4862 Final Exam
Solutions

1. 64K RAM → $\frac{64K}{8}$ bytes = 8KB (64K is the number of bits)
(This memory might be scattered ie. 64K bits at 64K locations).
We also need the memory organization (number of bits per location) and the type of RAM (SRAM or DRAM).

SRAM	DRAM
• cell is RS flipflop	• 1-2 transistors /cell
• 4-6 transistors/cell	• must be refreshed
• fast	• slower
• low density	• high density
	• cheaper per bit

DRAM Addressing

The address lines are time multiplexed. First the row, and then the column address values are placed on the lines. The Row Address Strobe ($\overline{RAS}$) indicates the presence of row values; the Column Address Strobe ($\overline{CAS}$) indicate the presence of column values.

2. PIT IC

5ms period → $f = \frac{1}{5ms} = 200Hz$

$\therefore$ Count = $\frac{2MHz}{200Hz}$ = 10000 (= 2710H)

Control word = 0011X100 = 00110100b = 34H

```
MOV  AL, 34H
OUT  0A3H, AL
MOV  AX, 10000
OUT  0A0H, AL
MOV  AL, AH
OUT  0A0H, AL
```

3.

	# bytes (# of bus cycles)	clock cycles
MOV AX, 0 (↑ 16bit register)	3	4
SUB AX, AX	2	3
XOR AX, AX	2	3

In both cases, SUB and XOR are faster than MOV. Either instruction should be used.

The INTEL CPUs do not support memory-to-memory operations. Thus the 8088/8086 can not subtract or exclusive or a memory location with itself.

4. With memory-mapped I/O, an I/O device uses memory accesses to read or write data. Thus $\overline{MEMR}$ and $\overline{MEMW}$ lines are used. The effect is that IN and OUT are not used to access the ports, but MOV and other instructions are.

For the given address lines, $2^8 = 256$ mirror images exist.

MOV BYTE PTR ES:[BX], 80H

Example physical address = 80007H

$\therefore$ (ES) = 8000H

(BX) = 0007H

5. When an NMI occurs:

- CPU finishes current instruction
- flags are saved to the stack
- TF & IF are cleared
- PUSH CS
- PUSH IP
- Type 2 (NMI) called ∴ location 00008H

  (IP) ← [00008H]

  (CS) ← [0000AH]
- Instruction as CS:IP executed.

Memory locations 00008H to 0000BH are used.

00008	A2
00009	25
0000A	00
0000B	F3

6. There are 3 T1 clock cycles. The second bus cycle contains 4 clock cycles. The extra clock cycle must be an IDLE state, and not a WAIT state, because the $\overline{RD}$ line goes high on the fourth clock cycle, Thus the fourth clock is T4. The next clock cycle is not a T1 state, since ALE does not become active. Thus the EU must be busy executing a long instruction, and the BIU has filled the queue, so that no bus activity can occur on that cycle.

Bonus Question

System A	System B
233MHz CPU	200 MHz CPU
33MHz bus	50 MHz bus

For most operations, system B will be the faster system. Although the CPU is about 14% slower, and thus instructions execute more slowly, the bus is about 51% faster. This means that the CPU can retrieve instructions and data much faster in system B. So in applications where movement of data is more important, such as multimedia or server applications, system B will be significantly faster because the BIU-equivalent will be busier than the EU-equivalent.

PART II

7(a) I. Determine machine code:

PUSH AX : $01010\underbrace{000}_{\text{REG}} = \boxed{50}$

OR

$1111\,1111\ \underbrace{11}_{\text{MOD}}110\underbrace{000}_{\text{R/M}}$ = FF F0

MOV AX, [BX+DI] : $1000\,10\overset{}{\underset{d\,w}{11}}\ \ \underbrace{00}_{\text{MOD}}\ \underbrace{000}_{\text{REG}}\ \underbrace{001}_{\text{R/M}} = \boxed{8B\ 01}$

(Part marks : Mem to Acc : $1010\,000\underset{w}{1}$ A5 16 → A1 A5 16)

OUT 50H, AX : $1110\,011\underset{w}{1}\ \ \underset{5}{0101}\ \underset{0}{0000} = \boxed{E7\ 50}$

POP AX : $01011\underbrace{000}_{\text{REG}} = \boxed{58}$

OR

$1000\ 1111\ \underbrace{11}_{\text{MOD}}000\underbrace{000}_{\text{R/M}}$ = 8F C0

Stack : SS:SP = 4600:1000 = $\boxed{47000}$

∴ First access is at 46FFE

Data : DS:(BX+DI) = 4000:(0600+10A5) = $\boxed{416A5}$ (odd)

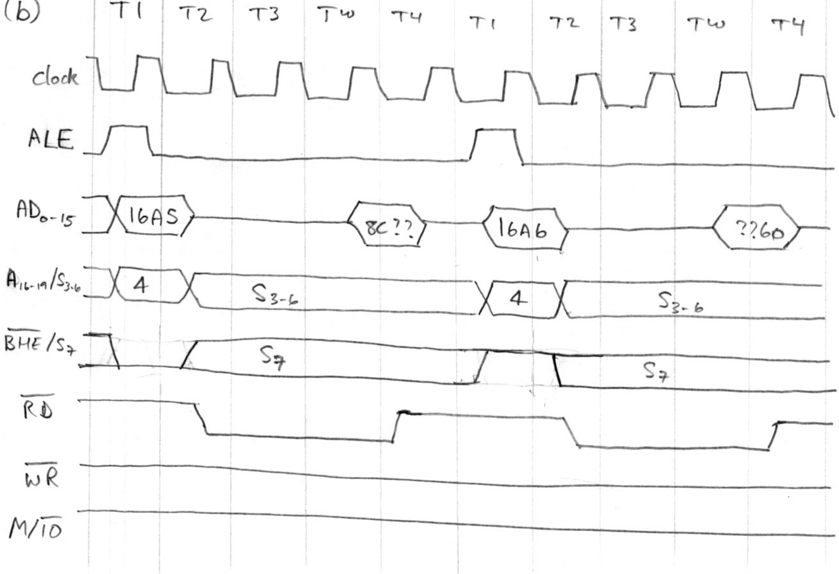
Code : CS:IP = F200:15BA = $\boxed{F35BA}$ (even)

Machine is an 8086 (16-bit):

assumed -- (same)

Address	F35BA	46FFE	F35BC	416A5	416A6	F35BE	00050	46FFE
Data	8B 50	1997	E7 01	8C ??	?? 60	58 50	60 8C	1997
Type	MEMR	MEMW	MEMR	MEMR	MEMR	MEMR	IOW	MEMR
		push		mov			out	pop

7(b)



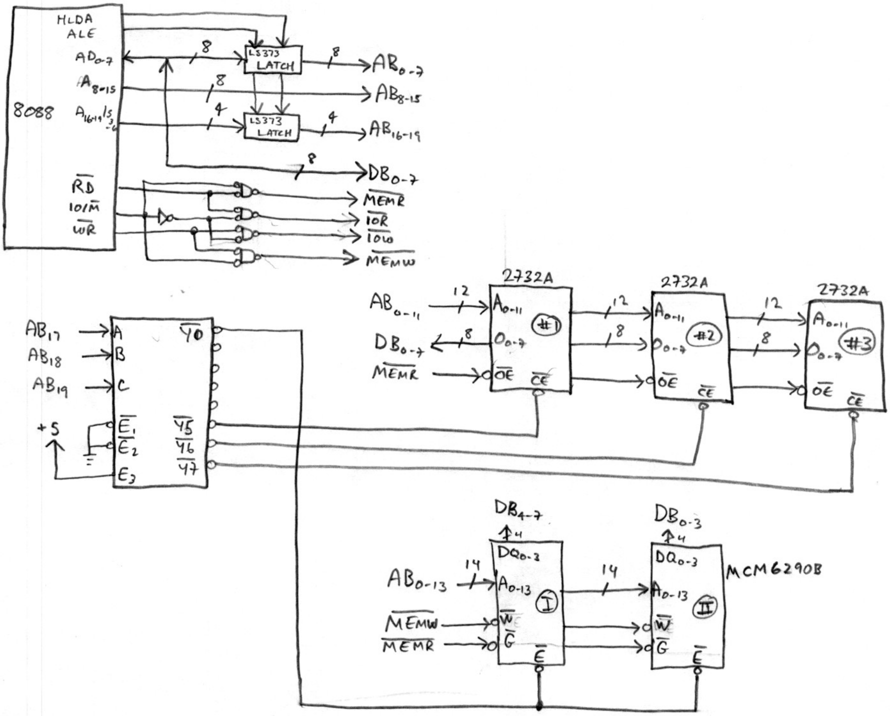
(c) (IP) = 15BA + 6 bytes = 15C0 (depends on exact machine code)
(SP) = 1000 (no change)
(AX) = 1997 (no change)

These instructions do not affect any flags.

8.(a) Need 16KB RAM ∴ use 2 16K×4 SRAM

Need 10 KB ROM ∴ use 3 4K×8 ROM (total of 12KB).

(b)



(c) SRAMs : 000x xx?? ???? ???? ???? : 00000 - 03FFF
04000 - 07FFF
⋮
1C000 - 1FFFF } 8 MI.

ROM #0 : 101x xxxx ???? ???? ????: A0000 - A0FFF
A1000 - A1FFF
⋮
BF000 - BFFFF } 32 MI.

ROM #1 : C0000 - C0FFF
C1000 - C1FFF
⋮
DF000 - DFFFF } 32 MI

ROM#3 : E0000 - E0FFF
E1000 - E1FFF
⋮
FF000 - FFFFF } 32 MI ← system can be booted.

8.(d) SRAM: $I_{IL} = 1\mu A$ $I_{IH} = 1\mu A$ $C_{IN} = 6pF$

EPROM: $I_{IL} = 10\mu A$ $I_{IH} = 10\mu A$ $C_{IN} = 6pF$

∴ TOTALS: $I_{IL} = 3 \times 10\mu A + 2 \times 1\mu A + 45\mu A = 77\mu A$

$I_{IH} = \qquad = 77\mu A$

$C_{IN} = 3 \times 6pF + 2 \times 6pF + 20pF = 50pF$

For CPU: $I_{OL} = 2.5mA > 77\mu A$ ∴ OKAY

$I_{OH} = 400\mu A > 77\mu A$ ∴ OKAY

$C_{LOAD} = 100pF > 50pF$ ∴ OKAY

∴ No need to buffer the line

(e) Address to Data valid

$3T_{CLCL} - T_{CLAV} - T_{latch} - T_{DVCL}$

$= 3(125) - 50 - 30 - 5 = 290ns > 250ns = t_{ACC}$ ∴ OKAY

Chip enable to Data valid

$3T_{CLCL} - T_{CLAV} - T_{latch} - T_{Decoder} - T_{DVCL}$

$= 290ns - 40ns = 250ns = 250ns = t_{CE}$ ∴ OKAY (barely)

Output enable to Data Valid

$2T_{CLCL} - T_{CLRL} - T_{DVCL} - T_{OR}$

$= 250ns - 70 - 5 - 10 = 165ns > 100ns = t_{OE}$ ∴ OKAY

The EPROM (barely) works with no WAIT STATES. Slowing the crystal frequency a "little bit" will cause the second equation to have extra time, and not be so barely successful.

8.(f)

```
         ; Routine to setup and control a signal crossing
         ; Set up PPI
         MOV   AL, 82H
         OUT   63H, AL          ; send control word
         MOV   AL, 0
         OUT   60H, AL          ; Turn off bit 0, port A

         ; Set up PIT, counter 0
         MOV   AL, 36H
         OUT   73H, AL          ; Counter 0, mode 3, 2 bytes
         MOV   AX, 40000        ; Count value = 40000
         OUT   70H, AL
         MOV   AL, AH
         OUT   70H, AL
         ; Set up PIT, counter 1
         MOV   AL, 56H
         OUT   73H, AL          ; counter 1, mode 3, 1 byte
         MOV   AL, 100          ; count value = 100
         OUT   71H, AL

         ; Main Loop:
Check:   IN    AL, 61H          ; check Port B
         AND   AL, 0000 0011b   ; bits 0 or 1 on?
         JZ    No_train
Train:   MOV   AL, 0000 0001b   ; train is present, so turn on signal
         JMP   Signal
No-train: MOV  AL, 0000 0000b   ; No train at crossing
Signal:  OUT   60H, AL          ; turn signal on or off
         JMP   Check            ; repeat check for train.
```

PPI Control Word

10000010 = 82H
AC BC

PIT Control Word

Counter 0: 0011X110 = 36H
Count value = 40000 d

Counter 1: 0101X110 = 56H
Count value = 100 d