

HCMUT ECE 120 Discussion Section 9: LC-3 Instructions

In this discussion, you will be given a sequence of binary words that correspond to LC-3 instructions and you will be asked to convert each binary word to a corresponding LC-3 instruction. You will then explain the function performed by the sequence of the given instructions.

You should read Chapter 4 and §5.1-5.2 from Patt & Patel before attending this discussion.

LC-3 Instruction Set

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ADD ⁺	0001				DR			SR1		0	00			SR2			DR ← SR1 + SR2; set NZP
ADD ⁺	0001				DR			SR1		1				imm5			DR ← SR1 + SEXT(imm5); set NZP
AND ⁺	0101				DR			SR1		0	00			SR2			DR ← SR1 AND SR2; set NZP
AND ⁺	0101				DR			SR1		1				imm5			DR ← SR1 AND SEXT(imm5); set NZP
BR	0000				n		z	p									IF ((n·N)+(z·Z)+(p·P)) THEN PC ← PC + SEXT(PCoffset9)
JMP	1100				000			BaseR						000000			PC ← BaseR
JSR	0100				1												R7 ← PC PC ← PC + SEXT(PCoffset11)
JSRR	0100				0		00	BaseR						000000			R7 ← PC PC ← BaseR
LD ⁺	0010				DR												DR ← M[PC + SEXT(PCoffset9)]; Set NZP
LDI ⁺	1010				DR												DR ← M[M[PC + SEXT(PCoffset9)]]; Set NZP
LDR ⁺	0110				DR			BaseR						offset6			DR ← M[BaseR + SEXT(offset6)]; Set NZP
LEA ⁺	1110				DR												DR ← PC + SEXT(PCoffset9); Set NZP
NOT ⁺	1001				DR			SR						111111			DR ← NOT(SR); Set NZP
RET	1100				000			111						000000			PC ← R7
RTI	1000																IF (PSR[15]==0) THEN PC ← M[R6]; R6 ← R6 + 1; TEMP ← M[R6]; R6 ← R6 + 1; PSR ← TEMP
ST	0011				SR												M[PC + SEXT(PCoffset9)] ← SR
STI	1011				SR												M[M[PC + SEXT(PCoffset9)]] ← SR
STR	0111				SR			BaseR						offset6			M[BaseR + SEXT(offset6)] ← SR
TRAP	1111				0000												R7 ← PC PC ← M[ZEXT(trapvect8)]

superscript "+" denotes instructions that update the condition bits NZP

1. LC-3 Instructions

Shown below is a snapshot of a portion of the contents of the LC-3 memory for addresses x3000-x3004. These addresses contain a short program. The 16-bit addresses of, and data in, the RAM are encoded in hexadecimal representation. In this discussion, you will interpret the contents of the RAM, trace the program, and determine its functionality.

address	data
x3000	x927F
x3001	x5842
x3002	x96FF
x3003	x5903
x3004	x993F

- 1) Translate the contents of the RAM into its LC-3 instructions and write them in RTL notation. A copy of the encoding for the LC-3 instruction set appears on page 1. The first one has been done for you as an example.

Address	Binary Instruction (translate from hex above)	LC-3 Instruction	RTL (be specific to this instruction)
x3000	1001 0010 0111 1111	NOT R1, R1	$R1 \leftarrow \text{NOT } R1$
x3001			
x3002			
x3003			
x3004			

- 2) Examine the sequence of your instructions. What function does the program perform? Your description should explain the high level behavior of the program in a single sentence and should not be a step-by-step description of what the program did. For example, "First the program adds R1 to R2 and stores it into R3..." is unacceptable.