



**ELEKTRİK MÜHENDİSLERİ ODASI İSTANBUL ŞUBESİ**  
**2004-2005 ÖĞRETİM YILI PROJE YARIŞMASI**

**BİLGİSAYAR KONTROLLÜ KAMERALI**  
**ROBOT KOL PROJESİ**

**Proje Yöneticisi : Yrd. Doç. Dr. Lale ÖZYILMAZ**

**HAZIRLAYANLAR**

**Diğer ÖZGÜR**

**H. Orhun TAŞKAYA**

**İstanbul, 2005**

# İÇİNDEKİLER

ÖNSÖZ .....	iv
GİRİŞ .....	1
Projede Kullanılan Malzemeler Hakkında Temel Bilgiler .....	2
1. ATMEL AT89C52 Mikrodenetleyicisi .....	2
1.1. AT89C52 Mikrodenetleyicisi Genel Özellikleri .....	2
1.1.1 Bacak Tanımlamaları.....	3
1.1.2. Hafıza Yapısı.....	5
1.1.2.1. Program ve Veri Hafızaların Ayrılması.....	5
1.1.2.2. Saklayıcılar.....	7
1.1.2.3.Bit Adreslenebilir Veri Hafıza.....	7
1.1.2.4.Özel Fonksiyon Saklayıcıları (SFR).....	7
1.1.2.5. Program Durum Kelimesi (PSW).....	8
1.1.2.6. Komut İşaretçisi (PC).....	9
1.1.3. CPU Zamanlaması.....	10
2. SERİ PORT İLE HABERLEŞME.....	11
2.1. Seri Haberleşme ve Ağ Yapıları.....	11
2.1.1. Seri Data Modu 0.....	12
2.1.2. Seri Data Modu 1.....	12
2.1.3. Seri Data Modu 2 (Çoklu İşlemci Modu).....	14
2.1.4. Seri Data Modu 3.....	14
2.2. Seri Porta Yazma.....	14
2.3. Seri Porttan Okuma.....	15
2.4. RS-232 İletişiminin Temel Kavramları.....	15
3. SERVO MOTORLAR.....	19

3.1. Servo Motor Temel Fonksiyonları.....	20
3.2. Servo Motor Kontrolü.....	20
4. ROBOT KOL PROJESİ ARAYÜZ PROGRAMI.....	22
4.1. Manuel Kontrol.....	23
4.2. Değer ile Sıralı Kontrol.....	23
4.3. Demo.....	24
4.4. Başlangıç Konumu.....	25
SONUÇ.....	26
KAYNAKLAR .....	27

## ÖNSÖZ

Sanayide yaşanan gelişmeler, insan gücü maliyetlerinin düşürülmesi ve kalite artırılmasının amaçlanması, insanları robot kol kullanmaya zorunlu kılmıştır. Günümüzde hemen hemen her fabrikanın üretim bandında, robot kollar insanların yerini almıştır ve almaya da devam edecektir. Yaşanan bu gelişmeler göz önünde bulundurularak robot kol projesi gerçekleştirilmiştir.

Hazırladığımız bu tez çalışmasında, yardımlarını hiçbir zaman esirgemeyen tez danışmanımız Yrd. Doç. Dr. Lale Özyılmaz'a, öğrenim hayatımız boyunca bizlere emeği geçen tüm hocalarımıza ve projemize katkılarından dolayı EMO İstanbul Şubesi'ne teşekkürü bir borç biliriz.

Ayrıca bugünlere gelmemizde en büyük paya sahip olan, bizleri her konuda daima destekleyen ailelerimize de teşekkür ederiz.

## GİRİŞ

Bu projenin gerçekleştirilmesindeki öncelikli amacımız, ulaşılamayan veya sağlık açısından zararlı bölgelerde bilgi toplama ve müdahale görevini üstlenebilecek bir makinenin gerçekleştirilmesidir. Bu makinenin bir robot kol olabileceğine karar verdik ve Bilgisayar “Kontrollü Kameralı Robot Kol projesini gerçekleştirdik.

Bilgisayar Kontrollü Kameralı Robot Kol üzerindeki kamera sayesinde bulunduğu ortamdan kontrol edildiği ortama görüntü aktarabilmektedir. Bu aktarım sayesinde kullanıcı kolun bulunduğu ortamı görebilmekte ve bulunduğu ortamdaki bilgisayar üzerinden kola istediği hareketleri yaptırabilmektedir.

Robot Kolun hareketleri üç farklı şekilde yaptırılmaktadır. Birincisi ara yüz programı üzerinde tuşlara mouse ile tıklanarak yapılan kontroldür. İkincisi klavye üzerinden önceden belirlenmiş tuşlara basılı tutularak yapılan kontroldür. Üçüncü ve son kontrol ise arayüz programında bulunan değer ile sıralı kontroldür.

Bilgisayar Kontrollü Kameralı Robot Kol Projesini geliştirirken hedefimiz uygulanması istenen tüm fonksiyonlara cevap verebilecek nitelikte bir prototip hazırlamaktır. Hazırladığımız prototipte kontrol uyumluluğu ve maliyet açısından RC Servo motorlar kullanılmıştır. Servo motorların kontrolleri ve bilgisayar haberleşmesinde Atmel 89C52 mikrodenetleyicisi kullanılmaktadır. Baskı devre tarafımızdan tasarlanıp çizim için Proteus Ares kullanılmıştır. Bilgisayar arayüz programı Visual Basic 6.0 da oluşturulmuştur. Kamera olarak USB Webcam kullanılmıştır. Arayüz programının çalışması ile birlikte kamera aktif olmakta ve aldığı görüntüyü bilgisayar ekranında açılan pencereye iletmektedir.

## PROJEDE KULLANILAN MALZEMELER HAKKINDA TEMEL BİLGİLER

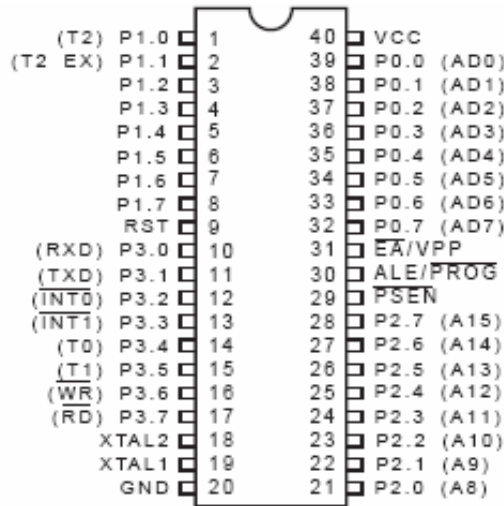
### 1. ATMEL AT89C52 Mikrodenetleyicisi

AT89C52 mikrodenetleyicisi düşük güçlü, RISC (reduced instruction set computer) mimarisi tabanlı bir mikrodenetleyicidir. Pek çok mikrodenetleyicide olduğu gibi AT89C52’de de Harvard mimarisi kullanılmıştır. Bu mimaride veri belleği ile komut belleği birbirinden bağımsızdır. Bu mimarinin aksine Von Neuman mimarisinde ise veri ve komutlar aynı bellek içerisinde yer almaktadır.

Bu yapı farklılığına rağmen her iki mimaride de saklayıcı olarak adlandırılan özel bellek alanları mevcuttur. Saklayıcı temel olarak işlemci içerisinde bulunur ve işlemcinin çok önemli bir yapıtaşıdır. Saklayıcıların erişim süreleri belleklere göre çok daha hızlıdır. Bu nedenle MCU’da kullanılan pek çok komut bu saklayıcılar kullanılarak gerçekleştirilmektedir. AT89C52’de kullanılan saklayıcılar bellek haritası içerisinde yer aldıklarından veri belleği olarak da kullanılabilirler.

AT89C52’de bellek haritasının bir başka bölümü de donanım kontrolü için ayrılmıştır. Bunlar Giriş/Çıkış (G/Ç) kontrol portları ya da G/Ç saklayıcıları olarak adlandırılır. Bu saklayıcılar kullanılarak değişik donanım kaynakları kontrol altında tutulabilir.

Aşağıdaki şekilde AT89C52 işlemcisinin 40-uçlu tümdevre şekli ve uçlarının lojik olarak gruplandırılmış gösterimi verilmiştir. Aşağıda sırasıyla, AT89C52’nin temel çekirdek mimarisinde yer alan, tümdevre uçları ve görevleri tanıtılacaktır.



Şekil 1.1 AT89C52 mikrodenetleyicisi pin tanımlamaları

#### 1.1. AT89C52 Mikrodenetleyicisi Genel Özellikleri

AT89C52 aşağıdaki özelliklere sahiptir:

- Kontrol uygulamalarına yönelik 8-bit CPU
- Yoğun Boolean işlemleri yapabilme (tek-bit lojik işlemler) özelliği

- 64K Program Hafıza (Program Memory) adres alanı
- 64K Veri Hafıza (Data Memory) adres alanı
- 8K tümdevre-üzeri (on-chip) Program Hafıza
- 256 Byte tümdevre-üzeri RAM
- 32 tane *iki* yönlü adreslenebilir I/O hatları
- 3 tane 16-bit Zamanlayıcı / Sayıcı (Timer / Counter)
- Full duplex UART (Universal Asynchronous Receiver Transmitter)
- İki öncelik seviyesine sahip 6-kaynak / 5-vektörlü kesme donanım yapısı

### 1.1.1 Bacak Tanımlamaları

#### Vcc

Besleme gerilimi.

#### Vss

Tümdevre toprağı.

#### Portlar

AT89C52'deki portların hepsi iki yönlüdür. Her bir port, bir latch (P0 ile P3 arasındaki bir SFR), bir çıkış sürücüsü (driver) ve bir giriş bufferına sahiptir.

#### Port 0 (P0)

8-bit open-drain iki yönlü I/O portudur. P0 uçlarına program ile lojik 1 yazılması durumunda, yüksek empedanslı giriş uçları olarak kullanılabilir.

P0 aynı zamanda, harici program hafızaya ve veri hafızaya erişimlerde, seçmeli (multiplexed) düşük-değerli Adres/Veri Yolu (AD<sub>7</sub>-AD<sub>0</sub>) olarak da kullanılabilir. Bu durumda çalışırken P0 dışarıya lojik 1 işareti göndermede, dahili pull-uplar olarak çalışan FET transistörlerini kullanır. P0, EPROM'lu ürünlerin programlanmasında, işlem kod bytelarını alır ve ROM ve EPROM program doğrulamasında (verification), işlem kod byteları, buradan dışarı çıkar. Program doğrulama işleminde, harici pull-up dirençlerine ihtiyaç vardır.

#### Port 1 (P1)

8-bit dahili pull-up dirençlere sahip iki yönlü I/O portudur. Uçlarına program ile lojik 1 yazılan P1, dahili pull-uplar ile yüksek lojik 1 seviyesine çekilir ve bu durumda giriş olarak kullanılabilir.

Tümdevre içinde bulunan dahili pull-uplar yüzünden harici olarak, pull-uplar ile düşük seviyeye çekilen uçlardan akım (veri kitaplarındaki I<sub>IL</sub>) tümdevreden dışarı doğru çıkar.

EPROM'lu ürünlerin programlanmasında, ROM ve EPROM'ların program doğrulama işlemleri esnasında P1, düşük-değerli adres bytelarını alır.

**Port 2 (P2)**

8-bit dahili pull-up dirençlere sahip iki yönlü I/O portudur. Uçlarına program ile lojik 1 yazılan P2, dahili pull-uplar ile yüksek lojik 1 seviyesine çekilir ve bu durumda giriş olarak kullanılabilir.

Tümdevre içinde bulunan dahili pull-uplar yüzünden, harici olarak, pull-uplar ile düşük seviyeye çekilen uçlardan akım (veri kitaplarındaki  $I_{IL}$ ) tümdevreden dışarı doğru çıkar.

Harici Program Hafızadan okumalarda ve 16-bit adresler kullanan (MOVX @DPTR) harici Veri Hafızaya erişimlerde, P2 yüksek-değerli adres byteını çıkarır. Bu durumda çalışırken, P2 dışarı lojik 1 işareti göndermede, dahili pull-up dirençlerini kullanır.

8-bit adresler kullanan (MOVX @Ri) Harici Veri Hafızaya erişimlerde, P2 portu, P2 SFR içeriklerini dışarıya gönderir.

EPROM'lu ürünlerin programlanmasında, ROM ve EPROM'ların program doğrulama işlemleri esnasında, P2 yüksek-değerli adres bytelarını alır.

**Port 3 (P3)**

8-bit dahili pull-up dirençlere sahip iki yönlü I/O portudur. Uçlarına program ile lojik 1 yazılan P3, dahili pull-uplar ile yüksek lojik 1 seviyesine çekilir ve bu durumda giriş olarak kullanılabilir.

Tümdevre içinde bulunan dahili pull-uplar yüzünden, harici olarak, pull-uplar ile düşük seviyeye çekilen uçlardan akım (veri kitaplarındaki  $I_{IL}$ ) tümdevreden dışarı doğru çıkar.

**Reset**

Osilatör çalışırken, 2 makine çevrimi kadar bu uç üzerindeki bir yüksek seviye (lojik 1) işlemciyi sıfırlar (reset).

**Xtal1**

Osilatör girişi.

**Xtal2**

Osilatör çıkışı.

**Ale / Prog (Address Latch Enable)**

Bu çıkış sinyali, harici hafızaya erişimlerde adresin düşük değerli byteını, çıkıştaki adres latchine tutturmada kullanılır. Bu uç aynı zamanda, EPROM'lu ürünlerin programlanması sırasında, program darbe girişidir (PROG).

**Ea / Vpp (External Access-EA )**



Bu uç düşük seviye (lojik 0) yapıldığında, işlemci 0000H'tan başlayıp FFFFH'a kadar hafıza hücrelerini, harici Program Hafızadan okuyacak hale getirilir. EA sinyali mutlaka 1 veya 0 seviyesinde olup boşta bırakılmamalıdır. Bu bacak aynı zamanda, EPROM'lu ürünlerin programlanması sırasında, programlama besleme voltajını (VPP) alır.

### **Psen** (Program Store Enable)

Bu çıkış, harici program hafızadan okuma sinyalidir. PSEN, harici program hafızadan okumalarda her makine çevriminde 2 kere aktif olur. Bununla beraber, harici veri hafızaya her erişimde PSEN sinyalinin 2 aktif durumu geçilmektedir. PSEN dahili program hafızadan okumalarda aktif değildir.

## **1.1.2. Hafıza Yapısı**

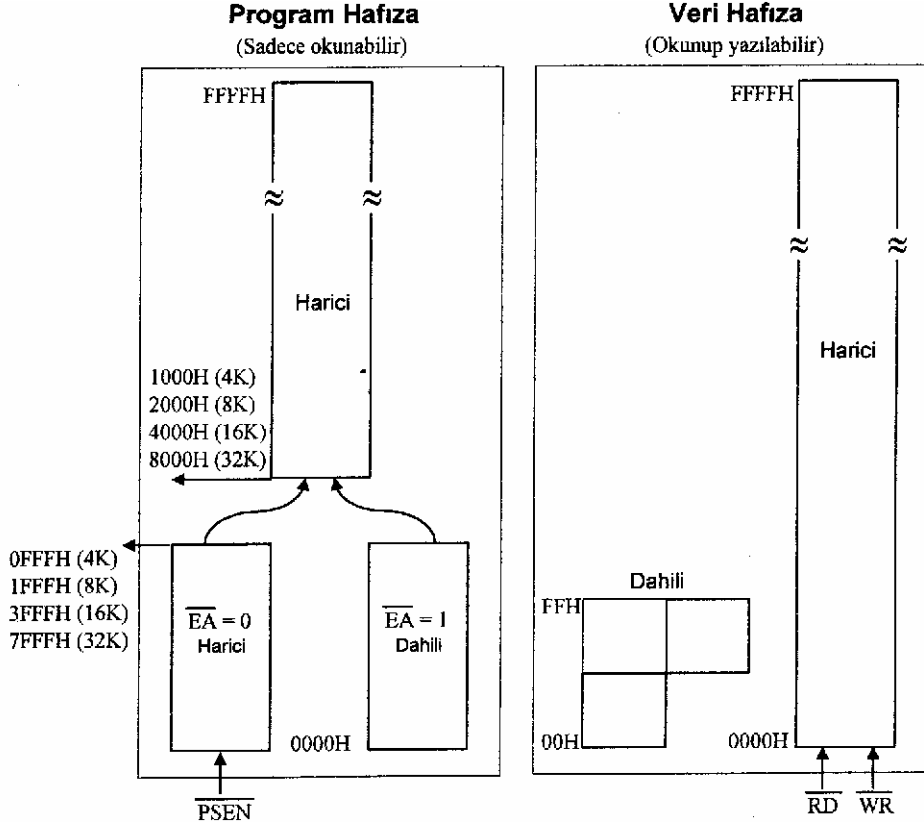
AT89C52 mikrodenetleyicisinin hafıza yapısı program ve hafızası olmak üzere iki kısımdan oluşmaktadır.

### **1.1.2.1. Program ve Veri Hafızaların Ayrılması**

AT89C52 mikroişlemcisinde program ve veri hafızasının lojik ayırımı, veri hafızasının 8-bit adreslerle erişimine imkan sağlar. Bu da, 8-bit AT89C52 MCU'su tarafından, verilerin daha hızlı erişilmesini, işlenmesini ve saklanmasını sağlar. Bununla beraber, harici 16-bit veri hafıza adresleri, AT89C52 DPTR (data pointer) saklayıcısı yoluyla da üretilebilmektedir.

Program hafıza, sadece okunabilir olup veri bu alana yazılamaz. Program Hafızasının boyu 64K byte kadar olabilir. Bu ailenin ROM veya EPROM'lu sürümlerinde, program hafızanın en düşük 4K, 8K..., 16K veya 32KB'lık kısmı, tümdevre üzerinde bulunabilir.

Veri hafıza, program hafızadan ayrı bir adres alanı işgal eder. 64KB'a kadar olan harici bir RAM hafıza, harici veri hafıza alanı içinde adreslenebilmektedir. CPU, harici veri hafızasına erişimlerde ihtiyaç oldukça, okuma ve yazma sinyalleri (RD/WR) üretmektedir.

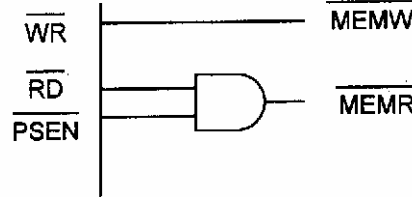


Şekil 1.2 AT89C52 hafıza haritası

AT89C52 yukarıda bahsedildiği gibi 64K harici Veri Hafızasını ve ayrıca 64K harici program hafızasını adresleyebilmektedir. Bunlar ayrı hafıza blokları olabilmektedir. Böylece toplam 128K'ya kadar hafıza, mikrodenetleyiciye eklenebilmektedir.

AT89C52, iki ayrı hafıza okuma sinyaline sahiptir, RD (P3.7) ve PSEN. Birincisi, harici Veri Hafızadan ve diğeri ise harici Program Hafızadan, bir byte okunurken aktif olmaktadır. Bu sinyaller aktif düşük (active low) sinyallerdir. Bütün harici kod, harici hafızadan okunur. Ek olarak, harici hafızadan gelen byte'lar MOVC gibi özel okuma komutlarıyla da okunabilir. Harici hafızadan okuma için, MOVX gibi ayrı komutlar da vardır. Yani, hafızanın hangi bloğundan okunacağını komutlar belirlemekte ve hafıza okuma çevriminde de, bu duruma karşı gelen sinyallerden RD veya PSEN aktif olmaktadır.

Bir hafıza bloğu, hem Veri Hafıza ve hem de Program Hafıza olarak, hafıza haritasına yerleştirilebilir. Şekil 1.3'te görüldüğü gibi, istenirse, RD ve PSEN sinyalleri bir AND kapısı ile birleştirilip kapı çıkışı, harici Program/Veri Hafıza okuma sinyali olarak kullanılarak, harici Program ve Veri Hafıza birleştirilebilir. Bu yol ile, iki girişten biri düşük olduğunda AND kapısının çıkışı düşük olacaktır. Bu sayede, aynı hafıza bloğundan okuma işlemi gerçekleştirilir.



Şekil 1.3 Harici program ve veri hafızaların birleştirilmesi

### 1.1.2.2. Saklayıcılar

Dahili RAM bölgesinde [00h..1Fh] arasındaki hafıza hücreleri, saklayıcılar için kullanılır. Her biri 8 saklayıcıdan oluşan 4 küme (bank) bulunur. Bir kümenin saklayıcıları, R0'dan R7'ye değişen gösterimle belirtilir. Saklayıcı kümeleri, 0, 1, 2 ve 3, dahili RAM bölgesinden sırasıyla [00h..07h], [08h..0Fh], [10h..17h] ve [18h..1Fh] adresleri arasında yer almaktadır.

### 1.1.2.3. Bit Adreslenebilir Veri Hafıza

Dahili Veri Hafızada, [20h..2Fh] arasındaki 16-byte'lık blok, bit adreslenebilir hafızadır. Bu blokta 128 bit bulunur. Her bit, 0h ile 7Fh arasında bir adrese sahiptir. Byte 20h'in bit 0 adresi 00h, ve byte 2Fh'in bit 7 adresi ise 7Fh'tir.

### 1.1.2.4. Özel Fonksiyon Saklayıcıları (SFR)

SFR'lar, mikrodenetleyici için veri saklama ve kontrol saklayıcıları olarak görev yaparlar. SFR'lar [80..FFh] arasındaki *byte* adreslere sahiptir. SFR'lar, [00h..7Fh] arasında dahili RAM bölgesinden sonraki, 128 byte'lık blokta yer alır. AT89C52 dahili RAM'ına ve SFR'lara doğrudan adresleme modu ile erişilebilir. Dahili RAM, saklayıcı-dolaylı adresleme yoluyla da erişilebilmektedir.

ACC, B saklayıcısı ve PSW, SFR'larda yer alır. PSW, elde bayrağı ve aktif saklayıcı kümesini seçmek için kullanılan 2-bit gibi sistem bayraklarını içerir. SFR'lardan SP, DPTR, DPL, DPH, PCON, TMOD, TL0, TL1, TH0, TH1 ve SBUF sadece byte-adreslenebilir saklayıcılardır. Yani bu SFR'lar, sadece byte olarak okunabilir, yazılabilir, üzerinde işlem görebilir ve karşılaştırılabilir.

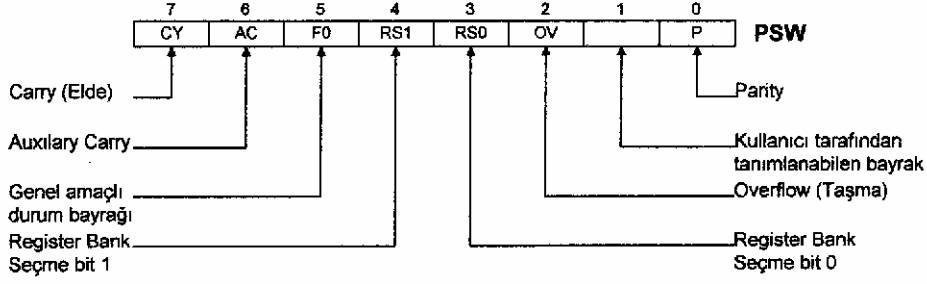
Geri kalan SFR'lar, ACC, B, PSW, P0, P1, P2, P3, IE, IP, SCON ve TCON aynı zamanda bit-adreslenebilir saklayıcılardır. Yani, bu saklayıcıların bitleri okunabilir, yazılabilir, üzerinde işlem görebilir ve karşılaştırılabilir. Bit-adreslenebilir SFR'ların (byte) adresleri 8'in katları şeklindedir. Bit adresi, SFR'in byte adresi ile SFR'daki adreslenen bit yerinin (location) toplanması ile elde edilir. Örneğin, ACC'nin 5'inci bitinin adresi E5h ve SCON'un 0'ıncı bitinin adresi ise 98h'tir.

**Tablo 1.1** Özel Fonksiyon Saklayıcıları. (SFR)

0F8H								0FFH
0F0H	B 00000000							0F7H
0E8H								0EFH
0E0H	ACC 00000000							0E7H
0D8H								0DFH
0D0H	PSW 00000000							0D7H
0C8H	T2CON 00000000	T2MOD XXXXXX00	RCAP2L 00000000	RCAP2H 00000000	TL2 00000000	TH2 00000000		0CFH
0C0H								0C7H
0B8H	IP XX000000							0BFH
0B0H	P3 11111111							0B7H
0A8H	IE 0X000000							0AFH
0A0H	P2 11111111							0A7H
98H	SCON 00000000	SBUF XXXXXXXX						9FH
90H	P1 11111111							97H
88H	TCON 00000000	TMOD 00000000	TL0 00000000	TL1 00000000	TH0 00000000	TH1 00000000		8FH
80H	P0 11111111	SP 00000111	DPL 00000000	DPH 00000000			PCON 0XXX0000	87H

#### 1.1.2.5. Program Durum Kelimesi (PSW)

Program durum kelimesi (Program Status Word), bir çalışma anında CPU'nun durumunu gösteren çeşitli durum bitlerini (Flags) içermektedir. Şekil 1.4'te gösterilen PSW, mikrodenetleyicinin SFR alanı içinde bulunur ve adresi D0h'tır. PSW, Carry (elde) biti, Auxilary Carry biti, 2 tane saklayıcı kümesi seçme biti, Overflow (Taşma) biti, Parity biti ve kullanıcı tarafından tanımlanabilen iki tane biti içermektedir. Aşağıda bu bitler tanıtılmaktadır.



Şekil 1.4 AT89C52 PSW saklayıcısı

**Carry (CY veya C)** bayrağı, aritmetik işlemlerde elde fonksiyonunu gerçekleştirir. C bayrağı, sonuç FFh değerini aşarsa 1'lenir aksi halde 0'lanır. Ayrıca bu bayrak birçok Boolean işleminde 1-bit "ACC" olarak görev alır.

**RS0** ve **RS1** bitleri, Şekil 1.4'te görülen 4 saklayıcı kümesinden birini seçmede kullanılır. Değişik komutlar, bu RAM bölgesine, R0 ve R7 arasında belirtilen saklayıcılar üzerinden erişmektedir. Dört saklayıcı kümesinden yalnız biri, bu bitlerin yürütme sırasındaki durumuna göre seçilir.

**Parity (P)** biti, ACC'deki 1'lerin sayısını yansıtır. Eğer ACC tek (odd sayıda 1'ler içeriyorsa, P = 1 olur (odd parity). Eğer ACC çift (even) sayıda 1'ler içeriyorsa, P = 0 olur (even parity). Böylece ACC'deki 1'ler ile P'nin toplamı her zaman çift olur.

**Auxiliary Carry (AC)** bayrağı, düşük 4-bitten (low nibble) yüksek 4-bite (bit 3'ten bit 4'e) bir elde (carry) olma durumunda 1'lenir, diğer durumda 0'lanır. Bu bayrak, BCD (Binary Coded Decimal) sayılarla işlem yaparken kullanılabilir.

**Overflow (OV)** biti ise, işaretli sayıların toplandığında veya çıkarıldıklarında oluşan bir durumu yansıtır. Taşma, işlem sonucunun hedef saklayıcıya sığmadığını gösterir. Örneğin, 8-bit saklayıcılarda işaretli sayı aritmetiğinde, 7FH (+127) 01H ile toplandığında sonuç 80H (-128) olur. Bu sonuç, işaretli toplama için OV-bayrağı ile belirtilen bir taşma durumudur ve bu sonucun pozitif bir sayı olarak yorumlanamayacağını belirtir. İşaretsiz işlemler için, OV-bayrağının bir önemi yoktur ve ihmal edilir.

PSW'daki son iki bit (F0 ve kullanıcı tarafından tanımlanabilen bayrak), bir işlem için atanmadığından, genel amaçlı durum bayrakları olarak kullanılabilir.

#### 1.1.2.6. Komut İşaretçisi (PC)

Program Counter (PC), hafızada okunacak bir sonraki byte komutuna işaret etmek için dahili olarak kullanılmaktadır. Doğrudan erişilememektedir. PC'nin değeri JMP veya CALL gibi dallanma komutları ile değişir. PC, indisli adreslemede Program Hafızadan okurken taban (base) saklayıcı olarak da kullanılabilir.

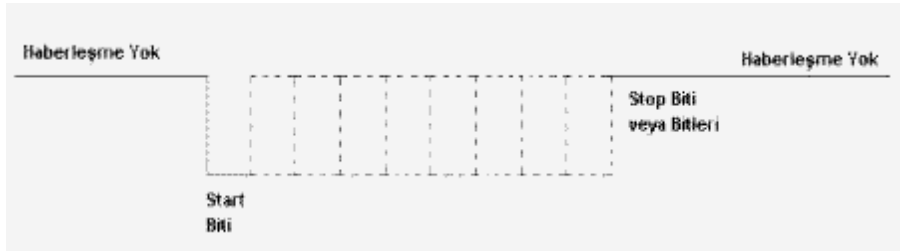
### **1.1.3. CPU Zamanlaması**

Bütün 8051 mikrodenetleyicileri tümdevre-üzeri osilatöre (dahili saat üreticine) sahiptir. Bu üreteç için, mikrodenetleyicinin XTAL1 ve XTAL2 giriş lerine bir kristal veya bir seramik resonatör bağlanır. Dahili saat üreteci, bir makine çevrimini (machine cycle) oluşturan durum (state) sinyallerini belirler.

## 2. SERİ PORT İLE HABERLEŞME

AT89C52 entegresinin güçlü özelliklerinden birisi de yapısında bulundurduğu seri haberleşme birimidir (UART). AT89C52 entegresinin seri bir haberleşme birimine sahip olması demek, AT89C52 ile standart RS-232 seri haberleşmesinin yapılabilmesidir. Böyle bir birim entegre içerisinde bir parça olarak verilmeseydi, seri bir hattan bilgi yollayıp okumak oldukça uğraştırıcı bir iş olarak karşımıza çıkacaktı. Seri bir hattan bilgi yollamak için bit değerini entegrenin bir bacağına yazmak, daha sonra dikkatli bir zamanlama kullanarak sırasıyla bitleri hatta yerleştirmek gerekecekti. Bir pin üzerinden seri bilgi okunması daha da külfetli bir iş olarak karşımıza çıkacaktı. Yollanan veya okunan her bir karakterin, başlangıç biti (start bit), bitiş biti veya bitleri (stop bit), parite kontrolünün AT89C52 programcısı tarafından yapılması gerekecekti. Fakat günümüz mikrodenetleyicilerin çoğunda seri haberleşme biriminin bulunması bu sorunu gidermiştir.

Entegrenin seri portunu kullanmak için yapılması gereken yalnızca birkaç işlem vardır. Bunun için entegrenin seri port iletişim modunu ve seri port haberleşme hızının seçilmesi gereklidir. Bu işlemler yapıldıktan sonra kalan tek şey yazılacak karakterin SBUF özel fonksiyon registerına göndermek ve okunacak karakteri de SBUF registerından almaktır. AT89C52 bir karakteri yolladığında, karakteri yolladığını; veya seri porttan bir karakter okunduğunda okunan bir karakterin olduğunu bildirecektir. Böylece programlayıcı karakterin bit bazında gönderilme işleminin sıkıntılarından kurtulmuş olmaktadır. Seri port asenkron iletişim, bir start biti, bir veya daha fazla stop biti kullanır.



Şekil 2.1 Seri haberleşme örneği

### 2.1. Seri Haberleşme ve Ağ Yapıları

Bir dizi bilgisayarı veya bir dizi işlemciyi birbirine bağlamak için uygun olarak birkaç temel yöntem vardır. Bunlardan en önemlileri, yıldız (star) ve loop (döngü) bağlama yöntemleridir.

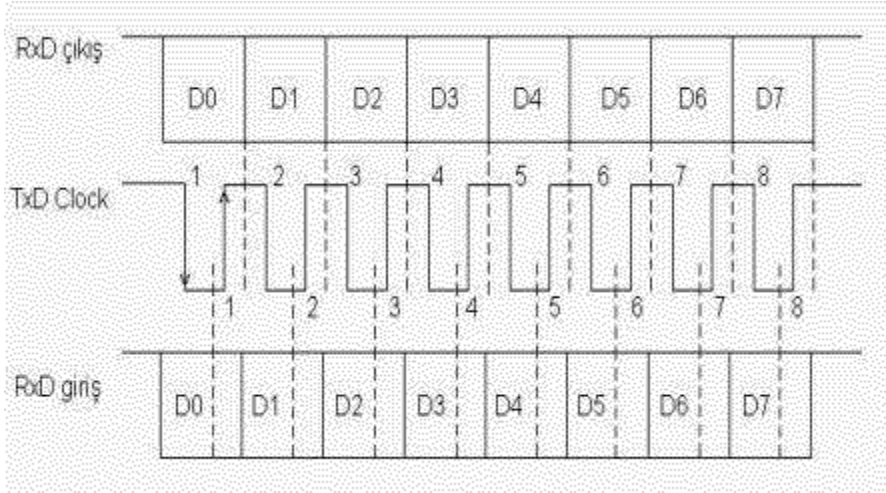
Yıldız bağlama yönteminde ana kontrolcünden birer hat tüm diğer ikincil kontrolcülere götürülür. Bu yöntem de tüm kontrol kararı ana kontrolcü tarafından verilir ve her bir ikincil kontrolcü yalnızca ana kontrolcü ile iletişimde bulunur.

Döngü (loop) bağlama yönteminde ise bir kontrol hattı ile ana kontrolcü tüm diğer kontrolcülere bağlanır. Hangi yöntemin seçileceği bu işi düzenleyen kişinin tercihine ve pek çok dış etkene bağlı olabilir.

Temel olarak yıldız bağlama yöntemi güvenilirlik ve hata bulma açısından döngü yöntemine göre daha üstündür. Döngü yöntemi ise maliyet ve kablo sayısı yönünden üstündür. Yıldız yöntemi eğer ikincil kontrolcülerin sayısı azsa ve ana kontrolcü ile ikincil kontrolcüler arasındaki mesafe azsa tercih edilen yöntem olabilir. Fakat ikincil kontrolcülerin sayısı ve bu kontrolcülerin ana kontrolcüye olan uzaklığı arttıkça tercih döngü yöntemine doğru gider.

### 2.1.1. Seri Data Modu 0

AT89C52 entegrenin seri haberleşme biriminin seri data modu 0'a getirilmesi için SM0 ve SM1 bitleri 0 yapılır. Seri data modu 0'da haberleşme hızı sabittir ve osilatör frekansının 12'de birine eşittir. 12Mhz kristal kullanılan bir AT89C52 devresinde bu haberleşme hızı 1 Megabyte/s'dir. Bu haberleşme modunda RxD ucu hem karakterlerin okunmasında, hem de karakterlerin yollanmasında kullanılır. Entegre seri haberleşme modu 0'a getirildiğinde, TxD pininde ise kristal frekansının 1/12'sinde kare dalga sinyal bulunur. Bu sinyal yalnızca haberleşme sırasında aktif duruma gelir. Hatta haberleşme olmadığı zaman, TxD hattı lojik 1 seviyesinde bulunur. Entegreden data yollanırken, TxD hattının ilk yükselmesinden, bir clock sonra yollanacak karakterin ilk biti RxD hattına kaydırılır. Daha sonra TxD hattının her yükselişinden bir clock sonra diğer bitler RxD hattına konularak karakter hattan yollanmış olur. Entegreye data okunurken de, karakterler RxD hattından okunur. TxD hattında ise yine senkronizasyon sinyali bulunur. Gelen bitler TxD hattının düşen kenarından 2 veya 3 clock sonra seri buffere (SBUF ) alınır. Mod 0 bilgisayar ve AT89C52 arasında haberleşmeden daha ziyade AT89C52 ve mod 0 benzeri haberleşme kullanan diğer entegrelerle veya diğer 8051 işlemcilerle haberleşme için kullanılmak üzere düşünülmüş bir haberleşme modudur.



Şekil 2.2 Seri haberleşme

### 2.1.2. Seri Data Modu 1

Seri port konfigüre edilirken yukarıda da belirtildiği gibi program, seri port haberleşme hızını belirleyecek düzenlemeleri yapmalıdır. Yalnızca mod 1 ve mod 3'te seri port haberleşme hızının konfigürasyonuna gerek vardır. Mod 2 ve mod 4 seri port haberleşme hızını direkt olarak kristal frekansından ürettiğinden bu modlarda haberleşme hızı ile ilgili işlem yapmaya gerek yoktur.



Mod 1 ve mod 3'te seri port haberleşme hızı zamanlayıcı 1 taşmalarıyla belirlenir. Zamanlayıcı 1 çok sık taşıyorsa, haberleşme hızı yüksek daha seyrek taşıyorsa haberleşme hızı daha düşük olacaktır. Zamanlayıcı 1'i haberleşme hızını belirlemek üzere çok farklı biçimde programlama olanağı vardır. Fakat en çok kullanılan metot, zamanlayıcı 1'i 8 bit otomatik yükle modunda (zamanlayıcı modu 2) kullanmak ve TH1 registerına gerekli tekrar yükleme değerini kaydetmektir. TH1 registerına yüklenecek değeri bulmak için (PCON.7 = 0) iken aşağıdaki formülün kullanılması gerekmektedir.

$$TH1 = 256 - ((\text{Kristal frekansı} / 384) / \text{Haberleşme Hızı}) \quad (2.1)$$

Eğer (PCON.7 = 1) ise, haberleşme hızı iki katına çıkar. Dolayısıyla PCON.7 = 1 olduğu durumda kullanılması gereken formül;

$$TH1 = 256 - ((\text{Kristal frekansı} / 192) / \text{Haberleşme Hızı}) \quad (2.2)$$

olacaktır. Burada elde edilen değer tam sayı olması gerekir. Bu da ancak 11.059 Mhz kristal kullanarak mümkün olmaktadır. Dolayısıyla bu değeri TH1 registerına yazılarak 19,200 baud için gerekli işlemler tamamlanmış olunur. Ayrıca, 11,059 Mhz bir kristalle 19,200 haberleşme hızında haberleşebilmek için yapılması gerekenler sırasıyla şunlardır.

1. Seri portu mod 1 veya mod 3 e getirilmesi gerekir
2. Zamanlayıcı 1'i mod 2'ye getirilmesi gerekir
3. TH1'e 253 değerinin yazılması gerekir
4. PCON.7 (SMOD) bitini "1" yapılması gerekir

Şimdi yukarıdaki işlemleri, assembly dilinde yazılmak istenirse;

```
MOV SCON, #048H ; seri port mod 1
MOV TCON, #020H ; zamanlayıcı 1 mod 2
MOV TH1, #0FDH ; TH1 = 253
MOV TL1, #0FDH ; TL1 = 253
```

```
ORL PCON, #080H ; PCON SMODE biti = 1
```

Yukarıdaki hesaplardan da görüldüğü gibi, 12 MHz bir kristal kullanıldığında, zamanlayıcı 1 kullanılarak oluşturulabilecek maksimum haberleşme hızı 4800'dür. 12 MHz kristalde 4800 haberleşme hızını programlamak için yazılması gereken assembly kodu :

```
MOV SCON, #048H ; seri port mod 1
MOV TCON, #020H ; zamanlayıcı 1 mod 2
MOV TH1, #0F3H ; TH1 = 243
MOV TL1, #0F3H ; TL1 = 243
```

```
ORL PCON, #080H ; PCON SMODE biti = 1
```

AT89C52 entegresinin kullanıldığı devrelerde, standard olarak 11,059 Mhz kristali seçilir. Bu entegre üçüncü bir zamanlayıcıya sahiptir. Bu zamanlayıcı da haberleşme hızının kontrolünde kullanılabilir. Bu zamanlayıcı 12 MHz kristalde 9600 haberleşme hızında seri iletişime olanak vermektedir. Dolayısıyla bu entegrelerle gerçekleştirilen devrelerde 12 Mhz kristal kullanılabilir. Çünkü 12 Mhz kristal kullanıldığında, zaman ölçme hesapları çok daha kolay olmaktadır.

### 2.1.3. Seri Data Modu 2 (Çoklu İşlemci Modu)

Seri haberleşme modu 2, seri haberleşme modu 1'e benzer. Mod 2'de haberleşmede 11 bit kullanılır. Bu 11 bitin biri start biti, birisi stop biti ve 9 tane data bitidir. Gönderilen datanın 9'uncu biti SCON registerının TB8 bitine yazılır. Data okuma sırasında ise bu 9'uncu bit RB8 bitine gelir. Seri data modu 2'de iletişim hızı;

$$f_{\text{baud2}} = (2^{\text{SMOD}} / 64) \times f_{\text{clock}} \quad (2.3)$$

formülü ile hesaplanır. Dikkat edilirse, bu iletişim modunda da haberleşme hızı mod 1 haberleşme hızından çok daha yüksektir. Pek çok işlemci kullanılan bir ağ üzerinde, toplanan verilerin çok hızlı bir şekilde aktarılabilmesine gerek vardır. Bu iletişim modunun en önemli özelliği kullanılan 9'uncu data bitinde yatmaktadır. Bu data biti sayesinde yollanan bir mesajın ağdaki belli işlemciler tarafından algılanması, diğerleri tarafından ise ihmal edilmesi sağlanabilmektedir. Seri iletişim modlarında, AT89C52 entegresine bir karakter geldiğinde karakterin son bitinin alınmasıyla RI biti 1 yapılmış olur. Seri data modunda RI bitinin 1 olabilmesi extra bazı şartlara da bağlanmıştır. RI bitinin set olabilmesi için, ya son data biti 1 olmalı ya da SCON registerındaki SM2 biti 0 olmalıdır. Dolayısıyla ağdaki bir işlemci gönderilen mesajın ağdaki tüm işlemciler tarafından alınmasını istiyorsa gönderdiği bilginin son bitini 1 yapmalıdır. Eğer ağdaki yalnız belli işlemcilerin bu mesajı alması isteniyorsa, mesaj yollayan işlemci gönderdiği bilginin son bitini 0 yapar. Bu sayede ağdaki işlemcilerden yalnızca SM2 biti 0 olanlar, gelen mesajı alırlar.

Bu sayede bir yerel ağ oluşturulup, zaman zaman sadece belli entegrelerin sorgulama sonuçlarına cevap verebilmeleri sağlanabilir. Bu yöntem işlemci ve bilgisayar ağlarında sık kullanılan, konuşan ve dinleyenlerin belirlenmesi esasına dayanan bir yöntemdir.

### 2.1.4. Seri Data Modu 3

Bu işlemci modu seri data modu 2 ile aynı çalışma mantığına sahiptir. Yalnızca bu modda haberleşme hızı sabit değildir ve aynı seri mod 1'de olduğu gibi zamanlayıcı 1 ile belirlenir.

## 2.2. Seri Porta Yazma

Seri port yukarıda açıklandığı şekilde konfigürasyonu yapıldığında, seri port yazma ve okuma için hazır hale gelmiştir.

Seri porta bir karakter yollamak için yapılması gereken sadece bu karakteri SBUF (99h) özel fonksiyon registerına yazmaktır. Örneğin seri porta "X" karakteri yollanmak isteniyorsa, verilmesi gereken komut;

MOV SBUF, #'X' komutudur.

Yukarıdaki komut işlemci tarafından okunduğunda, işlemci "X" karakterini seri port üzerinden yollayacaktır. Elbette ki seri porttan yollama işlemi anlık bir işlem değildir ve bir zaman dilimi içerisinde gerçekleşecektir. Standart AT89C52 bir seri çıkış bufferına sahip olmadığından, yeni bir karakter yollamadan önce, önceki karakterin yollandığından emin olmak zorundayız. AT89C52 entegresi bir karakterin yollanmasının bittiğini TI bitini "1" yaparak bildirir. Dolayısıyla bu bitin değeri "1" ise karakterin yollanması bitmiştir ve yeni bir karakter yollanabilir. Bu kontrol program kodu olarak aşağıdaki şekilde yazılır.

SENDCHR :

CLR TI ; Önce TI bitini temizle

MOV SBUF, #'A' ;Yollanacak karakteri seri porta yaz

JNB TI, \$ ;Karakter yollamasının bitişini bekle

RET ; Geri dön

Yukarıdaki üç komut ile bir karakter seri port üzerinden yollanır ve yollama işleminin bitişini beklenir.

### 2.3. Seri Porttan Okuma

Seri porttan okuma yapılması da seri porta yazma kadar kolay bir işlemdir. Seri porttan okuma yapılması için RI bayrağına bakılır. Bu bayrak "1" ise seri porttan bir karakter gelmiş demektir. Seri porttan gelen bu karakter SBUF registerinde bulunur.

Örneğin, program kodu bir karakter okumak üzere beklesin ve karakter geldiğinde bu karakteri akümülatöre okusun. Bunun için yazılması gereken kod yalnızca iki satır olacaktır.

GETCHR:

JNB RI, \$ ; AT89C52 RI bayrağının aktiflenmesini bekle

MOV A, SBUF ; Gelen karakteri akümülatöre oku

RET ; Geri dön

Yukarıdaki kodun birinci satırı RI biti "1" oluncaya kadar beklemektedir. Eğer seri hattın bir karakter gelirse RI biti "1" olacaktır . RI biti "1" olduğunda program ikinci satıra geçecek ve MOV komutunu işleme sokacaktır.

### 2.4. RS-232 İletişiminin Temel Kavramları

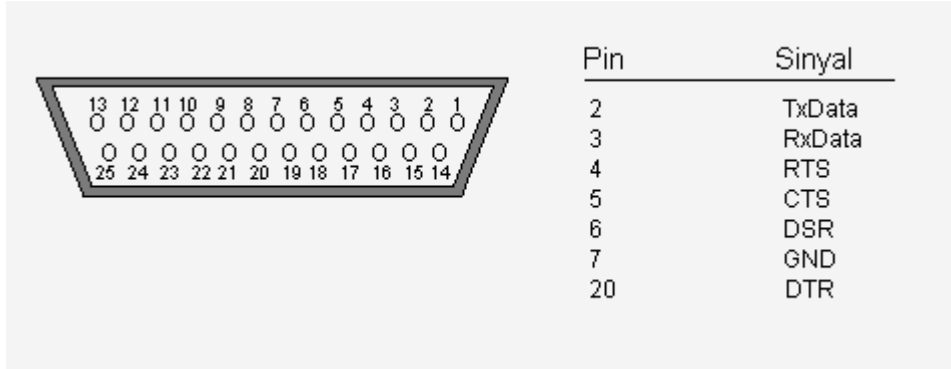
RS-232 bilgisayar dışındaki cihazların bilgisayar ile haberleşmelerinde en çok kullanılan iletişim standartlarından birisidir. RS-232 temel olarak bir seri iletişim birimidir. Seri iletişim biriminde karakterler bir hat üzerinden bit bit yollanır. Seri iletişimin paralel iletişime göre en

önemli üstünlüğü bağlantı kolaylığıdır. Bilgisayardan cihaza karakter yollamak 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 kablolanmaya olanak sağlamaktadır. Örneğin RS-422 1600 m uzunluğunda bir kablo üzerinden 1 megabit/saniye haberleşme hızlarını desteklemektedir.

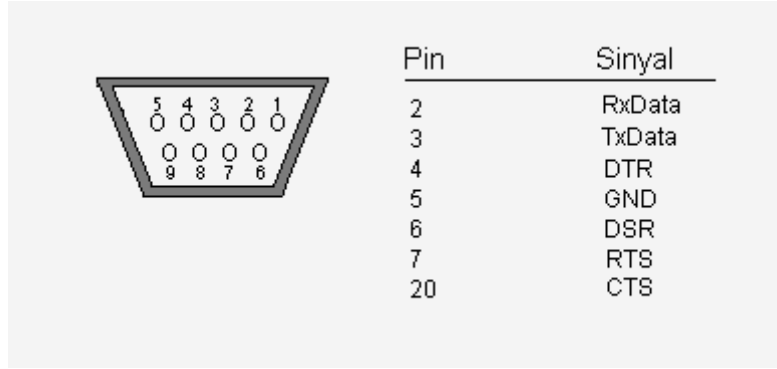
Her bilgisayar en az bir seri RS-232 iletişim birimine sahiptir. Yeni bilgisayarlarda RS-232 konnektörleri hemen hemen tamamen 9 pin erkek konnektörlerdir. Bilgisayar herhangi bir RS-232 portuna o portun adresi aracılığı ile ulaşır. Bilgisayarlarda bulunan RS-232 portunun adresine bilgisayar denetim masasından sistem seçeneği içinden araçlar menüsüne geçerek ulaşılır.

Bu aşamada biraz da RS-232 konnektörünün yapısından ve seri iletişimden bahsetmek faydalı olacaktır. 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.

Daha önce de belirtildiği gibi yeni bilgisayarlarda RS-232 iletişimi için 9 pin erkek konnektörler kullanılmıştır Fakat bazı bilgisayarlarda ve seri iletişim kurulacak bazı cihazlarda 25 pin erkek konnektörler de bulunabilmektedir. Aşağıda 25 pin ve 9 pin DB kablolarında RS-232 sinyallerinin nasıl bağlandığını görebilirsiniz.

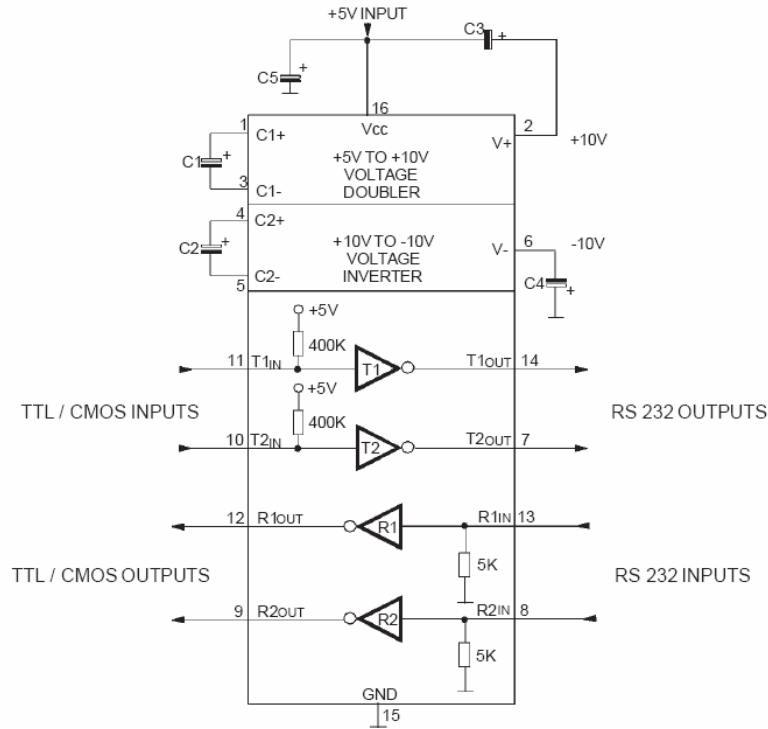


**Şekil 2.3** DB25S konnektör



Şekil 2.4 DB9S konnektör

Mikrodenetleyicilerle bilgisayarların iletişiminde temel olarak TxD, RxD ve GND hatları kullanılır. Pek çok mikrodenetleyicide, bilgisayarla haberleşmek üzere TxD ve RxD hatları bulunur. Fakat mikrodenetleyiciler de 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 dönüştürücü olarak kullanılan entegre ST232 entegresidir. ST232 entegresi 2 sürücü ve 2 alıcıdan oluşur.



Şekil 2.5 ST232 entegresi iç yapısı

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. Daha önce de belirtildiği gibi gereken kablo sayısı yalnızca 3 tür. Konnektörlerin 5 nolu uçları (GND) direkt olarak birbirine bağlanır. 2 (TxD) ve 3 (RxD) uçları ise birbirine çarpraz bağlanır. Böylece mikrodenetleyicinin TxD ucu

bilgisayarın RxD ucuna bağlanmasıyla arasındaki iletişim kurulmuş olur.

**Tablo 2.1** ST232 Pin Tanımlamaları

Pin No	Sembol	İsim ve Fonksiyonu
1	$C_{1+}$	Positive terminal for the first charge pump capacitor
2	$V+$	Doubled voltage terminal
3	$C_{1-}$	Negative terminal for the first charge pump capacitor
4	$C_{2+}$	positive terminal for the second charge pump capacitor
5	$C_{2-}$	Negative terminal for the second charge pump capacitor
6	$V-$	Çevrilmiş Gerilim Terminali
7	$T2_{OUT}$	Transmitter (Gönderici) 2 Çıkış Gerilimi
8	$R2_{IN}$	Receiver (Alıcı) 2 Giriş Gerilimi
9	$R2_{OUT}$	Receiver (Alıcı) 2 Çıkış Gerilimi
10	$T2_{IN}$	Transmitter (Gönderici) 2 Giriş Gerilimi
11	$T1_{IN}$	Transmitter (Gönderici) 1 Giriş Gerilimi
12	$R1_{OUT}$	Receiver (Alıcı) 1 Çıkış Gerilimi
13	$R1_{IN}$	Receiver (Alıcı) 1 Giriş Gerilimi
14	$T1_{OUT}$	Transmitter (Gönderici) 1 Çıkış Gerilimi
15	<b>GND</b>	Toprak
16	$V_{CC}$	Besleme Gerilimi

### 3. SERVO MOTORLAR

Servo motorlar, verilen girişe göre istenen açısal konuma gelen motorlardır. Servo motorların çok çeşitli uygulamalarda kullanılmasının, güvenilir olmasının yanında diğer nedenleri ise;

- Yüksek tork
- Doğru konumlama
- Kolay kurulum
- Kontrol kolaylığı
- Ekonomik oluşu

özelliklerine sahip olmasıdır.

Bir servo motor, yapı olarak dört kısımdan oluşmaktadır. Bunlar dc elektrik motoru, planetar dişli sistemi, geri besleme potansiyometresi ve dc motor pozisyon kumanda elektroniğidir.



**Şekil 3.1** Servo motor

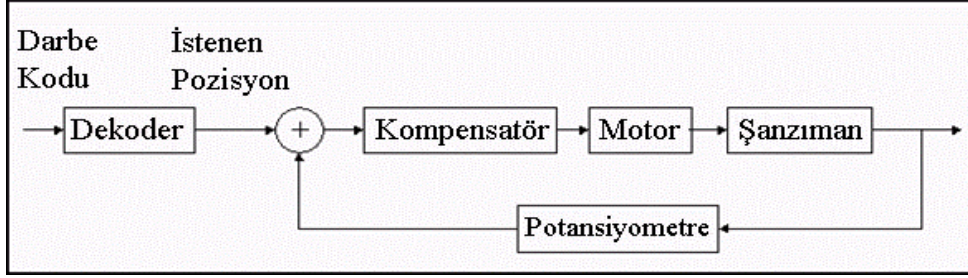
Dc motor herhangi bir dc oyuncak motorundan farklı olmayan çift mıknatıslı bir statora ve fırçalı bobin rotora sahiptir. Motor mili 1:200 ile 1:300 arası dönme oranına sahip bir dişli sistemine bağlanır, bu sayede oldukça yüksek bir tork değerine ulaşılır.

Dişli sisteminin çıkışında 5k'lık bir potansiyometre, mil konumunu elektronik kumanda devresine iletir. Elektronik devrenin görevi mil konumunu gelen veri konumuna gelinceye kadar motoru iletimde tutup tam yerinde durdurmaktadır.

Elektronik devre bu konumu algılamak için PWM (pulse width modulation) tekniğinden yararlanılmaktadır. Kumanda devresi kumanda çubuğunun konumuyla doğru orantılı olarak 1 ile 2 milisaniye arasında dalga genişliği değişen bir sinyali her 20 milisaniyede bir servoya gönderir. 1 milisaniye tam sol, 2 milisaniyede tam sağ pozisyonu ifade eder. Servo içindeki elektronik devre ilk önce gelen darbelerin darbe genişliğini ölçer, daha sonra potansiyometre konumuna bakar ve kendi darbe osilatörünün darbe genişliği gelen darbelerle eşitlenene kadar motoru hareket ettirir. Motorun durduğu konum kontrol çubuğunun tutulduğu konumla birebir aynıdır. Servolarda üç adet kablo dışarı çıkar. Bunlardan kırmızı olan +4.8V/5.0V arası besleme, siyah olan şase yani Ground, diğer kablo ise (turuncu yeşil veya beyaz olabilir) data girişidir.

### 3.1. Servo Motor Temel Fonksiyonları

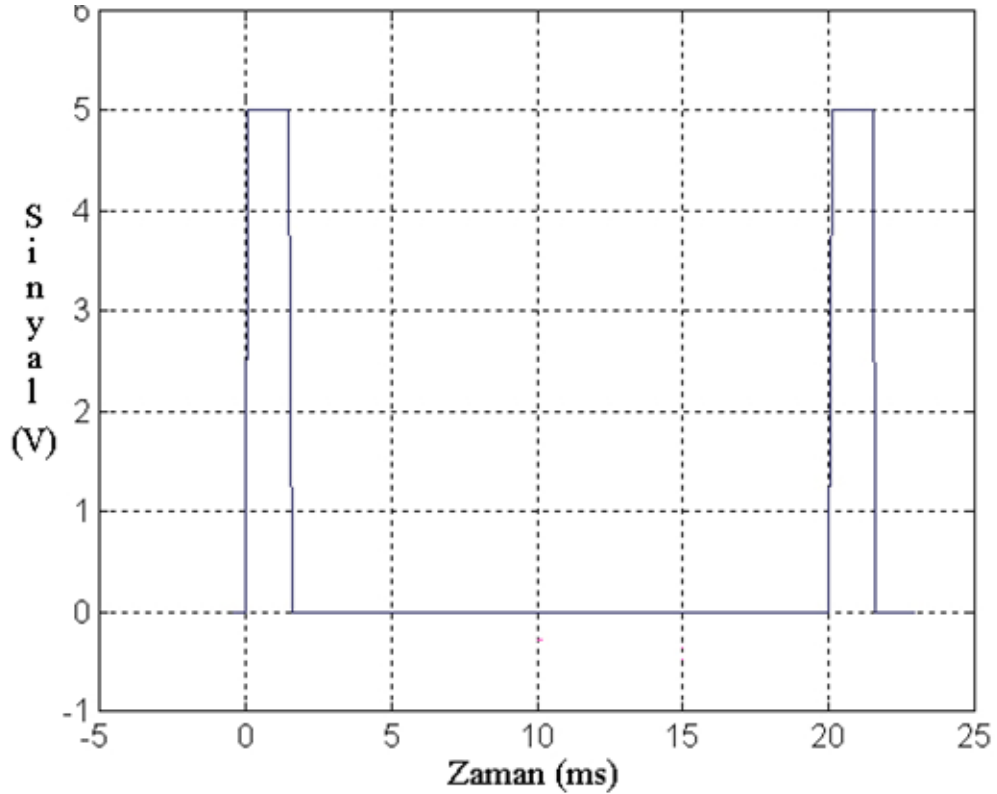
Servo motorun çalışma prensibi, gelen darbe koduna göre konum değiştirmektir. Aşağıda verilen blok diyagramı, servo motorun gerçekleştirdiği temel fonksiyonu çok iyi açıklamaktadır. İstenen konum ile servonun şaftının pozisyonu karşılaştırılır. Kompensatör ise gelen bu bilgiyi düzenler ve servo motora giriş işareti olarak ayarlar. Motorun şanzımana bağlı olmasından dolayı çıkışta düşük hızda bile yüksek tork gücü elde edilir. Şafta bağlı olan potansiyometrenin görevi ise geri besleme sinyalini sağlamaktır.



Şekil 3.2 Servo motorun çalışma şeması

### 3.2. Servo Motor Kontrolü

Servo motorlar 4.7 ile 7 volt besleme gerilimleri arasında çalışabilirler fakat verilen gerilim TTL seviyesinde olmalıdır. Servo motorlar yüksek güç ve sinyal için genellikle 5V'a yakın gerilimlerde çalıştırılırlar.



Şekil 3.3 Servo motor giriş darbesi örneği

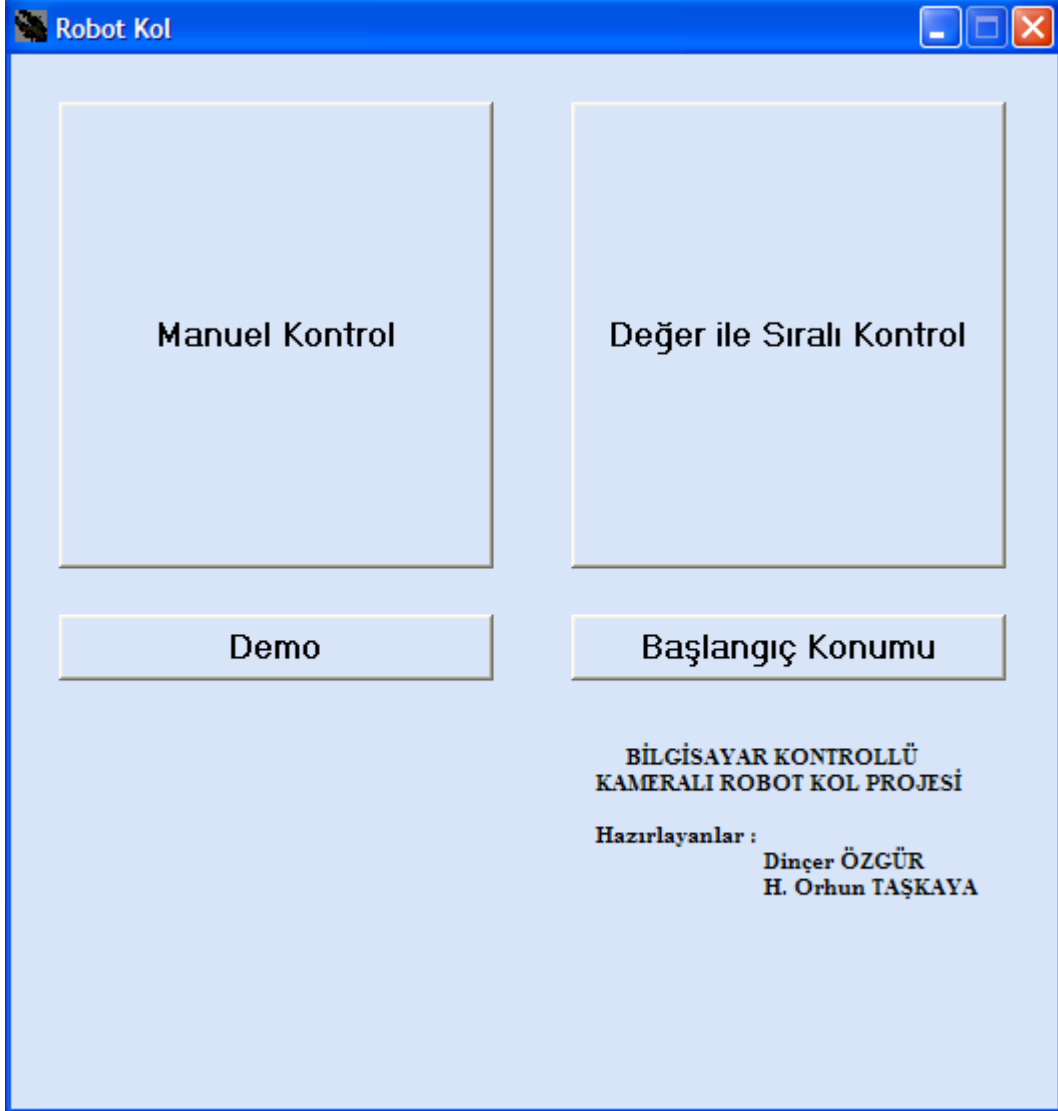


Yukarıdaki şekilde giriş işaretinin nasıl olacağı gösterilmiştir. Her 20 ms'de bir işaret gönderilir. Gönderilen darbenin genişliği servo motorun konumunu belirleyen bilgidir. Örneğin 1.5 ms'lik bir darbe servo motorun orta konuma gelmesini sağlar.

Konum bilgisini veren darbeyi üretmek için çeşitli yöntemler mevcuttur. Darbe üretici olarak sinyal jeneratörü, dijital devreler, mikrodenetleyici veya RC transmitter örnek olarak verilebilir. Bu kaynaklardan kolay bulunması nedeniyle özellikle mikrodenetleyici (MCS-51serisi, Motorola 68HC11, Zilog Z-80 vs.) veya PIC kullanmak iyi bir sonuç elde edilmesini sağlar. Eğer daha karmaşık bir yapı kurulacaksa PC ile arayüz gerçekleştirmek darbe üretimi açısından çok daha kolay olacaktır.

#### 4. ROBOT KOL PROJESİ ARAYÜZ PROGRAMI

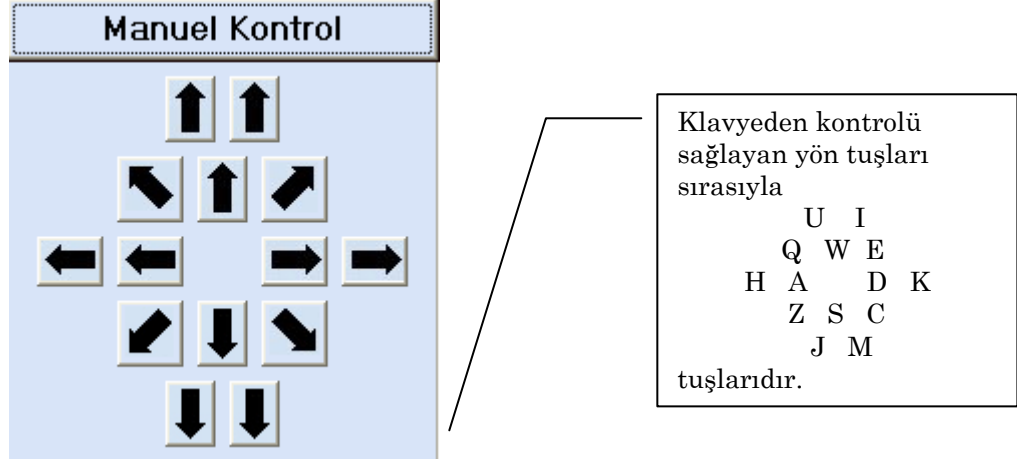
Bu projede, bilgisayardan arayüz programı ile mikrodenetleyici arasında haberleşerek robot kol kontrolü yapılmaktadır. Hazırlanan arayüz programı ile robot kolun kontrolü temel olarak; eş zamanlı etkileşim ve belirlenmiş pozisyonların arka arkaya yapılması olarak iki kısma ayrılmıştır. Diğer kontroller ise önceden hazırlanmış demo hareketleri ve robot kolu başlangıç konumuna getirmek için bulunan butonlar ile yapılmaktadır.



Şekil 4.1 Robot kol arayüz programı

#### 4.1. Manuel Kontrol

Manuel kontrolde, arayüz programında bulunan kontrol (yön) butonları ve klavye üzerinden önceden belirlenmiş olan 12 ayrı tuşla robot kolun kontrolü yapılabilmektedir.



Şekil 4.2 Manuel kontrol

İki ayrı kontrol alternatifine sahip olan robot kolun kontrolü eğer klavyeden yapılacaksa, klavye tuşlarının aktif olabilmesi için öncelikle **Manuel Kontrol** butonunun seçili durumda olması gerekmektedir. Klavye tuşlarının aktivasyonu sağlandıktan sonra klavyeden belirlenen tuşlar basılı tutulduğu sürece, robot kol tuşun kontrolündeki hareketi gerçekleştirir. Bu işlem arayüzdeki yön butonlarıyla (mouse aracılığıyla) yapıldığında ise her basma sonunda robot kol, butonun yönettiği yönde bir grad hareket etmektedir. Kolun bir yönde hareket edebileceği maksimum açı değeri ise 200 grad olarak belirlenmiştir.

#### 4.2. Değer ile Sıralı Kontrol

Değer ile sıralı kontrolde yapılması istenen hareketler, hareketi yapacak olan servonun numarası ve girilen açı değerleri ile seçim sırasına göre yapılır. Değer girilebilmesi için hareket ettirilmesi istenen servonun sahip olduğu kutucuk işaretlenmesi gerekmektedir.



Şekil 4.3 Değer ile sıralı kontrol

Bu kontrolde 10 tane pencere bulunduğundan maksimum 40 farklı servo hareketi yapmaya olanak sağlamaktadır. Pencere değişimleri için sağ alt köşede bulunan ileri ve geri yön butonlarını kullanmak gerekmektedir. Basılan butona göre bir ileri ya da bir geri pencereye geçilmektedir. Ters kinematik enerjisini söndürebilmek için, aynı pencere içindeki servolar arasında geçiş 10 ms gecikmeyle gerçekleştirilmiştir. Farklı pencerelere geçerken ise bu gecikme 500 ms olarak ayarlanmıştır. Toplam pencere sayısı belirlenmiş olduğundan ilk pencerede geri pencere butonu, son pencere olan 10. pencerede ise ileri pencere butonu inaktif duruma geçmektedir. İstenen tüm değerler girildikten sonra **Başlat** butonuna basılmasıyla bu değerler robot kola gönderilmektedir.

#### 4.3. Demo

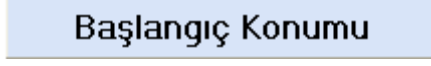
Robot kolun kontrol arayüzü programında bulunan demo butonuyla, robot kolun önceden hazırlanmış olan hareketleri yapması sağlanmaktadır. Demo butonu tıklandığında aşağı yönde 4 adet demo seçim butonu ekrana gelmektedir. Ekrana gelen seçim butonlarından seçilen butona göre robot kol, önceden programcı tarafından hazırlanan hareketleri sırasıyla yapmaktadır.



Şekil 4.4 Demo butonu

#### 4.4. Bařlangıç Konumu

Robot kolun, yapılan hareketlerden sonra konumunun düz hale getirilmesi veya robot kolun hareketlerinin tamamlanması sonucunda standby moduna getirilmesi için arayüz programında bařlangıç komutu butonu bulunmaktadır. Programda bulunan bařlangıç butonuna basılması ile bu işlem gerekleşmektedir.



Şekil 4.5 Bařlangıç konumu

## SONUÇ

Prototipi gerekleřtirilen Bilgisayar Kontrollü Kameralı Robot Kol Projesi'nin ihtiyaca yönelik řekilde daha büyük ve daha güçlü çeřitleri yapılabilir ve güvenli bir řekilde kullanılabilir.

Bu tür robot kollardan, sanayi ve endüstrinin çeřitli dallarında amaca yönelik řekilde programlanarak yararlanılabilir. Ayrıca bomba imha araçlarının üzerine entegre edilerek, insanları tehlikeye atmayacak řekilde bomba imha edici olarak çalışabilir.

Kablosuz haberleşme sistemi robot kola entegre edildiđi takdirde, insanların ulaşmasının maliyetli ve sağlık açısından kötü sonuçlar yaratacađı durumlarda robot kolun yapılması istenen görevleri uzaktan kontrol sistemi ile yapılmasına olanak sağlar.

Bu sistemlerin dışında gerekleřtirdiđimiz proje eğlence sektöründe ve oyuncak olarak çeřitli řekillerde kullanılabilir.

**KAYNAKLAR**

- [1] Gümüřkaya, H, (2002) , “Mikroiřlemciler ve 8051 Ailesi” , Alfa Yayınları
- [2] “MCS-51 Microcontroller Family Users Manual”, Intel
- [3] [www.ume.tubitak.gov.tr/lablar/ozel\\_olcumler/8051a5.htm](http://www.ume.tubitak.gov.tr/lablar/ozel_olcumler/8051a5.htm)
- [4] [www.seattlerobotics.org/guide/servos.html](http://www.seattlerobotics.org/guide/servos.html)
- [5] Karagülle, İ, Pala, Z , (2004) , “ Visual Basic 6.0 Pro” , Türkmen Yayınevi
- [6] Yanık, M , (2001) , “Microsoft Visual Basic 6.0 For Windows”, Beta Yayınları
- [7] AT89C52 Datasheet
- [8] ST232 Datasheet
- [9] Life View NoteCam 300 User’s Manual