

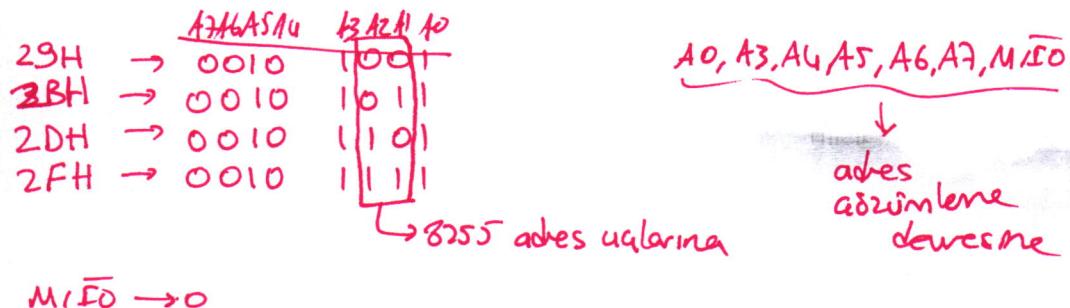
İsim: No:	2015/2 Mikroişlemci Sistemleri Vize 1 - 6 Nisan 2016 Süre: 90 dk	Soru 1 (30p)	Soru 2 (40p)	Soru 3 (30p)	Toplam (100p)
--------------	--	-----------------	-----------------	-----------------	------------------

Soru 1) Aşağıdaki boşlukları uygun şekilde doldurun (30p)

- a) 8086 veri yolu genişliği 16 bittir. (1p)
- b) 8086 adres yolu genişliği 20 bittir. (1p)
- c) 8086 için uygun hafıza arayüzü 512K × 16 kapasitesindedir. (2p)
- d) 8086 I/O uzayı 64K byte boyutundadır. (2p)
- e) 8086 hafıza uzayı 1M byte boyutundadır. (2p)
- f) M/I/O ucu sadece izole I/O haritalamada adres çözümleme için kullanılır. (1p)
- g) 8088 için uygun hafıza arayüzü $1M \times 8$ olduğuna göre, 8088 adres yolu genişliği 20 bittir. (1p)
- h) 80486 için uygun hafıza arayüzü $1G \times 32$ olduğuna göre, 80486 adres yolu genişliği 32 bittir. (2p)
- i) 8086'da AD_i uçlarından adres bilgisi ayırtırılırken ALE ucu kullanılır. (1p)
- j) 8086'da AD_i uçlarından veri bilgisi ayırtırılırken DEN ve DT/R uçları kullanılır. (2p)
- k) 8086, veri ve program hafızalarının aynı veri yolunu kullanması açısından von Neumann mimarisindedir. (2p)
- l) 8086, anlamlı byte'ın hafızada saklanma yeri açısından little endian mimarisindedir. (1p)
- m) 8086, komut seti yapısı itibarıyle CISC mimarisindedir. (2p)
- n) Adres çözümlenmesi belirli bir zamanda sadece bir çevre birimin aktif olmasını sağlayarak, farklı entegrelerin aynı anda, aynı yola (bus) farklı veri yazmasını engeller. (2p)
- o) Mikroişlemcide aritmetik, lojik işlem sonuçlarını saklayan yazmaca akumulator denir. (1p)
- p) 8086'da adres ve veri bilgisinin aynı uçlarda farklı zamanlarda bulunması time multiplexing olarak adlandırılır. (2p)
- q) 8255 mod 1'de input yönlü ayarlamışsa port uçlarındaki verinin 8255'e otomatik olarak alınmasını sağlayan STB anlaşma (handshaking) ucudur. (1p)
- r) YTU amblemindeki yıldız 10 uçladur. (2p)
- s) 8 adet, aynı hacim ve şekildeki altın paradan biri hilelidir (ağırlığı daha az). Hileli parayı kesin olarak bulabilmek için iki kollu terazi ile en az 2 ölçüm yeterlidir. (2p)

Soru 2) Tek müşteri kuyruğu ve iki gişesi olan bir bankada, müşterilerin sağ veya sol gitmeye yönlendirilmesinde, görevlinin sağ veya sol isimli tuşlara basması ile bir göstergede sağa veya sola dönük ok işaretinin oluşturulması istenmektedir. Bu amaçla minimum modda çalışan 8086'lı bir sistemde 8255 yardımıyla 1 adet 8×8 nokta-matris-gösterge (dot-matrix-display) ve 2 adet buton sürelecektir. Not1: İzole I/O haritalama kullanın. Not2: Mikroişlemcinin adres, veri ve kontrol uçlarının uygun şekilde tutulmuş ve tamponlanmış olduğunu varsayıñ. (40p)

- a) 8255'in **29H** adresinden itibaren ardışık tek adreslere yerleştirilebilmesi için gerekli adres çözümleme devresini 1 adet 3×8 dekoder ve gerekli lojik kapılar kullanarak gerçekleyin. Not: Sorunun altındaki boşluğu ve 6. sayfayı çözüm için kullanabilirsiniz. (5p)



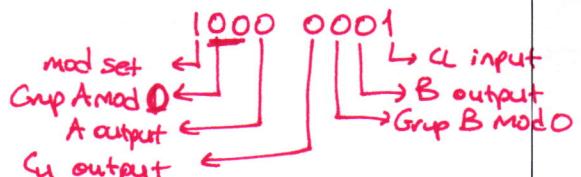
Adres çözümlene montajı 6. sayfada arzılmıştır.

- b) **8x8 nokta-matris-gösterge**, 8 satır 8 sütuna yerleştirilmiş 64 adet LED'den oluşan bir yapı olup, aynı sütundaki LED'ler ortak anot ucuna ve aynı satırındaki LED'ler ise ortak katot ucuna sahiptir. 2 adet buton ise pull-up dirençlerle sürülmektedir. 8x8 nokta-matris-göstergenin ortak anot ucları 8255 PORTA'ya, 8x8 nokta-matris-göstergenin ortak katot ucları 8255 PORTB'ye ve 2 adet buton ise sırası ile PORTC0 (sol tuş) ve PORTC1 (sağ tuş) bağlanmıştır. 8255'i Mod 0'da bahsedilen bağlantılara göre uygun şekilde koşullayan CONTROL WORD değerini bulun ve bu değerin 8255'e yazılabilmesi için gerekli assembly kodunu yazın. (5p)

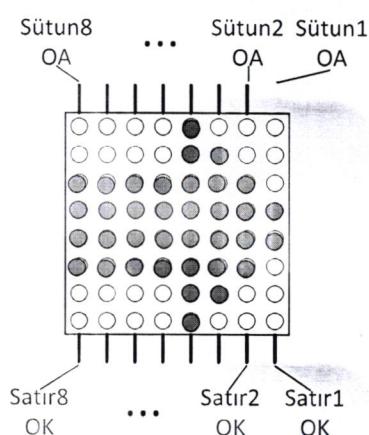
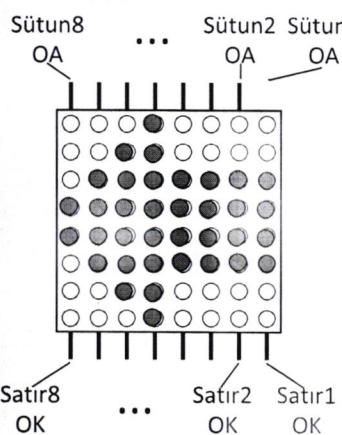


PortA ile OA uclarinda sütun secatlı yapılacak \Rightarrow output
PortB ile OK uclarından yakılacak istenen deger aklılarak \Leftrightarrow output
PortCL ile düğmeler okunacak \Rightarrow Input
8255 mod 0 kullanılacak.

MOV AL, 1000 0001B
OUT 2FH, AL



- c) 8x8 nokta-matris-göstergede bir değer oluşturulmak istendiğinde öncelikle her seferinde sadece 1 sütun aktif olacak şekilde sırasıyla tüm sütunlar taranmalıdır. Herhangi bir sütun aktif iken o sütunda gösterilmek istenen değer ortak katot uçlarından gönderilir. Buna göre 8x8 nokta-matris-göstergede sola bakan ok oluşturmak için verilmesi gereken değerleri bit bazında ve bu 8 bite karşılık gelen HEX değeri aşağıdaki tabloda doldurun. (5p)



Aktif Sütun	HEX Karşılık	OK ucu Satır 8	OK ucu Satır 7	OK ucu Satır 6	OK ucu Satır 5	OK ucu Satır 4	OK ucu Satır 3	OK ucu Satır 2	OK ucu Satır 1
Sütun1	C3H	1	1	0	0	0	0	1	1
Sütun2	C3H	1	1	0	0	0	0	1	1
Sütun3	C3H	1	1	0	0	0	0	1	1
Sütun4	C3H	1	1	0	0	0	0	1	1
Sütun5	00H	0	0	0	0	0	0	0	0
Sütun6	81H	1	0	0	0	0	0	0	1
Sütun7	C31H	1	1	0	0	0	0	1	1
Sütun8	E7H	1	1	1	0	0	1	1	1

- d) 8255'in uçlarına, 8086, adres çözümleme devresi, 8x8 nokta-matris-gösterge, 2 adet buttondan gelen bağlantıları çizerek gösterin. Not: 6 nolu sayfada çizin. (5p)
e) Önceki şıkta yaptığınız bağlantıları göz önüne alarak, Sağ ve Sol butonlar sadece bırakıldığı anda göstergede ilgili ok işaretleri yanmasını sağlayan assembly kodunu yazın. (20p)

e)

```
STAK SEGMENT PARA STACK 'STACK'  
DW 20 DUP(?)  
STAK ENDS
```

```
DATA SEGMENT PARA 'DATA'  
SOL_OK DB 0C3H, 0C3H, 0C3H, 0C3H, 00H, 81H, 0C3H, 0E7H  
SAG_OK DB 0E7H, 0C3H, 81H, 00H, 0C3H, 0C3H, 0C3H, 0C3H  
BOS DB OFFH, OFFH, OFFH, OFFH, OFFH, OFFH, OFFH, OFFH  
NUM_OF_COLS DB 8  
DATA ENDS
```

```
CODE SEGMENT PARA 'CODE'  
ASSUME CS:CODE, DS:DATA, SS:STAK  
BASLA:  
MOV AX, DATA  
MOV DS, AX
```

```
MOV AL, 81H  
OUT 2FH, AL  
LEA BX, BOS
```

```
ENDLESS:  
CALL OK_YAK  
IN AL, 2DH  
TEST AL, 01H  
JNZ SAG_KOTROL  
SOL_BASIK:  
IN AL, 2DH  
TEST AL, 01H  
JZ SOL_BASIK  
LEA BX, SOL_OK
```

```
SAG_KOTROL:  
IN AL, 2DH  
TEST AL, 02H  
JNZ TUS_YOK
```

```
SAG_BASIK:  
IN AL, 2DH  
TEST AL, 02H  
JZ SAG_BASIK  
LEA BX, SAG_OK
```

```
TUS_YOK:  
JMP ENDLESS
```

```
OK_YAK PROC NEAR  
MOV CX, 8  
XOR SI, SI  
MOV AL, 01H  
TEKRAR:  
PUSH AX  
MOV AL, OFFH  
OUT 2BH, AL  
POP AX  
OUT 29H, AL  
PUSH AX  
MOV AL, [BX + SI]  
OUT 2BH, AL  
INC SI  
POP AX  
SHL AL, 1  
LOOP TEKRAR  
RET  
OK_YAK ENDP
```

```
CODE ENDS  
END BASLA
```

Soru 3) Minimum modda çalışan 8086 mikroişlemci sisteminin bellek uzayına E8000H adresinden FFFFFH adresine kadar olan alan 16Kx8 kapasiteli SRAM'lerden oluşan RAM bloğu ve F8000H adresinden FFFFFFFH adresine kadar olan alana ise 16Kx8 kapasiteli EPROM'lardan oluşan bir ROM bloğu yerleştirilmek isteniyor. Not1: İzole I/O haritalama kullanın. Not2: Mikroişlemcinin adres, veri ve kontrol uçlarının uygun şekilde tutulmuş ve tamponlanmış olduğunu varsayıın. (30p)

- RAM ve ROM bloklarının kapasitelerini belirtin. (5p)
- Kullanılan SRAM ve EPROM'ların adres ucu sayılarını belirtin. (5p)
- İsteneden RAM ve ROM bloklarını oluşturmak için kaçar adet SRAM ve EPROM kullanılması gerektiğini bulun. (5p)
- RAM ve ROM bloklarının bellek uzayında yerleşimini sağlayan adres çözümleme devresini tek bir 3x8 dekoder ve gerekli lojik kapılar kullanarak gerçekleyin ve mikroişlemciden hafıza birimlerine giden tüm uç bağlantılarını çizerek gösterin. (15p)

Not: SRAM entegresinde bulunan uçlar: D7-D0 (Veri yolu), Adres yolu, \overline{CS} , \overline{OE} , \overline{WE}
EPROM entegresinde bulunan uçlar: D7-D0 (Veri yolu), Adres yolu, \overline{CS} , \overline{OE}

a) ROM bloğu : $F8000H \rightarrow 1111\ 1000\ 0000\ 0000\ 0000$
 $FFFFFH \rightarrow 1111\ 1111\ 1111\ 1111\ 1111$

\hookrightarrow Kapasite $\Rightarrow 2^{15} = 32\text{ K byte}$

15 ua
değişiyor

RAM Bloğu : $E8000H \rightarrow 1110\ 1000\ 0000\ 0000\ 0000$
 $F7FFFH \rightarrow 1111\ 0111\ 1111\ 1111\ 1111$

17 ua değişiyor $\Leftarrow \Rightarrow$ YANLIŞ

$E8000H \rightarrow 1110\ 1000\ 0000\ 0000\ 0000$
 $EFFFFFH \rightarrow 1110\ 1111\ 1111\ 1111\ 1111$

15 ua değişiyor

$F0000H \rightarrow 1111\ 0000\ 0000\ 0000\ 0000$
 $F7FFFH \rightarrow 1111\ 0111\ 1111\ 1111\ 1111$

15 ua değişiyor

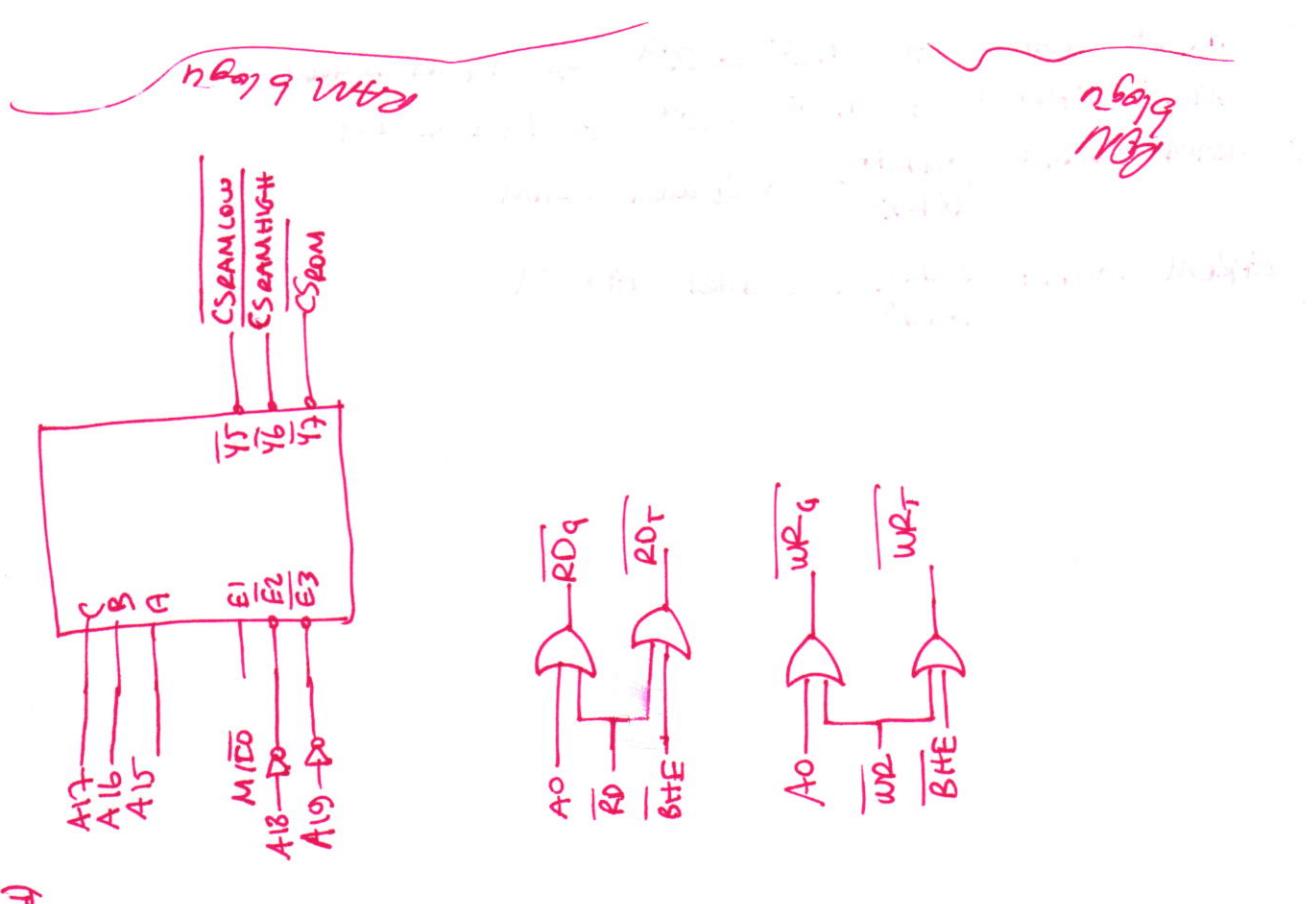
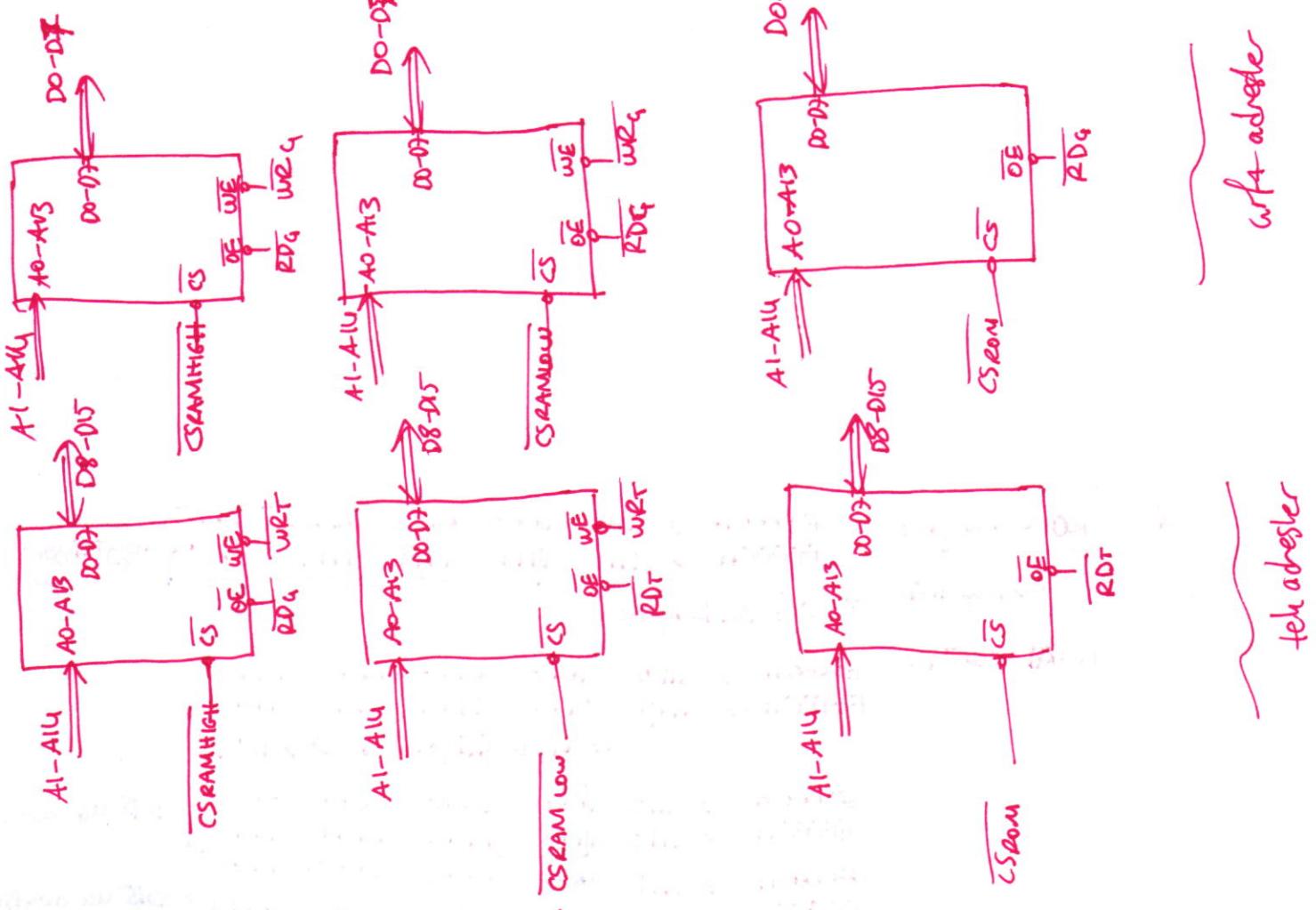
\rightarrow Kapasite = $2 \cdot 2^{15} = 64\text{ K byte}$

b) $16K \times 8\text{ SRAM} \Rightarrow 2^4 \cdot 2^{10} = 2^{14} \Rightarrow 16\text{ adres ucu}$

$16K \times 8\text{ EPROM} \Rightarrow 2^4 \cdot 2^{10} = 2^{14} \Rightarrow 16\text{ adres ucu}$

c) SRAM sayısı: $\frac{64\text{ KB}}{16\text{ Kx8}} = 4\text{ adet SRAM}$

EPROM sayısı: $\frac{32\text{ KB}}{16\text{ Kx8}} = 2\text{ adet EPROM}$



İsim:
No:

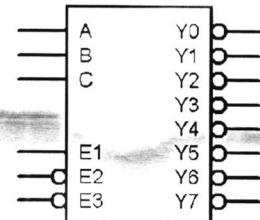
2015/2 Mikroişlemci Sistemleri

Vize 1 - 6 Nisan 2016

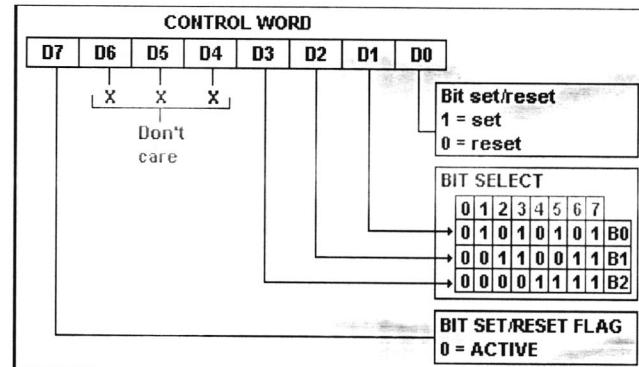
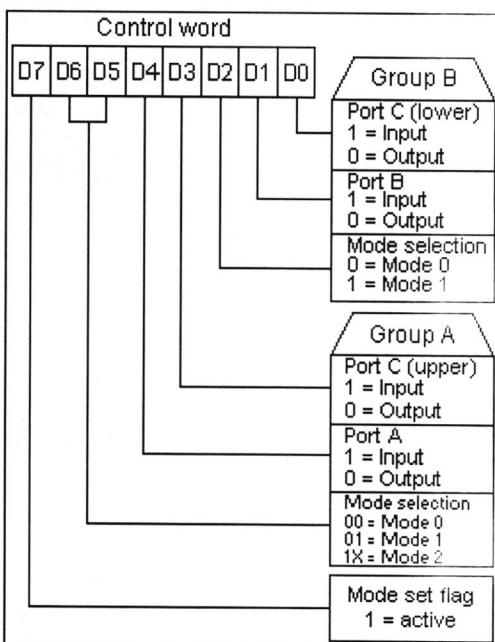
Süre: 90 dk

74138 3x8 Dekoder Fonksiyon Tablosu													
INPUTS			OUTPUTS						SELECTED OUTPUT				
ENABLE		SELECT	Y0	Y1	Y2	Y3	Y4	Y5	Y6	Y7			
E1	E2	E3	C	B	A	Y0	Y1	Y2	Y3	Y4	Y5	Y6	Y7
L	X	X	X	X	X	H	H	H	H	H	H	H	H
X	X	H	X	X	X	H	H	H	H	H	H	H	H
X	H	X	X	X	X	H	H	H	H	H	H	H	H
H	L	L	L	L	L	L	H	H	H	H	H	H	H
H	L	L	L	L	H	H	L	H	H	H	H	H	Y0
H	L	L	L	H	L	H	H	L	H	H	H	H	Y1
H	L	L	L	H	L	H	H	H	H	H	H	H	Y2
H	L	L	L	H	H	H	H	L	H	H	H	H	Y3
H	L	L	H	L	L	H	H	H	H	L	H	H	Y4
H	L	L	H	L	H	H	H	H	H	H	L	H	Y5
H	L	L	H	H	L	H	H	H	H	H	H	L	Y6
H	L	L	H	H	H	H	H	H	H	H	H	L	Y7

X : Don't Care, L : Low, H : High



8255 Komut Yazmacı



8255 Port Seçimi

A ₁	A ₀	Port selected
0	0	port A
0	1	port B
1	0	port C
1	1	control register

6

