METRO STATE
UNIVERSITY

ICS 232 Computer Organization & Architecture
Homework 4 - Chapter 4 Part 1 - 10 points
Due Date: 6/7/2023

Name:

Note: Please post your homework to ICS232 D2L on or before the due date.

Chapter 4 – MARIE: An Introduction to a Simple Computer

Essential Terms and Concepts

4. Where are registers located and what are the different types?

- CPU

1) Accumulator /AC
2) Memory address register / MAR
3) Memory buffer register / MBR
4) Program Counter / PC
5) Instruction register / IR
6) Input register/ InREG
7) Output register/ OutREG

10. What is a bus cycle?

- sequence of events that starts with an address being output on the system bus followed by read/write data transfer

17. Explain the difference between memory-mapped I/O and instruction-based I/O.

- memory-mapped: memory-specific instructions
- instruction-based: only in/out instructions are accessed

20. Why is address alignment important?

- because it will result in hardware traps if the instruction address aren't properly aligned

21. List and explain the two types of memory interleaving and the differences between them.

- high-order interleaving: · the high-order 4 bits select the chip
- low-order interleaving: · memory address contains the address of interest · the low-order 4 bits select the chip

33. How does interrupt driven I/O work?

- when an input is available, the CPU is interrupted from its work to take care of the data

38. What is a stack? Why is it important for programming?

- stack: used to implement functions, parsers, expression evaluation, and backtracking algorithms
- stacks are used in systematic memory management

METRO STATE
UNIVERSITY

ICS 232 Computer Organization & Architecture
Homework 4 - Chapter 4 Part 1 - 10 points
Due Date: 6/7/2023

Exercises

1. What are the main functions of the CPU?

fetch - decode - execute - write back

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

- ALU is part of the CPU that carries out arithmetic and logical operations on the operands in computer instruction words
- main function: arithmetic & logic operations

5. How many bits are required to address a 4M × 16 bits main memory if

a) Main memory is byte-addressable? $4M \times 16 = 2^{2(4)} \times 2^{20(1M)} \times 2^{1(2 \text{ bytes}, 16 \text{ bits})} = 2^{23}$ so 23-bits

b) Main memory is word-addressable? $4M \times 16 = 2^{2(4)} \times 2^{20(1M)} \times 2^{0(1 \text{ word}, 16 \text{ bits})} = 2^{22}$ so 22-bits

13. A digital computer has a memory unit with 24 bits per word. The instruction set consists of 150 different operations. All instructions have an operation code part (opcode) and an address part (allowing for only one address). Each instruction is stored in one word of memory.

a) How many bits are needed for the opcode? $2^7 = 128$ $128 < 150$, so 128 bits

b) How many bits are left for the address part of the instruction? $24 - 8 = 16$ 16 bits

c) What is the maximum allowable size for memory? $2^{16} = 65536$ Max number is 65535

d) What is the largest unsigned binary number that can be accommodated in one word of memory?

$2^{15} = 32768 - 1$
$= 32767$

21. 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

- MARIE handle 16-bit data, so AC must be 16-bit wide
- However, MARIE's memory is limited to 4096 address locations, so MAR only needs to be 12 bit wide to hold the largest address

METRO STATE
UNIVERSITY

ICS 232 Computer Organization & Architecture
Homework 4 - Chapter 4 Part 1 - 10 points
Due Date: 6/7/2023

27. Write the assembly language equivalent of the following MARIE machine language instructions:

a) 0111000000000000 HALT

b) 1011001100110000 330

c) 0100111101001111 F4F

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

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

```
if     load      x
       subt      1
       skipcond  800
       jump      endif
then   load      x
       add       x
       store     x
       clear
       store     x
endif  load      y
       add one
       store     1
       halt
```

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

```
X = 1;
while X < 10 do
   X = X + 1;
endwhile;
```

```
       org       100
       load      1
       store     x
test   subt      10
       skipcond  400
       jump
       loop
       load      x
       add       1
       store     x
       jump halt
```

Prepare for next class by continuing to read Chapter 4 – MARIE: An Introduction to a Simple Computer.

Start working on Your Group Project

Start working on Project 1