

BLG222E - Computer Organization

Project 3

Part 1:

Add hardwired control unit to the simple computer that you have designed in Project #2.

- Implement both regular instruction cycle and interrupt cycle as shown in Figure 1

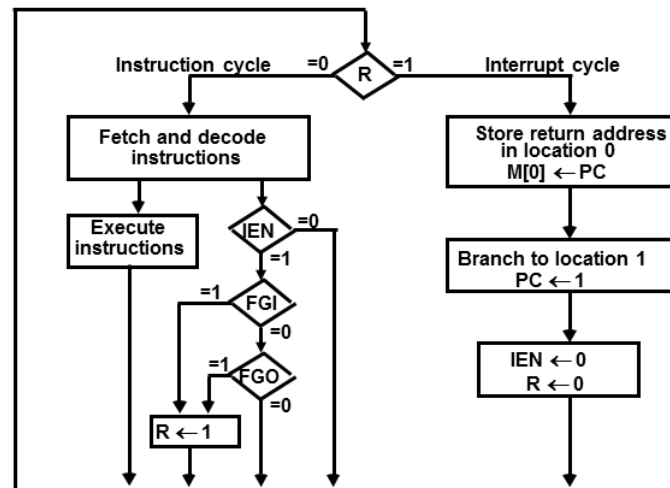
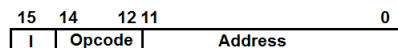


Figure 1: Fetch, decode and execute cycles of simple computer with interrupt.

- Make sure that you use the same instruction format of the textbook that is given in Figure 2.

Memory-Reference Instructions (OP-code = 000 ~ 110)



Register-Reference Instructions (OP-code = 111, I = 0)



Input-Output Instructions (OP-code = 111, I = 1)

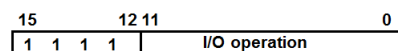


Figure 2: Instruction format.

- Your control unit should implement all instructions listed in Table 1

Symbol	Hex Code		Description
	I = 0	I = 1	
AND	0xxx	8xxx	AND memory word to AC
ADD	1xxx	9xxx	Add memory word to AC
LDA	2xxx	Axxx	Load AC from memory
STA	3xxx	Bxxx	Store content of AC into memory
BUN	4xxx	Cxxx	Branch unconditionally
BSA	5xxx	Dxxx	Branch and save return address
ISZ	6xxx	Exxx	Increment and skip if zero
CLA	7800		Clear AC
CLE	7400		Clear E
CMA	7200		Complement AC
CME	7100		Complement E
CIR	7080		Circulate right AC and E
CIL	7040		Circulate left AC and E
INC	7020		Increment AC
SPA	7010		Skip next instr. if AC is positive
SNA	7008		Skip next instr. if AC is negative
SZA	7004		Skip next instr. if AC is zero
SZE	7002		Skip next instr. if E is zero
HLT	7001		Halt computer
INP	F800		Input character to AC
OUT	F400		Output character from AC
SKI	F200		Skip on input flag
SKO	F100		Skip on output flag
ION	F080		Interrupt on
IOF	F040		Interrupt off

Table 1: Instruction set.

Part 2:

- Read Chapter #6 (“Programming the Basic Computer”) from your textbook.
- Write a program in assembly using instruction set given in Figure 1 that performs bitwise “and” operation on 10 numbers that are stored in memory between addresses 0x800 – 0x809 and write the result to the address 0x80A.
- Convert your program into hexadecimal machine codes manually. Enter the machine codes to your RAM. The program should reside at address 0x100.
- Initialize your PC to 0x100. Enter the following values to your RAM.

Address	Data
0x800	0xCA7D
0x801	0xCC88
0x802	0xA375
0x803	0x3B6F
0x804	0xCB51
0x805	0x3663
0x806	0xC929
0x807	0x3822
0x808	0x9EFA
0x809	0x5AE3

Execute your program and check your result that is at address 0x80A.

Groupwork:

Group work is expected for this project. Same group (from the previous project) of students should design together. You might be asked to make a 15-minute demonstration of your design with a few test cases.

What to turn in:

Implement your design for the register and simple computer in **logisim** software, upload a single compressed (zip or rar) file to ninova before the deadline. Only one student from each group should submit the project file. This compressed file should contain:

- the student number&names of the students in the group
- design (.circ) file for the simple computer with interrupt and hardwired control unit.
- assembly program that is written for part 2.
- machine code of your program exported from logisim software. You can do this by right clicking on the RAM and selecting “Save Image” from the context menu.
- a short report that lists of control inputs and corresponding functions of the simple computer