

Changes to LC3

Fall 2006

CIT 593

C - 1

Changes in LC3 Simulator/ISA

- **Multiply instruction**
 - Used reserved opcode
 - MUL R#, R#, R# or MUL R#, R#, #
- **Subtraction instruction**
 - SUB R#, R#, R# is psuedo instruction
 - It is equivalent to 2 LC3 instructions
 - There is no immediate i.e. SUB R#, R#, #
 - because we can do ADD R#, R#, # -

CIT 593

C - 2

RET vs. RTT

- **RET**

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RET	1	1	0	0	0	0	0	1	1	1	0	0	0	0	0	0
	JUMP							R7								

- Return from subroutine
- Stores PC + 1 in R7, however does not unset the privilege bit
- i.e. PSR[15] remains 1 (Supervisor mode stays on)
- This allows user to access memory locations that it should have access to.

CIT 593

C - 3

RET vs. RTT (contd..)

- **RTT**

- Similar to RET
- Except that it clears PSR[15]
- That is brings the system into user mode.
- Use especially for TRAP returns

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RTT	1	1	0	0	0	0	0	1	1	1	0	0	0	0	0	1

CIT 593

C - 4

Changes to lc3os.asm

- lc3os.asm uses RET for TRAP routine return
- lc3os2.asm uses RTT for TRAP routine return
 - However the IN trap routine does not work correctly as it calls the other trap routines with RTT. Why??

CIT 593

C - 5

IN Routine

```

TRAP_IN
    ST R7,OS_IN_SAVE_R7 ; save R7 (no need to save R0, since we
                        ; overwrite later
    LEA R0,TRAP_IN_MSG ; prompt for input
    PUTS
    GETC                ; read a character
    OUT                 ; echo back to monitor
    ST R0,OS_SAVE_R0   ; save the character
    AND R0,R0,#0       ; write a linefeed, too
    ADD R0,R0,#10
    OUT
    LD R0,OS_SAVE_R0   ; restore the character
    LD R7,OS_IN_SAVE_R7 ; restore R7
    RTT
    
```

Why will this not work?

```

GETC
    LDI R0,OS_KBSR
    ; wait for a keystroke
    BRzp TRAP_GETC
    LDI R0,OS_KBDR
    ; read it and return
    RTT
    
```

CIT 593

C - 6

PUTS is rewritten

```

TRAP_PUTS
    ST R0,OS_SAVE_R0
    ST R1,OS_SAVE_R1
    ST R7,OS_SAVE_R7
    ADD R1,R0,#0

TRAP_PUTS_LOOP
    LDR R1, R0, #0
    BRz TRAP_PUTS_DONE
TRAP_PUTS_WAIT
    LDI R3, OS_DSR
    BRzp TRAP_PUTS_LOOP
    STI R1, OS_DDR
    ADD R0, R0, #1
    BRnzp TRAP_PUTS_LOOP

TRAP_PUTS_DONE
    LD R0,OS_SAVE_R0
    LD R1,OS_SAVE_R1
    LD R7,OS_SAVE_R7
    RET

TRAP_PUTS
    ST R0,OS_SAVE_R0
    ST R1,OS_SAVE_R1
    ST R7,OS_SAVE_R7
    ST R3,OS_SAVE_R3

TRAP_PUTS_LOOP
    LDR R1, R0, #0
    BRz TRAP_PUTS_DONE
TRAP_PUTS_WAIT
    LDI R3, OS_DSR
    BRzp TRAP_PUTS_LOOP
    STI R1, OS_DDR
    ADD R0, R0, #1
    BRnzp TRAP_PUTS_LOOP

TRAP_PUTS_DONE
    LD R0,OS_SAVE_R0
    LD R1,OS_SAVE_R1
    LD R7,OS_SAVE_R7
    LD R3,OS_SAVE_R3
    RTT
    
```

hw4 uses PUTS, and hence rewritten

CIT 593

C - 7