CG2028 COMPUTER ORGANIZATION

TUTORIAL 1: MICROPROCESSOR CONCEPTS

WEEK 8, GU JING

Q1: Given a binary pattern in some memory location, is it possible to tell whether this pattern represents a machine instruction or a number?

Q1: Given a binary pattern in some memory location, is it possible to tell whether this pattern represents a machine instruction or a number?

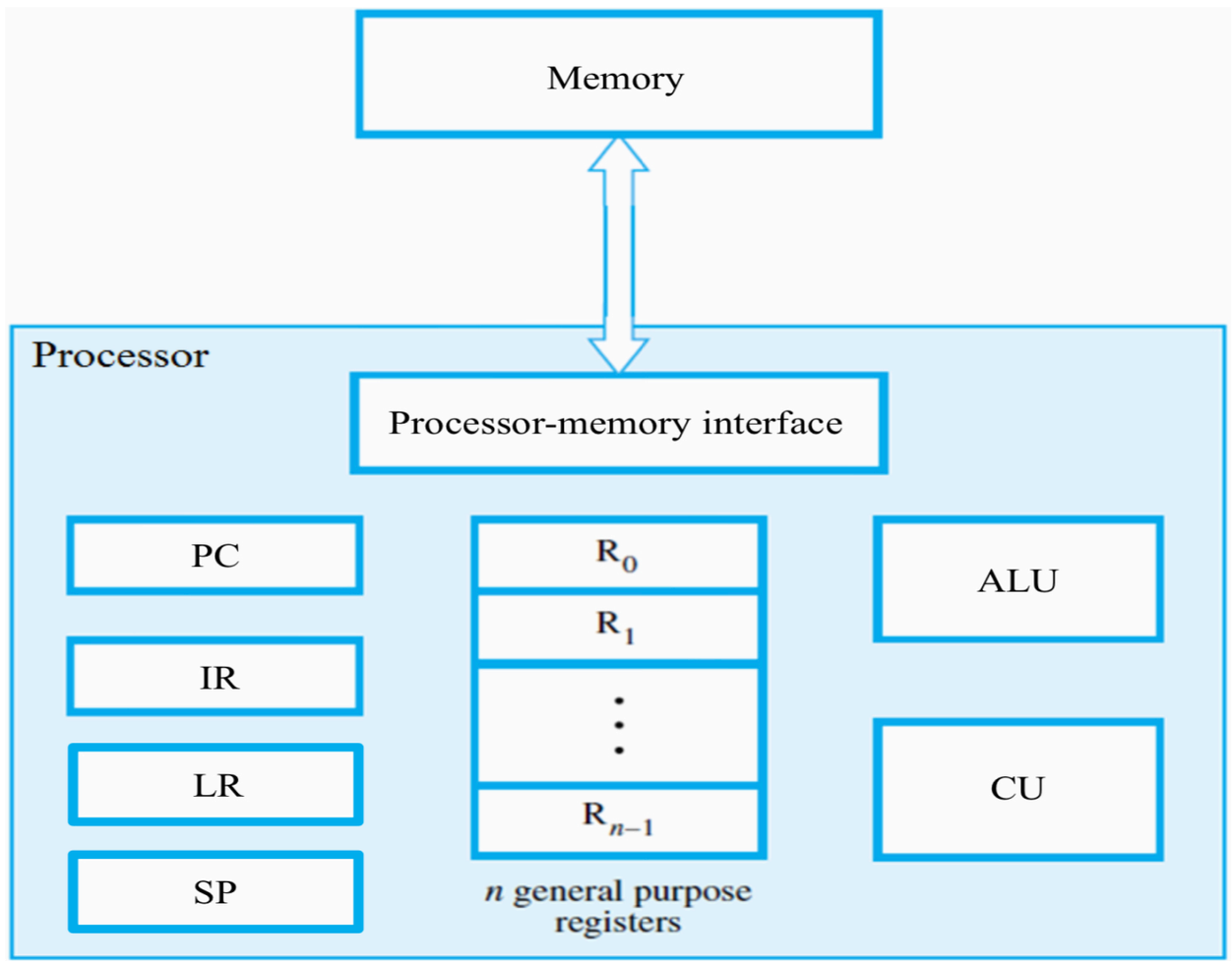
Ans: No, any binary pattern can be interpreted as a number or as an instruction.

Then how PC knows whether it is an instruction or a number?
Ans: by its address.

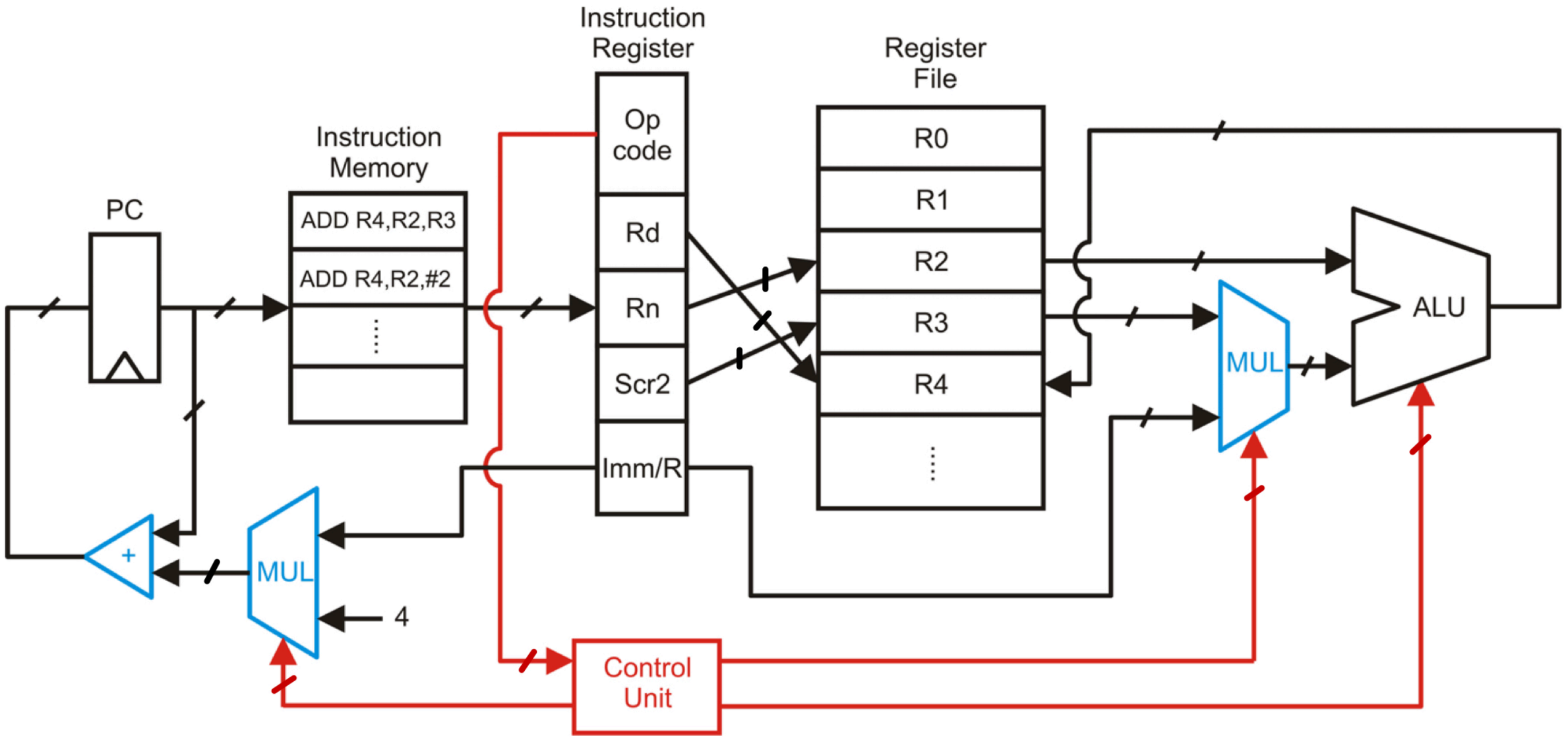
Q2: List the steps needed to execute the machine instruction

Add R4, R2, R3

in terms of the transfers between components shown in the figure on slide 13 of Lecture 1 and simple control commands. Assume that the address of the memory location containing this instruction is initially in register PC.



Q2: Add R4, R2, R3



* ***Incomplete circuit of ARM processor where memory instructions (e.g. LDR/STR) are not designed.***
* ***More details about hardware design of ARM processor are covered in CG3207.***

Q2: Add R4, R2, R3

Ans:

1. Send the address of the instruction word from register PC to the memory circuits and issue a Read control command;
2. Wait until the requested word has been retrieved from the memory, then load it into register IR, where it is interpreted (decoded) by the control circuitry to determine the operation to be performed;
3. Increment the contents of register PC to point to the next instruction in memory;
4. Send the contents of register R2 and R3 to the ALU and issue an ADD command to the ALU;
5. Send the sum from the output of the ALU to register R4.

Q3: A program to add a number stored at location A to another number stored at location B on a RISC computer is given by:

```
Load R2, A
Load R3, B
Add R4, R2, R3
```

Explain why the program cannot simply be written as:

```
Add R4, A, B
```

Q3: A program to add a number stored at location A to another number stored at location B on a RISC computer is given by:

Load R2, A
Load R3, B
Add R4, R2, R3

Explain why the program cannot simply be written as:

Add R4, A, B

Ans: In a RISC computer, (1) each instruction must occupy only a single word and (2) only Load and Store instructions are used to access memory operands.
Hence, the operands for arithmetic/logic instructions must be in registers, or one of them may be given explicitly in instruction word.

Q4: A program comprising of several machine (binary) encoded instructions are stored in main memory at the locations shown. In order to repeat the execution of the sequence of instructions found in locations 0x00000100 to 0x0000010C when a certain condition is satisfied, a conditional branch instruction (at 0x00000110) is used in the program.

What is the effect of the conditional branch instruction on the contents of the Program Counter (PC) register in the processor when the condition is satisfied?

Processor	
PC	
IR	
:	
R2	
R3	
R4	
:	

Main Memory	
0x00000100	Load R2, A
0x00000104	Load R3, B
0x00000108	Add R4, R2, R3
0x0000010C	Store R4, C
0x00000110	Conditional branch
0x00000114	:

Q4:

Processor	
PC	
IR	
:	
R2	
R3	
R4	
:	

Main Memory	
0x00000100	Load R2, A
0x00000104	Load R3, B
0x00000108	Add R4, R2, R3
0x0000010C	Store R4, C
0x00000110	Conditional branch
0x00000114	:

Ans:

Before the conditional branch instruction is executed, the content of the PC is the address of the next instruction, i.e. 0x00000114.

When the conditional branch instruction is executed and the condition is satisfied, the content of the PC is replaced with 0x00000100. This causes the “Load R2, A” instruction to be fetched and executed (again). The PC is automatically incremented and the subsequent instructions will also be executed (again).

Q5: Numbers are stored in the registers and main memory of a computer in two's complement representation. If the content of R2 is 0xFFFFFFF8 and the content of R3 is 0x00000007, what are

(i) the content of R4, and
(ii) the status of the N, Z, V and C condition code flags

after the "Add R4, R2, R3" instruction is executed?

Q5: Explanation of V (Overflow) flag:

Q5: Numbers are stored in the registers and main memory of a computer in two's complement representation. If the content of R2 is 0xFFFFFFF8 and the content of R3 is 0x00000007, what are

(i) the content of R4, and
(ii) the status of the N, Z, V and C condition code flags

after the "Add R4, R2, R3" instruction is executed?

Ans:

0xFFFFFFF8 is -8 in decimal. 0x00000007 is +7 in decimal. Adding these two values results in 0xFFFFFFFF = -1 in decimal (result is in R4). So the N flag is set to 1 by the processor, and Z, V and C condition code flags are 0.

Revision of Condition Flags: N, Z, C, V

N (Negative)	**Set if MSB is 1**
Z (Zero)	**Set if all bits are 0**
C (Carry)	**Set if there is a Carry from MSB**
V (Overflow)	**Set if both MSB are the same and the resul'ts MSB flips:** **1. $MSB_1 = 0, MSB_2 = 0, MSB_R = 1$: (+ve) + (+ve) = (-ve)** **2. $MSB_1 = 1, MSB_2 = 1, MSB_R = 0$: (-ve) + (-ve) = (+ve)**

Example: how the flags change when adding 0xFFFFFFFF and 0x00000003?

N	Z	C	V

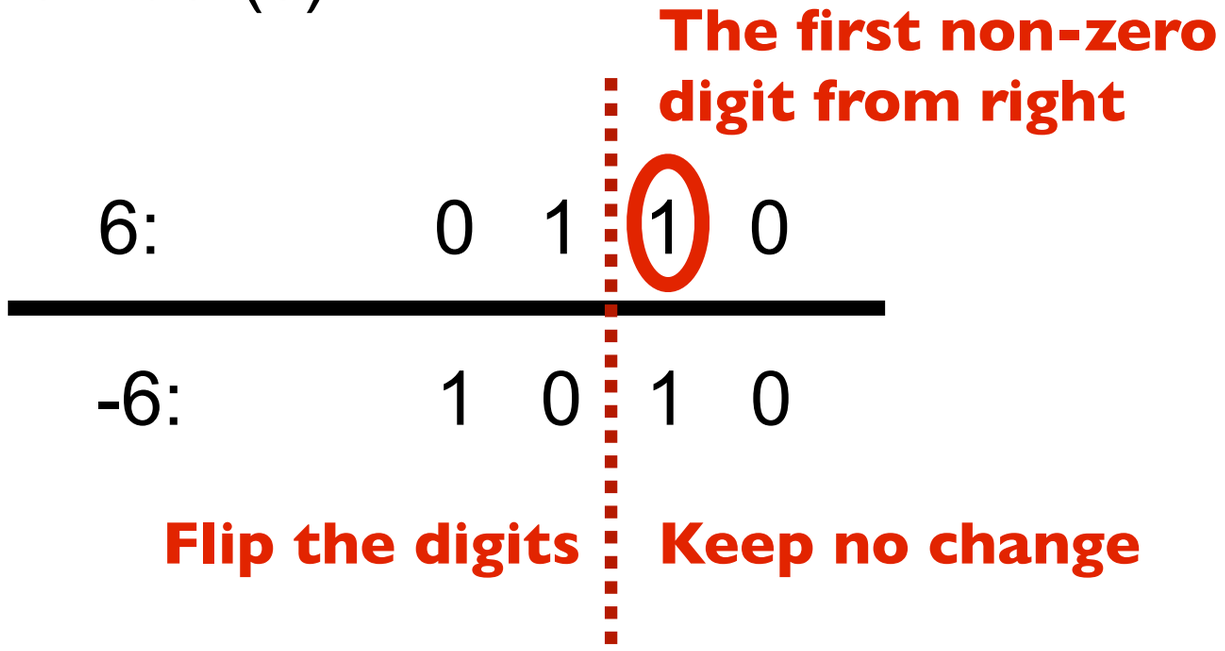
Revision of Two's Complement Conversion

Example: Convert signed number 6 (0110) to -6:

Method 1: Flip and +1

0 1 1 0 → 1 0 0 1 → 1 0 1 0

Method 2: Keep the right side number(s) from the first non-zero digit, flip the remaining left side number(s)



THE END

For Consultation: eleguji@nus.edu.sg
Office: E4-03-10