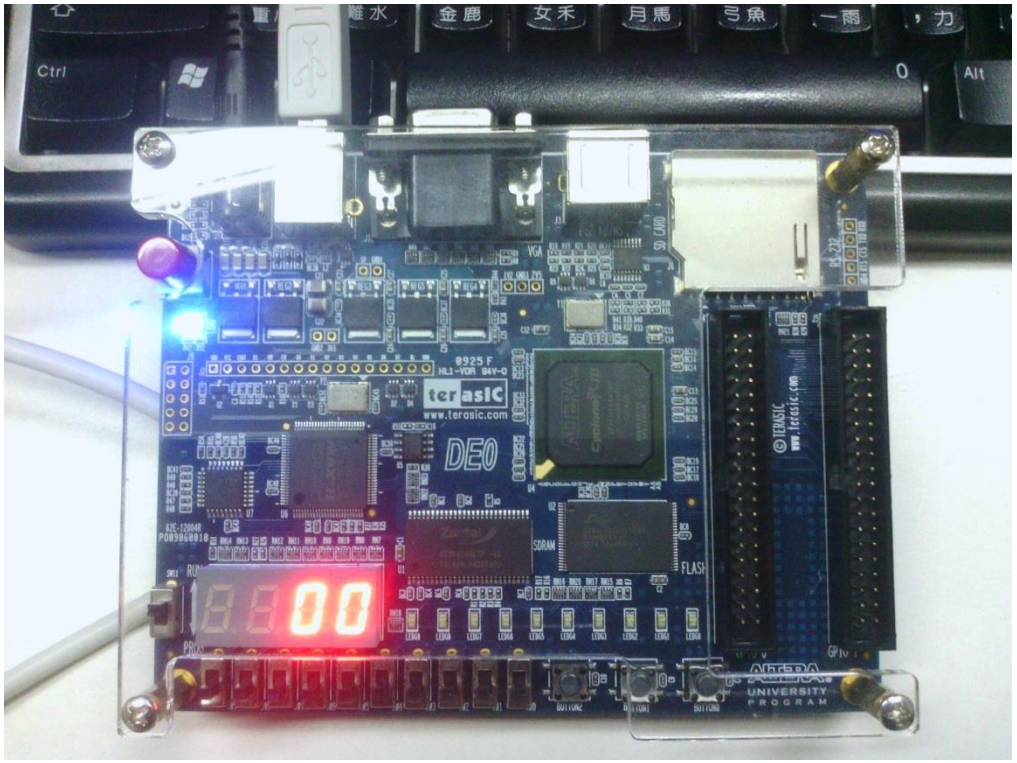
Reading Switches, Writing LEDs

黃永廣
助教：劉柏廷、郭柏興、林嘉偉、陳昶宏
雲科電子
2014/9/29~10/3

Introduction of Lab Resources

- NO FOOD, NO DRINKS in lab
- Violators will fail the course
- LC3-Editor
- LC3 Simulator
- DE0 FPGA board from Altera/Terasic, similar equipment is used by Ivy League universities such as Columbia U. and U. of Michigan
- A LC3 soft core is extended to be used with DE0
- Quartus II Ver. 9 (Web Edition) from Altera

DE0 FPGA board from Altera/Terasic



Input Output Devices

- Memory-Mapped IO: Each device is asssigned a memory address
- Green LEDs
- Switches
- 7-segments
- Pushbuttons: Push Button0 (rightmost) to reset

Reading Switches, Writing LEDs

- .ORIG x3000 ; file = “lab1a.asm”
- LD R1, Saddr ; R1 ←m[Saddr] = 0xFFFC = addr of SWs
- LDR R3, R1, #0 ; R3 ←m[0xFFFC+0] ; read switches
- LD R2, Laddr ; R2 ←m[Laddr] = 0xFFFD = addr of LEDs
- STR R3, R2, #0 ; m[0xFFFD+0] ← R3 = switches; write LEDs

- Saddr .FILL 0xFFFC ; Memory address for switches
- Laddr .FILL 0xFFFD ; Memory address for LEDs
- .END

Task 1: I/O with Switches and LED

- Input a number from switches SW3 to SW0
  - Read value from SW3_0, show number on LED3_0
- Signed by TA when task is done

Task 2:
Pseudo Code: Sum ← 1+2+…+10

- Sum ← 0;
- Num ← 1;
- Count ← 10;
- While (Count != 0) {
- Sum ← Sum + Num;
- Num ← Num+1;
- Count ← Count-1;
- }

LC3 Code: $Sum_N \leftarrow 1+2+\ldots+N$

```
    .ORIG x3000
    AND R0,R0,#0          [addr=x3000]
    ADD R1,R0,#1          [addr=x3001]
    LD  R2,N              [addr=x3002]
Loop  ADD R0,R0,R1        [addr=x3003]
    ADD R1,R1,#1          [addr=x3004]
    ADD R2,R2,#-1         [addr=x3005]
    BRp Loop              [addr=x3006]
Bye ST R0, Sum            [addr=x3007]
Forever BRnzp Forever     [addr=x3008]
N     .FILL #3 ; Memory for N [addr=x3009]
Sum .FILL #0 ; Store sum  [addr=x300A]
    .END
```

Learn to Use LC3 Simulator
(all numbers expressed in hexadecimal)

PC	R0	R1	R2	R3	R4	R5	R6	R7	R8	addr	Updated m[addr]
3000	0										
3001		1									
3002											
3003											
3004											
3005											
3006											
3001											
…											

Task 2: I/O with Switches and LED

- Input a number N with switches SW7 to SW0
  - Read SW7_0, store into memory location N
- Compute Sum_N = 1+2+..+N and store the result into memory location Sum
- Show Sum on LED9_0
- Write down your assembly code with comments on the lab worksheet
- Compute Sum_N, where N=40 to 45 and record each sum shown on LED9_0 on a table on the worksheet
- Report any unusual observations and explain why they occur

Result Table

N	40	41	42	43	44	45
Sum_N						

Program Steps for Task 2

- Read the value of N from switches
- Save the value into N
- Compute $Sum_N = 1+2+\ldots+N$ with the given program
- Write Sum_N to LEDs

Task 3: Find Remainder

- Input two numbers, N with switches SW7_0 and M as a number in memory (e.g., M: .FILL #10)
- Find the remainder after N is divided by M
- Store the remainder into memory at Result
- Then show the remainder in led LED7_0

- E.g., if N=63, M=10, then Remainder = 3;
- if N=38, M=10, then Remainder = 8;
- if N=63, M=75, then Remainder = 63;

The Remainder after Dividing N by M

- Loop:
- D := N – M
- If D<0, then Goto EndLoop //BRn EndLoop
- N := N – M //Else D >= 0
- Goto Loop
- EndLoop: Result := N;
- Forever: Goto Forever

How to Subtract?

- 2's complement
- Suppose register R1 = M, where M is an integer
- How to get R1 := -M and R2 := R2-M
  - NOT R1,R1 ; means R1 := -M - 1
  - ADD R1,R1,#1 ; means R1 := -M
  - ADD R2,R2,R1 ; means R2 := R2 - M