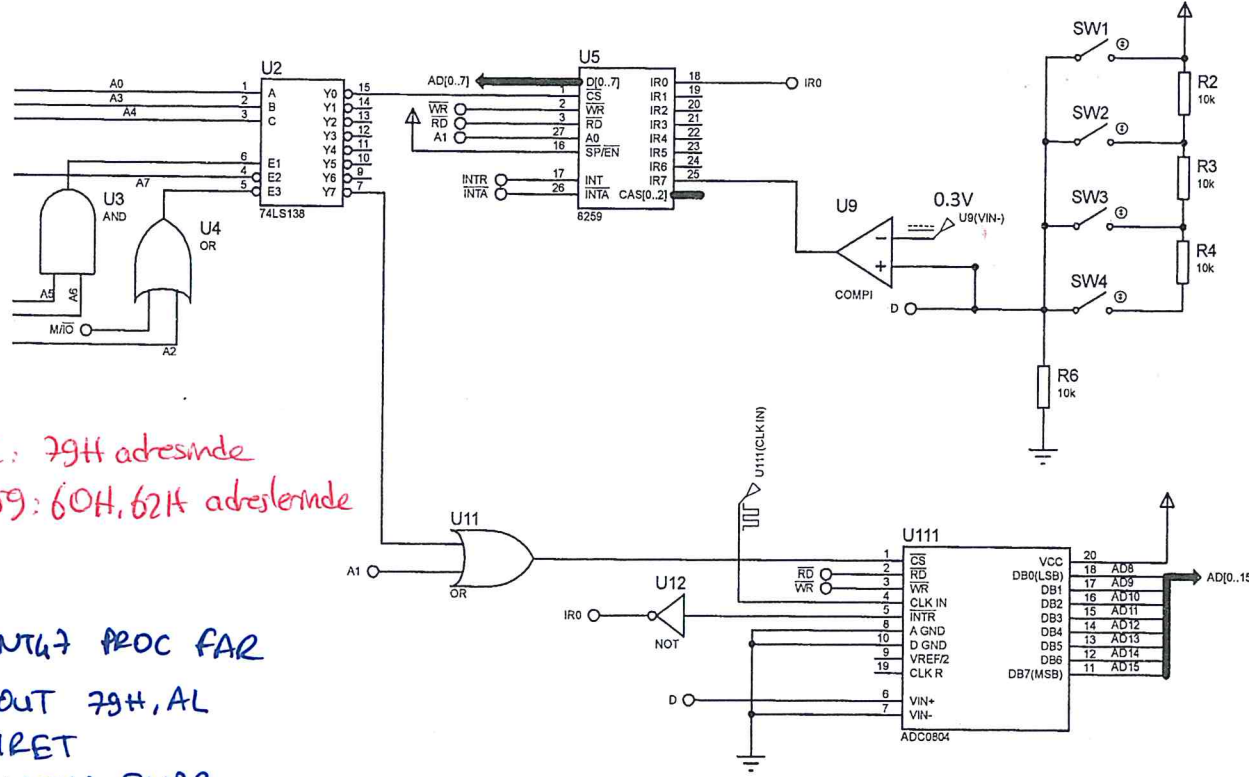


Soru 6) ADC0804, 8259A ve bir karşılaştırıcı (comparator) kullanılarak kesme tabanlı bir tuş tarama sistemi oluşturulmak istenmektedir. 4 adet buton bir direnç dizisi yardımıyla şekildeki gibi bağlanmıştır. SW1, SW2, SW3, SW4 tuşlarından herhangi birine basıldığında D isimli düğümde farklı analog gerilim değerleri elde edilmektedir. Karşılaştırıcının + ucuna D düğümü, - ucuna ise 0.3V bağlanmıştır. D düğümündeki gerilim 0.3V'tan büyük olduğunda karşılaştırıcı çıkışı lojik 1 (5V), diğer durumlarda 0V olmaktadır. ADC'nin analog girişine D düğümü bağlıdır. Karşılaştırıcı çıkışı 8259A'da IR7 ucuna bağlıdır. ADC'nin *INTR* ucu ise 8259'da IR0 ucuna bağlıdır. Verilenlere ilişkin kesme tabanlı tuş tarama mantığı assembly kodunu aşağıdaki adımları dikkate alarak yazın. (35p)

X=5V	Gerilim Değeri	0-255 arası karşılığı
X	5.00	255
X/2	2.50	128
X/3	1.67	85
X/4	1.25	64

- a) ADC0804 ve 8259A'nın yerleştirilmiş olduğu adresleri bulun. (5p)  
b) Herhangi bir butona basıldığında 47H tipindeki kesme servisi programının çağırarak ayarlamayı yapın. Bu kesme servisi programında ADC dönüşümü başlatın. (10p)  
c) ADC dönüşümü bitince 40H tipindeki kesmenin çağırılmasını sağlayan ayarları yapın. 40H tipindeki kesme servisi programında hangi butona basıldığını BL yazmacında saklayın. (10p)  
d) 8259A ayarlarını yapın. (10p)



a) ADC: 79H adresinde  
8259: 60H, 62H adreslerinde

b,c,d)

NEWINT47 PROC FAR

OUT 79H,AL  
IRET

NEWINT47 ENDP

NEWINT40 PROC FAR

IN AL,79H

MOV BL,1

CMP AL,0BFH

JA CİKİS

MOV BL,2

CMP AL,6AH

JA CİKİS

MOV BL,3

CMP AL,6AH

JA CİKİS

MOV BL,4

CİKİS:

IRET

NEWINT40 ENDP

START PROC FAR

XOR AX,AX

MOV ES,AX

LEA AX,NEWINT47

MOV WORD PTR ES:[11CH],AX

MOV WORD PTR ES:[11EH],CS

LEA AX,NEWINT40

MOV WORD PTR ES:[100H],AX

MOV WORD PTR ES:[102H],CS

MOV AL,13H

OUT 60H,AL

MOV AL,60H

OUT 62H,AL

MOV AL,03H

OUT 62H,AL

STI

ENDLESS: JMP ENDLESS

RET

START ENDP

İsim :	2017/1 BLM3621	S1	S2	S3	S4 veya S5	S6	Σ
No :	Mikroişlemci Sistemleri	(10p)	(20p)	(30p)	(25p)	(35p)	(120p)
İmza :	Final - 8 Ocak 2018						
	Süre: 90 dk						

NOT1 : Tüm sorular için izole hafıza haritalama (isolated mapping) kullanın.

NOT2 : Tüm  $\mu P$  uçlarının uygun şekilde ayrıştırılmış ve tutulmuş olduğunu varsayın.

NOT3: Seçmeli sorularda (1,2,4 ve 5) istenenden fazla şık cevaplanmışsa, öncelikle hatalı çözümler değerlendirilecektir.

Soru 1) Aşağıdaki ifadelerden sadece 5 tanesini seçerek Doğru-Yanlış olduklarını işaretleyin. (5x2p=10p)

- a) Kesme servisi programları FAR tipinde tanımlanmak zorundadır. (D) (Y)  
b) Kesme servisi programı dönüşünde RET komutu kullanılabilir. (D) (Y)  
c) NMI ucundan gelen kesme istekleri IF = 0 yapılarak maskelenebilir. (D) (Y)  
d) Kesme vektör tablosu hafıza uzayında 1KB yer kaplar. (D) (Y)  
e) MOV AX, 0F0F0H komutlarının işletilmesi sonucunda INTO oluşur. (D) (Y)  
f) 8255 arayüzünde bir port ile LED sürülmek isteniyorsa, port giriş yönlü ayarlanır. (D) (Y)  
g) 8086 CISC mimarisinde, von Neumann hafıza düzeninde bir mikro işlemcidir. (D) (Y)  
h) 8254 arayüzü zamanlayıcı olarak kullanılabilir fakat sayıcı olarak kullanılamaz. (D) (Y)  
i) 8086 için uygun ve doğru hafıza yapılandırması 512Kx16 şeklindedir. (D) (Y)

Soru 2) Aşağıdaki sorularda boşlukları uygun şekilde doldurun. Sadece 5 tanesini seçerek cevaplayın. (5x4p=20p)

- a) 8086'da adres değerlerini ayrıştırıp tutmak için ALE fiziki ucu kullanılır.  
b) Kesme vektör tablosu hafıza uzayının 0H adresinden başlar.  
c) 8086 I/O uzayı 64K byte boyutundadır.  
d) 8086'da adres ve veri bilgisinin aynı uçlarda farklı zamanlarda bulunması zaman çakışması olarak adlandırılır.  
e) Kesme vektör tablosunda, ISR adresi (CS ve IP olarak) bilgisi tutulur.  
f)  $V_{REF} = 5V$  olan 8 bitlik ADC'nin gerilim çözünürlüğü (ölçebileceği en küçük gerilim farkı) 5/256 değerindedir.  
g) 8251 arayüzü I/O uzayında 2 byte yer kaplar.  
h) Mikroişlemci, hafıza, kontrol ve I/O birimlerini tümleşik olarak içeren yapıya MC denir.  
i) DAC0830 arayüzü hafıza uzayında 1 byte yer kaplar.  
j) 8086, anlamlı byte'ın hafızada saklanma yeri açısından little endian mimarisindedir.  
k) 32 bit veri yoluna ve 32 bit adres yoluna sahip 80486 için uygun hafıza yapılandırması 1G x 32 şeklindedir.



l) : Meltdown & Spectre



**Soru 3)** Minimum modda çalışan 8086  $\mu$ P sisteminin bellek uzayına E8000H adresinden F7FFFH adresine kadar olan alana 4 adet SRAM'den oluşan bir RAM bloğu ve F8000H adresinden FFFFFH adresine kadar olan alana ise 2 adet EPROM'dan oluşan bir ROM bloğu yerleştirilmek isteniyor. (30p)

- RAM ve ROM bloklarının kapasitelerini belirtin. (5p)
- Kullanılan SRAM ve EPROM'ların adres ucu sayılarını belirtin. (5p)
- Kullanılan SRAM ve EPROM'ların kapasitesini belirtin. (5p)
- RAM ve ROM bloklarının bellek uzayında yerleşimini sağlayan adres çözümleme devresini tek bir 3x8 dekode ve gerekli lojik kapılar kullanarak gerçekleyin. (10p)
- Bir SRAM ve bir EPROM'a mikroislemciden ve adres çözümleme devresinden giden tüm uç bağlantılarını çizerek gösterin. (5p)

a)  $F8000H \leftarrow (F7FFFH+1)$   
 $E8000H$   
 $10000H \Rightarrow 2^{16} \Rightarrow 64K$  hafıza görü  
RAM: **64KB** her hafıza görü 8bit olduğundan

$100000H \leftarrow (FFFFFH+1)$   
 $F8000H$   
 $008000H \Rightarrow 2^{15} \Rightarrow 32K$  hafıza görü

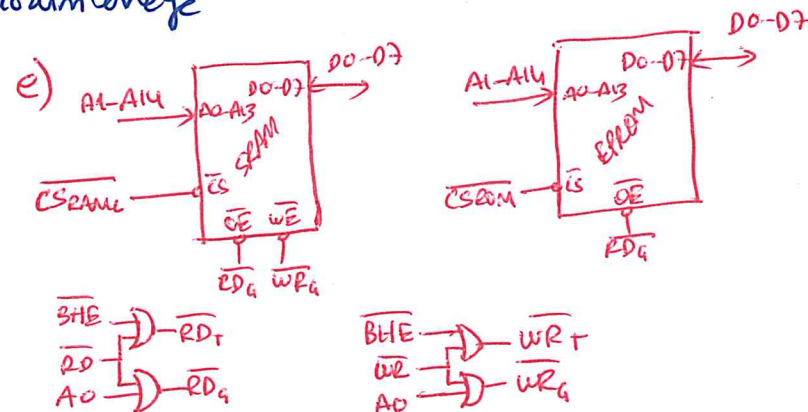
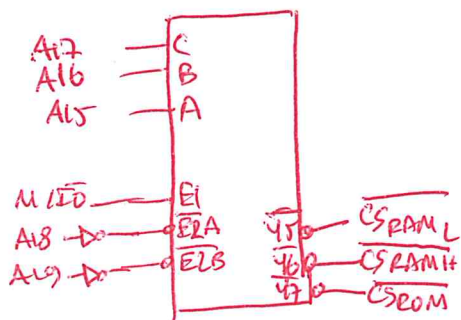
ROM: **32KB** her hafıza görü 8bit olduğundan

c)  $\frac{64K \times 8}{4} = 16K \times 8$  SRAM kapasitesi  
 $\frac{32K \times 8}{2} = 16K \times 8$  EPROM kapasitesi

b)  $16K \times 8 \rightarrow 8$  veri ucu hem EPROM hemde SRAM için  
 $\rightarrow 16$  adres ucu

d) RAM: E8000H : 1110 1000 0000 0000 0000  
 EFFFFH : 1110 1111 1111 1111 1111  
 F0000H : 1111 0000 0000 0000 0000  
 F7FFFH : 1111 0111 1111 1111 1111  
 ROM: F8000H : 1111 1000 0000 0000 0000  
 FFFFFH : 1111 1111 1111 1111 1111

→ çift tek adresleri ayırmak için  
 → SRAM ve EPROM adres uçlarına  
 → adres çözümlemeye

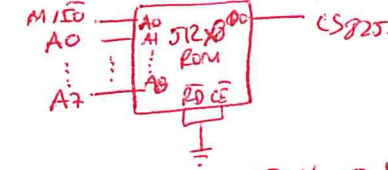


**Soru 4)** A8H adresinden itibaren ardışık çift adreslere yerleştirilmiş bir 8255 yardımıyla, 15 adet kapıdan kaçının açık olduğunun bir 7 parçalı gösterge ile hex. değer olarak yazdırılması istenmektedir (0:tüm kapılar kapalı, ..., F:tüm kapılar açık). Bunun için ortak katotlu bir 7 parçalı gösterge 8255 PORTA'ya uygun şekilde bağlanmıştır. 15 adet kapıya bağlı binary sensörlerin (kapı açık ise lojik 1, kapı kapalı ise lojik 0 gönderiyor) çıkışları ise boş olan PORTB7-B0 ve PORTC6-C0 uçlarına bire bir bağlanmıştır. İsteneni gerçeklemek için: (25p)

- 8255 için uygun adres çözümleme devresini 512 byte (512x8) kapasiteli bir ROM kullanarak gerçekleyin. (ROM içeriğinin ne olması gerektiğini belirtin) (10p)
- 8255'yi, 7 parçalı gösterge ve sensörleri sürüp/okuyabilecek uygun şekilde ayarlan assembly komutlarını yazın. (7p)
- Veri kesiminde tanımlı DIGIT2SEGMENT isimli dizide sırasıyla her hex basamak değerine karşılık 7 parçalı gösterge değerlerinin tutulduğunu varsayarak, 15 kapıdan kaçının açık olduğunu 7 parçalı göstergede yakan assembly kodunu yazın. (8p)

SORU4 ve SORU5'ten sadece 1 tanesini seçerek cevaplayın.

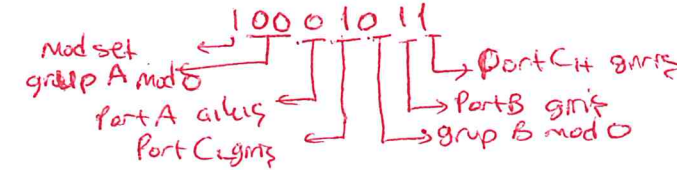
a) A8H, AAH, ACH, AEH: 8255 adresleri



ROM içeriği: 150H, 150H, 158H, 15CH  
 adreslerinde FEH  
 diğer adreslerde FFH mevcut

b) PortA çıkış, Port B ve PortC giriş

MOV AL, 10001011B  
 MOV 0AEH, AL



c) TEKRAR: XOR BX, BX  
 IN AL, 0AAH

MOV CL, 8  
 L1: RCR AL, 1  
 ADC BX, 0  
 LOOP L1  
 IN AL, 0ACH  
 MOV CL, 7  
 L2: RCR AL, 1  
 ADC BX, 0  
 LOOP L2  
 MOV AL, DIGIT2SEGMENT[BX]  
 OUT 0A8H, AL  
 JMP TEKRAR

**Soru 5)** Aşağıdaki çıkılarda istenen işlemleri gerçekleyen assembly komutlarını yazın. (25p)

- 8086'da NMI ucuna, basıldığında lojik 1, bırakıldığında lojik 0 değeri üreten bir buton bağlanmıştır. Butona kaç kere basıldığını belirleyebilmek için gerekli kesme alt programını ve vektör tablosunu uygun şekilde değiştiren kod parçasını yazınız. (15p)
- A9H adresinden itibaren ardışık tek adreslere yerleştirilmiş 8255 için, GRUPB'yi mod 1'de giriş yönlü, GRUPA'yı ise mod1'de çıkış yönlü ayarlayan assembly kodunu yazın. (10p)

SORU4 ve SORU5'ten sadece 1 tanesini seçerek cevaplayın.

a) NEWINTNMI PROC FAR

INC AX

IRET

NEWINTNMI ENDP

START PROC FAR

XOR AX, AX

MOV ES, AX

LEA AX, NEWINTNMI

MOV WORD PTR ES:[8H], AX

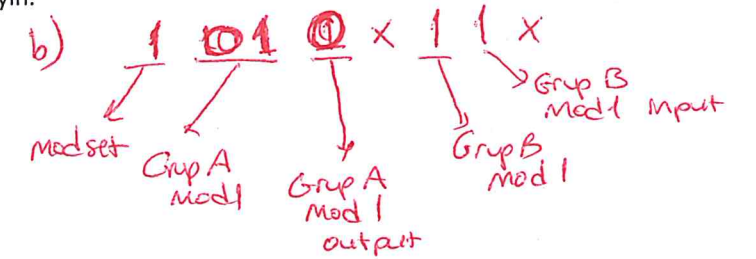
MOV WORD PTR ES:[AH], CS

XOR AX, AX

ENDLESS:

JMP ENDLESS

START ENDP



MOV AL, 10100110B  
 OUT 0AFH, AL