Practical List for BHCS02
Computer System Architecture

(Use Simulator – CPU Sim 3.6.9 or higher version for the implementation)

1. Create a machine based on the following architecture:

IR	DR	AC	AR	PC	S	I	E
16 bits	16 bits	16 bits	12 bits	12 bits	1 bit	1 Bit	1 Bit

Memory	Instruction format
4096 words 16 bits per word	0 3 4 15 Opcode (3 bits) | Address (12 bits)

Basic Computer Instructions

Memory Reference			Register Reference		Input-Output		
Symbol	Hex		Symbol	Hex	Symbol	Hex	
AND	0xxx	Direct Addressing	CLA	E800	INP	F800	
ADD	2xxx		CLE	E400	OUT	F400	
LDA	4xxx		CMA	E200	SKI	F200	**Optional**
STA	6xxx		CME	E100	SKO	F100	
BUN	8xxx		CIR	E080	ION	F080	
BSA	Axxx		CIL	E040	IOF	F040	
ISZ	Cxxx		INC	E020			
AND_I	1xxx	Indirect Addressing	SPA	E010			
ADD_I	3xxx		SNA	E008			
LDA_I	5xxx		SZA	E004			
STA_I	7xxx		SZE	E002			
BUN_I	9xxx		HLT	E001			
BSA_I	Bxxx						
ISZ_I	Dxxx						

Refer to chap 5 of Morris Mano for description of instructions. Create the microoperations and associate with instructions as given in chapter. (Except interrupts).

Design the register set, memory and the instruction set. Use this machine for the assignments of this section.

2. Create a Fetch routine of the instruction cycle.

3. Write an assembly program to simulate ADD operation on two user-entered numbers.

4. Write an assembly program to simulate SUBTRACT operation on two user-entered numbers.

5. Write an assembly program to simulate the following logical operations on two user-entered numbers.

1. AND 2. OR 3. NOT 4. XOR 5. NOR 6. NAND

6. Write an assembly program to simulate MULTIPLY operation on two user-entered numbers.

7. Write an assembly program for simulating following memory-reference instructions.

1. ADD 2. LDA 3. STA 4. BUN 5. ISZ

8. Write an assembly language program to simulate the machine for following register reference instructions and determine the contents of AC, E, PC, AR and IR registers in decimal after the execution: 1. CLA 2. CMA 3. CME 4. HLT

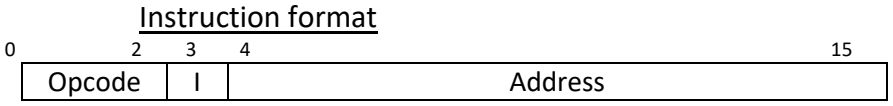
9. Write an assembly language program to simulate the machine for following register reference instructions and determine the contents of AC, E, PC, AR and IR registers in decimal after the execution: 1. INC 2. SPA 3. SNA 4. SZE

10. Write an assembly language program to simulate the machine for following register reference instructions and determine the contents of AC, E, PC, AR and IR registers in decimal after the execution: 1. CIR 2. CIL

11. Write an assembly program that reads in integers and adds them together; until a negative non-zero number is read in. Then it outputs the sum (not including the last number).

12. Write an assembly program that reads in integers and adds them together; until zero is read in. Then it outputs the sum.

13. Create a machine for the following instruction format:



The instruction format contains a 3-bit opcode, a 1-bit addressing mode and a 12-bit address. Write an assembly program to simulate the machine for addition of two numbers with I= 0 (Direct Address) and address part = 082. The instruction to be stored at address 022 in RAM, initialize the memory word with any decimal value at address 082. Determine the contents of AC, DR, PC, AR and IR in decimal after the execution.

14. Simulate the machine for the memory-reference instruction referred in above question with I= 1 (Indirect Address) and address part = 082. The instruction to be stored at address 026 in RAM. Initialize the memory word at address 082 with the value 298. Initialize the memory word at address

298 with operand 632 and AC with 937. Determine the contents of AC, DR, PC, AR and IR in decimal after the execution.

15. The instruction format contains 3 bits of opcode, 12 bits for address and 1 bit for addressing mode. There are only two addressing modes, I = 0 is direct addressing and I = 1 is indirect addressing. Write an assembly program to check the I bit to determine the addressing mode and then jump accordingly.