CS/ECE 252: INTRODUCTION TO COMPUTER ENGINEERING
COMPUTER SCIENCES DEPARTMENT
UNIVERSITY OF WISCONSIN-MADISON

Prof. Mark D. Hill
TAs Marc de Kruijf & Sanghamitra Roy

Midterm Examination 4
In Class (50 minutes)
Wednesday, May 9, 2007
Weight: 15%

CLOSED BOOK, NOTE, CALCULATOR, PHONE, & COMPUTER.

The exam in two-sided and has **NINE** pages, including two blank pages and a copy of the ***LC-3 Instruction Set handout*** on the final page (please feel free to detach this final page, but insert it into your exam when you turn it in).

Plan your time carefully, since some problems are longer than others.

NAME: _____

ID# _____

Problem Number	Maximum Points	Points Awarded
1	5	SR
2	4	SR
3	4	SR
4	8	MK
5	4	SR
6	3	MK
7	2	MK
Total	30	

Problem 1 (5 points)

An assembly language LC-3 program is given below:

```
        .ORIG   x3003
        LEA     R1, DATA
        LDR     R2, R1, #0
LOOP    ADD     R2, R2, #-3
        BRzp    LOOP
        HALT

DATA    .FILL   x000C
        .END
```

a. Create a symbol table for the program:

Symbol	Address
LOOP	**x3005**
DATA	**x3008**

b. How many times will the instruction at the memory address labeled LOOP execute?

5 times (R2 = 12, 9, 6, 3, 0)

Problem 2 (4 points)

a. What is the purpose of the HALT statement?

A HALT instruction stops the execution of the program and returns to the OS.

b. Is it meaningful to have more than one HALT statement in a single-file LC-3 program? Explain.

Yes. A program may have more than one logical exit.

c. What is the purpose of .END pseudo-op?

The .END pseudo-op tells the assembler where the program ends. Any string that occurs after that will be disregarded and not processed by the assembler.

d. Is it meaningful to have more than one .END in a single-file LC-3 program? Explain.

No. Any .END after the first .END will never be processed by the assembler.

Problem 3 (4 points)

Regarding the assigned reading "RFID Inside" on RFID implants:

a. Give two different potential benefits of RFID implants.

I. To be used as a life saving device in an emergency

II. To be used as a source of authentication for security

b. Give two different potential drawbacks of RFID implants.

I. Invasion of employee's privacy

II. An employee should have the right to bodily integrity

c. In what way was Wisconsin mentioned in the article?

Wisconsin passed a bill in May 2006, to prohibit requiring anyone to have a microchip implanted.

Problem 4 (8 points)

The following program calculates the sum of absolute values of two numbers and stores the sum in R4. The subroutine at the label "ABS" finds the absolute value of the argument.

```
      .ORIG  x3000     ; Instructions start at x3000;
      AND R4, R4, #0   ; Clearing R4
      LD  R1, VAL1
      LD  R2, VAL2

      ADD R0, R1, #0   ; Prepare argument VAL1 (fill)
      JSR ABS          ; Call subroutine ABS
      ADD R4, R0, #0   ; Add Abs(VAL1) to R4
      ADD R0, R2, #0   ; Prepare argument VAL2 (fill)
      JSR ABS          ; Call subroutine ABS
      ADD R4, R4, R0   ; Add Abs(VAL2) to R4
      HALT

                       ; Argument passed in register R0 (fill)
ABS   ST  R4, SaveR    ; Save register R4 (fill)
      ADD R0, R0, #0   ; Set condition code based on R0
      BRzp     NEXT
      NOT R4, R0
      ADD R0, R4, #1
NEXT  LD  R4, SaveR    ; Restore register R4 (fill)
      RET              ; Value is returned in register R0 (fill)

                       ; Values
SaveR .FILL    x0000
VAL1  .FILL    x0005   ;  5
VAL2  .FILL    xFFFB   ; -5
      .END
```

a. Fill in the blanks in the above program at all places indicated by "(**fill**)".

See above.

b. What is the value in register R4 at the end of program execution?

R4 contains 0x000A in hexadecimal, or '10' in decimal.

Problem 5 (4 points)

An LC-3 assembly language program is given below. Carefully read the program and answer the questions that follow. Adding comments will help in partial credit.

Label	**Assembly language instruction**
START	LDI R1, KBSR ; **Test for character input**
	BRzp START ;
	LDI R0, KBDR ;
LOOP	LDI R1, DSR ; **Test output register ready**
	BRzp LOOP ;
	STI R0, DDR ;
	HALT ;
KBSR	.FILL xFE00 ; **Address of KBSR**
KBDR	.FILL xFE02 ; **Address of KBDR**
DSR	.FILL xFE04 ; **Address of DSR**
DDR	.FILL xFE06 ; **Address of DDR**
	.END

a. What does this program do?

The program accepts a character typed at the keyboard and displays the same character on the monitor.

b. What is the purpose of the KBSR?

Bit 15 of the KBSR (keyboard status register) controls the synchronization of the slow keyboard and the fast processor. When a key on the keyboard is struck the ASCII code for that key is loaded into KBDR[7:0] and the electronic circuits associated with the keyboard automatically set KBDR[15] to 1. When the LC-3 reads KBDR, KBSR[15] is automatically cleared allowing another key to be struck. If KBSR[15] = 1, the keyboard is disabled.

Problem 6 (3 points)

a. What does the JSR instruction do? How does it differ from a JMP instruction?

The JSR instruction stores the next PC value in R7 and then jumps to a subroutine via a PC-relative offset. The JMP instruction differs in that it does not store the next PC, and also in that it uses the base + offset addressing mode rather than PC-relative addressing.

b. Why must a RET instruction be used to return from a TRAP routine? Why won't a BR (Unconditional branch) instruction work instead?

TRAP routines need to be able to return to the instruction after the TRAP initiation. The location of this instruction will differ between TRAP instances, and could be anywhere. The RET instruction solves this problem by using the address stored in R7, which is the next PC address that was saved when the TRAP occurred. The BR instruction will always jump to the same PC-relative address, which cannot work in the general case. Also note that the RET instruction is base + offset and the BR instruction is PC-relative, so the BR instruction might have insufficient reach (partial credit answer).

Problem 7 (2 points)

In lecture, we discussed implementing a program denoted Halt(P,I).

a. What is Halt(P,I) supposed to do?

Halt(P,I) analyzes a program P running on input I to determine whether (i) P running on I halts (or terminates) or (ii) P running on I runs forever.

b. What are the alternative answers Halt(P,I) may return?

Halt(P) returns either "halts" (case i) or "does not halt" (case ii).

Scratch Sheet 1 (in case you need additional space for some of your answers)

Scratch Sheet 2 (in case you need additional space for some of your answers)

LC-3 Instruction Set (Entered by Mark D. Hill on 03/14/2007; last update 03/15/2007)

PC': incremented PC. setcc(): set condition codes N, Z, and P. mem[A]:memory contents at address A.
SEXT(immediate): sign-extend immediate to 16 bits. ZEXT(immediate): zero-extend immediate to 16 bits.
Page 2 has an ASCII character table.