DIGITAL DESIGN AND COMPUTER ORGANIZATION

Dr. Reetinder Sidhu
Department of Computer Science and Engineering

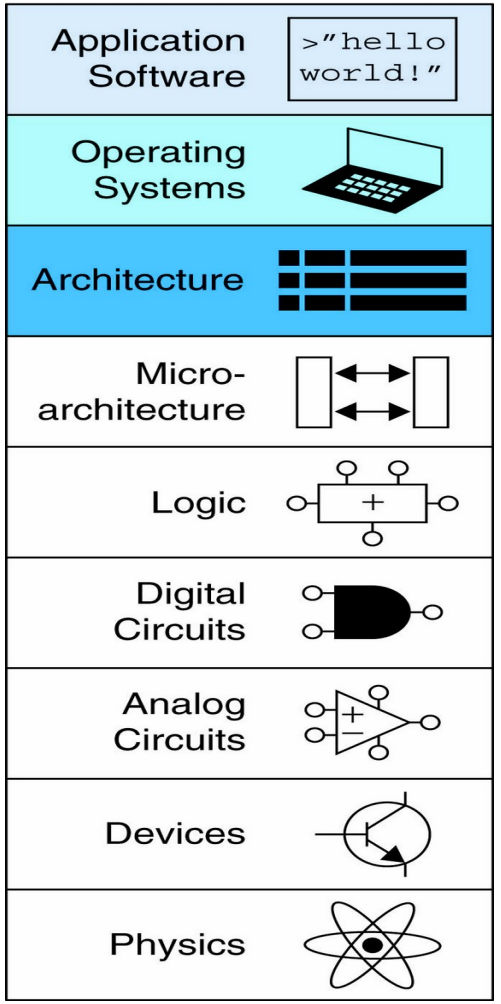
DIGITAL DESIGN AND COMPUTER ORGANIZATION

Addressing Modes

Dr. Reetinder Sidhu
Department of Computer Science and Engineering

Addressing Modes

- Introduction
- Assembly Language
- Machine Language
- Programming
- Addressing Modes
- Lights, Camera, Action: Compiling, Assembling, & Loading
- Odds and Ends



Addressing Modes

How do we address the operands?

- Register Only
- Immediate
- Base Addressing
- PC-Relative
- Pseudo Direct

Addressing Modes (Register Only)

Register Only

- Operands found in registers
  - **Example:** `add $s0, $t2, $t3`
  - **Example:** `sub $t8, $s1, $0`

Addressing Modes (Immediate)

Immediate

- 16-bit immediate used as an operand
  - **Example:** `addi $s4, $t5, -73`
  - **Example:** `ori  $t3, $t7, 0xFF`

Addressing Modes (Base)

Base Addressing

- Address of operand is:

  `base address + sign-extended immediate`

  - **Example:** `lw  $s4, 72($0)`
    - address = `$0 + 72`
  - **Example:** `sw  $t2, -25($t1)`
    - address = `$t1 - 25`

Addressing Modes (PC Relative)

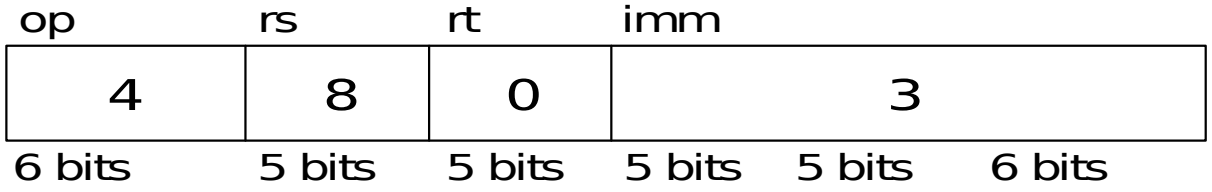
PC-Relative Addressing

```
0x10          beq   $t0, $0, else
0x14          addi  $v0, $0, 1
0x18          addi  $sp, $sp, i
0x1C          jr       $ra
0x20      else:     addi  $a0, $a0, -1
0x24          j     factorial
```

Assembly Code

```
beq $t0, $0, else
(beq $t0, $0, 3)
```

Field Values



Addressing Modes (Pseudo-direct)

Pseudo-direct Addressing

```
0x0040005C          j       sum
...
0x004000A0  sum:    add     $v0, $a0, $a1
```

J TA 0000 0000 0100 0000 0000 0000 1010 0000 (0x004000A0)

26-bit addr 0000 0000 0100 0000 0000 0000 1010 0000 (0x0100028)

0 1 0 0 0 2 8

Field Values

op	imm
3	0x0100028
6 bits	26 bits

Machine Code

op	addr	
000011	00 0001 0000 0000 0000 0010 1000	(0x0C100028)
6 bits	26 bits	