CDA 3103 Computer Organization
Homework 1

Assigned: May 25th, 2016 **Due:** 11:55pm via Canvas, May 31th, 2016

1 Problems

1. (5 pts) What are the essential building blocks for a computer?

2. (5 pts) What is the difference between computer organization and computer architecture?

3. (5 pts) Technical societies such as IEEE and ACM as well as other entities organize contests for Computer science and engineering students. Name a few such contests.

4. (5 pts) What are Charles Babbage and Ada Gordon famous for in the history of computers?

5. (10 pts)
   a) How many milliseconds (ms) are in 2 second?
   b) How many microseconds are in 5 second?
   c) How many nanoseconds (ns) are in 3 millisecond?
   d) How many microseconds are in 2 millisecond?
   e) How many nanoseconds are in 13 microsecond?
   f) How many kilobytes (KB) are in 2 gigabyte (GB)?
   g) How many kilobytes are in 6 megabyte (MB)?
   h) How many megabytes are in 10 gigabyte (GB)?
   i) How many bytes are in 10 megabytes?
   j) How many kilobytes are in 5 gigabytes?

6. (5 pts) In the von Neumann model, explain the purpose of the
   a) Processing unit,
   b) Program counter.

7. (5 pts) Under the von Neumann architecture, a program and its data are both stored in memory. It is therefore possible for a program, thinking a memory location holds a piece of data when it actually holds a program instruction, to accidentally (or on purpose) modify itself. What implications does this present to you as a programmer?

8. (5 pts) State Moore's law and Rock's Law. How is Rock's Law related to Moore's Law?

9. (5 pts) How does the fetch-decode-execute cycle work?

10.(5 pts) Explain why modern computers consist of multiple abstract levels.

2 Submission Requirements

The following requirements are for electronic submission via Canvas.

- Your solutions must be in a single pdf file with a file name yourname-hw1.
- Upload the file by following the link where you download the homework description on Canvas.
- If scanned from hand-written copies, then the writing must be legible, or loss of credits may occur.
- **Only submissions via the link on Canvas where this description is downloaded are graded. Submissions to any other locations on Canvas will be ignored.**

Steven
Romeiro

Computer Organization Homework 1

(1) • A processor to interpret & execute programs
• A memory to store both data & programs
• A mechanism for transferring data. I/O

(2) Computer organization:
- Encompass all physical aspects of computer systems.
- Examples: Computer design, control signals

Computer Architecture:
- Logical aspects of system implementation
- Combo of computer hardware & ISA
- ISA is an agreed upon interface between hardware & Software

(3) ACM Student Research Competition
ACM International Collegiate Programming Contest
IEEE Xtreme Programming Competition
IEEE Student Branch Web Site Contest
Source: ACM & IEEE websites

(4) Charles Babbage - The Difference Engine also designed the Analytical Engine
Ada, Countess of Lovelace - Name "Gordon" is incorrect. Suggested a plan for how Babbage's Analytical Engine would calculate numbers. Also suggested a binary system for storing data.

(5) a) 2000 ms b) 5,000,000 μs c) 3,000,000 ns
d) 2,000 μs e) 13,000 ns f) $2^{30}/2^{10} = 2 * 2^{20}$ KB
g) $2^{20}/2^{10} = 6 * 2^{10}$ or 6,000 kB h) $2^{30}/2^{20} = 10 \times 2^{10}$ MB
i) 10×2^{20} bytes j) $2^{30}/2^{10} = 5 \times 2^{20}$ KB

(6) a) Processing Unit – Known as the Central Processing Unit (CPU) contains the Control Unit responsible for fetching & sending info, the Arithmetic Logic Unit responsible for calculations, Registers which are small storage areas & the Program Counter that is responsible for correctly telling the control Unit which instruction comes next
b) Program Counter – Determines where the instruction the control unit needs is located & which instruction line is next.

(7) Proper understanding of how memory is accessed is crucial to a programmer so that he or she doesn't do this. It is also their responsability to Implement safety measures against those trying to exploit a program.

(8) Moore's Law - The density of silicon chips doubles every 18 months.
Rock's Law - The cost of capital equipment to build semi-conductors will double every four years.
These laws are directly related by the fact that either the target goal of doubling density of chips in 18 months needs to be extended or the costs must be met to maintain expectations. Either one of these laws must govern the other.

(9) Program Counter tells the control unit which instruction to grab next, control unit fetches & decodes the data & passes it to the ALU to process the data. The processed information is either sent back to memory or it's temporarily stored in the registers for further manipulation. This process is then repeated again for the next instruction

(10) This design draws similarities with a programming concept of dividing up a large problem into small segments & developing systems that handle each segment. In computer architecture this is done via levels of abstraction of virtual machines. Each level excecutes it's own set of instructions by calling upon a lower level machine to carry out whatever task is necessary. This hierarchy system allows each abstract level to handle small segments until the large problem is resolved

$(AB)' \cdot (B+C)' = (A'+B')\cdot(B'C')$

$A'B'C' + B'B'C' = A'B'C' + B'C' = (A+1)B'C'$

$= \overline{BC}$

CDA 3103 Computer Organization

Homework #3

Due Date: June 17, 2016

$(x'+y+z)'$

$x \cdot y' \cdot z'$

1 Problems

1. (10 Points) Construct a truth table for the following:
   a) $xyz + x(yz)' + x'(y+z) + (xyz)'$
   b) $(x+y')(x'+z')(y'+z')$

2. (5 Points) Using DeMorgan's Law, write an expression for the complement of F if $F(x, y, z) = (x'+y)(x+z)(y'+z)'$.

   $[(x'+y)(x+z)(y'+z)']'$

3. (10 Points) Show that $x = xy + xy'$
   a) Using truth tables
   b) Using Boolean identities

   $\cancel{(x'+y)'\,(x+}$

   $(x'+y)' + (x+z)' + (y'+z)'$

   $x' \cdot y' + x' \cdot z' + (y'+z)$

   $x \cdot y'$

4. (10 Points) Simplify the following functional expressions using Boolean algebra and its identities. List the identity used at each step.
   a) $F(x, y, z) = x'yz + xz$

   b) $F(x, y, z) = (x'+y+z')' + xy'z' + yz + xyz$

5. (5 Points) The truth table for a Boolean expression is shown below. Write the Boolean expression in Canonical sum-of-products and Canonical product-of-sum forms.

x	y	z	F
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

6. (15 Points) Create the Kmaps and then simplify for the following functions:
   a) $F(x, y, z) = x'y'z' + x'yz + x'yz'$

b) $F(x,y,z) = x'y'z' + x'yz' + xy'z' + xyz'$
c) $F(x,y,z) = y'z' + y'z + xyz'$

7. (15 Points) Write a simplified expression for the Boolean function defined by each of the following Kmaps.
a)

wx \ yz	00	01	11	10
00	1	0	0	1
01	1	0	0	1
11	0	0	1	0
10	1	0	1	0

b)

wx \ yz	00	01	11	10
00	1	1	1	1
01	0	0	1	1
11	1	1	1	1
10	1	0	0	1

c)

wx \ yz	00	01	11	10
00	0	1	0	1
01	0	1	1	1
11	1	1	0	0
10	1	1	0	1

AND OR

$A \cdot 0 = 0$ $A + 0 = A$

$A \cdot 1 = A$ $A + 1 = 1$

$A \cdot A = A$ $A + A = A$

$A \cdot A' = 0$ $A + A' = 1$

Misc

$A'' = A$

$A + AB = A$

$A + \bar{A}B = A + B$

$A + (BC) = (A+B)(A+C)$

Steven
Romeiro

Homework #3

1. Truth table

a) $xyz + x(yz)' + x'(y+z) + (xyz)' =$

X	Y	Z	xyz	x(yz)'	x'(y+z)	(xyz)'	xyz + x(yz)' + x'(y+z) + (xyz)'
0	0	0	0	0	0	1	1
0	0	1	0	0	1	1	1
0	1	0	0	0	1	1	1
0	1	1	0	0	1	1	1
1	0	0	0	1	0	1	1
1	0	1	0	1	0	1	1
1	1	0	0	1	0	1	1
1	1	1	1	0	0	0	1

b) $(x+y')(x+z)(y'+z)' =$

X	Y	Z	(x+y')	(x+z)	(y'+z)'	(x+y')(x+z)(y'+z)'
0	0	0	1	0	0	0
0	0	1	1	1	0	0
0	1	0	0	0	1	0
0	1	1	0	1	0	0
1	0	0	1	1	0	0
1	0	1	1	1	0	0
1	1	0	1	1	1	1
1	1	1	1	1	0	0

② Use DeMorgans Law $f(x,y,z) = (x'+y)(x+z)(y'+z)'$

$f'(x,y,z) = [(x'y)(xz)(y'z)']'$

$f'(x,y,z) = (x'y)' + (xz)' + (y'z)$

$\boxed{f'(x,y,z) = (xy') + (x'z') + (y'z)}$

③ Show $X = XY + XY'$

a) Using truth tables

X	Y	Y'	XY	XY'	XY+XY'
0	0	1	0	0	0
0	1	0	0	0	0
1	0	1	0	1	1
1	1	0	1	0	1

b) Using Boolean Iden.

$X = X(Y+Y')$ distrib

$X = X(1)$ Inverse

$\boxed{X = X}$

④ Simplify with Boolean Identity

a) $F(x,y,z) = X'YZ + XZ$

$= Z(X'Y + X)$ distrib.

Rules

#11 $Z(X+Y)$

$\boxed{XZ + YZ}$ Distrib

AND	OR
1 $A \cdot 0 = 0$	5 $A + 0 = A$
2 $A \cdot 1 = A$	6 $A + 1 = 1$
3 $A \cdot A = A$	7 $A + A = A$
4 $A \cdot \bar{A} = 0$	8 $A + \bar{A} = 1$

MISC

9 $\bar{\bar{A}} = A$	11 $A + \bar{A}B = A + B$
10 $A + AB = A$	12 $(A+B)(A+C) = A + (BC)$

b) $F(X,Y,Z) = (X'+Y+Z')' + XY'Z' + YZ + XYZ$

$XY'Z + XY'Z' + YZ + XYZ$ (DeMorgans)

$XY'(Z+Z') + YZ + XYZ$ (Distributive)

$XY'(1) + YZ(1+X)$ (Inverse & Distributive

$XY' + YZ(1+X)$ (Identity)

$XY' + YZ(1)$ (Dominance Law)

$XY' + YZ$ (Identity)

$(X+Z)(Y'+Y)$ (Distributive)

$\boxed{(X+Z)}$ (inverse/Identity)

(5) $\boxed{SOP = X'Y'Z' + X'YZ' + XY'Z + XYZ' + XYZ}$

$\boxed{POS = (X+Y+Z')(X+Y'+Z')(X'+Y+Z)}$

(6) KMAPS for

a) $F(X,Y,Z) = \underset{000}{X'Y'Z'} + \underset{011}{X'YZ} + \underset{010}{X'YZ'}$

X \ YZ	00	01	11	10
0	1	0	1	1
1	0	0	0	0

$\boxed{F(X,Y,Z) = X'Z' + X'Y}$

b) $F(X,Y,Z) = \underset{000}{X'Y'Z'} + \underset{010}{X'YZ'} + \underset{100}{XY'Z'} + \underset{110}{XYZ'}$

X \ YZ	00	01	11	10
0	1	0	0	1
1	1	0	0	1

$\boxed{F(X,Y,Z) = Z'}$

c) $F(X,Y,Z) = \underset{00}{Y'Z'} + \underset{01}{Y'Z} + \underset{110}{XYZ'}$

$X \backslash YZ$	$\bar{Y}\bar{Z}$ 00	$\bar{Y}Z$ 01	YZ 11	$Y\bar{Z}$ 10
$\bar{x}$ 0	1	1	0	0
x 1	1	1	0	1

$\boxed{F(X,Y,Z) = Y' + XZ'}$

7)

$WX \backslash YZ$	00	01	11	10
00	1	0	0	1
01	1	0	0	1
11	0	0	1	0
10	1	0	1	0

3 Groups

$\boxed{F(W,X,Y,Z) = W'Z' + X'Y'Z' + WYZ}$

b)

$WX \backslash YZ$	00	01	11	10
00	1	1	1	1
01	0	0	1	1
11	1	1	1	1
10	1	0	0	1

4 Groups

$\boxed{F(W,X,Y,Z) = \overline{W}\overline{X} + WX + XY + \overline{X}\overline{Z}}$

c)

$WX \backslash YZ$	00	01	11	10
00	0	1	0	1
01	0	1	1	1
11	1	1	0	0
10	1	1	0	1

4 Groups

$\boxed{F(W,X,Y,Z) = W\overline{Y} + \overline{Y}Z + \overline{W}XY + \overline{X}Y\overline{Z}}$

Steven
Romeiro

Homework 2

(1) Convert 25.84375

$25/2 = 12\ r\ 1 \rightarrow 1 \quad 1 \leftarrow .84375 \times 2 = 1.68750$

$12/2 = 6\ r\ 0 \rightarrow 0 \quad 1 \leftarrow .68750 \times 2 = 1.37500$

$6/2 = 3\ r\ 0 \rightarrow 0 \quad 0 \leftarrow .375 \times 2 = 0.750$

$3/2 = 1\ r\ 1 \rightarrow 1 \quad 1 \leftarrow .75 \times 2 = 1.50$

$1/2 = 0\ r\ 1 \rightarrow 1 \quad 1 \leftarrow .5 \times 2 = 1.0$

$= 11001.11011$

to six places $= \boxed{11001.1}$

(2) Convert hex AC12

$\underbrace{1010}_{A}\ \underbrace{1100}_{C}\ \underbrace{0001}_{1}\ \underbrace{0010}_{2}$

$\boxed{AC12_{16} = (10101100\ 00010010)_2}$

(3)

	Signed Mag	One's	Two's
a) 60 =	00111100	00111100	00111100
b) -60 =	10111100	11000011	11000100

(4) Decimal value of 10011110 if

a) Unsigned: 158_{10}

b) Signed Mag: -30_{10}

c) One's Comp: -97_{10}

d) Two's Comp: -98_{10}

(5) Given 11111100 & 01110000

a) Unsigned: 11111100 > 01110000

b) Two's Comp: 01110000 > 11111100

c) Signed Mag: 11111100 < 01110000

(6)

a) Signed Mag:	b) One's Comp	Two's Comp
0 = 0000	0 = 0000	0 = 0000
1 = 0001	1 = 0001	1 = 0001
2 = 0010	2 = 0010	2 = 0010
3 = 0011	3 = 0011	3 = 0011
4 = 0100	4 = 0100	4 = 0100
5 = 0101	5 = 0101	5 = 0101
6 = 0110	6 = 0110	6 = 0110
7 = 0111	7 = 0111	7 = 0111
-0 = 1000	-7 = 1000	-8 = 1000
-1 = 1001	-6 = 1001	-7 = 1001
-2 = 1010	-5 = 1010	-6 = 1010
-3 = 1011	-4 = 1011	-5 = 1011
-4 = 1100	-3 = 1100	-4 = 1100
-5 = 1101	-2 = 1101	-3 = 1101
-6 = 1110	-1 = 1110	-2 = 1110
-7 = 1111	-0 = 1111	-1 = 1111

7 a) Signed: $-(2^{x-1}-1)$ to $+(2^{x-1}-1)$

b) One's Comp: $-(2^{x-1}-1)$ to $+(2^{x-1}-1)$

c) Two's Comp: $-(2^{x-1})$ to $+(2^{x-1}-1)$

8 Using arithmetic shifting

a) double 00010101 = $\boxed{00101010}$

b) Quadruple 01110111 = $\boxed{\text{Overflow unless extra byte is added } 00000001\ 11011100}$

c) Didide by half 11001010 = $\boxed{11100101}$

9 Decode ASCII

74, 79, 72, 78, 32, 68, 79, 69

J O H N D O E

$\boxed{\text{message = JOHN DOE}}$

10 3 info bit & 1 Parity bit code, odd parity:

0001

0010

0100

0111

1000

1011

1110

(11) Odd parity is being used. The 8th bit is being used as a parity bit to make the sum of all the "ON" bits an odd number.

(12) a)

```
          100111 (39)
        + 111001 (57)
carry 1   100000  -> Overflow
```

b)

```
  10110 (22)
-   101 (5)
  10001 = 17
```

(13) X = 87 & Y = 25

a) X + Y

```
  01010111 (87)
+ 00011001 (25)
  01110000 = 112
```

b) X − Y

```
  01010111 (87)
- 00011001 (25)
  00111110  62
```

c) −X + Y

```
  10101001 (-87)
+ 00011001 (25)
  11000010 (-62)
```

d) −X − Y

```
  10101001 (-87)
  11100111 (-25)
  10010000  -112
```

CDA 3103 Computer Organization

Homework #4

Assigned: June 30, 2016 **Due**: 11 : 59pm via Canvas, July 6, 2016

1. What are the main functions of the CPU?

2. How is the ALU related to the CPU? What are its main functions?

3. How many bits are required to address a 4M X 32 main memory if $2+20$
   a) Main memory is byte-addressable? → $4M = 2^2 \times 2^{20} = 2^{22} = 22$ bits ?
   b) Main memory is word-addressable? →

4. How many bits are required to address a 2M X 8 main memory if
   a) Main memory is byte-addressable?
   b) Main memory is word-addressable?

5. Assume a 2^{21} byte memory:
   a) What are the lowest and highest addresses if memory is byte-addressable?
   b) What are the lowest and highest addresses if memory is word-addressable, assuming a 16-bit word?
   c) What are the lowest and highest addresses if memory is word-addressable, assuming a 32-bit word?

6. Explain the steps of the fetch-decode-execute cycle. Your explanation should include what is happening in the various registers.

7. Combine the flowcharts that appear in Figures 4.11 and 4.12 so that the interrupt checking appears at a suitable place.

8. Explain why, in MARIE, the MAR is only 12 bits wide while the AC is 16 bits wide. Hint: Consider the difference between data and addresses.

9. Use a few sentences to answer each of the following questions. The answers can be found in section 4.1 to 4.7 in the textbook.
   a) What purpose does the data path in a CPU serve?
   b) What does the control unit in a CPU do?
   c) Where are registers located and what are are the different types?
   d) How does the ALU know which function to perform?
   e) Explain the differences between data buses, address buses, and control buses.
   f) Explain the relation between clock cycle time and clock frequency.
   g) Explain the differences between memory-mapped I/O and instruction-based I/O.
   h) Explain the differences between byte addressable and word addressable.
   i) Explain what interrupts are, and list four reasons that trigger interrupts.

Steven
Romeiro

Homework 4

(1) Fetching instructions, decoding them & performing the indicated sequence of operations

(2) The ALU one of the CPU's main parts. It carries out logic operations & arithmetic operations during program execution.

(3) a) $4M = 2^2 \times 2^{20} = 2^{22}$

$W = 32bits = 4bytes = 2^2$

$2^{22} \times 2^2 = 2^{24}$ items $= \boxed{24 bits}$

b) $4M = 2^2 \times 2^{20} = 2^{22}$

$32bits = 1 word = 2^0$

$2^{22} \times 2^0 = 2^{22}$ items $= \boxed{22 bits}$

(4) a) $2M = 2^1 \times 2^{20} = 2^{21}$

$W = 8 bits = 1 byte = 2^0$

$2^{21} \times 2^0 = 2^{21}$ items $= \boxed{21 bits}$

b) $2M = 2^1 \times 2^{20} = 2^{21}$

$8 bits = 1 word = 2^0$

$2^{21} \times 2^0 = 2^{21}$ items $= \boxed{21 bits}$

⑤ a) $\boxed{0 \text{ Lowest } - (2^{21}-1) \text{ highest}}$

b) 16 bit words = 2 bytes = 2^1

$2^{21} \div 2^1 = 2^{20}$

$\boxed{0 \text{ Lowest } - (2^{20}-1) \text{ highest}}$

c) 32 bit words = 4 bytes = 2^2

$2^{21} \div 2^2 = 2^{19}$

$\boxed{0 \text{ lowest } - (2^{19}-1) \text{ highest}}$

⑥ Fetch: Copy contents of PC to MAR

Fetch instruction in MAR & place in IR.

Increment PC by 1

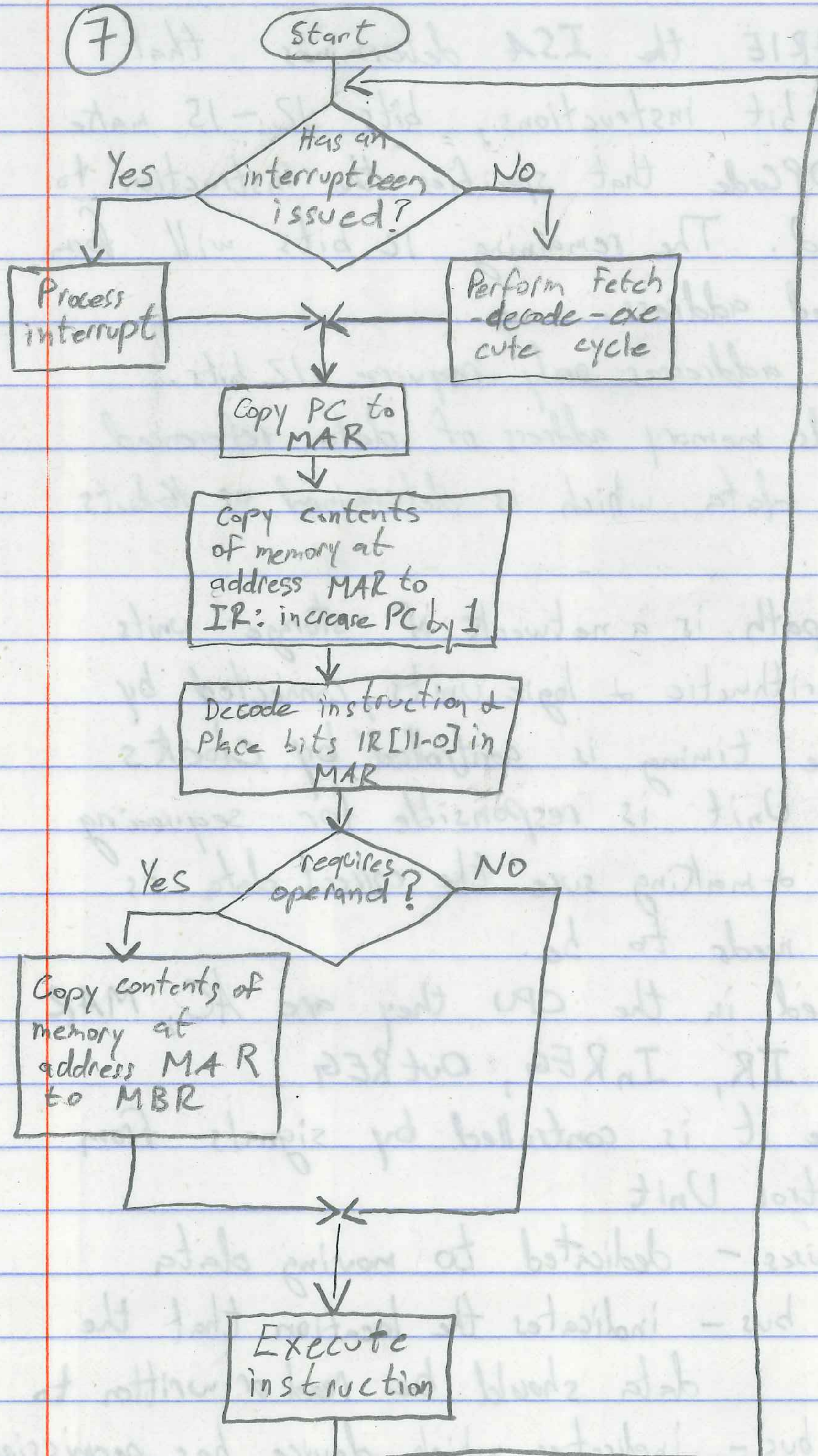
Decode: Copy the right 12 bits of IR to MAR. Decode the left 4 bits to determine OPCode

Execute: If necessary, use MAR address get data & place it in MBR or AC.

Then execute instruction MBR <— M[MAR]

& execute actual instruction

7



Start
Has an interrupt been issued?
Yes
No
Process interrupt
Perform Fetch-decode-execute cycle
Copy PC to MAR
Copy contents of memory at address MAR to IR: increase PC by 1
Decode instruction & Place bits IR[11-0] in MAR
requires operand?
Yes
NO
Copy contents of memory at address MAR to MBR
Execute instruction

(8) In MARIE the ISA determines that of the 16 bit instructions, bits 12 – 15 make up the OPCode that specifies the instruction to be executed. The remaining 12 bits will form the required address.

Therefore addresses only require 12 bits.

MAR holds memory address of data referenced

AC holds data which is determined as 16 bits

(9) a) data path is a network of storage units (registers) arithmetic & logic units, connected by buses where timing is controlled by clocks

b) Control Unit is responsible for sequencing operations & making sure the correct data is where it needs to be.

c) Located in the CPU they are AC, MAR MBR, PC, IR, InREG, OutREG

d) Because it is controlled by signals from the Control Unit

e) data buses – dedicated to moving data

address bus – indicates the location that the data should be read or written to

Control bus – indicates which device has permission to use the bus & for what purpose

f) Clock cycles – time between clock ticks for instruction performance

Clock frequency – measured in hertz & is the reciprocal of Clock Cycles

g) Memory mapped I/O – registers in the interface appear in the computers memory map & there is no difference between accessing memory & I/O device

Instruction based I/O – CPU has specialized instructions that perform input & output.

h) byte addressable – which means each individual has a unique address

Word addressable – which means each word has its own address

i) Interrupts – are events that alter the normal flow of execution in the system.

Reasons – I/O requests, Arithmetic errors, Hardware malfunction, Page faults.

SCRAP

26

10		

16 bits

2^{26}

2^{16} cache

ORG 100
Load X
Subt One
Skipcond 000
Jump Else
Then, load X
ADD Y
Store Y
load X
Clear
Jump Endif
Else, load Y
ADD One
Store Y
Endif, Halt
X, Dec 5
Y, Dec 3
One, Dec 1

MBR — PC
MAR — X
MMAR — MBR

4M x 16

byte = $2^2 \times 2^{20} \times \frac{2^4}{2^3}$ = 23 bits

word = 22 bits

1M x 8

$2^0 \times 2^{20} \times 2$ = 20

128 word 8-way HOI

uses 8 banks = 2^3 → 3 bits per chip

128 words = 2^7 → 7 bits per word addr

3 bits MSB chip 4 bit for offset

each chip holds 2^4 words = 16 words

4 modules = 2^2 = 2 bits for chip select

128 words = 2^7 5

4096 bytes 2048 1024 512 256

2^{12}

256 = $2^8 * 2^4$ = 16 chips

each addr = 8 bits

2M x 16 built using 1 256K x 8 chip

$2^1 \times 2^{20} = 21$

$2^8 \times 2^{10} = 2^{18}$

$\frac{2M \times 16}{256K \times 8} = \frac{2^{21} \times 2^4}{2^{18} \times 2^3} = \frac{2^{25}}{2^{21}} = 2^4$

16 chips

2 chips, 16 chips ~~= 2^4 = 4 bits addr~~

$\frac{2M}{256k} = \frac{21}{18} = 2^3$ = 8 banks

MAR — X
MBR — MMAR
AC — MBR
MAR — X
MBR — MMAR
AC — MBR + AC

MAR — ~~AC~~
MBR — AC
MMAR — MBR
MBR — X

2^{16}

19 + 7 = 26

		AC
	ORG 100	
	load One	1
	Store X	1
loop,	load X	1 2 3 4 5
	ADD One	2 3 4 5 6
	Store X	2 3 4 5 6
	Subt Five	-3 -2 -1 0 1
	Skipcond 800	-3 -2 -1 0 1
	Jump loop	↓
	Halt	1 → Final result in AC
X,	Dec 0	
One,	Dec 1	
Five,	Dec 5	

1108
3109
9106
3109
2107
7000
3108
9103
0023
0001

RTL load
MAR ← X
MBR ← M[MAR]
AC ← MBR
Store X
MAR ← X
MBR ← AC
M[MAR] ← MBR

3

Load
MAR ← X
MBR ← M[MAR]
MAR ← MBR
MBR ← M[MAR]
AC ← MBR
Store I
MAR ← X
MBR ← M[MAR]
MAR ← MBR
MBR ← AC
M[MAR] ← MBR

1 X → 104
3 Y → 105
2 Z → 106
7000

D = 10A
000
108

16 2K x 8bit chips = memory 32K

$32 = 2^5$ & $K = 2^{10}$

$2^4 = \quad = 2^{15}$ = 15 bits for each address

4 bits to select chip

2M x 32 if byte

$2^1 \times 2^{20} \times \frac{2^5}{2^3} = 2^1 \times 2^{20} * 2^2 \rightarrow 2^{23}$ 23 bits per add.

word $2^1 \times 2^{20} \times \cancel{2^5}$ = ~~26~~ bits 2^{21} = 21 bits needed

$$\begin{array}{r} 00FC \\ +\ 000E \\ \hline 10A \end{array}$$

12 14

$$\begin{array}{r} 26 \\ -16 \\ \hline 10 \end{array} = A$$

Due: 11 : 59pm ,July 12th, 2016

1. Consider the MARIE program below.
   a. List the hexadecimal code for each instruction.
   b. Draw the symbol table.
   c. What is the value stored in the AC when the program terminates?

Hex Address	Label	Instruction
200	Begin,	LOAD Base
201		ADD Offs
202	Loop,	SUBT One
203		STORE Addr
204		SKIPCOND 800
205		JUMP Done
206		JUMPI Addr
207		CLEAR
208	Done,	HALT
209	Base,	HEX 200
20A	Offs,	DEC 9
20B	One,	HEX 0001
20C	Addr,	HEX 0000

6 > 0

-skip
~~AC < 0~~
AC = 0
AC > 0

2. Given the instruction set for MARIE in this chapter, decipher the following MARIE machine language instructions. (Write the assembly language equivalent.)

   a. 0011000000000110
   b. 1011000000001011
   c. 0011000000001001

X-1 < 1

else

-ve
0

3. Write the following code segment in MARIE's assembly language:

```
if X < 1 then
    Y = X + Y;
    X = 0;
endif;
Y = Y + 1;
```

-1, -2, -3, 0

x - 1
check for < 0
true
false

x = x - 1 = 6

800

x - 1 = x +ve -ve

4. Write the following code segment in MARIE assembly language:

```
X = 1;
while X < 5 do
    X = X + 1;
endwhile;
```

Y ≤ 5

1 - 5 = -ve

load one
store X = 1

one 1
five 5

AC = 1
MBR = 5

Steven
Romeiro

Homework 5

(1)

1	209
3	20A
4	20B
2	20C
8	800
9	208
C	20C
A	000
7	000
	200
	009
	0001

a) Hex Codes for instruction

b) Symbol Table

Address	Symbol
200	Begin
202	Loop
208	Done
209	Base
20A	Offs
20B	One
20C	Addr

c) Final AC value = $\boxed{208}$

(2) a. 0011 0000 0000 0110

$\underbrace{3}_{\text{4-bit opcode}}\ \underbrace{0\ 0\ 6}_{\text{12-bit address}}$

$\boxed{\text{ADD}\quad 006}$

b. 1011 0000 0000 1011

$\underbrace{B}_{\text{4-bit opcode}}\ \underbrace{0\ 0\ B}_{\text{12-bit address}}$

$\boxed{\text{ADDI}\quad 00B}$

C. $\underbrace{\underbrace{0011}_{3}}_{\text{OP CODE}}\ \underbrace{\underbrace{0000}_{0}\ \underbrace{0000}_{0}\ \underbrace{1001}_{9}}_{\text{12-bit address}}$

$\boxed{\text{ADD 009}}$

(3)

Hex Address	Label	Instruction
100	If,	load X
101		Subt One
102		Skipcond 000
103		Jump Else
104	Then,	load X
105		ADD Y
106		Store Y
107		load X
108		Clear
109		Jump Endif
10A	Else,	load Y
10B		ADD One
10C		Store Y
10D	Endif,	Halt
10E	X,	Dec 5
10F	Y,	Dec 3
110	One,	Dec 1

* Note: modified loop for proper incrementation

(4)

Hex Address	Label	Instruction
100		load One
101		Store X
102	Loop,	load X
103		ADD One
104		Store X
105		Subt Five
106		Skipcond 800
107		Jump Loop
108		Halt
109	X,	Dec 0
10A	One,	Dec 1
10B	Five,	Dec 5