Sitthiphol Yuwanaboon

John O'Neal

CS 130

March 9 2021

Readings: Finish 2.2 after PA 2.2.9;

section 2.3;

and section 4.1

Do all of the PA's in section 2.2 after PA 2.2.9 (*25 pts.*);

section 2.3; and section 4.1

NOTE: I'll download these from *zybooks for grading*.

2.1 [5] <COD §2.2> For the following C statement, write the corresponding LEGv8 assembly code. Assume that the C variables `f`, `g`, and `h`, have already been placed in registers `X0`, `X1`, and `X2` respectively. Use a minimal number of LEGv8 assembly instructions.

```
f = g + (h - 5);
```

2.2 [5] <COD §2.2> Write a single C statement that corresponds to the two LEGv8 assembly instructions below.

```
ADD f, g, h
ADD f, i, f
```

2.3 [5] <COD §§2.2, 2.3> For the following C statement, write the corresponding LEGv8 assembly code. Assume that the variables `f`, `g`, `h`, `i`, and `j` are assigned to registers `X0`, `X1`, `X2`, `X3`, and `X4`, respectively. Assume that the base address of the

arrays `A` and `B` are in registers `X6` and `X7`, respectively.

```
B[8] = A[i - j];
```

2.4 [10] <COD §§2.2, 2.3> For the LEGv8 assembly instructions below, what is the corresponding C statement? Assume that the variables `f`, `g`, `h`, `i`, and `j` are assigned to registers `X0`, `X1`, `X2`, `X3`, and `X4`, respectively. Assume that the base address of the arrays `A`and `B` are in registers `X6` and `X7`, respectively.

```
LSL X9, X0, #3        // X9 = f*8
ADD X9, X6, X9        // X9 = &A[f]
LSL X10, X1, #3       // X10 = g*8
ADD X10, X7, X10      // X10 = &B[g]
LDUR X0, [X9, #0]     // f = A[f]

ADDI X11, X9, #8
LDUR X9, [X11, #0]
ADD X9, X9, X0
STUR X9, [X10, #0]
```

III. Define instruction fetch, data fetch, instruction execution cycle. Give an example from a load instruction from a Z80 load instruction. (*10 pts.*)

1. Instruction Fetch
   The fetch step is the same for each instruction: The CPU sends the contents of the PC to the MAR and sends a read command on the address bus. In response to the read command (with address equal to PC), the memory returns the data stored at the memory location indicated by the PC on the data bus.
2. The instruction cycles
   he cycle that the central processing unit (CPU) follows from boot-up until the computer has shut down in order to process instructions
3. Data Fetch
   Fetch is the retrieval of data by a software program, script, or hardware device. After being retrieved, the data is moved to an alternate location or displayed on a screen

Sample of instruction from a Z80 load instruction

LD A, 5
This loads the **A** register with the number 5

LD D, B
This transfer content of the **B** register to the **D** register

Sitthiphol Yuwanaboon
CS 139

2.1) f is placed in X0

g is placed in X1

h is placed in X2

$f = g + (h - 5)$

X_3 a temporary register

SUB X_3, X_2, #5

ADD X_0, X_1, X_3

evaluate $g + (h - 5)$ and store in X_0 register

2.2 ADD f, g, h → $f = g + h$

ADD f, i, f ⇒ $f = i + f$ (previous instruction)

single C statement

$f = i + g + h$

2.3)

SUB X9, X3, X4 temp register get (i-j)

LSL X9, X9, #3 X9 = (i-j)*8

ADD X9, X9, X6 X9 = X9 + X6

LDUR X10, [X9, #0] X10 = A[i-j]

STUR X10, [X7, #64] store B[7]

= A[i-j] into B[7] offset 7*8
= 64

2.4)

LSL → shift left operator

LSL X9, X0, #3 shift 3 bits

ADD X9, X6, X9 Adding above multiplied value to base address

LSL X10, X1, #3 Shifting variable g by 3 bits

ADD X10, X7, X10 Adding the multiplied value to the base address of B, meaning accessing address of B[g]

LDUR X0, [X9, #0] copied data stored in mem[X9+0] to X0

ADDI X11, X9, #8, Storing data at A[F] into A[F] to A[F+1]

LDUR X9, [X11, #0] copied the data stored at mem[X11+0] into X9

ADD X9, X9, X0 Adding A[F] + A[F+1]

STUR X9, [X10, #0] copied data stored in X9 to Mem[X10, #0]

Store result into B[g]

So the ending statement

B[g] = A[F] + A[F+1]