mikrodenetleyicileri, fiyatlarının ucuz olması, çok geniş ürün ailesine sahip olması, EEPROM (FLASH) belleğe sahip olup kullanım kolaylığı sağlaması vb. birçok özellikleri sayesinde gün geçtikçe kullanımı yaygınlaşmaktadır. Bir mikrodenetleyici ile sistem tasarımı gerçekleştirildiğinde hem kolay hem de ucuz olur. Buna göre tasarım ve program geliştirme aşamaları uzun olmakla birlikte PIC mikrodenetleyicileri ile endüstriyel sistemler için ucuz ve etkin çözümler elde etmek mümkündür.

Tasarımı gerçekleştirilen devrenin laboratuvar ve gerçek ortam testleri göstermiştir ki devrenin gürültü, ısı, iş yoğunluğundaki performansı oldukça iyidir.

Devrede seri iletiřim protokolü olarak gerilim esaslı RS232 yerine akım esaslı RS485 kullanılarak iletim hat mesafesi artırılabilir ancak kullanılacak kablonun burgulu olması geređi ve piyasa řartlarında burgulu ve esnek kablo bulma zorlukları ile maliyetin yükselmesi gibi dezavantajlar göz ardı edilmemelidir.

Tasarımı gerçekteřtirilen mikrodenetleyicili seri iletiřim kartları sadece asansörde deđil küçük deđiřiklikler yapılarak herhangi bir PC ye gerek olmadan birçok otomasyon sistemlerinde, bina güvenlik sistemlerinde, üretim bantlarının kontrolünde kullanılabilir.

KAYNAKLAR

1. Kan, İ. G., “Asansör Tekniği Cilt1”, **Birsen**, 1997
2. Gülercan, M., “Asansör Panel Verilerinin Radyo Dalgalarıyla İletimi” Hacettepe Üniversitesi, Fen Bilimleri Enstitüsü,2002
3. Aybey, Z., “Asansörde Seri Haberleşme (1)”, **Asansör Dünyası Dergisi**, Sayı 59, (2002)
4. Altınbaşak, O., Mikrodenetleyiciler Ve PIC Programlam, **Atlas Yayın Evi**, (2001)
5. Haykin, S., “Communication Systems Third Edition”, Mcmaster University
6. Yıldırım, M.,. “Seri İletişim Teknikleri-I.”, **Network Aylık Veri İletişimi ve Telekomünikasyon Dergisi**, Sayı.7, Sayfa.84-85,.(1997).
7. Yıldırım, M., “Seri İletişim Teknikleri-II Asenkron İletişim”, **Network Aylık Veri İletişimi ve Telekomünikasyon Dergisi**, Sayı.8, Sayfa.68-70, (1997).
8. Yıldırım, M., “Seri İletişim Teknikleri-III Senkron İletişim”, **Network Aylık Veri İletişimi Ve Telekomünikasyon Dergisi**, Sayı.9, Sayfa.70-73, (1997).
9. Svecko R, Ratej B, “Microcontroller Contactless Identification System And Access Control”, **Informacije Midem-Journal Of Microelectronics Electronic Components And Materials** 31 (3): 185-190, (2001)
10. Aybey, Z., “Asansörde Seri Haberleşme (2), **Asansör Dünyası Dergisi**, Sayı 60, (2002)
11. Aybey, Z., Asansörde Seri Haberleşme (3), **Asansör Dünyası Dergisi**, Sayı 61, (2002)
12. Öner, D., İletişim Kuramı I Ders Notları, **İstanbul Üniversitesi**, (2001)
13. Öner, D., İletişim Kuramı II Ders Notları, **İstanbul Üniversitesi**, (2001)
14. Acar, O., “Akıllı Bina Otomasyon Sistemlerinde Seri İletişim – Ethernet Entegrasyonu”, **Kocaeli Üniversitesi, Fen Bilimleri Enstitüsü**, (2004).
15. Huang YS, Chung SL, Jeng MD, “Modeling And Control Of Elevators By Statecharts”, **Asian Journal Of Control** 6 (2): 242-252 (2004).

16. Dinçer, G., "PIC Microcontroller Uygulama Devreleri 2. Baskı", **Era Bilgi Sistemleri Ve Yayıncılık**, (2000).
17. Microchip, "Microchip Databook", **Microchip Technology Inc.**, (1994).
18. Carr, J. J., "Desinger's Handbook Of Instrumentation And Control Circuits", **Academic Press.**, San Diego, London, 493-496, (1991).
19. Zhu Wd., Ni J., "Energetics And Stability Of Translating Media With An Arbitrarily Varying Length", **Journal Of Vibration And Acoustics-Transactions Of The Asme** 122 (3): 295-304, (2000)
20. Microchip, Microchip Technical Library CD-ROM Third Edition, **Microchip Technology Inc.**, Arizona, (2001)

EKLER

EK -1 Kontrol Kartı Programı

```

//*****//
#include <16F877.h>
#define device PIC16F877 ADC=10
#define fuses XT,NOPROTECT,NOWDT,PUT,NOLVP
#define use delay(clock=4000000)
#define use rs232(baud=9600, xmit=PIN_c6,rcv=PIN_c7)
#define rom 0x2100={00,0,00,00,00,00,00,00,00,00,00,00,00,00}

#define byte trisd=0x88
#define byte trise=0x89
#define byte trisc=0x87
#define byte trisb=0x86
#define byte trisa=0x85
#define byte porta=5

#define byte portb=6
#define byte portc=7
#define byte portd=8
#define byte porte=9

//programın tanımlamaları
#define byte abcd1=portd // multiplexerin data select girişleri
#define byte segment1=porta // PORTA çıkışındaki ledeler
#define bit w=portb.1 // multiplexerin çıkış pini
#define bit SR_clock=portb.4 // shift registerin clock pals pini!!bu bitleri çıkış yapmayı unutma
#define bit SR_data=portb.0 // shift registerin seri data pini!!!bu bitleri çıkış yapmayı unutma
#define bit hata_led=portb.5
#define define buffer_size 6
#define define BUFFER_RCSIZE 6
#define define BUFFER_TRSIZE 6
static boolean
timer_flag,sn_10,sn_p,T_d,gray_f=false,gray_f1=false,Tus_D1;
static boolean stop_bit,start_bit,b_flag1,b_flag,hata;
static int sh_kod[17]=
{0x01,0x02,0x04,0x08,0x10,0x20,0x40,0x80,0x01,0x02,0x04,0x08,0x10,0x20,0x40,0x80,0x00};//,0xE7,0X7D,0X39,0X5E,0X79,0X40};
static int
s[7],buffer_rce[BUFFER_RCSIZE],buffer_rc[BUFFER_RCSIZE],buffer_rceindex,buffer_rcindex,buffer_tr[BUFFER_TRSIZE],buffer_trindex,buffer[BUFFER_SIZE],buffer_index;

```

```

static int data_get,tus_D_old;
static int gray,m,led,latch1,bos1,bos2,bos11,bos22;
static int sh_reg_k,rh_h,rh_l,reg_count,tus_d,abcd,TUS_d2;
static int sn10,durum,saat,dak,sn,sn_count,tus_d1=0,tus_dh=0;
c
static int rx,x;
#int_rtcc
interrupt_altyordami() {
    timer_flag=1;
}
#int_rda
//disable_interruptions(INT_RTCC);
void serial_isr() {
    int i,j;
    data_get=getc();
    if((data_get==0x0F)&&(buffer_rcindex == 0)){
        start_bit=true;
    }
    if(data_get==0xF0){
        stop_bit=true;
    }
    if((start_bit)&&(buffer_rcindex != (buffer_rcsize))){
        buffer_rc[buffer_rcindex]=data_get;
        buffer_rcindex=buffer_rcindex+1;
    }
    if ((stop_bit)&&(buffer_rcindex == (buffer_rcsize))){
//if1
        start_bit=false;
        stop_bit=false;
        b_flag1=true; //hat saglam; iletim
var....
        for(i=1;i<=buffer_rcsize;i++){

            if(buffer_rce[buffer_rceindex]==buffer_rc[buffer_rceindex]){
                buffer_rceindex++;
                b_flag=true;
            }//end of if2
            else {
                i=(buffer_rcsize);
                buffer_rceindex=0;
                for(j=1;j<=buffer_rcsize;j++){

```

```

        buffer_rce[buffer_rceindex]=buffer_rc[buffer_rceindex];
            buffer_rceindex++;
    }
        buffer_rceindex=0;
        b_flag=false;
        hata=true;
    }//end of else
}//end of for
    buffer_rceindex=0;
    buffer_rcindex=0;

} //end of if1
} //end of void

void periyodik_is() {           // Saat altyordamı

    sn_count++;

    if(!(sn_count%25))
        sn_10=true;
    if (sn_count==250) {
        sn_count=0;
        sn_p = true;
        sn++;
        if (sn==60){
            dak++;
            sn=0;
            if (dak==60){
                saat++;
                dak=0;
                if (saat==23)
                    saat=0;
            }
        }
    }
}

void multiplexer(){           // 74AS150 Multiplexer altyodamı
    byte m,rsn,rx,x;

    rsn=3;

```

```

//output_d(0x00); // RB0, RB1, RB2, RB3 data select
pinleri
/*for(m=0;m<=15;m++){
    abcd=m;
    if(w==1){ // multiplexerin çıkış pini (RB1)
        Tus_D1=true;
        Tus_db[rsn]=abcd;
        tus_d=abcd;
        rsn++;
        //m=15;
    }
    else{
        if (Tus_D1){
            Tus_D1=false;
            Tus_DH=0;
            Tus_DL=0;
            rsn=0;
            T_D=true;
        }
    }
    //abcd=m;
} //end of for
*/
//if(tus_d1){
for (rx=0;rx=16;rx++){
x=tus_db[rx];
if(x<=7){
    bit_set(tus_d1,x);
}
else{
    bit_set (tus_dh,x-8);
}
}
//}
} //end of multiplexer

void sh_reg(){
    byte r,n;
    rh_h=tus_dh;
    rh_l=tus_d1;
    for(r=1;r<=8;r++) {
        if ( !bit_test(rh_h,7)){
            SR_data=1; // seri giriş
        }
    }
}

```

```

        else{
            SR_data=0;
        }
        SR_clock=0;    // clock palsi
        SR_clock=1;
        rh_h=rh_h<< 1;
    }//end of for
    for(n=1;n<=8;n++) {
        //sh_reg_k1 = ((sh_reg_k)&&(0x01));
        if ( !bit_test(rh_l,7)){
            SR_data=1;    // seri giriş
        }
        else{
            SR_data=0;
        }
        SR_clock=0;    // clock palsi
        SR_clock=1;
        rh_l=rh_l<< 1;

    }//end of for
} //end of void
//*****transmit alt
yordamı*****
void buffer_transmit(){

    //if (sn_p){
        //sn_p=false;
        buffer_tr[0]=0x0F;
        buffer_tr[1]=latch1;//+0x30;
        buffer_tr[2]=Tus_D;//+0x30;
        buffer_tr[3]=bos1;
            buffer_tr[4]=bos2;//+0x30;
        buffer_tr[5]=0xF0;//+0x30;

        while (buffer_trindex != buffer_trsize){
            putchar(buffer_tr[buffer_trindex]);
            buffer_trindex=buffer_trindex+1;
        }
        buffer_trindex=0;
    //}
} //end of buffer()

//*****

```

```

main(){
    sn_p=false;
    sn_10=false;
    gray_f=false;
    saat=dak=sn=0;
    buffer_index=0;
    sh_reg_k=0;
    set_rtcc(0);

    setup_counters( RTCC_INTERNAL, RTCC_DIV_16);
    enable_interrupts(INT_RTCC);
    enable_interrupts(GLOBAL);
    enable_interrupts(INT_RDA);
    set_tris_b (0xc2);
    set_tris_a (0x00);
    set_tris_d (0x00);
    set_tris_c (0xc7);
    output_A(0x00);
    //output_B(0x00);
    //output_D(0x00);
    output_E(0x00);
    //output_D(0xF0);
    while(TRUE){
        portA=0xFF;
        portE=0xFF;
        if (timer_flag) {
            timer_flag=false;
            periyodik_is();
            //multiplexer();
            for (rx=0;rx=16;rx++){
                x=tus_db[rx];
                if(x<=7){
                    bit_set(tus_d1,x);
                }
                else{
                    bit_set (tus_dh,x-8);
                }
            }//end of for
            tus_d1=5;
            sh_reg();

            if(sn_p)
            {
                sn_p=false;
            }
        }
    }
}

```

```
        porta=sn;
    }
}

} //end of while
} //end of main
```

Ek -2

Kabin Kartı İçin Programı

```

//*****//
#include <16F877.h>
#define PIC16F877 ADC=10
#define fuses XT,NOPROTECT,NOWDT,PUT,NOLVP
#define use delay(clock=4000000)
#define use rs232(baud=9600, xmit=PIN_c6,rcv=PIN_c7)
#define rom 0x2100={00,0,00,00,00,00,00,00,00,00,00,00,00,00}
#define byte trisd=0x88
#define byte trise=0x89
#define byte trisc=0x87
#define byte trisb=0x86
#define byte trisa=0x85
#define byte porta=5
#define byte portb=6
#define byte portc=7
#define byte portd=8
#define byte porte=9

//programın tanımlamaları
#define byte abcd=porta//portd // multiplexerin data select girişleri
#define bit SR_clock_d=porte.0 //b.4 // shift registerin clock pals pini!!bu bitleri çıkış yap
#define bit SR_data_d=porte.1 //b.0 // shift registerin seri data pini!!!bu bitleri çıkış yap
#define bit w=portc.5 //b.1 // multiplexerin çıkış pini
#define bit pompa=portc.2
#define bit mknts=portc.4
#define bit sgmnt=porte.2
#define bit SR_clock=porta.4 // shift registerin clock pals pini!!bu bitleri çıkış yapmayı unutma
#define bit SR_data=porta.5 // shift registerin seri data pini!!!bu bitleri çıkış yapmayı unutma
#define bit hata_led=portc.3
#define BUFFER_RCSIZE 7
#define BUFFER_TRSIZE 7
#define BUFFER_RCESIZE 7
#define BUFFER_SIZE 7
static boolean
timer_flag,sn_10,sn_p,T_d,gray_f=false,gray_f1=false,Tus_D1;
static boolean stop_bit,start_bit,b_flag1,b_flag,hata,l_d;

```

```

static int gray_dec[16]=
{0,1,3,2,7,6,4,5,15,14,12,13,8,9,11,10}; //gray_to_decimal donusum
tablosu
static int gray_dec1[16]=
{10,11,9,8,13,12,14,15,5,4,6,7,2,3,1,0}; //gray_to_decimal donusum
tablosu
//static int ss_kod[16]=
{0x3F,0x06,0x5B,0x4F,0x66,0x6D,0x7D,0x07,0x7F,0x6F,0xBF,0x86,0xDB,0x
CF,0xE6,0xED}; //,0xE7,0x7D,0x39,0x5E,0x79,0x40};
static int sh_kod[17]=
{0x01,0x02,0x04,0x08,0x10,0x20,0x40,0x80,0x01,0x02,0x04,0x08,0x10,0x
20,0x40,0x80,0x00}; //,0xE7,0x7D,0x39,0x5E,0x79,0x40};
static int
    buffer_rce[BUFFER_RCSIZE],buffer_rc[BUFFER_RCSIZE],buffer_rcei
ndex,buffer_rcindex,buffer_tr[BUFFER_TRSIZE],buffer_trindex,buffer[B
UFFER_SIZE],buffer_index;
static int data_get;
static int graye,gray,m,sh_reg_d,rh_d,led;
static int sh_reg_k,rh_h,rh_l,reg_count,tus_d,sayi=15;
static int
sn10,durum,saat,dak,sn,sn_count,tus_D_olD,R1,T1,T3,sn_bag,sn_bag_cou
nt;
static int
tusL,tusH,tuseL,tuseH,tuseL_olD,tuseH_olD,ledH,ledL,ledL_olD,ledH_ol
D;
static int ss_kod[16]=
{0x4F,0x5B,0x06,0x3F,0x06,0x5B,0x4F,0x66,0x6D,0x7D,0x07,0x7F,0x6F,0x
BF,0x86,0xDB}; //,0xE7,0x7D,0x39,0x5E,0x79,0x40};
#int_rtcc
interrupt_altyordami() {
    timer_flag=1;
}
#int_rda
//disable_interrupts(INT_RTCC);
void serial_isr() {
    int i,j;
    hata_led=0;
    data_get=getc();
    if((data_get==0x0F)&&(buffer_rcindex == 0)){
        start_bit=true;
    }
    if(data_get==0xF0){
        stop_bit=true;
    }
}

```

```

if((start_bit)&&(buffer_rcindex != (buffer_rcsize))){
    buffer_rc[buffer_rcindex]=data_get;
    buffer_rcindex=buffer_rcindex+1;
    hata_led=0;
}
if ((stop_bit)&&(buffer_rcindex == (buffer_rcsize))){
//if1
start_bit=false;
stop_bit=false;
b_flag1=true; //hat saglam; iletim
var....
for(i=1;i<=buffer_rcsize;i++){

if(buffer_rce[buffer_rceindex]==buffer_rc[buffer_rceindex]){
    buffer_rceindex++;
    b_flag=true;
} //end of if2
else {
    i=(buffer_rcsize);
    buffer_rceindex=0;
    for(j=1;j<=buffer_rcsize;j++){

buffer_rce[buffer_rceindex]=buffer_rc[buffer_rceindex];
    buffer_rceindex++;
    }
    buffer_rceindex=0;
    b_flag=false;
    hata=true;
} //end of else
} //end of for
    buffer_rceindex=0;
buffer_rcindex=0;
} //end of if1
} //end of void

void periyodik_is() { // Saat altyordam

    sn_count++;
    sn_bag_count++;
    if(!(sn_count%25))
        sn_10=true;
    if(sn_bag_count==250){
        sn_bag_count=0;

```

```

        sn_bag++;
    }
    if (sn_count==250) {
        sn_count=0;
        sn_p = true;
        sn++;
        sn_bag++;
        if (sn==60){
            dak++;
            sn=0;
            if (dak==60){
                saat++;
                dak=0;
                if (saat==23)
                    saat=0;
            }
        }
    }
}

void sh_reg_T(int sh_reg_d ){                // 74LS164 Shift Register
Altyordamı
    //output_D(0x00);
    byte d;
    rh_d=sh_reg_d;
    rh_d=ss_kod[rh_d];
    for(d=1;d<=8;d++) {
        //if (!bit_test(rh_d,7)){
        if (bit_test(rh_d,7)){
            SR_data_d=1;    // seri giriş
        }
        else{
            SR_data_d=0;
        }
        SR_clock_d=0;    // clock palsi
        SR_clock_d=1;
        rh_d=rh_d<< 1;
    }//end of for
} //end of void

void gray_kod(){
    gray=portd;

```

```

//gray=swap(gray);
gray=gray^0xff;
gray=gray&0x0f;
if(mknts==1){
    if((gray==0)||(gray==1)||(gray==3))
        sgmnt=1;
    else
        sgmnt=0;
    gray=gray_dec[gray]; //grayden decimale donusum
}
else{
    if((gray==15)||(gray==14)||(gray==12))
        sgmnt=1;
    else
        sgmnt=0;
    gray=gray_dec1[gray]; //grayden decimale donusum
}
if(gray!=graye){
    graye=gray;
    gray_f=true;
}
}
void multiplexer(){ // 74AS150 Multiplexer altyodam
byte m;
tusL=0;
tusH=0;
for(m=0;m<=15;m++){
    abcd=m;
    Tus_D1=false;
    if(w==1){ //w multiplexerin
        Tus_D1=true;
        //T_D=false;
        if(m<=7){
            tusL=sh_kod[m];
            tuseL=tuseL|tusL;
        }
        if(m>7){
            tusH=sh_kod[m];
            tuseH=tuseH|tusH;
        }
    }
    else{
        if(m<=7){

```

```

        tusL=sh_kod[m];
        tusL=tusL^0xFF;
        tuseL=tuseL&tusL;
    }
    if(m>7){
        tush=sh_kod[m];
        tush=tush^0xFF;
        tuseH=tuseH&tush;
    }
}
} //end of for
if((tuseL==tuseL_old)&&(tuseH==tuseH_old))
    T_D=false;
else{
    tuseL_old=TuseL;
    tuseH_old=TuseH;
    T_D=true;
}
} //end of multiplexer

```

```

void sh_reg(int sh_reg_k ,int sh_reg_k1){
    byte r,n;
    rh_h=sh_reg_k;
    rh_l=sh_reg_k1;
    for(r=1;r<=8;r++) {
        //if ( !bit_test(rh_h,7)){
        if ( bit_test(rh_h,7)){
            SR_data=1;    // seri giriş
        }
        else{
            SR_data=0;
        }
        SR_clock=0;    // clock palsi
        SR_clock=1;
        rh_h=rh_h<< 1;
    } //end of for
    for(n=1;n<=8;n++) {
        //sh_reg_k1 = ((sh_reg_k)&&(0x01));
        //if ( !bit_test(rh_l,7)){
        if ( bit_test(rh_l,7)){
            SR_data=1;    // seri giriş
        }
        else{

```

```

        SR_data=0;
    }
    SR_clock=0;    // clock p̄alsi
    SR_clock=1;
    rh_l=rh_l<< 1;
} //end of for
} //end of void
//*****transmit alt
yordamı*****
void buffer_transmit(){

    //if (sn_p){
        //sn_p=false;
        hata_led=1;
        buffer_tr[0]=0x0F;
        buffer_tr[1]=T1;//g0,g1,g2,g3,mv1,mv2,mntk,ayks
        buffer_tr[2]=T3;//+0x30;
        buffer_tr[3]=tuseH;    //rev,ayon,yyon,ok1,ok2,ok3,ok4
            buffer_tr[4]=tuseL;//+0x30;
        buffer_tr[5]=0x00;//+0x30;
        buffer_tr[6]=0xF0;//+0x30;

        while (buffer_trindex != buffer_trsize){
            putc(buffer_tr[buffer_trindex]);
            buffer_trindex=buffer_trindex+1;
        }
        buffer_trindex=0;
    //}
} //end of buffer()

//*****

main(){

    setup_counters( RTCC_INTERNAL, RTCC_DIV_16);
    enable_interrupts(INT_RTCC);
    enable_interrupts(GLOBAL);
    enable_interrupts(INT_RDA);
    set_tris_e (0x00);
    set_tris_b (0xff);
    set_tris_a (0x00);
    set_tris_d (0xff);
    set_tris_c (0xf0);

```

```

sn_p=false;
sn_10=false;
buffer_index=0;
sh_reg_k=0;
set_rtcc(0);
hata_led=1;
sh_reg(0,0);
gray_kod();
gray_f=false;
sh_reg_T(gray);

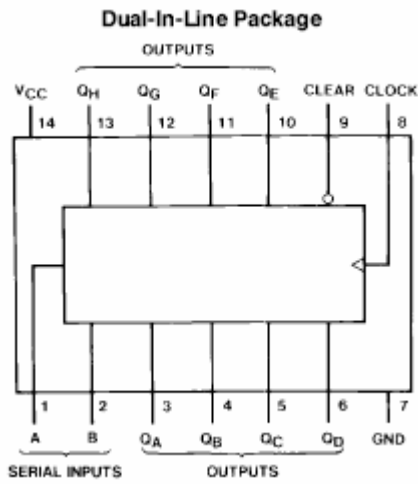
while(TRUE){
    if (timer_flag) {
        timer_flag=false;
        periyodik_is();
        T1=portd; //g0,g1,g2,g3,mv1,mv2,mtk,ayks
        T3=portb; //rev,ayon,yyon,ok1,ok2,ok3,ok4
        if (sn_10){
            sn_10=false;
            buffer_transmit();
        }
        gray_kod();
        if (gray_f){
            gray_f=false;
            sh_reg_T(gray);
            buffer_transmit();
        }
        multiplexer();
        if(T_D){
            T_D=false;
            buffer_transmit();
        }
        if (b_flag){
            b_flag=false;
            sn_bag=0;
            sn_bag_count=0;
            hata_led=1;
            R1=buffer_rce[1];
            ledH=buffer_rce[3];
            ledl=buffer_rce[4]; //2 kez teyyid edilmiş saat bilgisi
            R1=R1 & 0x07;
            portc=portc&0xF8;
            portc=portc|R1;
        }
    }
}

```


Ek - 3

74LS164'ün özellikleri:

- Tipik olarak saat frekansı 36MHz
- Tipik olarak güç dağılımı 80 mW
- Enable/disable seri girişler
- Asenkron temizleme
- Tamamen tamponlanmış saat pulsü ve seri girişler



Inputs		Outputs				
Clear	Clock	A	B	QA	QB	... QH
L	X	X	X	L	L	... L
H	L	X	X	QA0	QB0	... QH0
H	↑	H	H	H	QA _n	... QG _n
H	↑	L	X	L	QA _n	... QG _n
H	↑	X	L	L	QA _n	... QG _n

H = High Level (steady state), L = Low Level (steady state)

X = Don't Care (any input, including transitions)

↑ = Transition from low to high level

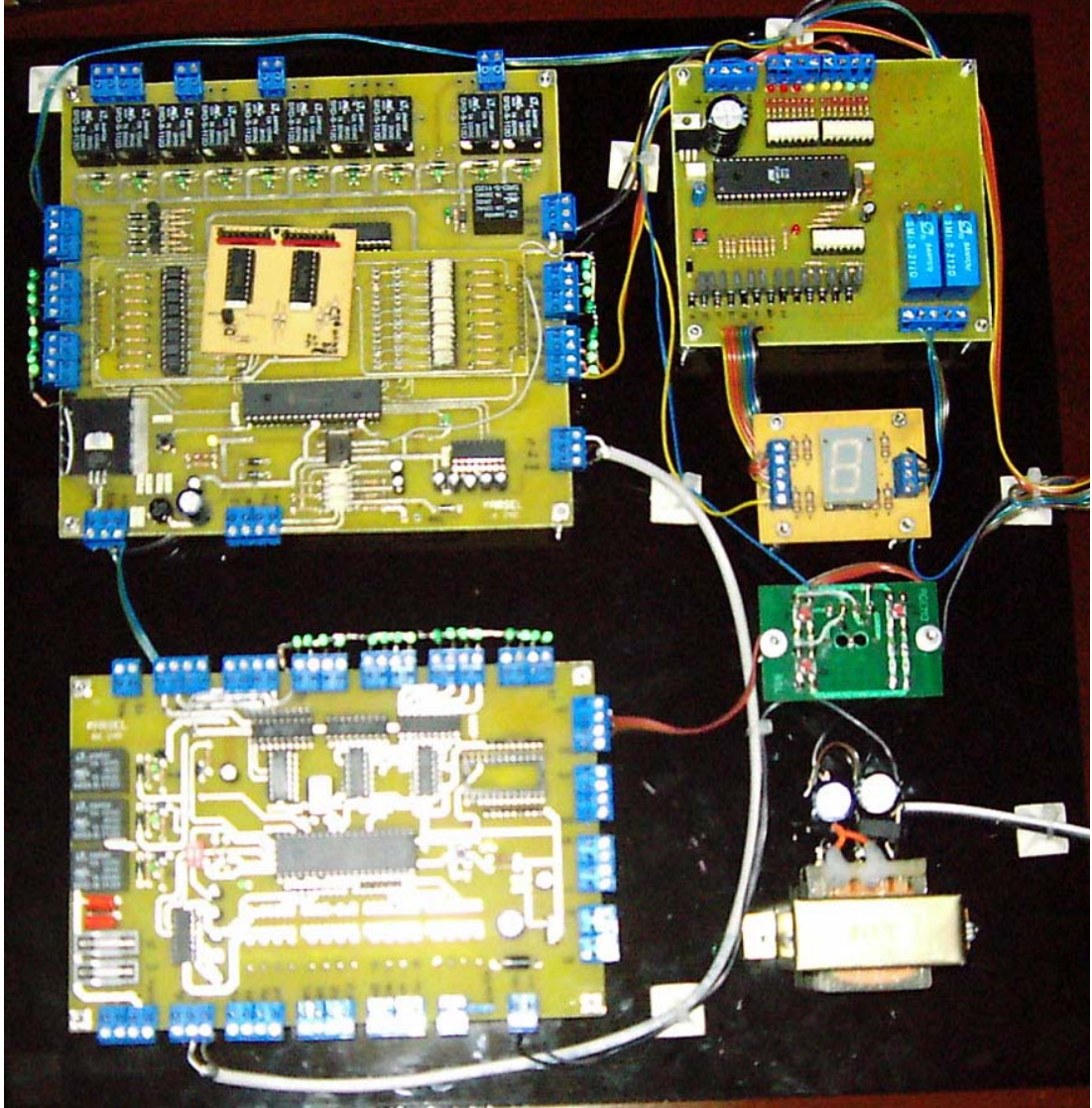
QA₀, QB₀, QH₀ = The level of QA, QB, or QH, respectively, before the indicated steady-state input conditions were established.

QA_n, QG_n = The level of QA or QG before the most recent ↑ transition of the clock; indicates a one-bit shift.

74LS164'ün bağlantı şeması ve fonksiyon tablosu

Ek-4

Prototipe Ait Resimler

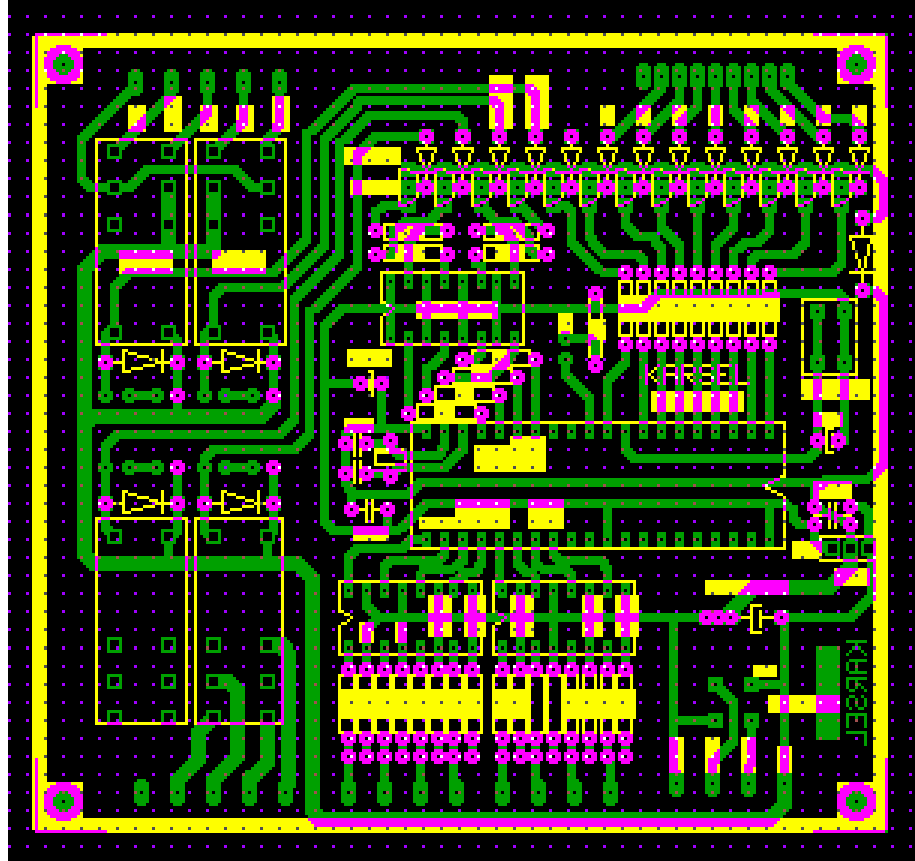




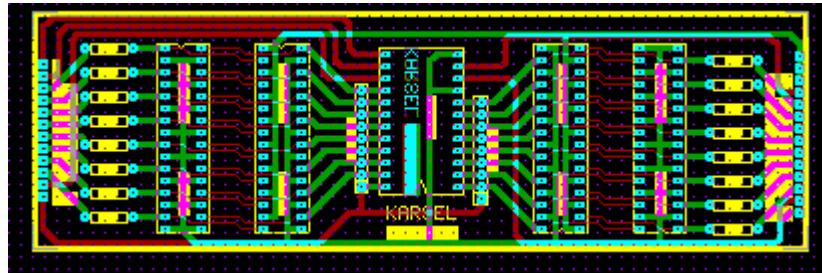
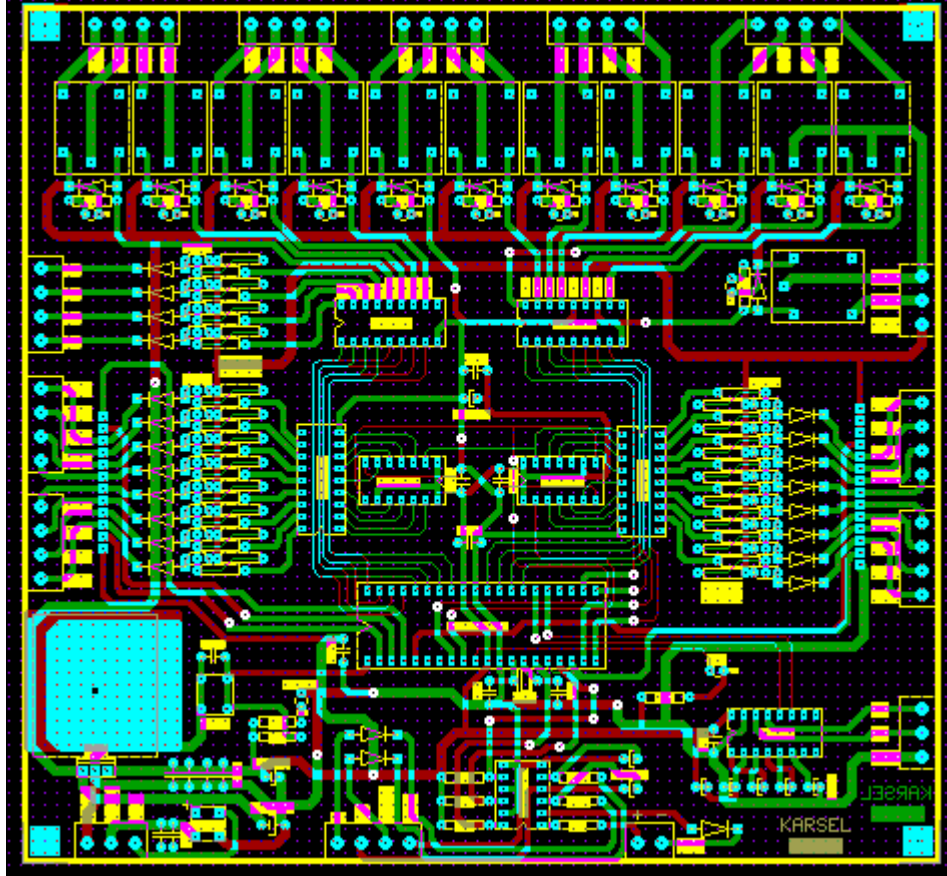
Ek -5

Baskı Devre Şemaları

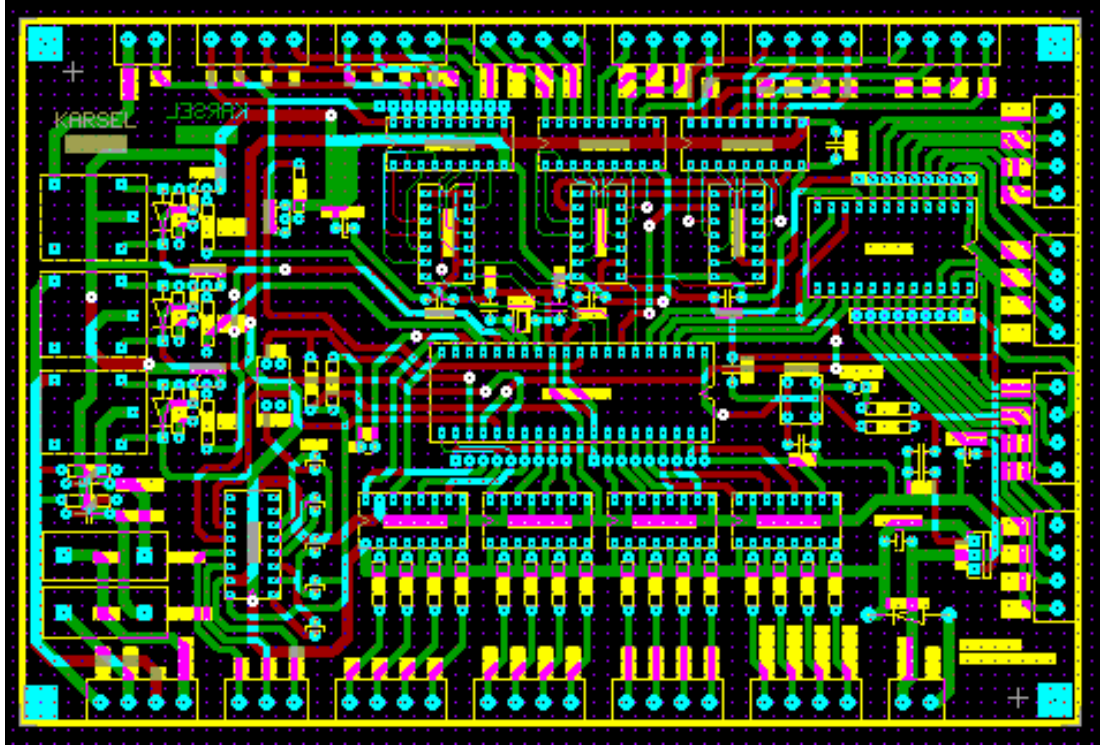
Kumanda kartı baskı devre şeması



Kontrol kartı baskı devre şeması



Kontrol kartı baskı devre şeması



ÖZGEÇMİŞ

1979 yılında Kırşehir'de doğdu. İlk, orta ve lise öğrenimini Ankara'da tamamladı. Lisans eğitimini 1997-2001 yılları arasında Gazi Üniversitesi Teknik Eğitim Fakültesi Elektronik Bilgisayar Eğitimi Bölümünde tamamladı. Aynı yıl Gazi Üniversitesi Fen Bilimleri Enstitüsü Elektronik Bilgisayar Eğitimi Anabilim dalında yüksek lisans eğitimine başladı. Halen Gazi Üniversitesi Fen Bilimleri Enstitüsünde araştırma görevlisi olarak çalışmaktadır.