



**YILDIZ TEKNİK ÜNİVERSİTESİ**  
ELEKTRİK - ELEKTRONİK FAKÜLTESİ  
ELEKTRONİK ve HABERLEŞME MÜHENDİSLİĞİ BÖLÜMÜ

# **Mikroişlemcili Sistem Tasarımı Laboratuvarı** **Mikroişlemci Temelli Sistem Yazılım/Donanım ve** **Endüstriyel Uygulamalar**

Hazırlayan  
**Yrd. Doç. Dr. Tuncay UZUN**

İSTANBUL , 2005

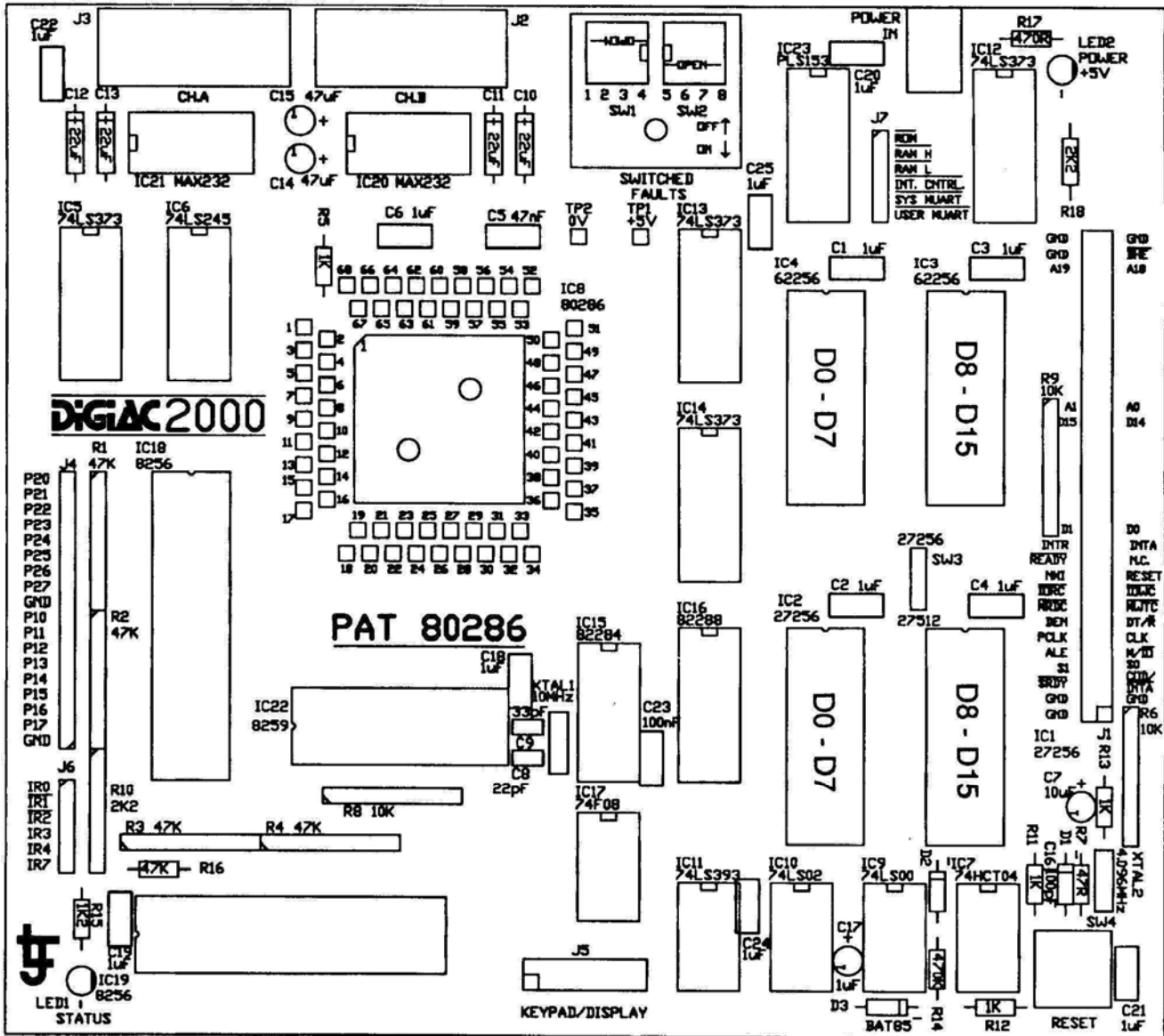
## MİKROİŞLEMCİ EĞİTİM SİSTEMİNİN TANITIMI

### Giriş

Mikroişlemci eğitim sistemi, 1250mA, 5V DC güç kaynağı ile çalışan klavye / gösterge birimi bulunan 80286 mikroişlemci temelli bir sistemdir. Bu sistem ile 80x86 temelli mikroişlemci temelli sistem yazılım/donanım uygulamalarını öğrenmek, incelemek ve tasarlamak amacıyla kullanılabilir. Kendi işletim yazılımı bulunan bağımsız ve/veya seri iletişim arabirimi ile IBM uyumlu kişisel bilgisayar (PC) destekli çalışabilir. Donanım modüler yapıda, işletim ve PC yazılımı ise hata ayıklama yeteneğindedir.

### Donanım Özellikleri

Eğitim sisteminin ana kartı, üzerinde bulunan tümleşik devre ve bağlantı uçlarıyla birlikte aşağıda görülen biçimdedir.



### Mikroişlemci

Eğitim sistemi, 10MHz sistem saat frekansı ile çalışan 80286 mikroişlemcisi kullanır. Sistem saati mikroişlemcinin içinde ikiye bölünür ve gerçek işlemci saati 5MHz olur.

**EPROM**

İki tane 32KB (27256) EPROM tümleşik devresi, eğitim setinin işletim programı, sistem test yazılımı ve uygulama programlarını saklar. Sistemin genişlemesi için iki tane 64KB (27512) EPROM tümleşik devresi kullanılarak sağlanabilir.

**RAM**

İki tane 32KB (62256) RAM tümleşik devresi, 62KB kullanıcı programları ve yığın, kalan 2KB ise sistem değişkenleri ve yığını için kullanılır.

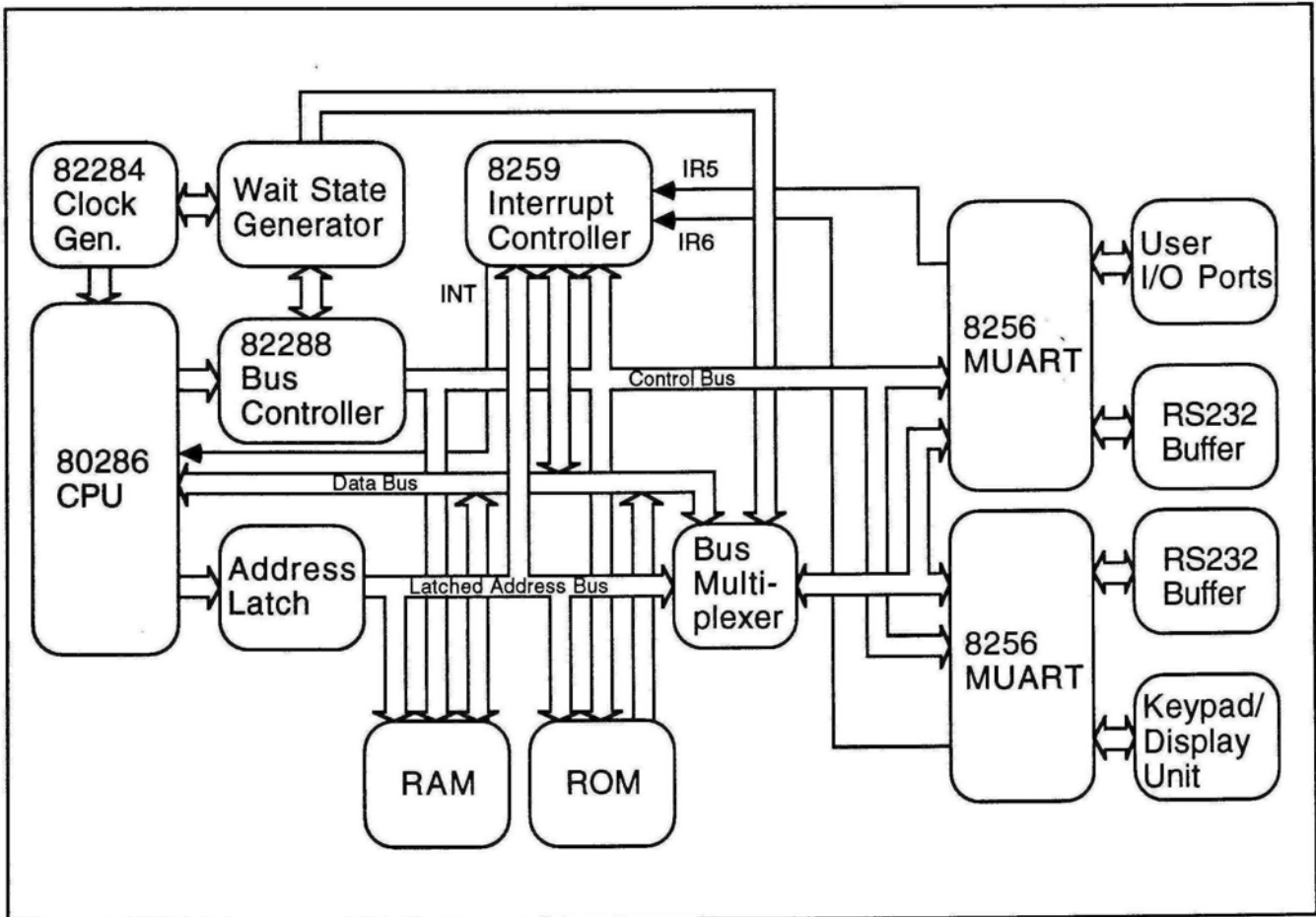
**Kullanıcı Giriş/Çıkışı**

Eğitim sistemi, biri sistem diğeri kullanıcı için olmak üzere iki tane 8256 MUART (Multifunction Universal Asynchronous Receiver Transmitter, Çok işlevli Genel amaçlı Asenkron Alıcı Gönderici) tümleşik devresini kullanır. Kullanıcı tümleşik devresi kullanıcıya, dışarıda bulunan cihazları denetlemek için 16 TTL uyumlu giriş/çıkış ucu sağlar.

**Sayıcı/Zamanlayıcı RS-232 Arabirim**

Kullanıcı 8256 tümleşik devresi 5 tane sayıcı/zamanlayıcı sağlar. Biri sistem, diğeri kullanıcı için olmak üzere iki tane iki yönlü RS232 arabirimi sağlanmıştır.

Eğitim sisteminin blok diyagramı aşağıda gösterilmiştir.

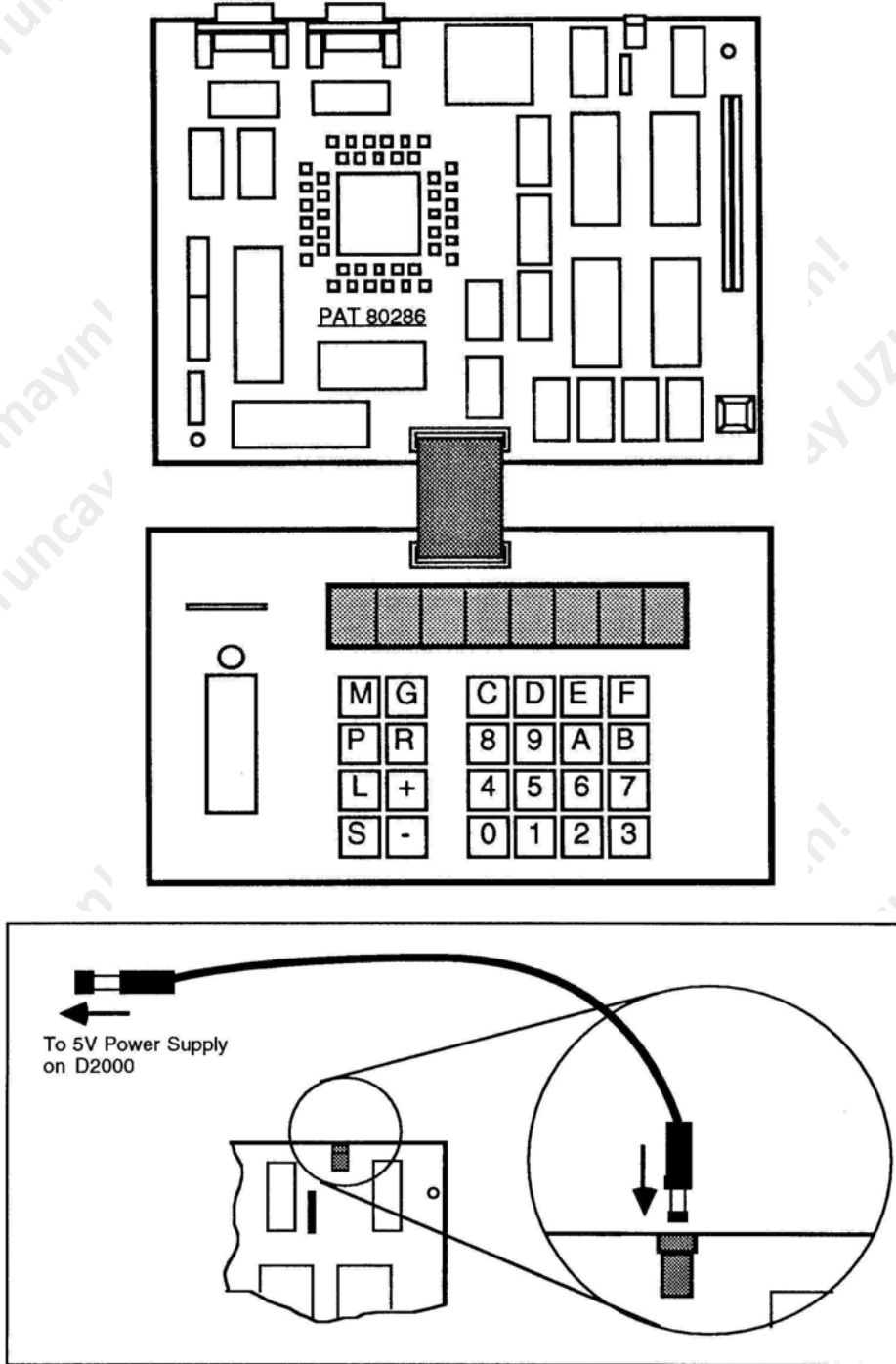


### Klavye/Gösterge Birimi

Klavye/gösterge birimi, 8 haneli 7 parçalı LED gösterge ve 16 tuşlu onaltılık tuş takımı ve 8 tuşlu denetim tuş takımından oluşur. Birim yassı kablo üzerinden bağlayıcı ile ana karta bağlanır.

### Sistemin Kurulması

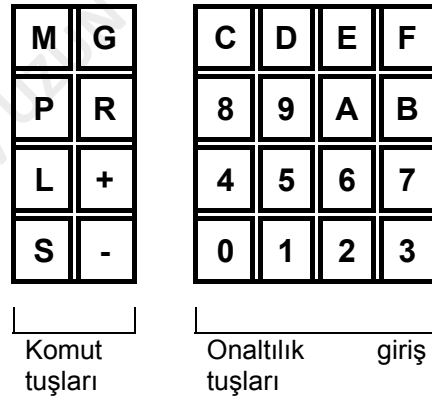
İlk önce klavye/gösterge birimi ana karta aşağıda görülen biçimde bağlanır. Sonra eğitim sistemi, bir 1250mA, 5V DC güç kaynağı aşağıda gösterilen şekilde bağlanarak çalışır duruma getirilebilir.



Güç kaynağı açıldığında göstergede "ready" (=hazır) iletisi görülür. Bu sistemin kullanıma hazır olduğunu bildirir.

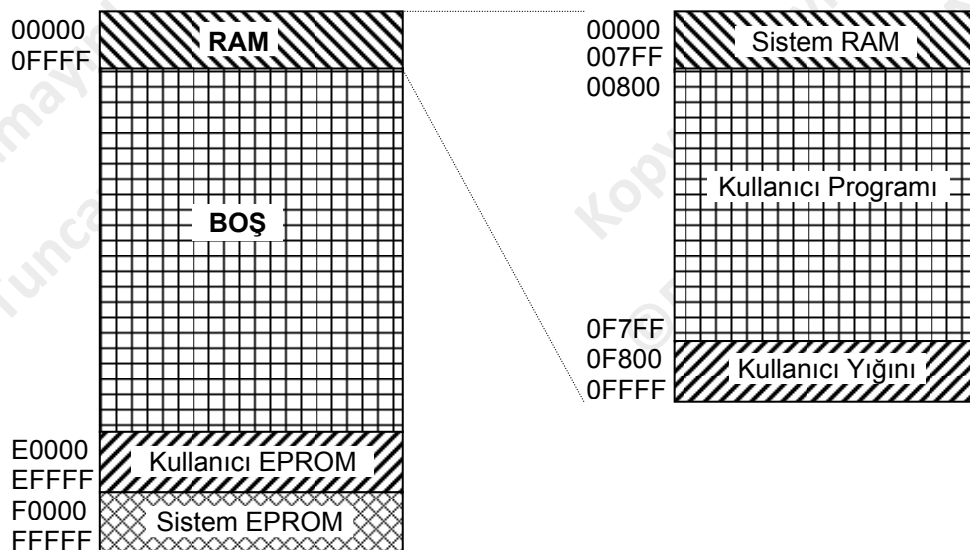
## Tuş Takımı İşlevleri

Eğitim sisteminin tuş takımı aşağıda görüldüğü gibi 8 tane portakal renkli komut tuşu ve 16 krem renkli onaltılık giriş tuşu olmak üzere iki kısımdan meydana gelmiştir.



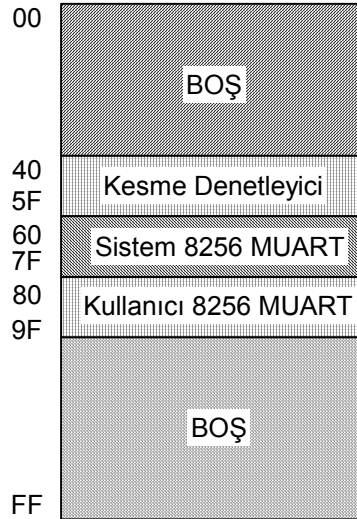
- M** Program ve veri girmek için kullanılır. Kullanıcının gireceği adresin mantıksal kısmı ve içeriği görülebilir ve düzeltilebilir. Adresin parça (DS) değeri 0080H olarak varsayılmıştır. İstenirse bu değer **S** tuşuna basılarak değiştirilebilir.
- G** Bellekteki bir programın, kullanıcı tarafından mantıksal başlangıç adresinin girilerek çalıştırılmasını sağlar. Adres hesabında kod yazmaç içeriği (CS) dikkate alınır.
- P** Kullanıcının giriş/çıkış port yazmaçlarını görmesini ve düzeltmesini sağlar.
- R** Bu tuşun iki işlevi vardır. İlk basıldığında 80286 yazmaçlarının içerikleri incelenebilir ve değiştirilebilir. Diğer yazmaçların görülmesi veya değiştirilmesi “+” ve “-” tuşları ile sağlanır. İkinci defa basıldığında, yazılı programın çalıştığı anda oluşacak kırılma noktası çizelgesinin görülmesi ve değiştirilmesi sağlanır.
- S** RS232 için ilk şartlandırmayı sağlar. Basıldığında değişik RS232 parametreleri görüntülenir ve değiştirilebilir.
- L** RS232 portu yoluyla program veya veri yüklenmesi için kullanılır.
- +** Görüntülenen bellek veya giriş/çıkış adresinin artırılması için kullanılır.
- Görüntülenen bellek veya giriş/çıkış adresinin azaltılması için kullanılır.

## Eğitim Sisteminin Bellek Haritası



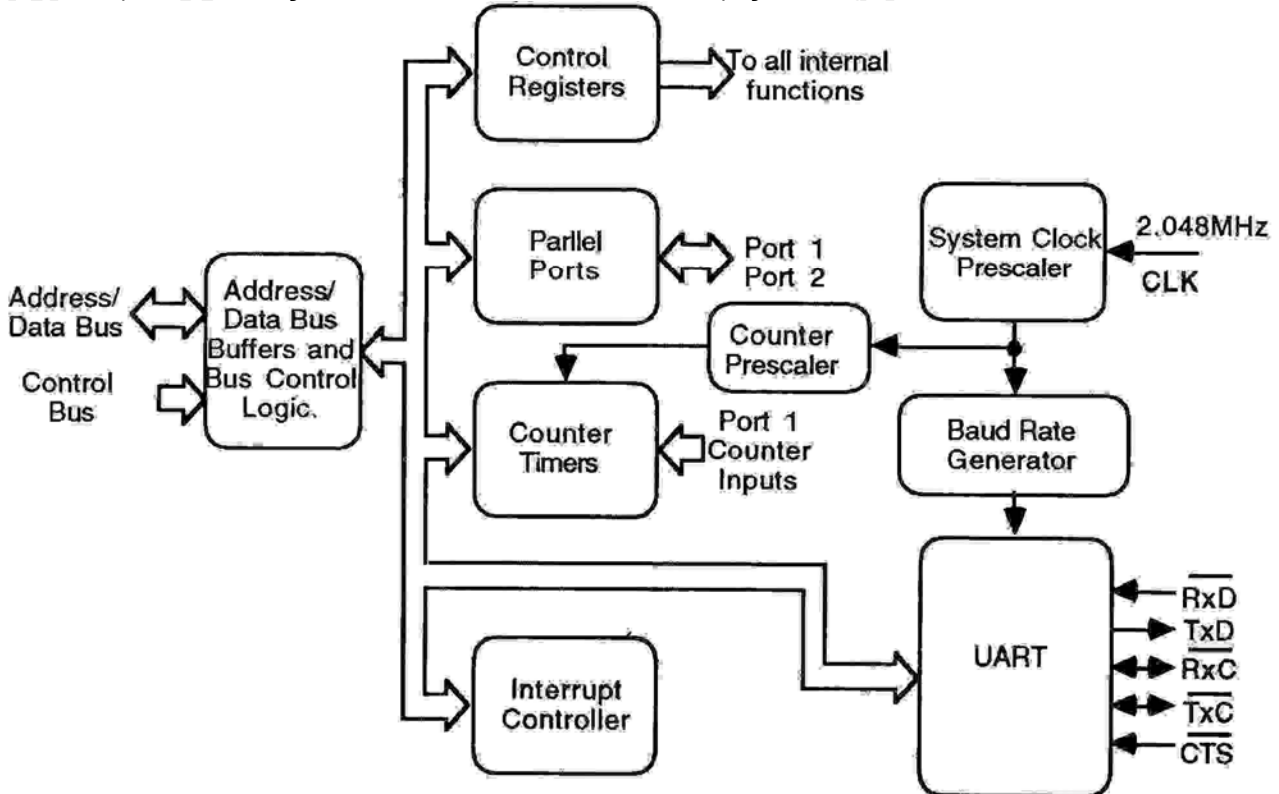
### Eğitim Sisteminin Giriş/Çıkış Haritası

Eğitim sisteminin giriş/çıkış haritası, 64 KB içinde 256 bayt giriş/çıkış adresi oluşacak biçimde gerçekleştirilmiştir. Aşağıda eğitim sisteminin giriş/çıkış haritası görülmektedir.



### 8256 MUART KULLANIMI

PAT 80286 mikroişlemci eğitim sistemi iki adet 8256 MUART (Multifunction Universal Asynchronous Receiver and Transmitter = Çok İşlevli Genel Amaçlı asenkron Alıcı ve Gönderici) tüm devresi kullanır. Her bir 8256 MUART Zamanlayıcı/Sayıcı, Paralel Portlar ve bir UART (Genel Amaçlı Asenkron Alıcı ve Gönderici) içerir.



MUART tümleşik devrelerinden biri sistem tarafından kullanılır. Bu MUART'ın RS232 portu, PAT mönitor kullanılacağı zaman PAT' in terminal bağlantısında kullanılır. Sistem MUART' ı iki adet 8-bit giriş/çıkış (G/Ç) portunu, tuş taraması, LED durumlarının kontrolü, sistem ve kullanıcı RS 232 portları için RTS sinyallerinin sağlanmasında kullanır.

Diğer MUART tümüyle kullanıcıya ayrılmıştır. RS232 portuna erişim bir 9 yollu D-bağlayıcı (9 way D-Connector) ve iki 8-bitlik giriş/çıkış portuna erişim 0.1" aralıklı bacaklarla sağlanır. 8-bitlik giriş/çıkış portları (port 1 ve port 2) mikroişlemcinin dış dünyayla bağlantısını sağlamak için uygundur.

### **SERİ HABERLEŞME**

8256 MUART'ın seri iletişim kısmı bir full-duplex UART içerir. Bu UART 50-19200 bit/saniye aralığındaki standart baud değerlerinde çalışmayı mümkün kılmaktadır. Baud değeri üretimi bir programlanabilen baud değeri üretici tarafından yapılır. UART değişik uzunluktaki bitlerle işlem yapılması, başlangıç ve bitiş bitlerinin (start/stop) oluşturulması ve eşlik bitlerinin üretimi için programlanabilir.

### **PARALEL GİRİŞ/ÇIKIŞ (I/O)**

MUART genel amaçlı iki I/O portu içerir. Port 1' in 8- biti teker teker giriş veya çıkış olarak programlanabilir. Port 2' nin 8- biti sadece 4- bitlik iki grup olarak giriş veya çıkış olarak programlanabilir.

### **SAYICILAR / ZAMANLAYICILAR**

MUART içerisinde 5 adet 8- bitlik sayıcı/zamanlayıcı vardır. MUART sayıcı olarak kullanılacağı zaman saat girişi, MUART saat darbeleri bir programlanabilen ön ölçek devresinden (prescaler) geçilerek elde edilir. Sayıcı saat darbesi 16 kHz veya 1 kHz olacak şekilde programlanabilir. Sayıcı ve zamanlayıcıların 4 tanesi 16- bitlik sayıcı/zamanlayıcı oluşturmak için ark arkaya bağlanabilir.

### **KESMELER**

Kullanıcı MUART'ın ve Sistem MUART'ın kesme çıkış uçları sırasıyla 8259 kesme kontrolcüsünün IR5 ve IR6 girişlerine bağlanmıştır. Sayıcı/Zamanlayıcı ve UART' tan gelen yedi kesme kaynağı geçerlidir. Bir kesme üretildiğinde 8259 kesme kontrolcüsü CPU veri yoluna kesme numarasını çıkarır. Kesme numarası, sistem ve kullanıcı MUART' ı için ayrı ayrı olacaktır. Kesme servis programı uygun MUART'ı seçer ve kesmenin nedenini belirlemek için dönüş değerini kullanır.

### **MUART KOŞULLANMASI**

MUART koşullanması gerekli değildir. Çünkü bu işlem reset işlemi sırasında PAT monitör programı tarafından yapılmaktadır. Bu yüzden kullanıcı programı toplam koşullanmanın sadece gerekli olan küçük bir kısmını değiştirebilir. Tüm koşullandırmalar için 8256 MUART'ın kataloguna bakmak gerekmektedir.

### **MUART YAZMAÇLARI**

MUART giriş/çıkış haritasında sadece 16- bayt' lik adresler vardır ve sadece çift adresler kullanılmaktadır. Bunun anlamı MUART yazmaçları toplamda 32 bayt uzunluğundadır. Bazı yazmaçlar, diğer yazmaçların okuma ve yazma fonksiyonları farklı olmasına rağmen aynı fonksiyonlara sahiptir. Aşağıdaki tablo tüm MUART yazmaçları için tek tek bit tanımlamaları yaparak okuma ve yazma fonksiyonlarını göstermektedir. Tabloda verilen adresler kullanıcı MUART'ı için geçerlidir. Sistem MUART'ı 0060H-007EH adresinde bulunmaktadır.

### 8256 MUART READ registers

Register Name	Register Address									
Command 1	<table border="1" style="display: inline-table;"> <tr> <td>L1</td><td>L0</td><td>S1</td><td>S0</td><td>BRKI</td><td>BITI</td><td>8086</td><td>FRQ</td> </tr> </table>	L1	L0	S1	S0	BRKI	BITI	8086	FRQ	0080H
L1	L0	S1	S0	BRKI	BITI	8086	FRQ			
Command 2	<table border="1" style="display: inline-table;"> <tr> <td>PEN</td><td>EP</td><td>C1</td><td>C0</td><td>B3</td><td>B2</td><td>B1</td><td>B0</td> </tr> </table>	PEN	EP	C1	C0	B3	B2	B1	B0	0082H
PEN	EP	C1	C0	B3	B2	B1	B0			
Command 3	<table border="1" style="display: inline-table;"> <tr> <td>0</td><td>RxE</td><td>IAE</td><td>NIE</td><td>0</td><td>SBRK</td><td>TBRK</td><td>0</td> </tr> </table>	0	RxE	IAE	NIE	0	SBRK	TBRK	0	0084H
0	RxE	IAE	NIE	0	SBRK	TBRK	0			
Mode	<table border="1" style="display: inline-table;"> <tr> <td>T35</td><td>T24</td><td>T5C</td><td>CT3</td><td>CT2</td><td>P2C2</td><td>P2C1</td><td>P2C0</td> </tr> </table>	T35	T24	T5C	CT3	CT2	P2C2	P2C1	P2C0	0086H
T35	T24	T5C	CT3	CT2	P2C2	P2C1	P2C0			
Port 1 Control	<table border="1" style="display: inline-table;"> <tr> <td>P17</td><td>P16</td><td>P15</td><td>P14</td><td>P13</td><td>P12</td><td>P11</td><td>P10</td> </tr> </table>	P17	P16	P15	P14	P13	P12	P11	P10	0088H
P17	P16	P15	P14	P13	P12	P11	P10			
Interrupt Enable	<table border="1" style="display: inline-table;"> <tr> <td>L7</td><td>L6</td><td>L5</td><td>L4</td><td>L3</td><td>L2</td><td>L1</td><td>L0</td> </tr> </table>	L7	L6	L5	L4	L3	L2	L1	L0	008AH
L7	L6	L5	L4	L3	L2	L1	L0			
Interrupt Address	<table border="1" style="display: inline-table;"> <tr> <td>D7</td><td>D6</td><td>D5</td><td>D4</td><td>D3</td><td>D2</td><td>D1</td><td>D0</td> </tr> </table>	D7	D6	D5	D4	D3	D2	D1	D0	008CH
D7	D6	D5	D4	D3	D2	D1	D0			
Receiver Buffer	<table border="1" style="display: inline-table;"> <tr> <td>D7</td><td>D6</td><td>D5</td><td>D4</td><td>D3</td><td>D2</td><td>D1</td><td>D0</td> </tr> </table>	D7	D6	D5	D4	D3	D2	D1	D0	008EH
D7	D6	D5	D4	D3	D2	D1	D0			
Port 1	<table border="1" style="display: inline-table;"> <tr> <td>D7</td><td>D6</td><td>D5</td><td>D4</td><td>D3</td><td>D2</td><td>D1</td><td>D0</td> </tr> </table>	D7	D6	D5	D4	D3	D2	D1	D0	0090H
D7	D6	D5	D4	D3	D2	D1	D0			
Port 2	<table border="1" style="display: inline-table;"> <tr> <td>D7</td><td>D6</td><td>D5</td><td>D4</td><td>D3</td><td>D2</td><td>D1</td><td>D0</td> </tr> </table>	D7	D6	D5	D4	D3	D2	D1	D0	0092H
D7	D6	D5	D4	D3	D2	D1	D0			
Timer 1	<table border="1" style="display: inline-table;"> <tr> <td>D7</td><td>D6</td><td>D5</td><td>D4</td><td>D3</td><td>D2</td><td>D1</td><td>D0</td> </tr> </table>	D7	D6	D5	D4	D3	D2	D1	D0	0094H
D7	D6	D5	D4	D3	D2	D1	D0			
Timer 2	<table border="1" style="display: inline-table;"> <tr> <td>D7</td><td>D6</td><td>D5</td><td>D4</td><td>D3</td><td>D2</td><td>D1</td><td>D0</td> </tr> </table>	D7	D6	D5	D4	D3	D2	D1	D0	0096H
D7	D6	D5	D4	D3	D2	D1	D0			
Timer 3	<table border="1" style="display: inline-table;"> <tr> <td>D7</td><td>D6</td><td>D5</td><td>D4</td><td>D3</td><td>D2</td><td>D1</td><td>D0</td> </tr> </table>	D7	D6	D5	D4	D3	D2	D1	D0	0098H
D7	D6	D5	D4	D3	D2	D1	D0			
Timer 4	<table border="1" style="display: inline-table;"> <tr> <td>D7</td><td>D6</td><td>D5</td><td>D4</td><td>D3</td><td>D2</td><td>D1</td><td>D0</td> </tr> </table>	D7	D6	D5	D4	D3	D2	D1	D0	009AH
D7	D6	D5	D4	D3	D2	D1	D0			
Timer 5	<table border="1" style="display: inline-table;"> <tr> <td>D7</td><td>D6</td><td>D5</td><td>D4</td><td>D3</td><td>D2</td><td>D1</td><td>D0</td> </tr> </table>	D7	D6	D5	D4	D3	D2	D1	D0	009CH
D7	D6	D5	D4	D3	D2	D1	D0			
Status	<table border="1" style="display: inline-table;"> <tr> <td>INT</td><td>RBF</td><td>TBE</td><td>TRE</td><td>BD</td><td>PE</td><td>OE</td><td>PE</td> </tr> </table>	INT	RBF	TBE	TRE	BD	PE	OE	PE	009EH
INT	RBF	TBE	TRE	BD	PE	OE	PE			
	<table border="1" style="display: inline-table;"> <tr> <td style="width: 50%;">Bit 7</td> <td style="width: 50%;">Bit 0</td> </tr> </table>	Bit 7	Bit 0							
Bit 7	Bit 0									

### 8256 MUART WRITE registers

Register Name	Register Address									
Command 1	<table border="1" style="display: inline-table;"> <tr> <td>L1</td><td>L0</td><td>S1</td><td>S0</td><td>BRKI</td><td>BITI</td><td>8086</td><td>FRQ</td> </tr> </table>	L1	L0	S1	S0	BRKI	BITI	8086	FRQ	0080H
L1	L0	S1	S0	BRKI	BITI	8086	FRQ			
Command 2	<table border="1" style="display: inline-table;"> <tr> <td>PEN</td><td>EP</td><td>C1</td><td>C0</td><td>B3</td><td>B2</td><td>B1</td><td>B0</td> </tr> </table>	PEN	EP	C1	C0	B3	B2	B1	B0	0082H
PEN	EP	C1	C0	B3	B2	B1	B0			
Command 3	<table border="1" style="display: inline-table;"> <tr> <td>SET</td><td>RxE</td><td>IAE</td><td>NIE</td><td>END</td><td>SBRK</td><td>TBRK</td><td>RST</td> </tr> </table>	SET	RxE	IAE	NIE	END	SBRK	TBRK	RST	0084H
SET	RxE	IAE	NIE	END	SBRK	TBRK	RST			
Mode	<table border="1" style="display: inline-table;"> <tr> <td>T35</td><td>T24</td><td>T5C</td><td>CT3</td><td>CT2</td><td>P2C2</td><td>P2C1</td><td>P2C0</td> </tr> </table>	T35	T24	T5C	CT3	CT2	P2C2	P2C1	P2C0	0086H
T35	T24	T5C	CT3	CT2	P2C2	P2C1	P2C0			
Port 1 Control	<table border="1" style="display: inline-table;"> <tr> <td>P17</td><td>P16</td><td>P15</td><td>P14</td><td>P13</td><td>P12</td><td>P11</td><td>P10</td> </tr> </table>	P17	P16	P15	P14	P13	P12	P11	P10	0088H
P17	P16	P15	P14	P13	P12	P11	P10			
Set Interrupts	<table border="1" style="display: inline-table;"> <tr> <td>L7</td><td>L6</td><td>L5</td><td>L4</td><td>L3</td><td>L2</td><td>L1</td><td>L0</td> </tr> </table>	L7	L6	L5	L4	L3	L2	L1	L0	008AH
L7	L6	L5	L4	L3	L2	L1	L0			
Reset Interrupt	<table border="1" style="display: inline-table;"> <tr> <td>L7</td><td>L6</td><td>L5</td><td>L4</td><td>L3</td><td>L2</td><td>L1</td><td>L0</td> </tr> </table>	L7	L6	L5	L4	L3	L2	L1	L0	008CH
L7	L6	L5	L4	L3	L2	L1	L0			
Transmitter Buffer	<table border="1" style="display: inline-table;"> <tr> <td>D7</td><td>D6</td><td>D5</td><td>D4</td><td>D3</td><td>D2</td><td>D1</td><td>D0</td> </tr> </table>	D7	D6	D5	D4	D3	D2	D1	D0	008EH
D7	D6	D5	D4	D3	D2	D1	D0			
Port 1	<table border="1" style="display: inline-table;"> <tr> <td>D7</td><td>D6</td><td>D5</td><td>D4</td><td>D3</td><td>D2</td><td>D1</td><td>D0</td> </tr> </table>	D7	D6	D5	D4	D3	D2	D1	D0	0090H
D7	D6	D5	D4	D3	D2	D1	D0			
Port 2	<table border="1" style="display: inline-table;"> <tr> <td>D7</td><td>D6</td><td>D5</td><td>D4</td><td>D3</td><td>D2</td><td>D1</td><td>D0</td> </tr> </table>	D7	D6	D5	D4	D3	D2	D1	D0	0092H
D7	D6	D5	D4	D3	D2	D1	D0			
Timer 1	<table border="1" style="display: inline-table;"> <tr> <td>D7</td><td>D6</td><td>D5</td><td>D4</td><td>D3</td><td>D2</td><td>D1</td><td>D0</td> </tr> </table>	D7	D6	D5	D4	D3	D2	D1	D0	0094H
D7	D6	D5	D4	D3	D2	D1	D0			
Timer 2	<table border="1" style="display: inline-table;"> <tr> <td>D7</td><td>D6</td><td>D5</td><td>D4</td><td>D3</td><td>D2</td><td>D1</td><td>D0</td> </tr> </table>	D7	D6	D5	D4	D3	D2	D1	D0	0096H
D7	D6	D5	D4	D3	D2	D1	D0			
Timer 3	<table border="1" style="display: inline-table;"> <tr> <td>D7</td><td>D6</td><td>D5</td><td>D4</td><td>D3</td><td>D2</td><td>D1</td><td>D0</td> </tr> </table>	D7	D6	D5	D4	D3	D2	D1	D0	0098H
D7	D6	D5	D4	D3	D2	D1	D0			
Timer 4	<table border="1" style="display: inline-table;"> <tr> <td>D7</td><td>D6</td><td>D5</td><td>D4</td><td>D3</td><td>D2</td><td>D1</td><td>D0</td> </tr> </table>	D7	D6	D5	D4	D3	D2	D1	D0	009AH
D7	D6	D5	D4	D3	D2	D1	D0			
Timer 5	<table border="1" style="display: inline-table;"> <tr> <td>D7</td><td>D6</td><td>D5</td><td>D4</td><td>D3</td><td>D2</td><td>D1</td><td>D0</td> </tr> </table>	D7	D6	D5	D4	D3	D2	D1	D0	009CH
D7	D6	D5	D4	D3	D2	D1	D0			
Modification	<table border="1" style="display: inline-table;"> <tr> <td>0</td><td>RS4</td><td>RS3</td><td>RS2</td><td>RS1</td><td>RS0</td><td>TME</td><td>DSC</td> </tr> </table>	0	RS4	RS3	RS2	RS1	RS0	TME	DSC	009EH
0	RS4	RS3	RS2	RS1	RS0	TME	DSC			
	<table border="1" style="display: inline-table;"> <tr> <td style="width: 50%;">Bit 7</td> <td style="width: 50%;">Bit 0</td> </tr> </table>	Bit 7	Bit 0							
Bit 7	Bit 0									



## UYGULAMA MODÜLÜ

DT35, Mikroişlemci Uygulama Modülü, mikroişlemci temelli kontrol deneyleri için ilgilenilen uygulama alanlarını sağlar. Bu uygulama modülü, 8-bit mikroişlemci sistemi ile kullanılabilir şekilde tasarlanmıştır; fakat 16 TTL uyumlu I/O hattına sahip herhangi bir bilgisayar sistemi ile de kullanılabilir.

Uygulama Modülü birimleri aşağıda sıralanmıştır:

- 8-bit Dijital Analog Çevirici, DAC;
- 8-bit Analog Dijital Çevirici, ADC;
- DC Motor ( DAC tarafından hız geri beslemeli sürülür);
- Potansiyometre ( ADC tarafından okunur.);
- Optik Gönderici ( DAC tarafından kontrol edilir.) ve Optik Alıcı ( ADC tarafından okunur.);
- Piezzo Ses Üretici ( TTL uyumlu I/O portlarından direkt kontrol edilir.);
- Ultrasonik Alıcı/Verici ( İki TTL uyumlu I/O port ile direkt kontrol edilir.)

İki tane 8-bitlik port, PortA ve PortB olarak tasarlanmış ve 16 Monitör Led ile port hatlarının durumunun her zaman görülmesi sağlanmıştır.

PortA, çift yönlü veri portu olarak kullanılabilir ve ayrıca kontrol eden bilgisayar tarafından DAC'a 8-bit veri yazılmasında veya ADC'den 8-bit çıkış verilerinin okunmasında kullanılabilir.

PortB, Analog-Dijital ve Dijital-Analog çevrimlerin denetiminde (motordan geri beslemenin alınmasında, piezzo ses üreticinin sürülmesinde ve ultrasonik alıcı/vericinin kontrolünde) kullanılır.

## UYGULAMA MODÜLÜ BİRİMLERİ ve ÖZELLİKLERİ

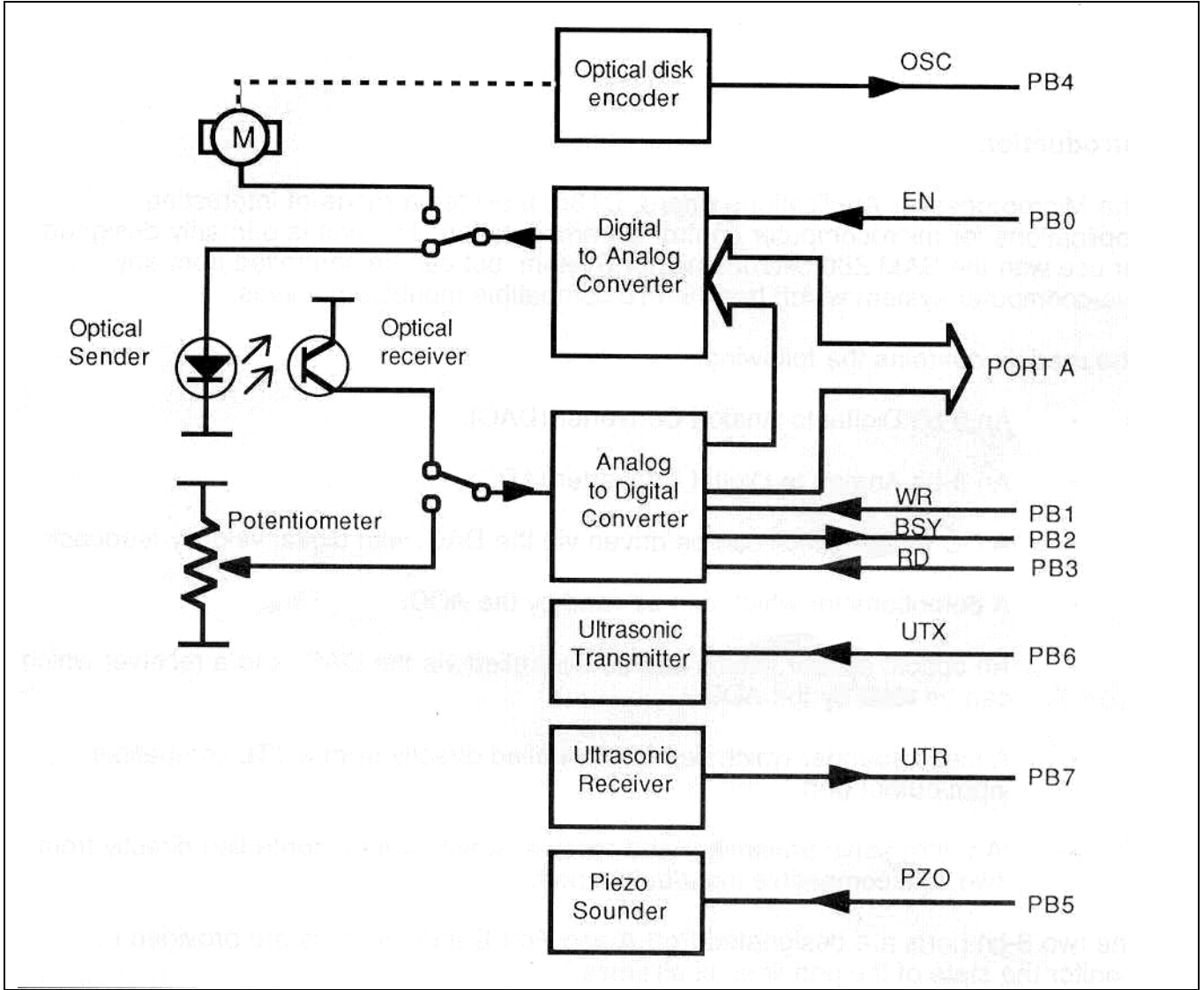
Güç Kaynakları :

- +5V (1A'den düşük akımda);
- -5V (100mA'den düşük akımda) ;

**Not:** Eğer motora max. sürüm voltajı uygulanırsa, çekilen akım 2.5-3.0A'e çıkar. Bu nedenle güç kaynağı bu akımı sağlayabilmelidir.

**Analog-Dijital Çevirici (ADC) :**

- Çevrim Tipi : Ardışık Yaklaşım;
- Çözünürlük : 8-bit;
- Giriş Voltajı : 0-2.55V;
- Çevrim Zamanı :  $\cong 9\mu s$ ;
- Lineerlik Hatası :  $\pm 0.5LSB$ ;
- Entegre Saat Frekansı : 1Mhz;



Uygulama Modülünün Blok Diyagramı

**Dijital-Analog Çevrici (DAC) :**

- Çıkış Voltajı : 0-2.55V;
- Lineerlik Hatası :  $\pm 0.5\text{LSB}$ ;
- Dönüşüm Zamanı :  $1.25\mu\text{s}$ ;
- Veri Hazır Olma Zamanı : 150ns;

**DC-Motor :**

- Motor Tipi : 3V DC Motor;
- Max. Sürüm Voltajı : 2.55V ( DAC' tan sağlanan);
- Geri Besleme : PB4 hattından, her dönüşte 2 pozitif darbe;
- Yük Kontrolü : Potansiyometre, benzetişimi yapılan yük için kontrol voltajını düzenler.

**Potansiyometre :**

- Çıkış Voltajı : 0-2.55V;

**Optik Gönderici :**

- Işık Yayma Gücü : 250 $\mu$ W;
- Tepe Dalga Boyu : 650 nm;
- Anahtarlama Zamanı : ( 20mA,  $t_r = 90$  ns :  $t_f = 40$ ns );
- Giriş Voltajı : 0-2.55V (DAC' tan sağlanan);

**Optik Alıcı :**

- Kollektör Karanlık Akımı : 10nA;
- Kollektör Aydınlık Akımı : 1mW/cm<sup>2</sup> , 650 nm = 065mA;
- Tepe Dalga Boyu : 830 nm;
- Tayf Dalga Boyu : 560 -980 nm;
- Kesim Frekansı : 170 kHz;
- Çıkış Voltajı : 2.55 Vmax;

**Piezzo Ses Üretici :**

- Kontrol İşareti : PB5 üzerinden TTL uyumlu giriş;
- Rezonans Frekansı : 4kHz;

**Ultrasonik Verici :**

- Rezonans Frekansı : 40 $\pm$ 1kHz;
- Çıkış : 106dB (0dB = 2.10<sup>-4</sup>  $\mu$ bar);

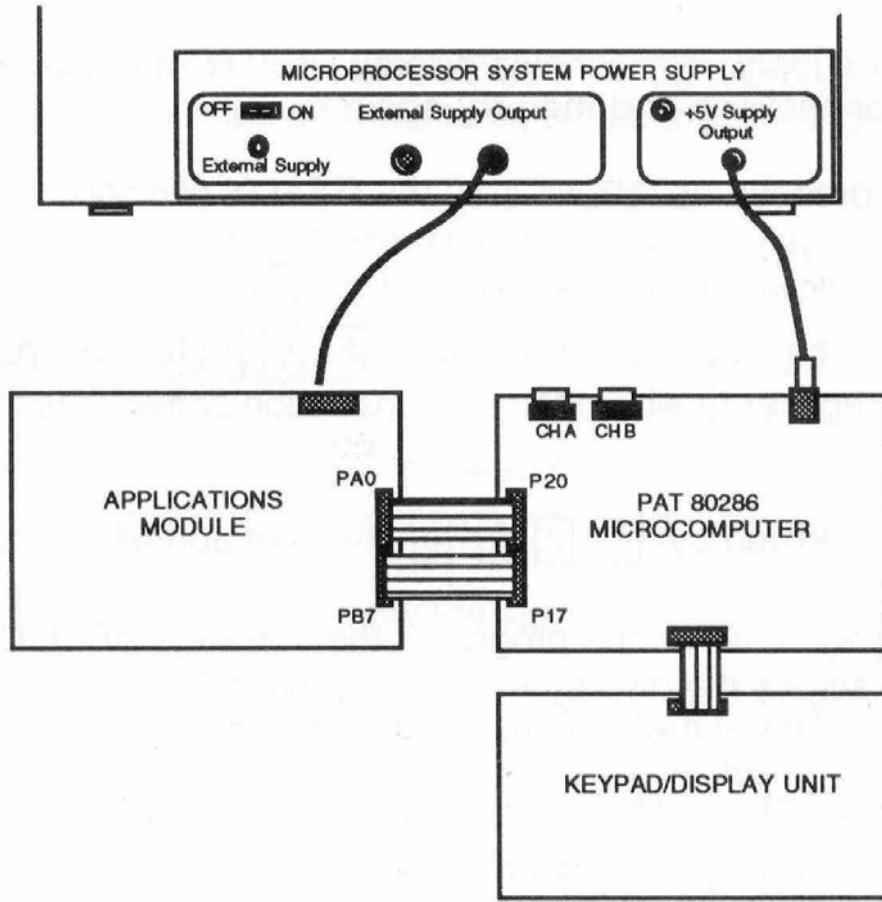
**Ultrasonik Alıcı :**

- Rezonans Frekansı : 40 $\pm$ 1kHz;
- Hassasiyet : -65dB (0dB = 1V/ $\mu$ bar);

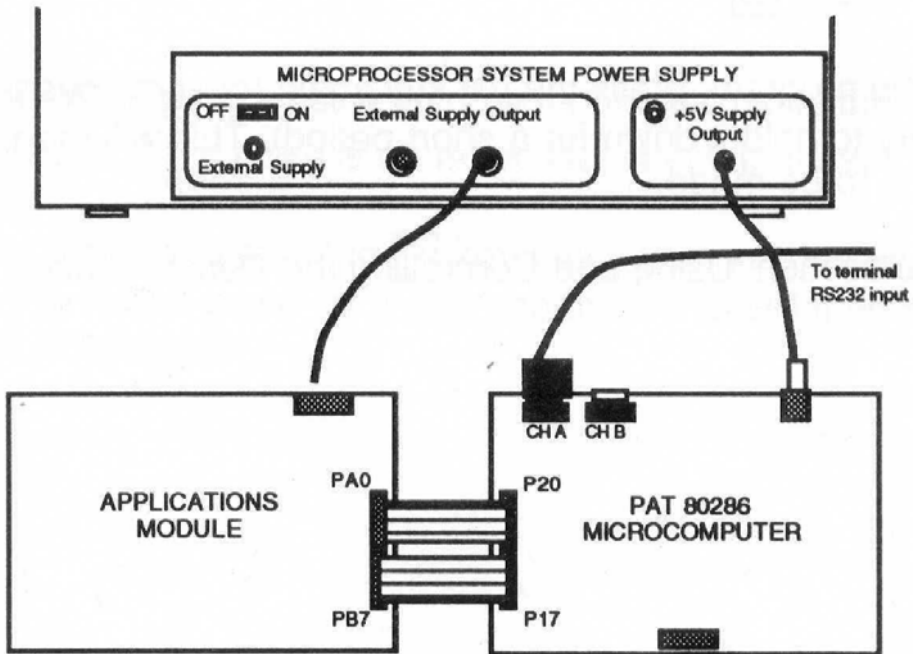
**GİRİŞ/ÇIKIŞ PORT (G/Ç,I / O PORT) BİLGİLERİ**

I/O port monitör LED'leri, iki 8 bitlik portun, PA0-7 ve PB0-7, her hattındaki durumu, hat Lojik "1" ise LED yanıyor, hat Lojik "0" ise LED sönük olacak şekilde gösterirler. Özel port hatları aşağıdaki gibi düzenlenmiştir.

Yön sütunu Uygulama Modülüne göre verilmiştir. Eğer modül bilgisayarın I/O portlarına bağlanmışsa, uygulama modülünün girişi (I) bilgisayarın çıkış (O) portuna bağlanmalı ve çıkış (O) hattı bilgisayarın giriş portuna bağlanmalıdır. Veri hatları, D0-D7, bilgisayarın DAC'a yazı yazmasına veya ADC'den bilgi okumasına göre giriş veya çıkış olarak kullanılabilir.

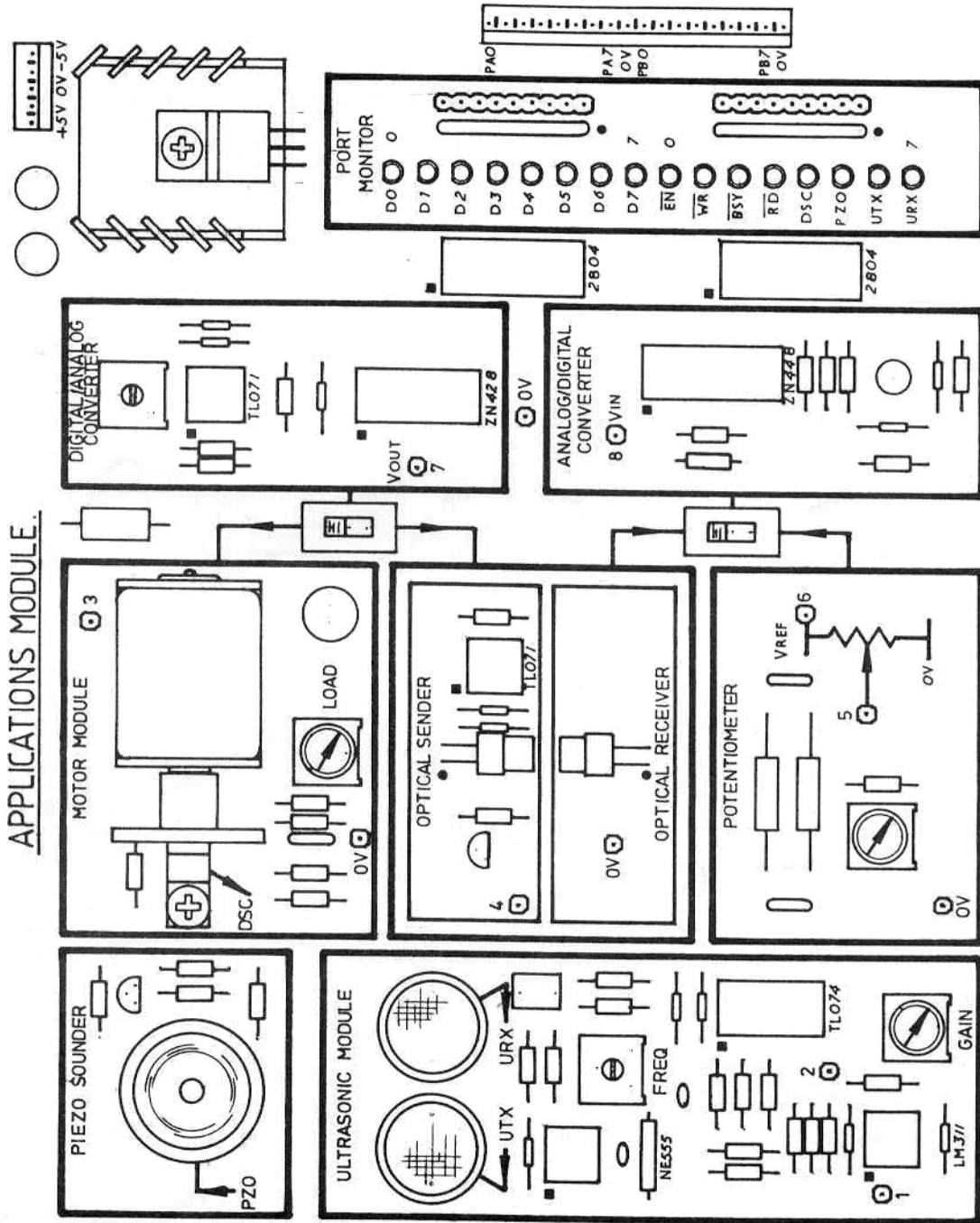


Uygulama Modülü Güç Kaynağı ve Tuş Takımı/Gösterge Modülü İle Bağlantısı



Uygulama Modülü Güç Kaynağı ve Terminal Bağlantısı

PORT	ATAMA	YÖN	FONKSİYON
PA0 PA1 PA2 PA3 PA4 PA5 PA6 PA7	DO D1 D2 D3 D4 D5 D6 D7	I/O I/O I/O I/O I/O I/O I/O I/O	Veri, DAC' a yazılır veya ADC' den okunur. D0=LSB D7=MSB
PB0	EN	I	DAC için tutucu aktif sinyali. Bu sinyal 0 iken, D0-D7 veri hattındaki bilgi sürekli olarak analoga çevrilir. Sinyal 1'e çekilince, veri yolundaki bilgi tutulmaz, analoga çevrilir ve çıkışlar bu değerde dondurulur.
PB1	WR	I	Bu hatta sinyalin negatif düşen kenarı (1→ 0) analog-dijital çevrimi başlatır.
PB2	BSY	O	Bu çıkış A/D çevrim başladığında "0" a düşer ve yaklaşık 9µs sonra "1" e çıkarak çevrimin tamamlandığını ve çıkışın okunabileceğini gösterir.
PB3	RD	I	Bu giriş ADC' nin çıkışlarının okunabilmesi için "0" a çekilmelidir. "1" seviyesinde iken ADC çıkışları yüksek-empedanstadır.
PB4	DSC	O	Bu, DC-motorda düzenlenen optik kesmeden gelen çıkıştır. Diskte iki delik vardır ve her dönüşte iki pozitif giden darbe üretir.
PB5	PZ0	I	Bu hat, piezzo ses üreticinin kontrol girişidir.
PB6	UTX	I	Bu girişteki lojik "1" seviye ultrasonik vericiyi anahtarlar ve Lojik "0" seviyede anahtarlamayı durdurur.
PB7	URX	O	Bu hat, ultrasonik alıcının çıkışıdır. Eğer alıcı tarafında, 40kHz'lik ses algılanırsa, bu çıkışa 40kHz kare dalga gönderir. Eğer ses algılanmazsa, çıkış Lojik "1" de sabit kalır.



Uygulama Modülünün Üstten Görüntüsü

**Deney No** : 1

**Deneyin Konusu** : Eğitim Sistemi

**Deneyin Amacı** : Eğitim Sistemi Donanımının İncelenmesi, Eğitim Sistemine Program Girilmesi ve Çalıştırılması

**Deney Öncesi Yapılacak İşlemler :**

Eğitim sisteminde kullanılan, 80286 mikroişlemcisinin genel özellikleri ve uç işlevleri, donanımı ve saat üretici (82284), yol denetleyici (82288) gibi yardımcı ve çevre birimi (8256,8259) tümleşik devreleri araştırılacak.

Eğitim sisteminde kullanılan, 80286 mikroişlemcisinin programlama modeli ve komut seti incelenecek.

**Deneyde Yapılacak İşlemler :**

Eğitim sisteminin kurulması, modüllerin birbiriyle bağlantıları “Eğitim Sisteminin Tanıtımı” metninde gösterilen şekilde yapılarak sistem çalışır duruma getirilecek. Sistemin bellek haritası incelenerek doğrulanacak.

Aşağıda verilen işlemlerin uygulaması yapılarak eğitim sisteminin kullanımı için gerekli komutların girilmesi öğrenilecek. Eğitim sistemine makina dili program girilecek ve çalıştırılacaktır.

**“M” bellek işlem komutu :**

- Mantıksal adres görülmesi veya değiştirilmesi
- Birinci  ile belirlenen mantıksal adresteki 1 bayt verinin görülmesi veya değiştirilmesi  
Not : DS parça yazmacı dikkate alınır.
- tuşları ile mantıksal bellek adresi bir artırılabilir veya azaltılabilir.

**“G” program çalıştırma komutu :**

- Makina dili programın mantıksal çalışma adresinin görülmesi veya değiştirilmesi
- Birinci  ile belirlenen mantıksal adresteki makina dili programın çalıştırılması  
Not : CS parça yazmacı dikkate alınır.
- tuşları ile çalışma adresi bir artırılabilir veya azaltılabilir.

**“P” giriş veya çıkış portu işlem komutu :**

- Giriş/Çıkış adresinin görülmesi veya değiştirilmesi
- Birinci  ile belirlenen giriş/çıkış adresindeki 1 bayt port verisinin görülmesi veya değiştirilmesi
- tuşları ile port adresi bir artırılabilir veya azaltılabilir.

**“R” yazmaç işlem komutu :**

- R 80286 yazmaçlarının görülmesi veya değiştirilmesi
- R  R makina dili program için çalışma sırasında kırılma noktalarının görülmesi veya değiştirilmesi
- +  - tuşları ile kırılma noktası seçimi yapılır.

**“S” Seri port işlem komutu :**

- S Seri veri iletişim portunun ilk koşullandırma bilgilerinin görülmesi veya değiştirilmesi
- +  - tuşları ile parametre seçimi yapılır.

**“L” Program yükleme komutu :**

- L Seri veri iletişim portundan makina dili program yükleme için sistemi hazır duruma getirir.

Aşağıda verilen programı, eğitim sistemine açıklamalardan yararlanarak giriniz ve çalıştırınız.

Satır	Adres	İşlem kodu	Makina dili kaynak program
1	0100		
2	0100		; 0080:0100 <sub>H</sub> adresinden başlayan, 4D <sub>H</sub> ve 67 <sub>H</sub> değerlerini toplayan ve
3	0100		; toplam sonucunu 0080:1000 <sub>H</sub> adresinde saklayan programı yazınız.
4	0100		
5	0100		
6	0100	ORG	0100H ;Programın başlangıç adresinin
7	0100		;belirlenmesi
8	0100		
9	0100	B0 4D	MOV AL,4DH ;4D <sub>H</sub> değerinin AL yazmacına
10	0102		;yüklenmesi
11	0102		
12	0102	04 67	ADD AL,67H ;67 <sub>H</sub> değerinin AL yazmacına
13	0104		;eklenmesi
14	0104		
15	0104	3E A2 00 10	MOV BAYT PTR DS:1000H,AL ;toplam sonucunun
16	0108		;0080:1000 <sub>H</sub> adresinde
17	0108		;saklanması
18	0108	BB 00 00	MOV BX,0000H ;Eğitim sistemine geri dönüş.
19	010B	B4 04	MOV AH,04H
20	010D	CD 28	INT 028H

Sisteme güç verildiğinde göstergede,

READY



Programın başlangıç adresi M 0 1 0 0 tuşları ile girildiğinde göstergede,

A d r 0 1 0 0

S tuşu ile DS parça yazmacı 0080 olduğu görülebilir veya düzeltilebilir.  
M M tuşları ile veri girişine hazır duruma gelindiğinde göstergede,

A 0 1 0 0 0 0

mantıksal adres

verinin içeriği

makina dili programın girilmesi:

B 0

+ 4 D

+ 0 4

. . .  
. . .  
. . .

+ C D

+ 2 8

G komutu çalıştırıldığında göstergede,

G 0 0 1 0 0

çalışma adresi

çalışma adresinin farklı olması durumunda 0 1 0 0 olarak değiştirilir.  
program ikinci G komutu ile çalıştırıldığında göstergede,

r u n n i n g

çalışan programı durdurmak için "RESET" anahtarına basılarak programın çalışması durdurulur ve başlangıç durumuna dönlür.

**Sorular:**

1. Segment yazmacının içeriği  $F800_H$  ve offset değeri de  $7200_H$  ise fiziksel adresi ve içeriğini bulunuz. Bu adres hangi bölgeye denk gelir ?
2.  $F000:11F9_H$  bellek bölgesinin içeriği nedir ?
3. Kullanıcı programları için ayrılan PAT bellek alanı neresidir ?
4. Verilen örnek programda AL yerine BL kullanılması durumunda hangi adresteki bilgilerin değiştirilmesi yeterlidir?
5. Sonucun saklandığı bellek adresini  $0090:1200_H$  olarak değiştirmek için neler yapılmalıdır?

**Deney No** : 2

**Deneyin Konusu** : Adresleme Modları

**Deneyin Amacı** : 80286 Adresleme Modlarının İncelenmesi

**Deney Öncesi Yapılacak İşlemler:**

80286 mikroişlemcisinin adresleme modları araştırılacak ve incelenecektir. Aşağıda verilen makina dili programlar incelenecek ve adresleme modları belirtilerek her satırın açıklaması yanına yazılacaktır.

**Deneyde Yapılacak İşlemler:**

Aşağıda verilen makina dili programların analizi, gerekli yazmaç ve bayraklar belirlenerek eğitim sistemi üzerinde çalıştırılarak yapılacaktır.

a) AX, BX, CX, DX yazmaç içeriklerini bellekte saklayan program.

Satır	Adres	İşlem kodu	Makina dili kaynak program
1	0100		ORG 0100H
2	0100	BE 00 30	MOV SI,3000H
3	0103	89 04	MOV [SI],AX
4	0105	89 5C 02	MOV [SI+2],BX
5	0108	89 4C 04	MOV [SI+4],CX
6	010B	89 54 06	MOV [SI+6],DX
7	010E	BB 00 00	MOV BX,0000H
8	0111	B4 04	MOV AH,04H
9	0113	CD 28	INT 28H

b) Bellekte bulunan bayt büyüklüğündeki sayıları toplayıp bellekte saklayan program.

1	0100		ORG 0200H
2	0200	BE 00 40	MOV SI,4000H
3	0203	8A 04	MOV AL,[SI]
4	0205	02 44 01	ADD AL,[SI+1]
5	0208	02 44 02	ADD AL,[SI+2]
6	020B	02 44 03	ADD AL,[SI+3]
7	020E	88 44 04	MOV [SI+4],AL
8	0211	BB 00 00	MOV BX,0000H
9	0214	B4 04	MOV AH,04
10	0216	CD 28	INT 28H

- c) Bellekte 0080:0300<sub>H</sub> ve 0080:0400<sub>H</sub> adresinde bulunan word büyüklüğündeki verilere sırasıyla 3000<sub>H</sub> ve 4000<sub>H</sub> ilave eden sonuçları 0080:0500<sub>H</sub> ve 0080:0600<sub>H</sub> adreslerinde saklayan makina dili programını tasarlayıp eğitim seti üzerinde çalıştırınız.
- d) Bellekte 0080:0200<sub>H</sub> ile 0080:0209<sub>H</sub> arasında bulunan 5 tane word büyüklüğündeki verinin yerleşim sırasını ters çeviren (5.veri 1.verinin adresine gelecek şekilde) makina dili programını tasarlayıp eğitim seti üzerinde çalıştırınız.

**Sorular :**

1. Adresleme modlarını kullanarak 0080:1000<sub>H</sub> ve 0080:1002<sub>H</sub> adreslerindeki kelimeleri toplayan ve sonucu 0080:1004<sub>H</sub> adresine yazan programı oluşturunuz.
2. İşlenecek verinin açık olarak verildiği 80286 adresleme modu nedir?
3. “ MOV 0215 [DI], BX “ komutunda kullanılan adresleme modu nedir?
4. SI yazmacı 1200<sub>H</sub> ve BX yazmacı 3020<sub>H</sub> değerleri var iken, “ MOV DX, 2400[ SI+BX] “komutu ile DX'e hangi adresten yükleme yapılmaktadır?
5. Aynı işlemin yapılabilirdiği adresleme modları arasında tercih yapılırken dikkate alınması gereken en önemli nokta nedir?
6. İçerik yoluyla adreslemenin (Implied Addressing) özelliği nedir ?

**Deney No** : 3

**Deneyin Konusu** : Makina Dilinde Aritmetik İşlem

**Deneyin Amacı** : Aritmetik İşlem Komut ve Programlarının İncelenmesi

**Deney Öncesi Yapılacak İşlemler :**

80286 mikroişlemcisinin aritmetik işlem komutları araştırılacak ve incelenecektir. Aşağıda verilen aritmetik işlem programları incelenerek her satırın açıklaması yanına yazılacaktır.

**Deneyde Yapılacak İşlemler :**

Aşağıda verilen makina dili programların analizi, gerekli yazmaç ve bayraklar belirlenerek eğitim sistemi üzerinde çalıştırılarak yapılacaktır.

a)

Satır	Adres	İşlem kodu	Makina dili kaynak program
1	0100		ORG 0300H
2	0300	B0 80	MOV AL,80H
3	0302	3E 2A 06 00	SUB AL,BAYT PTR DS:1000H
4	0307	3E A2 00 20	MOV BAYT PTR DS:2000H,AL
5	030B	BB 00 00	MOV BX,0000H
6	030E	B4 04	MOV AH,04H
7	0310	CD 28	INT 028H

b)

1	0100		ORG 0100H
2	0100	B0 4D	MOV AL,4DH
3	0102	04 67	ADD AL,67H
4	0104	3E A2 00 10	MOV BAYT PTR DS:1000H,AL
5	0108	BB 00 00	MOV BX,0000H
6	010B	B4 04	MOV AH,04H
7	010D	CD 28	INT 028H

c)

1	0100		ORG 0300H
2	0300	B8 E7 00	MOV AX,0E7H
3	0303	BB C3 00	MOV BX,0C3H
4	0306	B9 B9 00	MOV CX,0B9H
5	0309	F7 E3	MUL BX
6	030B	F7 F1	DIV CX
7	030D	3E A3 00 30	MOV DS:3000H,AX

8	0311	3E 89 16 02	MOV	DS:3002H,DX
9	0316	BB 00 00	MOV	BX,0000H
10	0319	B4 04	MOV	AH,04H
11	031B	CD 28	INT	028H

- d) Bellekte sırasıyla 0080:0400<sub>H</sub> ve 0080:0500<sub>H</sub> adresleinde bulunan word ve bayt büyüklüğündeki işaretli iki sayının sırasıyla toplamını 0600<sub>H</sub>, farkını 0602<sub>H</sub>, çarpımını 0604<sub>H</sub> ve bölüm sonucunu 0608<sub>H</sub> ve kalanı 060A<sub>H</sub> adresinde saklayan makina dili programını tasarlayıp eğitim seti üzerinde çalıştırınız.
- e) 0080:0200<sub>H</sub> adresinden başlayarak yerleşmiş en küçük AA<sub>H</sub> olmak üzere bayt büyüklüğündeki ardışıl 5 tane işaretli sayı için Gauss Kuralı' nın doğru olduğunu gösteren makina dili programını tasarlayınız ve eğitim seti üzerinde çalıştırınız. (Gauss Kuralı: 0' dan başlayan ardışıl n adet tamsayının toplamı  $(n*(n+1))/2$ ' dir. )

#### Sorular :

- 0080:0100<sub>H</sub> adresinden başlayan, 0080:1000<sub>H</sub> ve 0080:1002<sub>H</sub> adresindeki 16 bitlik bilgileri toplayan ve bu sonucu 0080:1004<sub>H</sub> adresine yazan programı oluşturunuz?
- ORG assembler komutunun işlevi nedir?
- Bölme işlemi sonucunda bölüm hangi yazmaçta saklanır?
- DIV, IDIV, MUL ve IMUL komutları arasındaki fark nedir ?
- $(E7_H \times C3_H)/B9_H$  işlemini yapıp bölümü 0080:3000<sub>H</sub>, kalanı 0080:3002<sub>H</sub> adresinde saklayan makina dili programını yazınız.

**Deney No** : 4

**Deneyin Konusu** : Döngü Komutları

**Deneyin Amacı** : Döngü Komut ve Programlarının İncelenmesi

**Deney Öncesi Yapılacak İşlemler :**

80286 mikroişlemcisinin döngü komutları araştırılacak ve incelenecektir. Aşağıda verilen döngü komutu içeren programlar incelenerek her satırın açıklaması yanına yazılacaktır.

**Deneyde Yapılacak İşlemler :**

Aşağıda verilen makina dili programların analizi, gerekli yazmaç ve bayraklar belirlenerek eğitim sistemi üzerinde çalıştırılarak yapılacaktır.

**a)** 0080:1200<sub>H</sub> bellek gözündeki verinin 0000<sub>H</sub> olması durumunda 0080:2100<sub>H</sub> bellek gözüne 33<sub>H</sub> ve sıfırdan farklı olması durumunda ise aynı bellek gözüne 55<sub>H</sub> yazan makina dili programı yazınız.

Satır	Adres	İşlem kodu	Makina dili kaynak program
1	0100		ORG 200H
2	0200	3E A1 00 12	MOV AX,DS:1200H
3	0204	0D 00 00	OR AX,0000H
4	0207	74 0A	JZ SIFIR
5	0209	3E C7 06 00 21 55 00	FARKLI: MOV DS:2100H,55H
6	0210	E9 07 00	JMP SON
7	0213	3E C7 06 00 21 33 00	SIFIR: MOV DS:2100H,33H
8	021A	BB 00 00	SON: MOV BX,0000H
9	021D	B4 04	MOV AH,04H
10	021F	CD 28	INT 028H
11	0221		END

**b)** 0080:1400<sub>H</sub> adresinden 0080:14FF<sub>H</sub> adresine kadar olan bellek gözlerini 77<sub>H</sub> değeri ile dolduran makina dili programı yazınız.

1	0100		ORG 0400H
2	0400	BB FF 14	MOV BX,14FFH
3	0403	C6 07 77	DEVAM: MOV BAYT PTR [BX],77H
4	0406	FE CB	DEC BL
5	0408	75 F9	JNZ DEVAM
6	040A	C6 07 77	MOV BAYT PTR [BX],77H
7	040D	BB 00 00	SON: MOV BX,0000H
8	0410	B4 04	MOV AH,04H
9	0412	CD 28	INT 028H
10	0414		END

c) 0080:1500<sub>H</sub> adresi ile 0080:15FF<sub>H</sub> adresi arasındaki bellek gözlerini aşağıda belirtilen biçimde dolduran makina dili programı yazınız.

Adres	Veri
0080:1500 <sub>H</sub>	00 <sub>H</sub>
0080:1501 <sub>H</sub>	01 <sub>H</sub>
0080:1502 <sub>H</sub>	02 <sub>H</sub>
.	.
.	.
.	.
0080:15FD <sub>H</sub>	FD <sub>H</sub>
0080:15FE <sub>H</sub>	FE <sub>H</sub>
0080:15FF <sub>H</sub>	FF <sub>H</sub>

1	0100			ORG	0500H
2	0500	BB FF 15		MOV	BX,15FFH
3	0503	B0 FF		MOV	AL,BAYT PTR 0FFH
4	0505	88 07	DEVAM:	MOV	BAYT PTR [BX],AL
5	0507	FE C8		DEC	AL
6	0509	FE CB		DEC	BL
7	050B	75 F8		JNZ	DEVAM
8	050D	88 07		MOV	BAYT PTR [BX],AL
9	050F	BB 00 00	SON:	MOV	BX,0000H
10	0512	B4 04		MOV	AH,04H
11	0514	CD 28		INT	028H
12	0516			END	

d) 0080:1000<sub>H</sub> adresinden başlayarak yerleştirilen verilerin 0080:2000<sub>H</sub> adresinden başlayarak transfer eden ve transfer edilecek verinin 00<sub>H</sub> olması durumunda transferi durduran makina dili programı yazınız.

Satır	Adres	İşlem kodu	Makina dili kaynak program
1	0100		ORG 0200H
2	0200		MEM1 EQU 0FFFH
3	0200		MEM2 EQU 1FFFH
4	0200	BE 00 00	MOV SI,0000H
5	0203	46	DEVAM: INC SI
6	0204	8A 84 FF 0F	MOV AL,MEM1[SI]
7	0208	88 84 FF 1F	MOV BAYT PTR MEM2[SI],AL
8	020C	0C 00	OR AL,00H
9	020E	75 F3	JNZ DEVAM
10	0210	BB 00 00	SON: MOV BX,0000H
11	0213	B4 04	MOV AH,04H
12	0215	CD 28	INT 028H
13	0217		END



e) 0080:2400<sub>H</sub> adresine; 0080:2300<sub>H</sub> adresindeki veri, 7F<sub>H</sub>'den küçük olması durumunda 01<sub>H</sub>, eşit olması durumunda AA<sub>H</sub> ve büyük olması durumunda ise FE<sub>H</sub> verisini yazan makina dili programı yazınız.

1	0100		ORG	0400H
2	0400	3E A0 00 23	MOV	AL,BAYT PTR DS:2300H
3	0404	3C 7F	CMP	AL,7FH
4	0406	74 16	JZ	ESIT
5	0408	72 0A	JC	KUCUK
6	040A	3E C7 06 00 24 FE 00	MOV	DS:2400H,0FEH
7	0411	E9 11 00	JMP	SON
8	0414	3E C7 06 00 24 01 00	MOV	DS:2400H,01H
9	041B	E9 07 00	JMP	SON
10	041E	3E C7 06 00 24 AA 00	MOV	DS:2400H,0AAH
11	0425	BB 00 00	SON:	MOV BX,0000H
12	0428	B4 04	MOV	AH,04H
13	042A	CD 28	INT	028H
14	042C		END	

f) 0080:2500<sub>H</sub> , 0080:2600<sub>H</sub> , 0080:2700<sub>H</sub> adreslerindeki verilerden en büyük olanı bulup 0080:2800<sub>H</sub> adresine yazan makina dili programı yazınız.

1	0100		ORG	0500H
2	0500	3E A1 00 25	MOV	AX,DS:2500H
3	0504	3E 3B 06 00 26	CMP	AX,DS:2600H
4	0509	73 04	JNC	KUCUK
5	050B	3E A1 00 26	MOV	AX,DS:2600H
6	050F	3E 3B 06 00 27	KUCUK1: CMP	AX,DS:2700H
7	0514	73 04	JNC	KUCUK2
8	0516	3E A1 00 27	MOV	AX,DS:2700H
9	051A	3E A3 00 28	KUCUK2: MOV	DS:2800H,AX
10	051E	BB 00 00	SON:	MOV BX,0000H
11	0521	B4 04	MOV	AH,04H
12	0523	CD 28	INT	028H
13	0525		END	

### Sorular :

- 0080:1000<sub>H</sub> ve 0080:1100<sub>H</sub> adreslerindeki kelimeleri toplayan ve sonuç FFFF<sub>H</sub>'ten büyük ise 0080:2000<sub>H</sub> adresine 99<sub>H</sub>, değil ise AA<sub>H</sub> yazan programı oluşturunuz?
- 0080:2418<sub>H</sub> adresindeki komut " JZ 2432H " ise, Program Sayıcıya eklenmesi gereken değer nedir?
- Relatif Adresleme için en büyük pozitif 16 bitlik ofset değeri nedir?
- 008:0500<sub>H</sub> ve 0080:0610<sub>H</sub> adresleri arasındaki bölgeyi 99<sub>H</sub> ile dolduran programı yazınız.
- 0080:3000<sub>H</sub> adresi herhangi bir bilgi içerebilir. Adresin içeriği C3<sub>H</sub> olana kadar artım veya azaltım yapan programı yazın.
- LOOP komutlarında döngü sayacı olarak kullanılan yazmaç hangisidir?
- Sayma işlemi sıfırlanmaya ve sıfır bayrağı (zero flag) temizleninceye kadar JUMP işlemi gerçekleştiren döngü komutu nedir?

8. Tanımlanan yazmaç veya etkin adresten, kaynak bölgedeki ( yazmaç veya adresteki ) bilgiyi, içerikleri değiştirmeden çıkaran, sadece bayrakları etkileyen komut nedir?
9. CMP işlemi sonucunda Z=1 ve C=1 ise karşılaştırılan elemanların büyüklükleri nasıldır?

**Deney No** : 5

**Deneyin Konusu** : Makina Dilinde Test ve Lojik İşlemler

**Deneyin Amacı** : Test, Lojik Komut ve Programlarının İncelenmesi

**Deney Öncesi Yapılacak İşlemler :**

80286 mikroişlemcisinin test ve lojik komutları araştırılacak ve incelenecektir. Aşağıda verilen test ve lojik komutu içeren programlar incelenerek her satırın açıklaması yanına yazılacaktır.

**Deneyde Yapılacak İşlemler :**

Aşağıda verilen makina dili programların analizi, gerekli yazmaç ve bayraklar belirlenerek eğitim sistemi üzerinde çalıştırılarak yapılacaktır.

a) 0080:1300<sub>H</sub> adresine; 0080:1200<sub>H</sub> adresindeki verinin 0. ,1. ,2. veya 3. bitinin değeri "0" ise, F0<sub>H</sub>, diğer durumda ise 77<sub>H</sub> verisini yazan makina dili programı yazınız.

Satır	Adres	İşlem kodu	Makina dili kaynak program
1	0100		ORG 0300H
2	0300	3E A0 00 12	MOV AL,BAYT PTR DS:1200H
3	0304	24 0F	AND AL,0FH
4	0306	34 0F	XOR AL,0FH
5	0308	A8 0F	TEST AL,0FH
6	030A	75 09	JNZ DOĞRU
7	030C	3E C6 06 00 13 77	YANLIŞ: MOV BAYT PTR DS:1300H,077H
8	0312	E9 06 00	JMP SON
9	0315	3E C6 06 00 13 F0	DOĞRU: MOV BAYT PTR DS:1300H,0F0H
10	031B	BB 00 00	SON: MOV BX,0000H
11	031E	B4 04	MOV AH,04H
12	0320	CD 28	INT 028H
13	0322		END

b) 0080:1400<sub>H</sub> adresine F800<sub>H</sub> verisini yazınız. Bu bellek gözündeki verinin içeriğini 12. biti "0" olana kadar döndüren makina dili programı yazınız.

1	0100		ORG 0400H
2	0400	3E D1 0E 00 14	KONTROL: ROR DS:1400H,01H
3	0405	3E A1 00 14	MOV AX,DS:1400H
4	0409	35 FF FF	XOR AX,0FFFFH
5	040C	A9 00 10	TEST AX,1000H
6	040F	74 EF	JZ KONTROL
7	0411	BB 00 00	SON: MOV BX,0000H
8	0414	B4 04	MOV AH,04H
9	0416	CD 28	INT 028H
10	0418		END

- c) 0080:1000<sub>H</sub> adresindeki word büyüklüğündeki veriyi 0080:1002<sub>H</sub> adresine ters çevirerek (en düşük anlamlı bit, en ağırlıklı bit olacak şekilde) yazan makina dili programını tasarlayınız.

Satır	Adres	İşlem kodu	Makina dili kaynak program
1	0100		ORG 0400H
2	0400		
3	0400	3E A1 00 10	MOV AX,DS:1000H
4	0404	B1 10	MOV CL,10H
5	0406	D1 D8 DONGU:	RCR AX,1
6	0408	D1 D3	RCL BX,1
7	040A	E2 FA	LOOP DONGU
8	040C	3E 89 1E EA	MOV DS:1002H,BX
9	0411	BB 00 00	MOV BX,0000H
10	0414	B4 04	MOV AH,04H
11	0416	CD 28	INT 028H

- d) 0080:0100<sub>H</sub> ile 0080:0109<sub>H</sub> arasında yer alan 10 adet bayt büyüklüğündeki veriler içinde çift olanların sayısını 0300<sub>H</sub> adresinde, bu çift sayıların adreslerini 0400<sub>H</sub>' den başlayarak saklayan aşağıdaki makina dili programının analizini eğitim seti üzerinde yapınız.

Satır	Adres	İşlem kodu	Makina dili kaynak program
1	0100		ORG 0500H
2	0500	SAYAC	EQU 0300H
3	0500		
4	0500	BE 00 01	MOV SI,0100H
5	0503	BF 00 04	MOV DI,0400H
6	0506	B3 02	MOV BL,02H
7	0508	B1 0A	MOV CL,0AH
8	050A		
9	050A	8A 04 DONGU:	MOV AL,[SI]
10	050C	F6 F3	DIV BL
11	050E	80 FC 00	CMP AH,00H
12	0511	74 06	JE CIFT_SAY
13	0513		
14	0513	46 GERI:	INC SI
15	0514	E2 F4	LOOP DONGU
16	0516	E9 0A 00	JMP BITIR
17	0519		
18	0519	3E 89 35 CIFT_SAY:	MOV DS:[DI],SI
19	051C	47	INC DI
20	051D	47	INC DI
21	051E	FE C2	INC DL
22	0520	E9 F0 FF	JMP GERI
23	0523		
24	0523	3E 89 16 00 BITIR:	MOV DS:SAYAC,DX

25	0528	BB 00 00	MOV BX,0000H
26	052B	B4 04	MOV AH,04H
27	052D	CD 28	INT 28H

e) 0080:0200<sub>H</sub> adresinde bulunan word büyüklüğündeki sayıda yer alan “0” ve “1” ‘lerin sayısını bulup, bu değerleri sırasıyla 0080:0300<sub>H</sub> ve 0301<sub>H</sub> adreslerinde saklayan makina dili programını tasarlayıp eğitim seti üzerinde çalıştırınız.

#### Sorular :

1. Akümülatör 4500<sub>H</sub> bilgisini içeriyorsa, “ XOR AX,0FFH “ komutunun işlenmesinden sonra hangi değeri alır?
2. Bir yazmaç B7<sub>H</sub> değerini içersin. Bu yazmaç üç kez sola kaydırılırsa yazmacın yeni değeri ne olur?
3. Akümülatör 4500<sub>H</sub> bilgisini içeriyorsa, “ TEST AX, 0FFH “ komutunun işlenmesinden sonra akümülatörün içeriği ve bayrakların durumu ne olur?
4. BL yazmacındaki 5.bitin “0” olup olmadığı nasıl test edilebilir ?
5. SHR ve SHL komutları hangi aritmetik işlemlerle aynı işleve sahiptir ?

**Deney No** : 6

**Deneyin Konusu** : Giriş/Çıkış Komutları

**Deneyin Amacı** : Giriş/Çıkış Komut ve Programlarının İncelenmesi

**Deney Öncesi Yapılacak İşlemler :**

80286 mikroişlemcisinin giriş/çıkış komutları araştırılacak ve incelenecektir. Aşağıda verilen giriş/çıkış komutu içeren programlar incelenerek her satırın açıklaması yanına yazılacaktır. Eğitim sisteminin monitör programında (=klavyenin okunması, göstergeye bilgi yazılması gibi işleri yapan işletim programı) kullanılan altprogramların çalıştırılması verilen ek bilgiler incelenerek öğrenilecektir.

**Deneyde Yapılacak İşlemler :**

Aşağıda verilen makina dili programların analizi, gerekli yazmaç ve bayraklar belirlenerek eğitim sistemi üzerinde çalıştırılarak yapılacaktır.

a) 0080:1000<sub>H</sub> adresindeki veri ile 0080:1100<sub>H</sub> adresindeki veriyi toplayan ve sonucu giriş / çıkış tümdevresinin 1. portuna yazan makina dili programı yazınız.

NOT: Makine dili programın ilk kısmında, bu ve bundan sonra verilen programlarda kullanılan değişkenlerin tanımlandığı bir eşitlik tablosu verilmiştir.

Satır Adres İşlem kodu Makina dili kaynak program

```

1 0100          INCLUDE PATCALLS.INC
2 0100  |
3 0100  |          ; yazmaç değerleri.
4 0100  |
5 0100  |          ; 80286 Eğitim sistemi
6 0100  |
7 0100  |          ; Çok kullanılan değerler:
8 0100  |          USER      EQU 40  ; Monitor kullanıcı fonksiyonu
9 0100  |          USRBSE     EQU 100H ; Kullanıcı program başlangıcı (org)
10 0100 |          USRSEG     EQU 80H  ; Kullanıcı program segmenti
11 0100 |          SYSSEG     EQU 0H   ; Sistem segmenti
12 0100 |          KYDBUF     EQU 47DH ; Tuş takımı/gösterge tamponu
13 0100 |
14 0100 |          ; Cihaz numaraları :
15 0100 |          DT1        EQU 0    ; /t1 cihazı
16 0100 |          DT2        EQU 1    ; /t2 cihazı
17 0100 |          DP         EQU 2    ; /p cihazı
18 0100 |          DKD        EQU 3    ; /kd cihazı
19 0100 |          DCAS       EQU 4    ; /cas cihazı
20 0100 |
21 0100 |          ; I/O port eşitlikleri :
22 0100 |          PIC0       EQU 40H   ; 8259 port 0

```

```

23 0100 |          PIC1          EQU  42H  ; 8259 port 1
24 0100 |
25 0100 |          ; Kullanıcı MUART tanımları :
26 0100 |          UCRREG1       EQU  80H  ; MUART komut yazmacı
27 0100 |          UCRREG2       EQU  82H  ; MUART komut yazmacı
28 0100 |          UCRREG3       EQU  84H  ; MUART komut yazmacı
29 0100 |          UMODEREG      EQU  86H  ; Mod yazmacı
30 0100 |          UPORT1CTL     EQU  88H  ; Port 1 yön denetimi
31 0100 |          UIRQEN        EQU  8AH  ; IRQ izin yazmacı
32 0100 |          UIRQADR       EQU  8CH  ; IRQ adres yazmacı
33 0100 |          URCVBUF       EQU  8EH  ; RS232 alıcı tamponu
34 0100 |          UPORT1        EQU  90H  ; Port 1
35 0100 |          UPORT2        EQU  92H  ; Port 2
36 0100 |          UTIMER1       EQU  94H  ; Timer 1
37 0100 |          UTIMER2       EQU  96H  ; Timer 2
38 0100 |          UTIMER3       EQU  98H  ; Timer 3
39 0100 |          UTIMER4       EQU  9AH  ; Timer 4
40 0100 |          UTIMER5       EQU  9CH  ; Timer 5
41 0100 |          USTATUS       EQU  9EH  ; Timer 6
42 0100 |
43 0100 |          ; Sistem altprogramları :
44 0100 |          READ          EQU  0
45 0100 |          READLN        EQU  1
46 0100 |          WRITE         EQU  2
47 0100 |          WRITLN        EQU  3
48 0100 |          EXIT          EQU  4
49 0100 |          PERR          EQU  5
50 0100 |
51 0100 |          ; Matematik altprogramları :
52 0100 |          AHEXTO        EQU  6
53 0100 |          ADECTO        EQU  7
54 0100 |          TOAHEX        EQU  8
55 0100 |          TOADEC        EQU  9
56 0100 |
57 0100 |          ; Kullanıcı altprogramları :
58 0100 |          RDCHAR        EQU  10
59 0100 |          RDBAYT        EQU  11
60 0100 |          WRCHAR        EQU  12
61 0100 |          WRBAYT        EQU  13
62 0100 |          GETIN         EQU  14
63 0100 |          WT1MS         EQU  15
64 0100 |          WTNMS         EQU  16
65 0100 |          CRLF          EQU  17
66 0100 |          CLRSCR        EQU  18
67 0100 |          LEDON         EQU  19
68 0100 |          LEDOFF        EQU  20
69 0100 |
71 0100 |          ORG 0300H
72 0300 | B0 FF          MOV AL,0FFH      ;port 1'in bütünüyle çıkış
73 0302 | E6 88          OUT UPORT1CTL,AL  ;olarak koşulllanması
74 0304 | 3E A0 00 10    MOV AL,BAYT PTR DS:1000H

```

```

75 0308 3E 02 06 00 11 ADD AL,BAYT PTR DS:1100H
76 030D E6 90 OUT UPORT1,AL ; sonucu port 1'e yaz
77 030F BB 00 00 SON: MOV BX,0000H
78 0312 B4 04 MOV AH,04H
79 0314 CD 28 INT 028H
80 0316 END

```

**b)** Giriş/çıkış tümdevresinin 1. portunun binary sayıcı gibi davranmasını sağlayan makina dili programı yazınız.

```

1 0100 .XLIST
2 0100 ORG 0500H
3 0500 B0 FF MOV AL,0FFH
4 0502 E6 88 OUT UPORT1CTL,AL
5 0504 B0 00 MOV AL,00H
6 0506 E6 90 CIKIS: OUT UPORT1,AL
7 0508 B9 FF FF MOV CX,0FFFFH
8 050B E2 FE GECIK: LOOP GECIK
9 050D FE C0 INC AL
10 050F E9 F4 FF JMP CIKIS
11 0512 END

```

**c)** Giriş/çıkış tümdevresinin 2. portunun her 1 saniye bir artan binary sayıcı gibi davranmasını sağlayan makina dili programı yazınız. Uygulama modülünün motor disk algılayıcısı sayma izin girişi olarak kullanılmıştır. Giriş bilgisi "0" ise sayım artarak devam edecek ve "1" olduğunda ise duracaktır.

```

1 0100 .XLIST
2 0100 ORG 0600H
3 0600 B0 00 MOV AL,00H
4 0602 E6 88 OUT UPORT1CTL,AL
5 0604 B0 03 MOV AL,03H
6 0606 E6 86 OUT UMODEREG,AL
7 0608 B0 00 MOV AL,00H
8 060A E6 92 OUT UPORT2,AL
9 060C E4 90 KONROL: IN AL,UPORT1
10 060E A8 10 TEST AL,10H
11 0610 75 FA JNZ KONTROL
12 0612 E4 92 SAY: IN AL,UPORT2
13 0614 FE C0 INC AL
14 0616 E6 92 OUT UPORT2,AL
15 0618 BB 05 00 MOV BX,05H
16 061B B9 FF FF MOV CX,0FFFFH
17 061E E2 FE GECIK: LOOP GECIK
18 0620 4B DEC BX
19 0621 75 FB JNZ GECIK
20 0623 E9 E6 FF JMP KONTROL
21 0626 END

```



d) Uygulama modülü üzerindeki motor disk dedektörünü giriş kabul eden ve girişin "1" olması halinde 2.porta bağlı tüm LED' leri yakan, girişin "0" olması halinde ise LED' lerin sönük kalmasını sağlayan makina dili programını tasarlayıp eğitim seti üzerinde çalıştırınız.

**Sorular:**

1. Bir mikrobilgisayar sisteminde veri giriş ve çıkışları ne üzerinden yapılır ?
2. Akümülatörde Port 2' den alınan bayt büyüklüğündeki veri saklanacaksa bu işlemin 80286 makina dili karşılığı nedir ?
3. Eğer Port 1 tümüyle çıkış olarak şartlanacaksa bunun için hangi 80286 makina dili komutlarına gereksinim olacaktır ?
4. Eğer LOOP komutu kullanılarak gecikme sağlanmak isteniyorsa CX yazmacına hangi hexadecimal sayı yüklenmelidir ? (NOT: 80286 eğitim sisteminin sistem saati 5MHz' dir ve LOOP komutunda herbir geçiş için 10 saat çevrimi gerekmektedir. )
5. Port 1' den C3<sub>H</sub>, Port 2' den 73<sub>H</sub> çıkışını verecek 80286 makina dili programını tasarlayınız.

**Deney No** : 7

**Deneyin Konusu** : Analogdan Sayısala Dönüştürme ve Sayısalda Analoga Dönüştürme

**Deneyin Amacı** : Eğitim sisteminin uygulama modülünde bulunan Analogdan Sayısala Dönüştürücü (ADC) ve Sayısalda Analoga Dönüştürücü (DAC) tümleşik devrelerinin çalışma yöntemlerinin, donanımlarının incelenmesi ve programlanmasının yapılmasıdır.

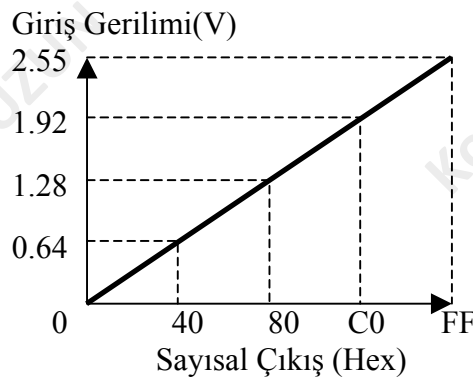
**Deney Öncesi Yapılacak İşlemler:**

Verilen programı inceleyerek her komutun açıklamasını kısaca yazınız.

**Analog Bilgiyi Sayısal Bilgiye Dönüştürücüler (ADC) :**

8-bitlik bir ADC 0-2.55V arasındaki analog bilgiyi 8-bit dijital bilgiye dönüştürür. Bir ADC anahtarı optik alıcının (optical receiver) veya potansiyometrenin çıkışının ADC 'in girişine bağlanmasına imkan verir.

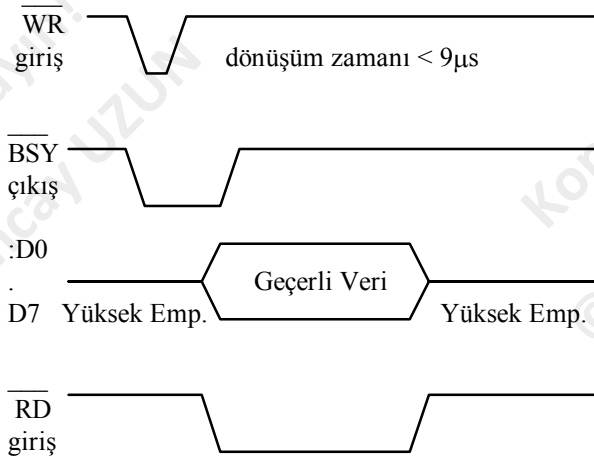
Seçilen giriş aşağıda gösterildiği gibi 8-bit dijital bilgiye dönüştürülür.



Bilgisayarın örnekselden sayısala dönüşüm yapabilmesi için aşağıdaki işlevleri gerçeklemesi gereklidir.

- Başlangıç olarak  $\overline{RD}$  ve  $\overline{WR}$  "high" seviyeye set edilmelidir.
- Dönüşümün başlatılabilmesi için  $\overline{WR}$  üzerine kısa süreli bir darbenin uygulanması gerekir. (1->0 >1) Bu bit "high" seviyeye gelinceye dek dönüşüm başlayamayacağından bu darbe süresi oldukça küçük tutulmalıdır. (200 ns)
- Dönüşümün tamamlandığını göstermek üzere  $\overline{BSY}$  hattının lojik 1 seviyesine dönmesi beklenmelidir. Bu süre maksimum 9  $\mu s$  olur. Eğer yazılımın gerçekleşmesi bu süreden daha uzunsa bu hat ihmal edilmelidir.
- ADC 'in çıkışlarının aktif hale getirilmesi için  $\overline{RD}$  'in lojik 0 yapılması gereklidir.
- A portu üzerinden (PA0-PA7) veri hatları (D0-D7) okunmalıdır.
- ADC 'in çıkışlarının pasif hale getirilmesi için,  $\overline{RD}$  lojik 1 yapılmalıdır.

Bu ardışıl işlemler grafiksel olarak aşağıda gösterilmiştir.



### Sayısal Bilgiyi Örneksel Bilgiye Dönüştürücüler (DAC) :

Eğer mikroişlemci temelli bir kontrol sistemi örneksel bir çıkış üretecekse, bir Sayısal Bilgiyi Örneksel Bilgiye Dönüştürücüye (DAC) gereksinim olacaktır. Bir DAC aldığı sayısal değere karşı çıkışında bir gerilim seviyesi oluşturur. 80286 eğitim seti üzerinde 8-bitlik DAC bulunmaktadır. Bu DAC 'nin çıkış gerilim seviyesi 0-2.55V arasında değişmektedir. Bu değişimi yukarıdaki şekilde görmek mümkündür. Görüldüğü gibi 0 ile 2.55V arasında  $FF_H = 255_{10}$  tane adım bulunmaktadır. Buna göre her  $01_H$  artış, çıkış gerilim seviyesini 10mV yükseltecektir. Çıkış, girişteki sayısal veriyle orantılı bir şekilde değişecektir.

Eğitim seti üzerinde bulunan DAC anahtarı, DAC çıkışının Optik Gönderici veya DC Motora uygulanmasını sağlar.

Sayısaldan Örneksele dönüşüm için aşağıdaki işlemlerin yapılması gerekir:

1. DAC 'yi aktif hale getirmek için B portunun (port 1) 0.biti  (PB0) "0" yapılmalıdır. Bu sayede dönüşümün başlatılabilmesi için gerekli EN sinyali üretilmiş olur.
2. A portuna (port 2) sayısal veri yazılmalıdır.

### Potansiyometreler :

Uygulama modülündeki  $10k\Omega$  potansiyometre ADC 'nin 2.55V 'lik referans çıkışı ile toprak arasında bağlı olup 0 ile 2.55V arasında değişen sürekli gerilim değerleri sağlar.

### Deneyde Yapılacak İşlemler :

- a) Uygulama modülündeki potansiyometrenin aldığı değere göre 2.porttan binary çıkış veren aşağıdaki makina dili programının analizini eğitim seti üzerinde yapınız.

Satır	Adres	İşlem kodu	Makina dili kaynak program
1	0100		ORG 0500H
2	0500	UMODEREG	EQU 086H
3	0500	UPOINTCTL1	EQU 088H
4	0500	UPOINT1	EQU 090H
5	0500	UPOINT2	EQU 092H
6	0500		
7	0500		
8	0500		
9	0500	B0 00	AGAIN: MOV AL,00H
10	0502	E6 86	OUT UMODEREG,AL
11	0504	B0 0A	MOV AL,0AH
12	0506	E6 88	OUT UPOINTCTL1,AL
13	0508		
14	0508	B0 02	MOV AL,02H
15	050A	E6 90	OUT UPOINT1,AL
16	050C	B0 00	MOV AL,00H
17	050E	E6 90	OUT UPOINT1,AL
18	0510	B0 02	MOV AL,02H
19	0512	E6 90	OUT UPOINT1,AL
20	0514		
21	0514		
22	0514	E4 90	CMPLTE: IN AL,UPOINT1
23	0516	A8 04	TEST AL,04H
24	0518	74 FA	JZ CMPLTE
25	051A		
26	051A		
27	051A	B0 F7	MOV AL,0F7H
28	051C	E6 90	OUT UPOINT1,AL
29	051E	E4 92	IN AL,UPOINT2
30	0520	88 C3	MOV BL,AL
31	0522		
32	0522	B0 80	MOV AL,08H
33	0524	E6 90	OUT UPOINT1,AL
34	0526	B0 03	MOV AL,03H
35	0528	E6 86	OUT UMODEREG,AL
36	052A	88 D8	MOV AL,BL
37	052C	E6 92	OUT UPOINT2,AL
38	052E		
39	052E	B9 00 07	MOV CX,0700H
40	0531	E2 FE	DELAY: LOOP DELAY
41	0533	E9 CA FF	JMP AGAIN

b) Port 2 üzerinde 0' dan başlayarak yaklaşık her 0.13 sn' de bir defa olmak üzere binary sayma yapan, sayma işleminde elde edilen sayısal bilgiyi DAC kanalıyla örneksel gerilime dönüştürüp Optik Gönderici veya DC Motora uygulayan aşağıdaki makina dili programının analizini eğitim seti üzerinde yapınız

Satır	Adres	İşlem kodu	Makina dili kaynak program
1	0100		
2	0400		ORG 0400H
3	0400	UMODEREG	EQU 086H
4	0400	UPOINTCTL1	EQU 088H
5	0400	UPOINT1	EQU 090H
6	0400	UPOINT2	EQU 092H
7	0400		
8	0400	B0 01	MOV AL,01H
9	0402	E6 88	OUT UPOINTCTL1,AL
10	0404	B0 03	MOV AL,03H
11	0406	E6 86	OUT UMODEREG,AL
12	0408	B0 00	MOV AL,00H
13	040A	E6 90	OUT UPOINT1,AL
14	040C		
15	040C	B0 00	MOV AL,00H
16	040E	E6 92	OUTPUT: OUT UPOINT2,AL
17	0410	B9 FF FF	MOV CX,0FFFFH
18	0413	E2 FE	DELY: LOOP DELY
19	0415	FE C0	INC AL
20	0417	E9 F4 FF	JMP OUTPUT

c) Tuştakımı üzerindeki "S" tuşuna basıldığında piezo sounder' ı aktif hale getiren aşağıdaki makina dili programının analizini eğitim seti üzerinde yapınız

Satır	Adres	İşlem kodu	Makina dili kaynak program
1	0100		INCLUDE PATCALLS.INC
2	0100		
3	0100		; yzmaç deęerleri.
4	0100		
5	0100		; 80286 Eđitim sistemi
6	0100		
7	0100		; Çok kullanılan deęerler:
8	0100		USER EQU 40 ; Monitör kullanıcı fonksiyonu
9	0100		USRBSE EQU 100H; Kullanıcı program başlangıcı (org)
10	0100		USRSEG EQU 80H ; Kullanıcı program parçası
11	0100		SYSSEG EQU 0H ; Sistem parçası
12	0100		KYDBUF EQU 47DH; Tuş takımı/gösterge tamponu
13	0100		
14	0100		; Cihaz numaraları :

```

15 0100 |          DT1      EQU  0   ; /t1 cihazı
16 0100 |          DT2      EQU  1   ; /t2 cihazı
17 0100 |          DP        EQU  2   ; /p cihazı
18 0100 |          DKD       EQU  3   ; /kd cihazı
19 0100 |          DCAS      EQU  4   ; /cas cihazı
20 0100 |
21 0100 |          ; I/O port eşitlikleri :
22 0100 |          PIC0      EQU 40H   ; 8259 port 0
23 0100 |          PIC1      EQU 42H   ; 8259 port 1
24 0100 |
25 0100 |          ; Kullanıcı MUART tanımları :
26 0100 |          UCRREG1   EQU 80H   ; MUART komut yazmacı
27 0100 |          UCRREG2   EQU 82H   ; MUART komut yazmacı
28 0100 |          UCRREG3   EQU 84H   ; MUART komut yazmacı
29 0100 |          UMODEREGEQU 86H   ; Mod yazmacı
30 0100 |          UPORT1CTLEQU 88H   ; Port 1 yön denetimi
31 0100 |          UIRQEN    EQU 8AH   ; IRQ izin yazmacı
32 0100 |          UIRQADR   EQU 8CH   ; IRQ adres yazmacı
33 0100 |          URCVBUF   EQU 8EH   ; RS232 alıcı tamponu
34 0100 |          UPORT1    EQU 90H   ; Port 1
35 0100 |          UPORT2    EQU 92H   ; Port 2
36 0100 |          UTIMER1   EQU 94H   ; Timer 1
37 0100 |          UTIMER2   EQU 96H   ; Timer 2
38 0100 |          UTIMER3   EQU 98H   ; Timer 3
39 0100 |          UTIMER4   EQU 9AH   ; Timer 4
40 0100 |          UTIMER5   EQU 9CH   ; Timer 5
41 0100 |          USTATUS   EQU 9EH   ; Timer 6
42 0100 |
43 0100 |          ; Sistem alt programları :
44 0100 |          READ      EQU  0
45 0100 |          READLN    EQU  1
46 0100 |          WRITE     EQU  2
47 0100 |          WRITLN    EQU  3
48 0100 |          EXIT      EQU  4
49 0100 |          PERR      EQU  5
50 0100 |
51 0100 |          ; Matematik alt programları :
52 0100 |          AHEXTO    EQU  6
53 0100 |          ADECTO    EQU  7
54 0100 |          TOAHEX    EQU  8
55 0100 |          TOADEC    EQU  9
56 0100 |
57 0100 |          ; Kullanıcı alt programları :
58 0100 |          RDCHAR    EQU 10
59 0100 |          RDBAYT    EQU 11
60 0100 |          WRCHAR    EQU 12
61 0100 |          WRBAYT    EQU 13
62 0100 |          GETIN     EQU 14
63 0100 |          WT1MS     EQU 15
64 0100 |          WTNMS     EQU 16
65 0100 |          CRLF      EQU 17

```

```

66 0100 |          CLRSCR   EQU   18
67 0100 |          LEDON    EQU   19
68 0100 |          LEDOFF   EQU   20
69 0100 |
70 0100 |          ORG 0800H

71 0800 B4 0E   CHECK:  MOV AH,GETIN
72 0802 CD 28          INT 28H
73 0804 3C FF          CMP AL,0FFH
74 0806 74 F8          JZ CHECK
75 0808 3C 53          CMP AL,053H
76 080A 75 F4          JNZ CHECK
77 080C
78 080C B0 20          MOV AL,20H
79 080E E6 88          OUT UPORT1CTL,AL
80 0810 B3 64          MOV BL,64H
81 0812
82 0812 B0 00   OUTPUT: MOV AL,00H
83 0814 E6 90          OUT UPORT1,AL
84 0816 B4 0F          MOV AH,WT1MS
85 0818 CD 28          INT 28H
86 081A
87 081A B0 20          MOV AL,20H
88 081C E6 90          OUT UPORT1,AL
89 081E B4 0F          MOV AH,WT1MS
90 0820 CD 28          INT 28H
91 0822 FE CB          DEC BL
92 0824 75 EC          JNZ OUTPUT
93 0826
94 0826 E9 D7 FF          JMP CHECK

```

d) Bir önceki deneye benzer şekilde Tuş takımı üzerinde “S” tuşuna basılması ve bunu takip eden yaklaşık 2sn içerisinde de buna ek olarak “R” tuşuna basılması halinde piezo sounder’ ı aktif hale getiren makina dili programını tasarlayınız ve eğitim seti üzerinde çalıştırınız.

#### Sorular :

1. Uygulama modülü üzerindeki DAC ’nin girişinde meydana gelecek  $05_H$  değerindeki bir artış, DAC ’nin çıkış voltajını ne kadar artırır ?
2. Uygulama modülü üzerindeki ADC’ den mikroişlemciye dönüşümün tamamlandığını göstermek üzere hangi sinyal gönderilir ?
3. Uygulama modülü üzerindeki ADC ’nin girişinde 2.0V ’luk bir gerilim varsa dönüşüm sonucunda hangi hexadecimal sayı elde edilecektir ?
4. Uygulama modülü üzerindeki ADC’ de örneksel bilgiden sayısal bilgiye geçiş adımları nelerdir ?
5. Örnekselden sayısala dönüşümde ve sayısaldan örneksele dönüşümde kullanılan yöntemler ve özellikleri nelerdir ?

**Deney No** : 8

**Deneyin Konusu** : DT-25 Tuştakımı/Gösterge Modülünün İncelenmesi

**Deneyin Amacı** : 80286 Temelli Bir Giriş/Çıkış Örneğinin İncelenmesi

### Deney Öncesi Yapılacak İşlemler:

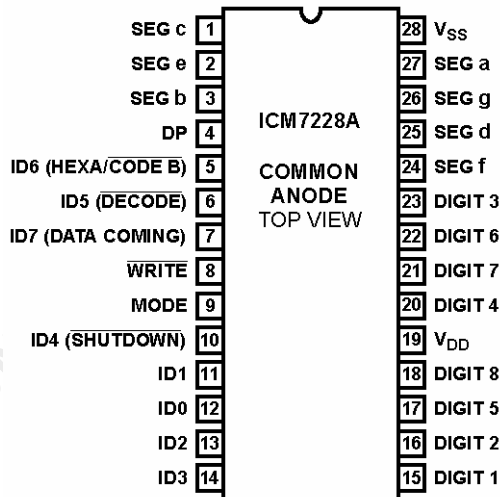
ICM7228A Dinamik 7-Parçalı Gösterge Sürücüsü ile P8256AH Çok Amaçlı Mikroişlemci Yardımcı Denetleyicisi (MUART) entegrelerinin çalışma şeklinin incelenerek DT-25 Modülünün çalışma mantığının teorik olarak incelenmesi.

### DT-25 Tuştakımı/Gösterge Modülü:

DIGIAC 2000 80286 eğitim setinin DT-25 Tuş takımı/Gösterge biriminin çalışması prensibi ICM7228 entegresi ile göstergelerin sürülmesi ve tuş takımının matris biçiminde sütunlardan sürülerek satırlardan okunması mantığıyla çalışır. DT-25 biriminin açık devre şeması ektedir.

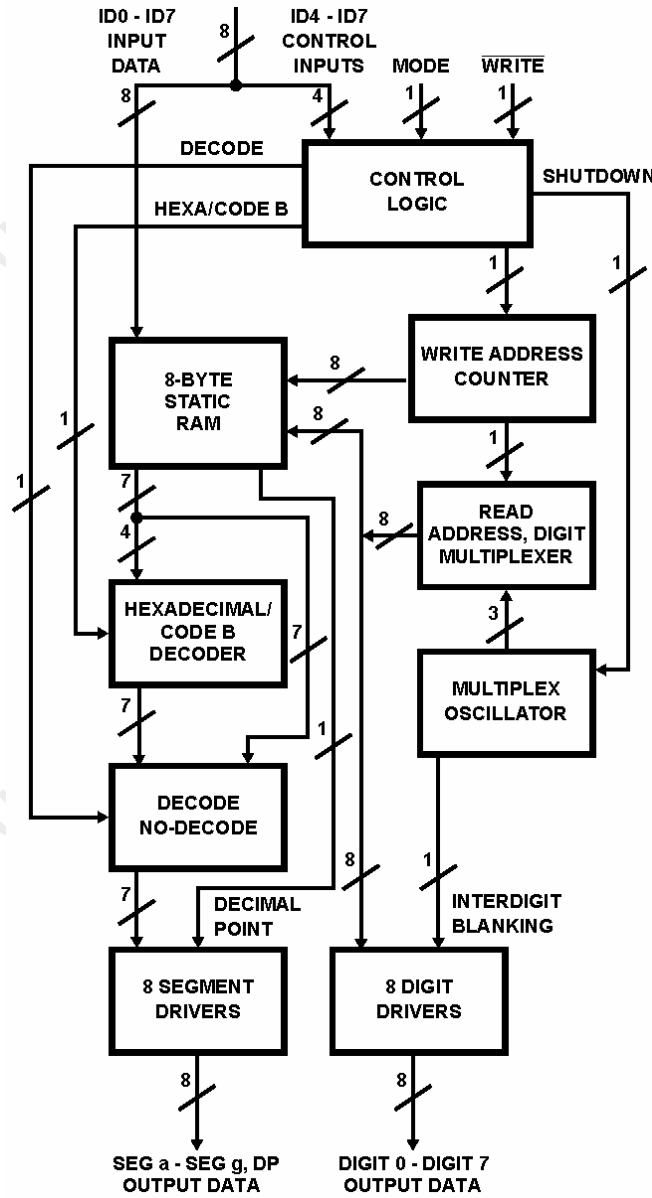
ICM7228 entegresi 8 adet 7-parçalı sayısal LED göstergeyi dinamik olarak tarayan bir mikroişlemci ara-yüzüdür. İy yapısının blok şeması Şekil 2'de gözükken çip üzerinde iki farklı tür 7-parçalı çözümleyici, çoğullayıcı tarama devresi, LED gösterge parça sürücüler, LED gösterge hane sürücüler ile 8 bayt büyüklüğünde statik RAM bulunmaktadır.

ICM7228A entegresi ortak anotlu göstergeleri sürebilir. Göstergelere iletilmek istenen veriler RAM'e ardışık 8-Hane veya Tek-Hane Güncelleme ile yapılabilir. Veriler Onaltılık modda, Kod B modunda veya Kodlanmamış modda iletebilir. Kodlanmamış modda veri iletmek ayırık parça kontrolü sağladığından nümerik olmayan karakterlerin de göstergeye iletilmesini mümkün kılar.



**Şekil 1** ICM7228A entegresinin uç yapısı.

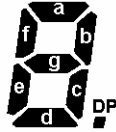




Şekil 2 ICM7228A entegresinin iç blok yapısı.

INPUT DATA LINES			SELECTED DIGIT
1D2	ID2	ID0	
0	0	0	DIGIT 1
0	0	1	DIGIT 2
0	1	0	DIGIT 3
0	1	1	DIGIT 4
1	0	0	DIGIT 5
1	0	1	DIGIT 6
1	1	0	DIGIT 7
1	1	1	DIGIT 8

Şekil 3 ICM7228A entegresinin Tek Hane Güncelleme modunda kullandığı kodlama tablosu.

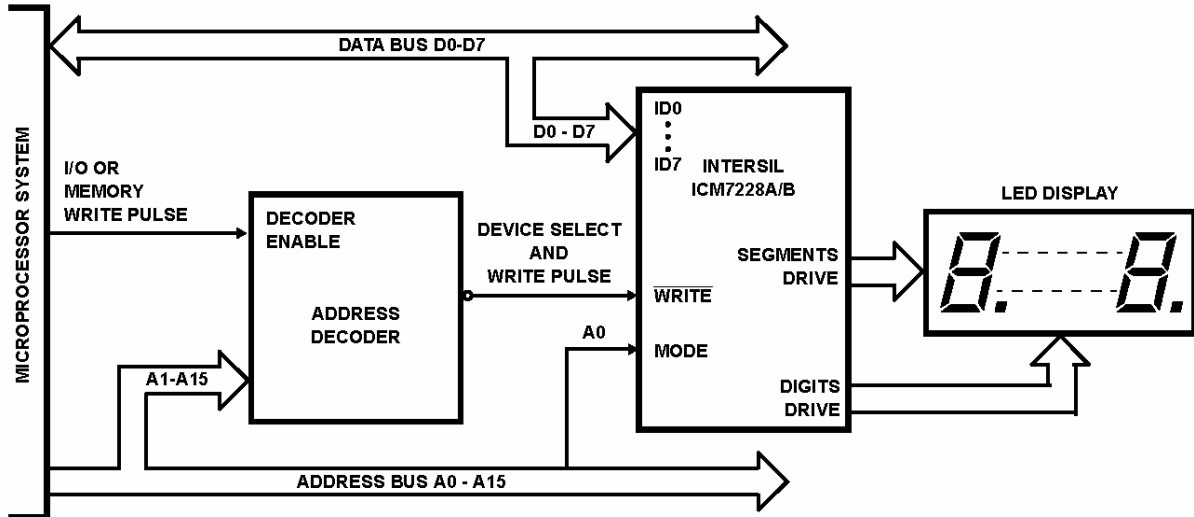


DATA INPUT	ID7	ID6	ID5	ID4	ID3	ID2	ID1	ID0
Controlled Segment	Decimal Point	a	b	c	e	g	f	d

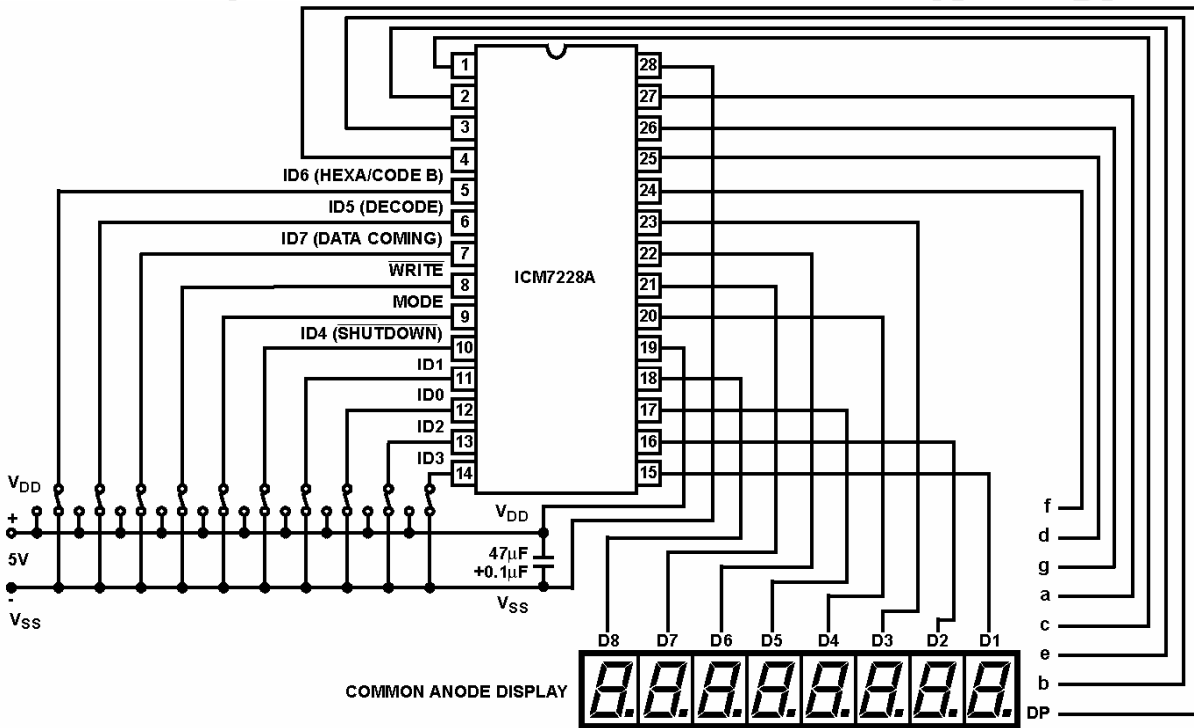
**Şekil 4** ICM7228A entegresinin kodlanmamış girişlerini karşılık segment çıkış tablosu.  
(Not: DT-25 modülünde bağlantılar belirtilen şekilde değil, ID0-ID6 girişlerine karşılık sırasıyla a-g segmentleri karşı düşecek şekilde yapılmıştır.)

**Tablo 1** ICM7228A entegresinin uç ayrıntıları.

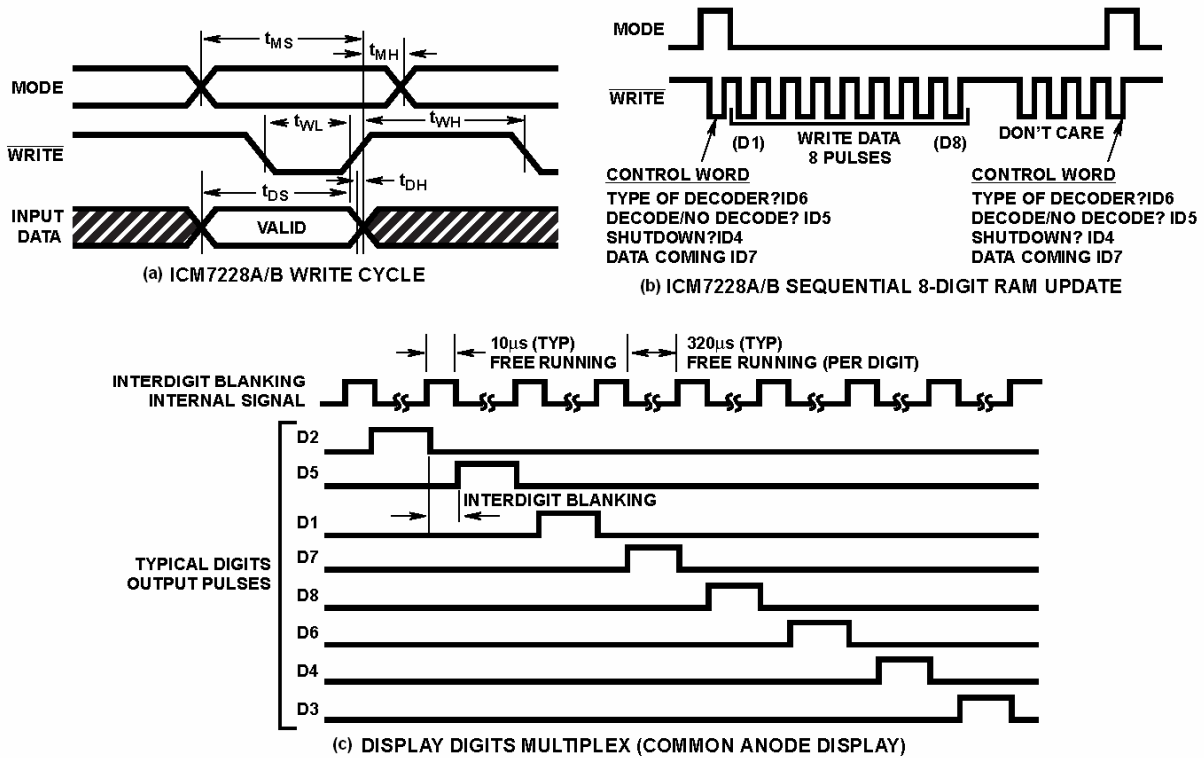
PIN NO.	NAME	FUNCTION	DESCRIPTION
1	SEG c	Output	LED Display Segments c, e, b and Decimal Point Drive Lines.
2	SEG e		
3	SEG b		
4	DP		
5	ID6, (HEXA/CODE B)	Input	When "MODE" Low: Display Data Input, Bit 7. When "MODE" High: Control Bit, Decoding Scheme Selection: High, Hexadecimal Decoding; Low, Code B Decoding.
6	ID5, (DECODE)	Input	When "MODE" Low: Display Data Input, Bit 6. When "MODE" High: Control Bit, Decode/No Decode Selection: High, No Decode; Low, Decode.
7	ID7, (DATA COMING)	Input	When "MODE" Low: Display Data Input, Bit 8, Decimal Point Data. When "MODE" High: Control Bit, Sequential Data Update Select: High, Data Coming; Low, No Data Coming.
8	WRITE	Input	Data Input Will Be Written to Control Register or Display RAM on Rising Edge of WRITE.
9	MODE	Input	Selects Data to Be Loaded to Control Register or Display RAM: High, Loads Control Register; Low, Loads Display RAM.
10	ID4, (SHUTDOWN)	Input	When "MODE" Low: Display Data Input, Bit 5. When "MODE" High: Control Bit, Low Power Mode Select: High, Normal Operation; Low, Oscillator and Display Disabled.
11	ID1	Input	When "MODE" Low: Display Data Input, Bit 2. When "MODE" High and "ID7 (DATA COMING)" Low: Digit Address, Bit 2, Single Digit Update Mode.
12	ID0	Input	When "MODE" Low: Display Data Input, Bit 1. When "MODE" High and "ID7 (DATA COMING)" Low: Digit Address, LSB, Single Digit Update Mode.
13	ID2	Input	When "MODE" Low: Display Data Input, Bit 3. When "MODE" High and "ID7 (DATA COMING)" Low: Digit Address, MSB, Single Digit Update Mode.
14	ID3	Input	When "MODE" Low: Display Data Input, Bit 4. When "MODE" High: RAM Bank Select (Decode Modes Only): High, RAM Bank A; Low, RAM Bank B
15	DIGIT 1	Output	LED Display Digits 1, 2, 5 and 8 Drive Lines.
16	DIGIT 2		
17	DIGIT 5		
18	DIGIT 8		
19	V <sub>DD</sub>	Supply	Device Positive Power Supply Rail.
20	DIGIT 4	Output	LED Display Digits 4, 7, 6 and 3 Drive Lines.
21	DIGIT 7		
22	DIGIT 6		
23	DIGIT 3		
24	SEG f	Output	LED Display Segments f, d, g and a Drive Lines.
25	SEG d		
26	SEG g		
27	SEG a		
28	V <sub>SS</sub>	Supply	Device Ground or Negative Power Supply Rail.



Şekil 5 ICM7228A entegresinin sisteme bağlantı şeması örneğinin blok diyagramı.



Şekil 6 ICM7228A entegresinin örnek test devresinin açık devre şeması.



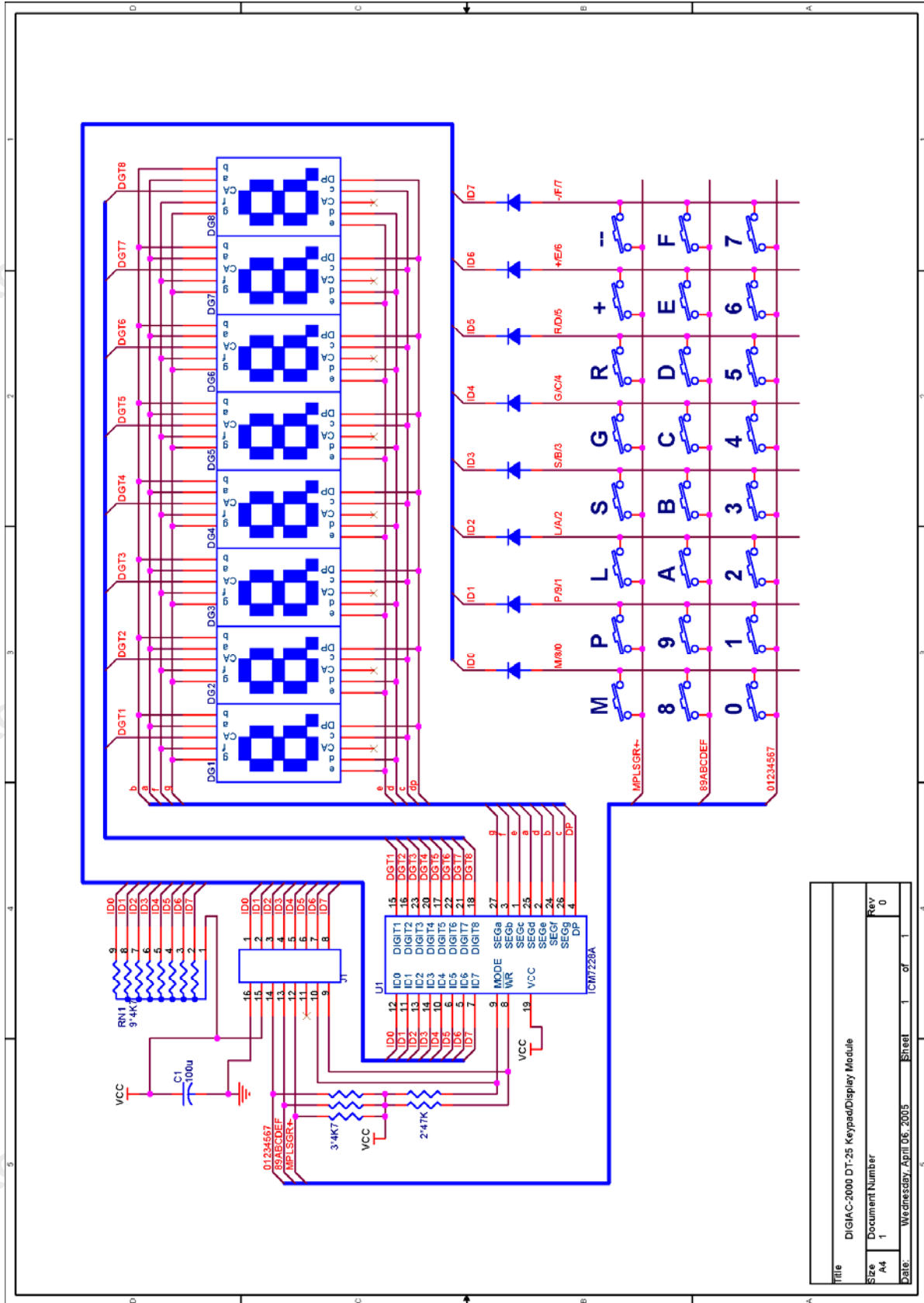
**Şekil 7** ICM7228A entegresinin zamanlama çizelgeleri. (a) Veri yazma çizelgesi. (b) Sıralı veri yazma örneği. (c) Entegrenin basamaklar arasındaki çoğullamayı gerçekleştirme sırası.

### Deneyde Yapılacak İşlemler:

- Deneyin ilk kısmında göstergede READY yazısı varken ve bir tuşa basılı değilken ID0-ID7 pinlerindeki dalga şekillerini osiloskop ekranında inceleyerek çizin ve yorumlayınız.
- Ekranda READY yazısı varken 0-7 arasındaki tuşlara sırasıyla basılı tutarak ID0-ID7 pinlerindeki dalga şeklini ayrı ayrı inceleyerek çizin. Bu tuşların birer ucu aynı düğüme bağlı olduğuna göre her bir tuşa basılma durumu için ortak uçtaki dalga şeklini inceleyerek çizin. Ölçüm sonuçlarınızı yorumlayınız.
- Göstergede READY yazısı gözükürken MODE ve  $\overline{WR}$  uçlarının durumlarını osiloskopta inceleyerek yorumlayınız.
- DIGIT1-DIGIT8 pinlerindeki işaretleri osiloskopta inceleyerek taramanın nasıl yapıldığını inceleyiniz ve yorumlayınız.
- Ekrana kendi isminizi yazmak istediğinizde DT-25 modülüne hangi sırayla hangi işaretlerin gönderilmesi gerektiğini tasarlayınız.

### Sorular :

- DT-25'in matris tuştakımı sürme tekniği nasıldır?
- ICM7228A entegresinin hangi özelliği tuştakımı sürme pinleri ile göstergeye veri iletme pinlerini ortaklaştırmamıza olanak sağlamıştır?
- DT-25 modülüne veri iletimi sırasında veriler 8256AH entegresinin portları ne şekilde kullanılmaktadır?
- ICM7228A entegresine kodlanmamış şekilde sırasıyla 1-8 arasındaki sayıların segment karşılıkları iletilmek istendiğinde yapılması gereken işlemler nelerdir?



Şekil 8 DT-25 Tuştakımı/Gösterge modülünün açık devre şeması.

**Deney No** : 9

**Deneyin Konusu** : Standart Tuş Takımı ve Gösterge Birimi Yazılımının Tasarlanması.

**Deneyin Amacı** : 8086 temelli bir mikroişlemcili sistem için basit kullanıcı giriş/çıkış birimi uygulamasının hazır donanım üzerinden yazılımla gerçekleştirilmesi.

**Deney Öncesi Yapılacak İşlemler:**

- a) Eğitim sisteminin tuş takımı ve gösterge birimini kullanarak tuş takımındaki onaltılık bir rakama basıldığında ilgili rakamı göstergenin en sağdaki hanesine yazan, 'S' tuşuna basıldığında programı sonlandıran, diğer tuşlardan birine basıldığında ise herhangi bir işlem yapmayan programı eğitim sisteminin alt programlarını kullanmadan tasarlayınız. Örneğin sırasıyla '5', 'F', ve '3' tuşlarına basıldığında aşağıdaki üç durum görülmelidir.

5

F

3

- b) Eğitim sisteminin tuş takımı okuma ve göstergeye yazma alt programlarını kullanarak eğitim sisteminin 'M' tuş fonksiyonunu yerine getiren, bellek gözlerinin içeriğinin görüntülenmesini ve değiştirilebilmesini sağlayan bir program tasarlayınız. İlk olarak göstergede aşağıdaki adres değiştirme yazısı görülecektir.

ADR - 0 100

Herhangi bir onaltılık rakam (0-F) tuşuna basıldığında mevcut adres değeri bir hane sola kaydırılarak basılan değer en sağdaki göstergede gösterilecektir. Örneğin 'F' tuşuna basıldığında göstergede aşağıdaki durum görülecektir

ADR - 100F

'+' veya '-' tuşlarına basıldığında göstergedeki adres bir artmalı veya bir azalmalıdır. Örneğin '+' tuşuna basıldığında gösterge aşağıdaki gibi görülecektir.

A 7-segment display showing the hexadecimal address 'A000'. The digits are 'A', '0', '0', and '0'.

'M' tuşuna basıldığında önceden göstergede görünmekte olan adres değeri en solda, adresin içeriği ise en sağda görünecek şekilde veri değiştirme yazısı göstergede görülecektir.

A 7-segment display showing the hexadecimal address '0000'. The digits are '0', '0', '0', and '0'.

Herhangi bir onaltılık rakam tuşuna (0-F) basıldığında mevcut adresin içindeki değer bir hane sola kaydırılarak basılan değer en sağdaki göstergede görünmelidir. Örneğin 'A', '2' ve 'B' tuşlarına sırayla basıldığında göstergede sırasıyla aşağıdaki üç durum görülecektir.

A 7-segment display showing the hexadecimal address '000A'. The digits are '0', '0', '0', and 'A'.

A 7-segment display showing the hexadecimal address '00A2'. The digits are '0', '0', 'A', and '2'.

A 7-segment display showing the hexadecimal address '002B'. The digits are '0', '0', '2', and 'B'.

'+' veya '-' tuşlarına basıldığında göstergedeki adres bir artmalı veya bir azalmalıdır. Örneğin '+' tuşuna basıldığında gösterge aşağıdaki gibi görülecektir.

A 7-segment display showing the hexadecimal address '0001'. The digits are '0', '0', '0', and '1'.

'M' tuşuna bir kez daha basıldığında göstergede tekrar adres değiştirme yazısı görünmelidir. Örneğin gösterge yukarıdaki durumda iken 'M' tuşuna basıldığında gösterge aşağıdaki gibi görülecektir.

A 7-segment display showing the hexadecimal address 'A001'. The digits are 'A', '0', '0', and '1'.

'S' tuşuna basıldığında göstergede veri veya adres değiştirme yazısı görünmesinden bağımsız olarak tuş takımı ve gösterge biriminin kontrolü eğitim sistemi programına geri bırakılmalıdır. Program içinde belirtilenler haricinde bir tuşa basılması durumunda program herhangi bir işlem yapmamalıdır.

**Not 1** İşlevi olan herhangi bir tuşa basılı tutulduğunda o tuşun yapması beklenen işlem yalnızca bir defa yapılmalıdır. Örneğin adres değiştirme yazısı göstergede bulunmakta iken '-' tuşuna basılı tutulduğunda adres değeri yalnızca bir azalmalıdır.

**Not 2** Kullanılacak olan alt programlar CLRSCR (12<sub>H</sub>), WRITE (02<sub>H</sub>), AHEXTO (06<sub>H</sub>), TOAHEX (08<sub>H</sub>), GETIN (0E<sub>H</sub>) ve EXIT (04<sub>H</sub>).

### Deneyde Yapılacak İşlemler:

- a) Tasarladığınız tuş takımı ve gösterge birimi kontrol programını eğitim sistemine aktararak çalıştırınız.
- b) Tasarladığınız 'M' tuş fonksiyonunu yerine getiren programı eğitim sistemine aktararak çalıştırınız.

### Sorular :

1. Eğitim sisteminin CLRSCR (12<sub>H</sub>), WRITE (02<sub>H</sub>), AHEXTO (06<sub>H</sub>), TOAHEX (08<sub>H</sub>), GETIN (0E<sub>H</sub>) ve EXIT (04<sub>H</sub>) alt programlarının çalışma mantığı nasıldır, açıklayınız.
2. WRITE (02<sub>H</sub>) eğitim sistemi alt programının benzerini, yalnızca standart giriş/çıkış biriminin göstergesine 8 baytlık veri aktaracak şekilde kendiniz tasarlayınız. Verilerin bellekteki 1000<sub>H</sub> adresinden başlayarak kodlanmamış şekilde saklandığı varsayılacaktır. (Programın akış diyagramını çizmek yeterlidir)
3. Gösterge sürücü 7228A kullanılmadığı durumda 8 adet göstergenin nasıl sürülebileceğini tasarlayın.



## Detailed Description

### System Interfacing and Data Entry Modes, ICM7228A and ICM7228B

The ICM7228A/B devices are compatible with the architectures of most microprocessor systems. Their fast switching characteristics makes it possible to access them as a memory mapped I/O device with no wait state necessary in most microcontroller systems. All the ICM7228A/B inputs, including MODE, feature a 250ns minimum setup and 0ns hold time with a 200ns minimum WRITE pulse. Input logic levels are TTL and CMOS compatible. Figure 9 shows a generic method of driving the ICM7228A/B from a microprocessor bus. To the microprocessor, each device appears to be 2 separate I/O locations; the Control Register and the Display RAM. Selection between the two is accomplished by the MODE input driven by address line A0. Input data is placed on the ID0 - ID7 lines. The WRITE input acts as both a device select and write cycle timing pulse. See Figure 1 and Switching Specifications Table for write cycle timing parameters.

The ICM7228A/B have three data entry modes: Control Register update without RAM update, sequential 8-digit update and single digit update. In all three modes a control word is first written by pulsing the WRITE input while the MODE input is high, thereby latching data into the Control Register. The logic level of individual bits in the Control Register select Shutdown, Decode/No Decode, Hex/Code B, RAM bank A/B and Display RAM digit address as shown in Tables 1 and 2.

The ICM7228A/B Display RAM is divided into 2 banks, called bank A and B. When using the Hexadecimal or code B display modes, these RAM banks can be selected separately. This allows two separate sets of display data to be stored and displayed alternately. Notice that the RAM bank selection is not possible in No-Decode mode, this is because the display data in the No-Decode mode has 8 bits, but in Decoded schemes (Hex/Code B) is only 4 bits (ID0 - ID3 data). It should also be mentioned that the decimal point is independent of selected bank, a turned on decimal point will remain on for either bank. Selection of the RAM banks is controlled by ID3 input. The ID3

logic level (during Control Register update) selects which bank of the internal RAM to be written to and/or displayed.

### Control Register Update without RAM Update

The Control Register can be updated without changing the display data by a single pulse on the WRITE input, with MODE high and DATA COMING low. If the display is being decoded (Hex/Code B), then the value of ID3 determines which RAM bank will be selected and displayed for all eight digits.

### Sequential 8-Digit Update

The logic state of DATA COMING (ID7) is also latched during a Control Register update. If the latched value of DATA COMING (ID7) is high, the display becomes blanked and a sequential 8-digit update is initiated. Display data can now be written into RAM with 8 successive WRITE pulses, starting with digit 1 and ending with digit 8 (See Figure 2). After all 8 RAM locations have been written to, the display turns on again and the new data is displayed. Additional write pulses are ignored until a new Control Register update is performed. All 8 digits are displayed in the format (Hex/Code B or No Decode) specified by the control word that preceded the 8 digit update. If a decoding scheme (Hex/Code B) is to be used, the value of ID3 during the control word update determines which RAM bank will be written to.

### Single Digit Update

In this mode each digit data in the display RAM can be updated individually without changing the other display data. First, with MODE input high, a control word is written to the Control Register carrying the following information; DATA COMING (ID7) low, the desired display format data on ID4 - ID6, the RAM bank selected by ID3 (if decoding is selected) and the address of the digit to be updated on data lines ID0 - ID2 (See Table 5). A second write to the ICM7228A/B, this time with MODE input low, transfers the data at the ID0 - ID7 inputs into the selected digit's RAM location. In single digit update mode, each individual digit's data can be specified independently for being displayed in Decoded or No-Decode mode. For those digits which decoding scheme (Hex/Code B) is selected, only one can be effective at a time. Whenever a control word is written, the specified decoding scheme will be applied to all those digits which selected to be displayed in Decoded mode.

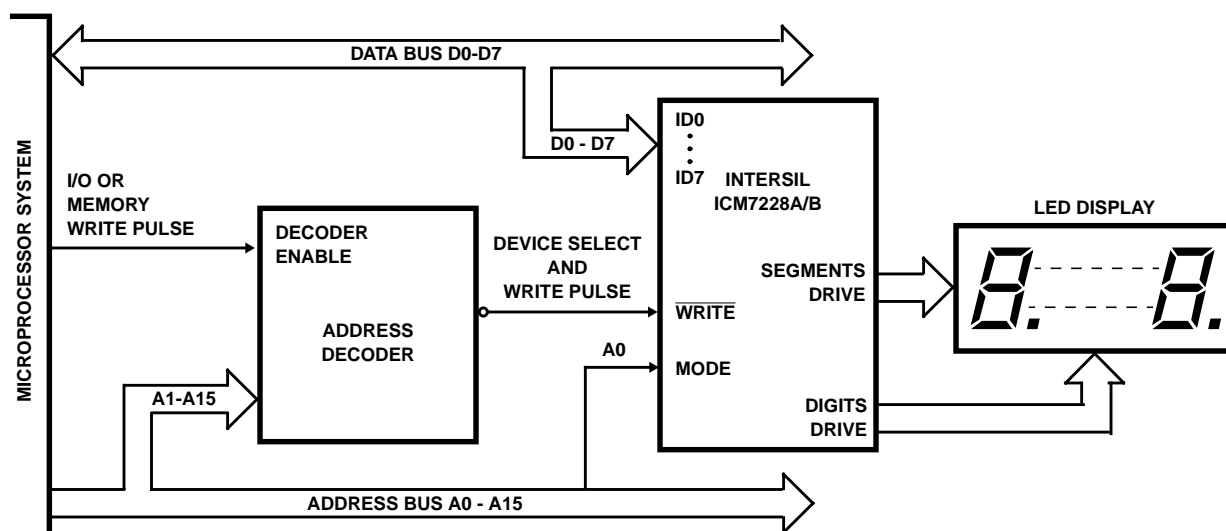


FIGURE 9. ICM7228A/B MICROPROCESSOR SYSTEM INTERFACING

# ICM7228

**TABLE 5. DIGITS ADDRESS, ICM7228A/B**

INPUT DATA LINES			SELECTED DIGIT
1D2	ID2	ID0	
0	0	0	DIGIT 1
0	0	1	DIGIT 2
0	1	0	DIGIT 3
0	1	1	DIGIT 4
1	0	0	DIGIT 5
1	0	1	DIGIT 6
1	1	0	DIGIT 7
1	1	1	DIGIT 8

### System Interfacing, ICM7228C and ICM7228D

The ICM7228C/D devices are directly compatible with the architecture of most microprocessor systems. Their fast switching characteristics make it possible to access them as a memory mapped I/O device with no wait state necessary in most microcontroller systems. All the ICM7228C/D inputs, excluding HEXA/CODE B/SHUTDOWN, feature a 250ns minimum setup and 0ns hold time with a 200ns minimum WRITE pulse. Input logic levels are TTL and CMOS compatible. Figure 10 shows a generic method of driving the ICM7228C/D from a microprocessor bus. To the microprocessor, the 8 bytes of the Display RAM appear to be 8 separate I/O locations. Loading the ICM7228C/D is quite similar to a standard memory write cycle. The address of the digit to be updated is placed on lines DA0 - DA2, the data to be written is placed on lines ID0 - ID3 and ID7, then a low pulse on WRITE input will transfer the data in. See Figure 3 and Switching Characteristics Table for write cycle timing parameters.

The ICM7228C/D devices do not have any control register, and also they do not provide the No Decode display format. Hexadecimal or Code B character selection and shutdown mode are directly controlled through the three level input at Pin 9, which is

accordingly called HEXA/CODE B/SHUTDOWN. See Tables 3 and 4 for input and output definitions of the ICM7228C/D devices.

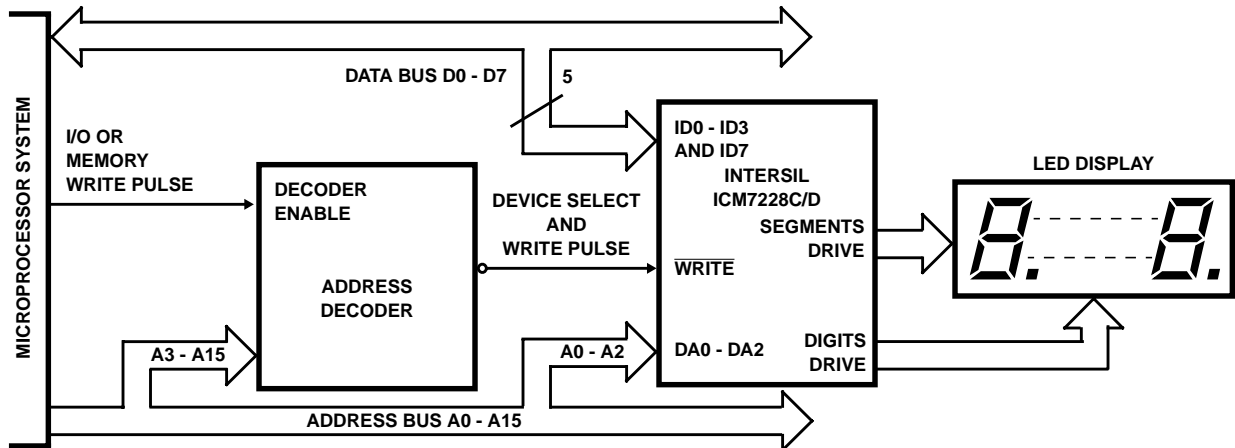
### Display Formats

The ICM7228A and ICM7228B have three possible display formats; Hexadecimal, Code B and No Decode. Table 6 shows the character sets for the decode modes and their corresponding input code.

The display formats of the ICM7228A/B are selected by writing data to bits ID4, ID5 and ID6 of the Control Register (See Table 1 and 2 for input Definitions). Hexadecimal and Code B data is entered via ID0-ID3 and ID7 controls the decimal point.

**TABLE 6. DISPLAY CHARACTER SETS**

INPUT DATA CODE			DISPLAY CHARACTERS		
ID3	ID2	ID1	ID0	HEXADECIMAL	CODE B
0	0	0	0	0	0
0	0	0	1	1	1
0	0	1	0	2	2
0	0	1	1	3	3
0	1	0	0	4	4
0	1	0	1	5	5
0	1	1	0	6	6
0	1	1	1	7	7
1	0	0	0	8	8
1	0	0	1	9	9
1	0	1	0	A	-
1	0	1	1	b	E
1	1	0	0	C	H
1	1	0	1	d	L
1	1	1	0	E	P
1	1	1	1	F	(Blank)



**FIGURE 10. ICM7228C/D MICROPROCESSOR SYSTEM INTERFACING**

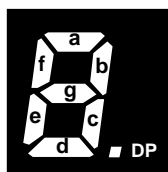


FIGURE 11. DIGITS SEGMENT ASSIGNMENTS

The No Decode mode of the ICM7228A and ICM7228B allows the direct segment-by-segment control of all 64 segments driven by the device. In the No Decode mode, the input data directly control the outputs as shown in Table 7.

TABLE 7. NO DECODE SEGMENT LOCATIONS

DATA INPUT	ID7	ID6	ID5	ID4	ID3	ID2	ID1	ID0
Controlled Segment	Decimal Point	a	b	c	e	g	f	d

An input high level turns on the respective segment, except for the decimal point, which is turned on by an input low level on ID7.

The No Decode mode can be used in different applications such as bar graph or status panel driving where each segment controls an individual LED.

The ICM7228C and ICM7228D have only the Hexadecimal and Code B character sets. The HEXA/CODE B/SHUTDOWN input, pin 9, requires a three level input. Pin 9 selects the Hexadecimal format when pulled high, the Code B format when floating or driven to mid-supply, and the shutdown mode when pulled low (See Tables 3 and 4). Table 6 also applies to the ICM7228C/D devices.

**Shutdown and Display Banking**

When shutdown, the ICM7228 enters a low power standby mode typically consuming only 1µA of supply current for the ICM7228A/B and 2.5µA for the ICM7228C/D. In this mode the ICM7228 turns off the multiplex scan oscillator as well as the digit and segment drivers. However, input data can still be entered when in the shutdown mode. Data is retained in memory even with the supply voltage as low as 2V.

The ICM7228A/B is shutdown by writing a control word with Shutdown (ID4) low. The ICM7228C/D is put into shutdown mode by driving pin 9, HEXA/CODE B/SHUTDOWN, low.

The ICM7228 operating current with the display blanked is within 100µA - 200µA for all versions. All versions of the ICM7228 can be blanked by writing Hex FF to all digits and selecting Code B format. The ICM7228A and ICM7228B can also be blanked by selecting No Decode mode and writing Hex 80 to all digits (See Tables 6 and 7).

**Common Anode Display Drivers, ICM7228A and ICM7228C**

The common anode digit and segment driver output schematics are shown in Figure 12. The common anode digit driver output impedance is approximately 4Ω. This provides a nearly constant voltage to the display digits. Each digit has a

minimum of 200mA drive capability. The N-Channel segment driver's output impedance of 50Ω limits the segment current to approximately 25mA peak current per segment. Both the segment and digit outputs can directly drive the display, current limiting resistors are not required.

Individual segment current is not significantly affected by whether other segments are on or off. This is because the segment driver output impedance is much higher than that of the digit driver. This feature is important in bar graph applications where each bar graph element should have the same brightness, independent of the number of elements being turned on.

**Common Cathode Display Drivers, ICM7228B and ICM7228D**

The common cathode digit and segment driver output schematics are shown in Figure 13. The N-channel digit drivers have an output impedance of approximately 15Ω. Each digit has a minimum of 50mA drive capability. The segment drivers have an output impedance of approximately 100Ω with typically 10mA peak current drive for each segment. The common cathode display driver output currents are only 1/4 of the common anode display driver currents. Therefore, the ICM7228A and ICM7228C common anode display drivers are recommended for those applications where high display brightness is desired. The ICM7228B and ICM7228D common cathode display drivers are suitable for driving bubble-lensed monolithic 7 segment displays. They can also drive individual LED displays up to 0.3 inches in height when high brightness is not required.

**Display Multiplexing**

Each digit of the ICM7228 is on for approximately 320µs, with a multiplexing frequency of approximately 390Hz. The ICM7228 display drivers provide interdigit blanking. This ensures that the segment information of the previous digit is gone and the information of the next digit is stable before the next digit is driven on. This is necessary to eliminate display ghosting (a faint display of data from previous digit superimposed on the next digit). The interdigit blanking time is 10µs typical with a guaranteed 2µs minimum. The ICM7228 turns off both the digit drivers and the segment drivers during the interdigit blanking period. The digit multiplexing sequence is: D2, D5, D1, D7, D8, D6, D4 and D3. A typical digit's drive pulses are shown on Figure 4.

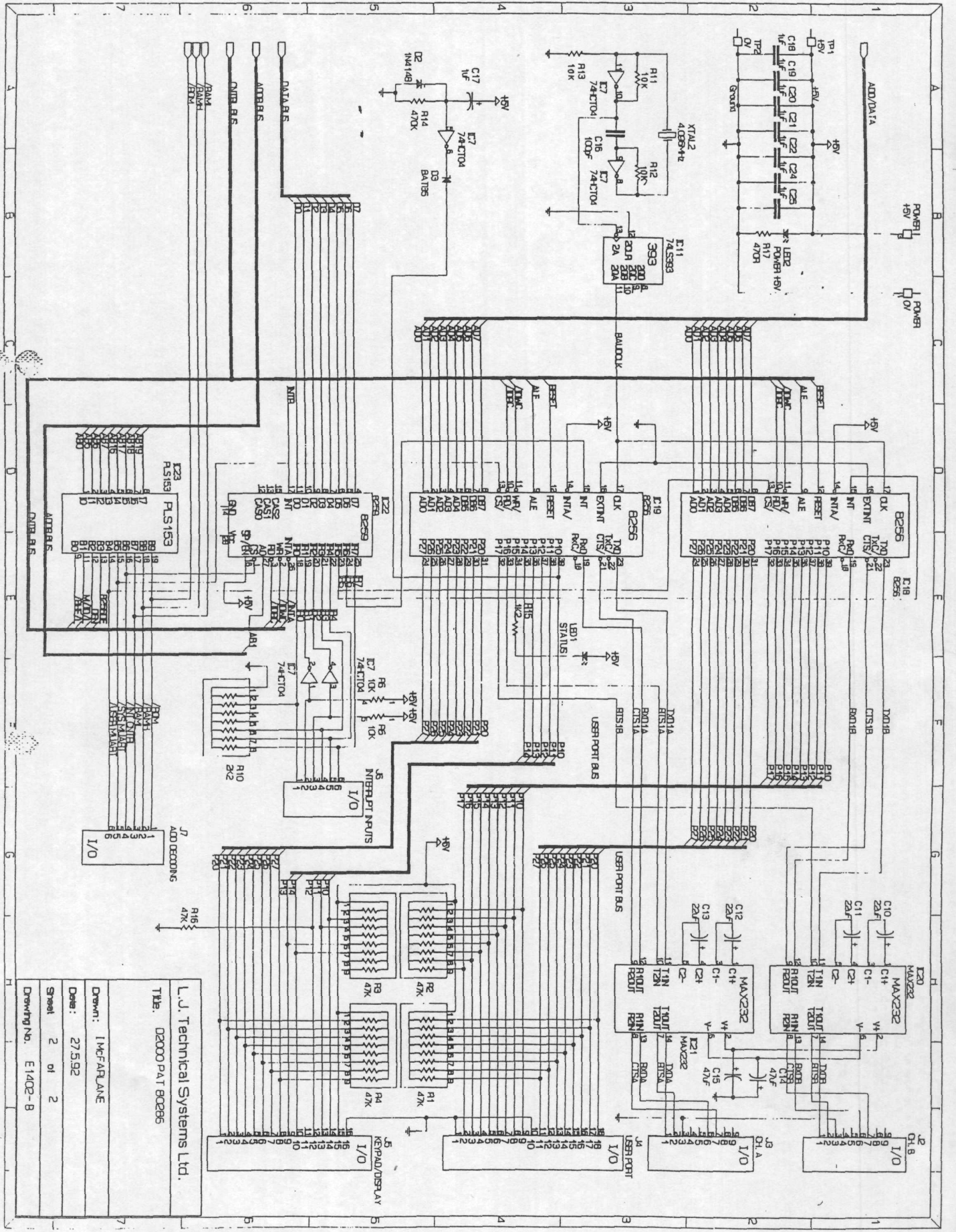
Due to the display multiplexing, the driving duty cycle for each digit is 12% (100 x 1/8) This means the average current for each segment is 1/8 of its peak current. This must be considered while designing and selecting the displays.

**Driving Larger Displays**

If very high display brightness is desired, the ICM7228 display driver outputs can be externally buffered. Figures 14 thru 16 show how to drive either common anode or common cathode displays using the ICM7228 and external driver circuit for higher current displays.

Another method of increasing display currents is to connect two digit outputs together and load the same data into both digits. This drives the display with the same peak current, but the average current doubles because each digit of the display is on for twice as long, i.e., 1/4 duty cycle versus 1/8.





L.J. Technical Systems Ltd.  
 T/Ws. 02000 PAT 80286

Drawn: IHC-FARLANE  
 Date: 27.5.92  
 Sheet 2 of 2  
 Drawing No. E1402-8

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57  
58  
59  
60  
61  
62  
63  
64  
65  
66  
67  
68  
69  
70  
71  
72  
73  
74  
75  
76  
77  
78  
79  
80  
81  
82  
83  
84  
85  
86  
87  
88  
89  
90  
91  
92  
93  
94  
95  
96  
97  
98  
99  
100

**EKLER :****EK 1. Eğitim Sistemi Hata Kodları:****No: Hata Mesajı ve Açıklaması**

---

01	Bilinmeyen komut.
02	Yazım hatası
03	Bilinmeyen yazmaç adı
04	Sınır dışına çıkan değer
05	Bilinmeyen cihaz adı
10	Daha fazla kırılma adresi tanımlanamıyor
11	Tanımlanmış kırılma adresi yok
12	Bellek sınırları aşılmış
20	Cihaz hazır değil
21	Okuma hatası
22	Yazma hatası
23	Kontrol toplamı (checksum) hatası
24	Bilinmeyen intel hex kayıt biçimi
25	Kayıt içinde bilinmeyen hex değer
26	Sıfıra bölme hatası
27	Örtülemez kesme (NMI)
28	Taşma ile kesme algılandı (INTO komutu)
29	BOUND komutu sınırı aşıldı (INT 5)
30	Geçersiz işlem kodu (OpCode)
31	İşlemci genişlemesi uygun değil (Nümerik işlemci komutu işlenmesi)
32	Parçanın aşılması
33	Koşullanmamış kesme vektörü
34	Tanımsız fonksiyon çağırılması

**EK 2. Eğitim Sistemi Alt programları :**

Eğitim sistemi, seri port kullanımı, matematik işlemler, klavyeden giriş ve göstergeye çıkış gibi alt programları kullanıcının faydalanabileceği şekilde sunmuştur.

No:	Giriş	Çıkış	Fonksiyon
00 <sub>H</sub>	AH=00 <sub>H</sub> (fonksiyon no) BL=cihaz no CX=okunacak verinin en fazla sayısı ES:DI=giriş tampon belleğin adresi	CX=okunan güncel bayt sayısı	Numarası belirtilen cihazdan belirtilen sayıda veri okur.
01 <sub>H</sub>	AH=01 <sub>H</sub> (fonksiyon no) BL=cihaz no CX=okunacak verinin en fazla sayısı ES:DI=giriş tampon belleğin adresi	CX=okunan güncel bayt sayısı	Bir önceki fonksiyona benzer, farkı CR (0D <sub>H</sub> ) gelene kadar beklemesidir.
02 <sub>H</sub>	AH=02 <sub>H</sub> (fonksiyon no) BL=cihaz no CX=yazılacak verinin sayısı ES:DI=tampon belleğin adresi	CX=yazılan güncel bayt sayısı	Numarası belirtilen cihazdan CX sayıda veri çıkışı yapılır.
03 <sub>H</sub>	AH=03 <sub>H</sub> (fonksiyon no) BL=cihaz no CX=yazılacak verinin sayısı ES:DI=tampon belleğin adresi	CX=yazılan güncel bayt sayısı	Numarası belirtilen cihazdan CR görene kadar veya CX sayıda veri çıkışı yapılır.
04 <sub>H</sub>	AH=04 <sub>H</sub> (fonksiyon no) BX=çağrılan programdan dönen hata kodu		programdan çıkar ve denetimi eğitim sistemine geri verir.
05 <sub>H</sub>	AH=05 <sub>H</sub> (fonksiyon no) BX=hata kodu		hata mesajını yazar.
06 <sub>H</sub>	AH=06 <sub>H</sub> (fonksiyon no) ES:DI=dizinin adresi	AX=hex değer ES:DI=bir sonraki adresi gösterir.	ES:DI adresindeki ASCII hex değeri hex değere dönüştürür.
07 <sub>H</sub>	AH=07 <sub>H</sub> (fonksiyon no) ES:DI=dizinin adresi	AX=hex değer ES:DI=bir sonraki adresi gösterir.	ES:DI adresindeki ASCII hex değeri ondalık değere dönüştürür.
08 <sub>H</sub>	AH=08 <sub>H</sub> (fonksiyon no) BX=dönüştürülecek değer CX=çıkıştaki hane sayısı ES:DI=tampon belleğin adresi	sonuç tampon bellekte, ES:DI=bir sonraki adresi gösterir.	BX' deki hex değeri ES:DI adresindeki ASCII hex değere dönüştürür.
09 <sub>H</sub>	AH=09 <sub>H</sub> (fonksiyon no) BX=dönüştürülecek değer CX=çıkıştaki hane sayısı ES:DI=tampon belleğin adresi	sonuç tampon bellekte, ES:DI=bir sonraki adresi gösterir.	BX' deki hex değeri ES:DI adresindeki ASCII ondalık değere dönüştürür.
0A <sub>H</sub>	AH=0A <sub>H</sub> (fonksiyon no)	AL=okunan karakterin ASCII kodu	giriş cihazından girilen verinin karakterini okur.
0B <sub>H</sub>	AH=0B <sub>H</sub> (fonksiyon no)	AL=okunan karakterin ASCII hex değeri	giriş cihazından girilen verinin karakterini okur.
0C <sub>H</sub>	AH=0C <sub>H</sub> (fonksiyon no) AL=yazılacak karakterin ASCII kodu		giriş cihazına verinin karakterini yazar.

No:	Giriş	Çıkış	Fonksiyon
0D <sub>H</sub>	AH=0D <sub>H</sub> (fonksiyon no) AL=yazılacak hex değer		giriş cihazına verinin hex değerini yazar.
0E <sub>H</sub>	AH=0E <sub>H</sub> (fonksiyon no)	AL=karakter kodu, tuşa basılmadıysa 255	tuş takımından girilen tuşun karakterini okur.
0F <sub>H</sub>	AH=0F <sub>H</sub> (fonksiyon no)		1 ms gecikme sağlar.
10 <sub>H</sub>	AH=10 <sub>H</sub> (fonksiyon no) BX=ms olarak bekleme süresi		BX' in değeri kadar ms olarak gecikme sağlar.
11 <sub>H</sub>	AH=11 <sub>H</sub> (fonksiyon no)		CR (0D <sub>H</sub> ) , LF (0A <sub>H</sub> ) değer çıkışını sağlar.
12 <sub>H</sub>	AH=12 <sub>H</sub> (fonksiyon no)		standart giriş cihazının göstergesini temizler.