Chapter 04: Instruction Sets and the Processor organizations

Lesson 05:

Basic Addressing Modes for operands

Objective

- Basic Addressing Modes

Addressing Modes

Addressing mode

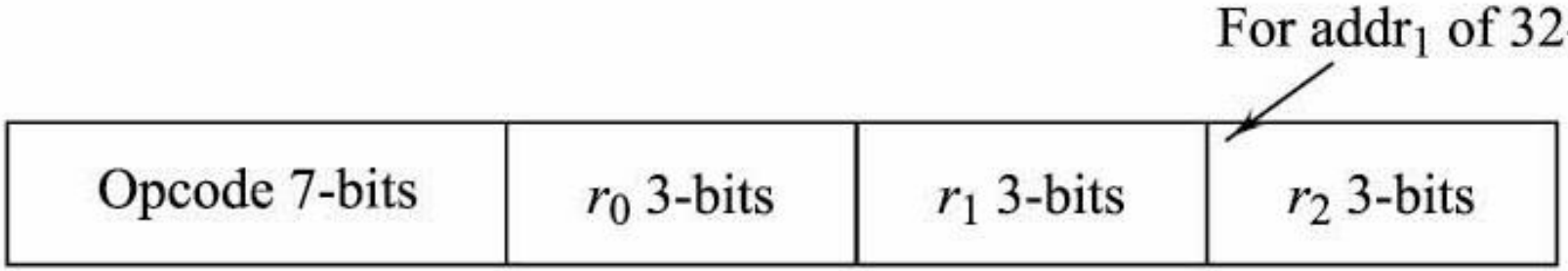
- Defined by an instruction opcode of an instruction lets the processor compute the address of the operands

Common Ways

- 4-ways— the instructions of an instruction set compute the address of the operands

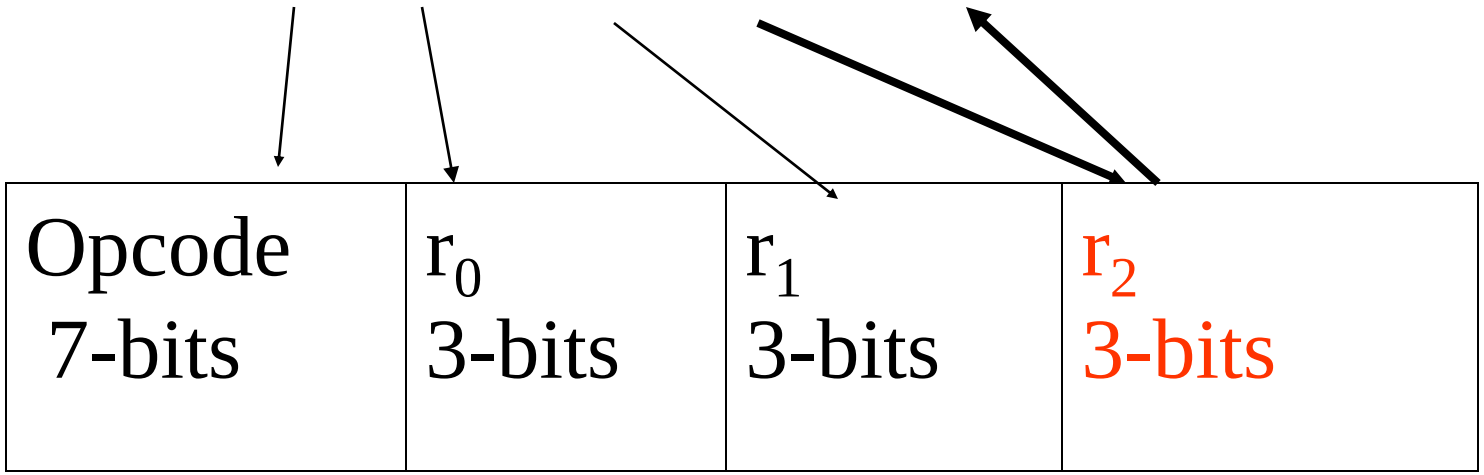
1. At a register
2. At a memory address specified in instruction
3. At a memory address pointed by a register
4. The immediate operand (part of instruction after the opcode bits)

Indirect Addressing Mode



Indirect Memory to Register Operation Format

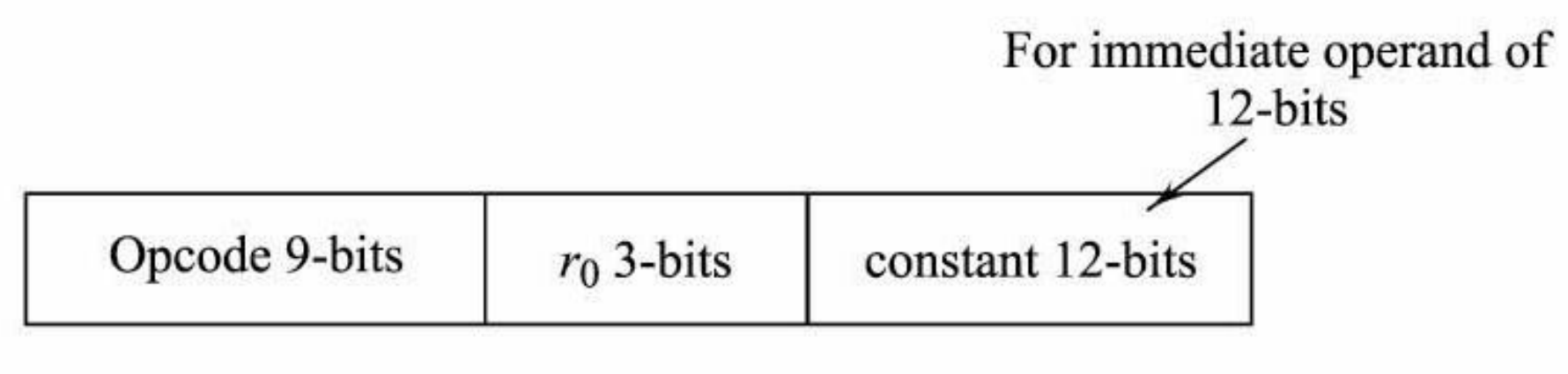
ADD r_0, r_1, r_2 {M [$addr_1$]},



4) Immediate operand

- Operand is at a part of the instruction immediate operand after the opcode of the instruction — ADD r1, #07, second source operand is immediate and is 07
- First source operand and destination is same and is a register operand in r1

Immediate operand

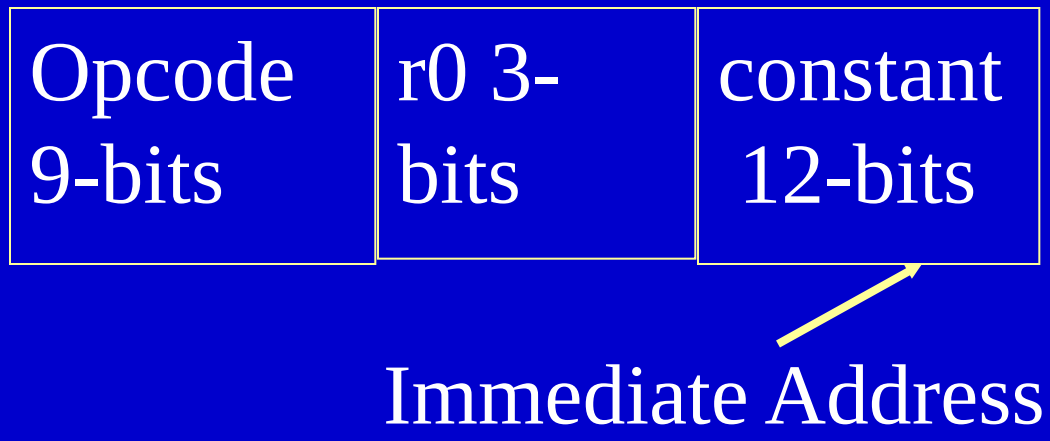


Constant (Immediate Operand) to Register Operation Format

- If register r_0 is specified by 3-bits and a constant is of 12 bits then instruction *MOV r0, #b11-b0* can be three-bytes (m = 3) long when the opcode field can have 24 – 3 – 12 = 9 bits. b11-b0 are 12 bits for a constant value to be moved in the register r_0. *r0* ← *#b11-b0*

Immediate Address

MOV r0, #constant



Memory address as operand

Register to memory or Memory to Register Operation Formats

- *I* = operation (M_0,.., M_i, r_0,.., r_j) or I = operation (r_0,.., r_j, M_0,.., M_i,)
- i = 0, 1 or 2 and j = 0, 1 or 2
- M─ means a memory address and r means a register
- M_i ─ means memory content at address i, $ADDR_i$

Memory address among 2^n locations

1) Memory address short or at a page or segment of memory – A memory address represented by few bits than *n*

MUL r_0, M [$addr_1$].

- Direct memory by short address ─ If register r_2 is specified by 4-bits and memory addresses by 20 bits (a short address for 32-bits), then MUL r_2, M [$addr_1$] can be four-bytes (m = 4) long when the opcode field can have 32 – 4 – 20 = 8 bits. [r_3 is implicit.] (r2-r3) pair ← r2 × M [$addr_1$].

Memory address among 2^n locations

2) Memory address long - A memory address represented by all address bits represented by *n* bits

MUL r_2, r_3, M [addr$_1$]

- Absolute long memory address─ If register pair r_2-r$_3$ is specified by 4-bits each and memory addresses by 32 bits, then MUL r_2, r_3, M [addr$_1$] can be six-bytes (m = 6) long when the opcode field can have 48 – 8 – 32 = 8 bits. r2-r3 ← r2 × M [addr$_1$]

Memory address among 2^n locations

3) Relative address— A memory address represented by relative displacement (plus or minus) from the next instruction address

Memory address among 2^n locations

4) Offset address— A memory address represented by an offset with respect to a base address, the offset plus base address bits displacement (plus or minus) from next instruction address

Memory address among 2^n locations

5) Base plus relative or offset, Index plus relative or offset, Base plus index plus relative or offset

Summary

We Learnt

- Addressing of operands from memory
- Register addressing
- Memory direct addressing
- Memory indirect addressing
- Absolute long address
- Relative address
- Offset (Displacement)
- Immediate addressing

End of Lesson 05 on

Basic Addressing Modes for operands