COMPUTER SCIENCE

Computer Organization and Architecture

Machine Instruction and Addressing Modes

Machine Instruction

Lecture_02

Vijay Agarwal sir

01 Machine Instruction

02 Instruction Set Architecture

Memory Concept

Memory : Byte Addressable

Processor : Data form is Word

Multiple cell Accessing is Required Parallely.

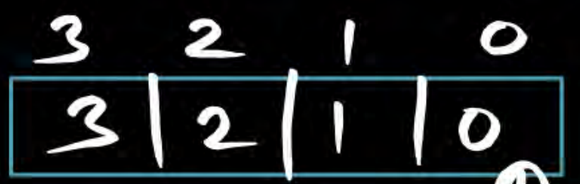
(e.g.) 16 bit Processor

(e.g.) 32 bit Processor.

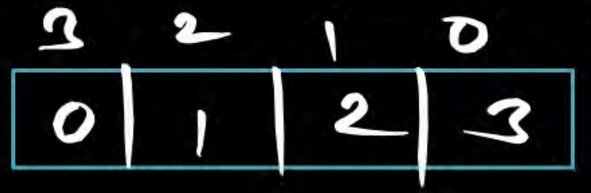
Endian Mechanism

Lower Address Contain LB
Higher Address Contain HB.

① Little Endian



② Big Endian



Eg) 32 bit Processor.

Higher Byte Position

Lower Byte Position

13 22 45 21

Store Memory Location 500.

Little Endian

503	502	501	500
13	22	45	21

500	21
501	45
502	22
503	13

Big Endian

503	502	501	500
21	45	22	13

500	13
501	22
502	45
503	21

Instruction

Instruction is a binary code (sequence of bit) which is designed inside the Processor to perform some operation.

Instruction Format

OPCODE ⇒ operational Code.

↳ Type of operation.

(e) If opcode is 2 bit then

2^2 ⇒ 4 operation performed.

00 ADD
01 MUL
10 INC
11 OR

If opcode is 3 bit then 2^3

⇒ 8 operation performed

000 → OR
001 → AND
010 → NOT
011 → INC
100 → DIV
101 → SUB
110 → ADD
111 → MUL.

Instruction Format

Note: n bit opcode can Represent 2^n operations

OPERAND : ⇒ 'DATA'

(OR)

OPERAND Reference (Address field) : Address of the operand.

eg: If Memory is 1KB then # Address bit = $\lceil \log_2 1KB \rceil$ or $2^{10}B$ ⇒ 10bit Addr

n bit Address line (Field)

Word Addressable

2^n Words.

eg 15 bit A.L

2^{15} Words = 32k Words.

32kW then 15 bit

Byte Addressable

2^n Byte

eg 15 bit A.L

2^{15} Byte = 32kB

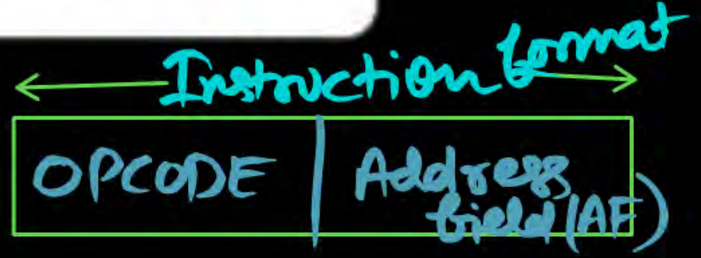
OR

If 32kB then 15 bit

Instruction Representation

- Opcodes are represented by abbreviations called mnemonics
- Examples includes:
  - ADD — Add
  - SUB — Subtract
  - MUL — Multiply
  - DIV — Divide
  - LOAD — Load data from memory
  - STORE — Store data to memory
- Operands are also represented symbolically
- Each symbolic opcode has a fixed binary representation

Instruction Format



Opcode (Mnemonics) : Type of operation / Instruction Supported

(eg) If opcode is 4 bit

then Total Number of operation / Instruction $= 2^4 = 16$ operation.

$10^3 = 1K = 2^{10}$

$10^6 = 1M = 2^{20}$

$10^9 = 1G = 2^{30}$

$= 1T = 2^{40}$

$= 1P = 2^{50}$

$= 1E = 2^{60}$

$= 1Z = 2^{70}$

$= 1Y = 2^{80}$

1bit $= 2^1 = 2$

2bit $= 2^2 = 4$

3bit $= 2^3 = 8$

4bit $= 2^4 = 16$

5bit $= 2^5 = 32$ (17-32)

6bit $= 2^6 = \underline{64}$ (33-64)

$\underline{7bit} = 2^7 = \underline{128}$ (65-128)

8bit $= 2^8 = 256$

9bit $= 2^9 = 512$

10bit $= 2^{10} = 1024 (1K)$

Instruction Format

Q.1 opcode is 6 bit

then # operation $= 2^6$

$=$ 64 operation/Instruction Ans

Q.2 If 100 Operation/Instruction

then

No. of bit in opcode field $= 100 = 2^n$

$=$ 7bit Ans

#of opcode bit $= \lceil \log_2 100 \rceil = \lceil 6.87 \rceil$

$= 7$bit ✗

Q.3 If memory is 256kB

then memory A.F Size in bit

$\lceil \log_2 256k \rceil$

18bit

OR $256kB = 2^8 \cdot 2^{10}B = 2^{18}B$

memory A.F $=$ 18 bit Ans

Q.4 If Processor has 26 Register

then # bits Required in

Reg AF $=$ 5bit Ans

$2^n = 26$

$\therefore$

Instruction Format

OP : Opcode.
AF/AI : Address field / Address Instruction

X4AI/4AF ✗ | OP | AF1 | AF2 | AF3 | AF4 | → Next Instn Address But we have PC for that.

Destn Source1 Source2

3AI/3AF | OP | AF1 | AF2 | AF3 |

OP AF1 AF2 AF3
ADD | R1 R2 R3 ; $R_1 \in R_2 + R_3$

2AI/2AF | OP | AF1 | AF2 |

ADD | R1 R2 ; $R_1 \in R_1 + R_2$

1AI/1AF | OP | AF1 |

ADD | R_L $AC \in AC + R_L$

0AI/0AF | OPCODE |

ADD Stack.

Instruction Format

Eg for ALU operation.

MUL
ADD

ALU operation

Elements of a Machine Instruction

Operation Code (opcode)

- Specifies the operation to be performed. The operation is specified by a binary code, known as the operation code, or opcode.

Source Operand Reference

- The operation may involve one or more source operands, that is, operands that are inputs for the operation

Destination

Result Operand Reference

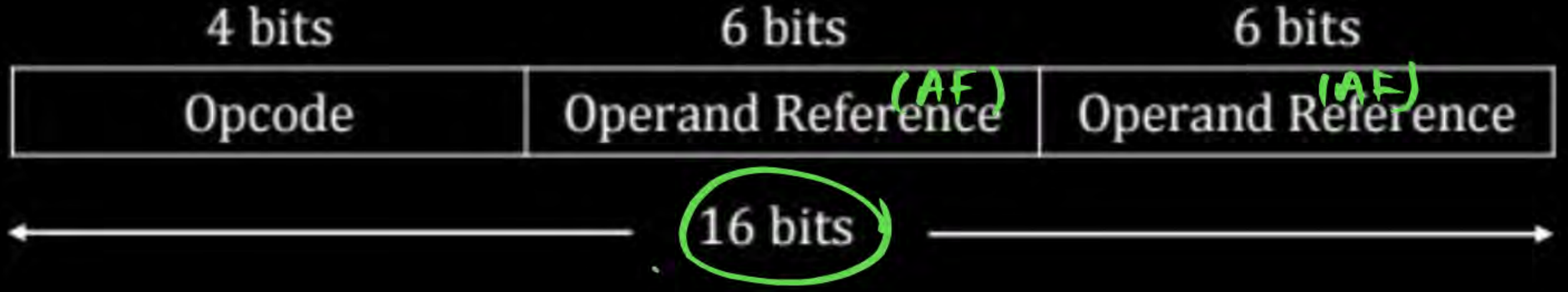
- The operation may produce a result

Next Instruction Reference

- This tells the processor where to fetch the next instruction after the execution of this instruction is complete

Instruction Representation

- Within the computer each instruction is represented by a sequence of bits
- The instruction is divided into fields, corresponding to the constituent elements of the instruction



Q.1

Consider a Hypothetical Processor which support 128 byte memory and instruction length is 16 bit.

(i) If 2AF(2AI(Address Instruction) same size) is used then How many total number of operation supported (formulated)?

(ii) If 1AF (Address field) is used then how many total number of operation supported formulated?

Soln

Instruction Size = 16 bit

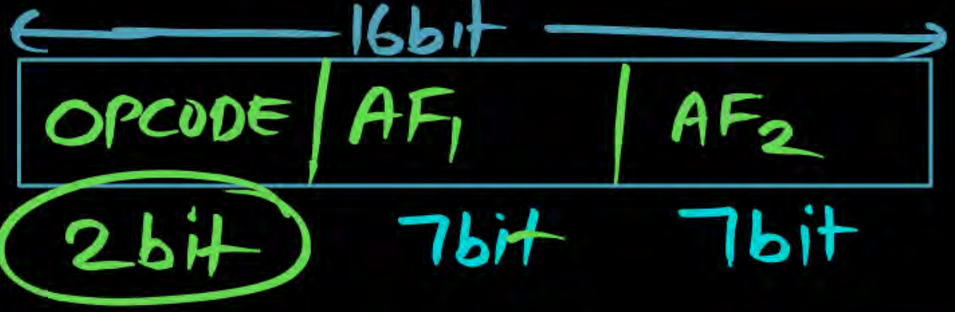
Memory = 128 Byte $[2^7 B]$

AF = 7bit

Soln I

2AI/2AF

Soln 1

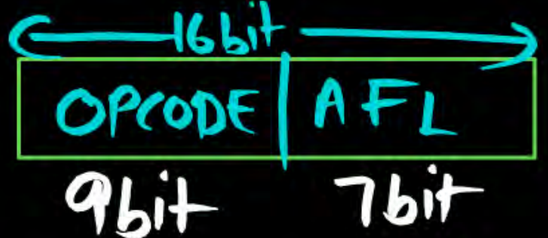


OPCODE = 16 − (7 + 7)

= 16 − 14 = 2bit

Total # operation = 2^2 = 4 operations/Instn Ans

Soln 2

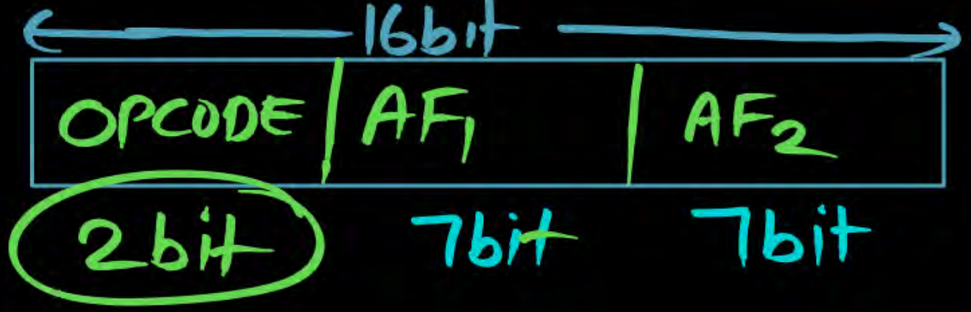


OPCODE = 16 − 7 = 9bit

Total # operation = 2^9 = 512 operation/Instn Ans

Soln I Asking maximum & Minimum operation ?
2AI/2AF

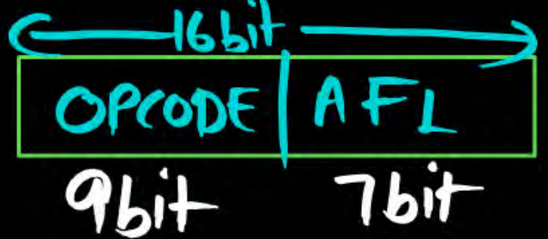
Soln 1



Minimum operation = 1
Maximum operation = 4

Total # operation = 2^2 = 4 operations/Instn Ans

Soln 2



Minimum operation = 1
Maximum operation = 512.

Total # operation = 2^9 = 512 operation/Instn Ans

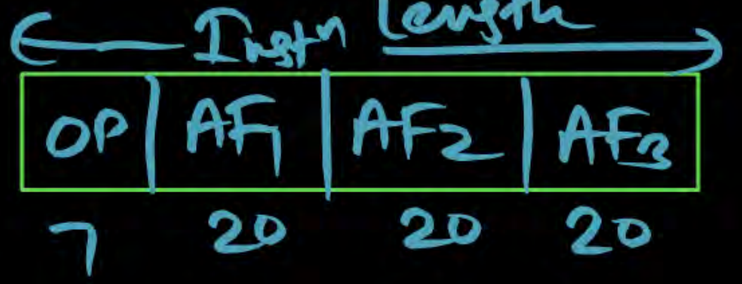
∴

Q. A Hypothetical Processor support 100 different operation and 3 address memory field (same size). Instruction is stored in 1 MB memory. Then what is the length of the instruction?

Soln

100 operation

then opcode = 7 bit



Memory Size = 1MB

$= 2^{20}$ Byte

then A.F = 20 bit

Instruction length = 7 + 20 + 20 + 20

= 67 bits Ans

Machine Instruction Characteristics

- The operation of the processor is determined by the instructions it executes, referred to as machine instructions or computer instructions
- The collection of different instruction that the processor can execute is referred to as the processor's instruction set **(ISA)**

Instruction Set of Size = (11)

OPcode = 4 bit

11 Distinct operation are Performed like

ADD, OR, AND, NAND, XOR, INC, MOV, STORE SUB, DIV, MUL.

Instruction Format

OP : Opcode.
AF/AI : Address field / Address Instruction

XYAI/YAF ✕ | OP | AF_1 | AF_2 | AF_3 | AF_4 → Next Instn Address But we have PC for that.

Destn Source1 Source2

OP AF_1 AF_2 AF_3

3AI/3AF | OP | AF_1 | AF_2 | AF_3 | ADD | R_1 R_2 R_3; $R_1 \in R_2 + R_3$

General Registers

2AI/2AF | OP | AF_1 | AF_2 | ADD | R_1 R_2; $R_1 \in R_1 + R_2$

Accumulator

1AI/1AF | OP | AF_1 | ADD | R_1 ; $AC \in AC + R_1$

0AI/0AF Stack | OPCODE | ADD ; Stack ;

Instruction Set Architecture Classification

1. Stack organization.
2. Single accumulator organization.
3. General register organization.

Instruction Set Architecture

CUP Organization is classified into 3 types based on the availability of ALU Operand (Data) (AF: Address field or AI: Address Instruction)

1. Stack-CPU **[0AF]** Stack Based org [Stack-CPU]
2. Accumulator-CPU **[1AF]** Accumulator Based org [AC-CPU]
3. General Register organization
   i. Reg-Memory reference CPU **[2AF]**
   ii. Reg-Reg reference CPU **[3AF]**

Q.1 Consider a Hypothetical CPU which supports 110 (operation) instruction, 50 registers and 512KB memory space. Instruction contain 2 register operands, Memory operands and 13 bit Immediate constant fields. Program contain 300 instruction. Memory storage space required in Bytes to store the program is _____.

$\frac{51}{8} = 6.4$ ✗ Exactly in Byte ✓

110 operation ⇒ Opcode = 7 bit

50 Register ⇒ Reg AF = 6 bit

512 kB Memory (2^{19} B) ⇒ MemAF = 19 bit

OPCODE	RegAF	RegAF	MemAF	Immediate fixed
7bit	6bit	6bit	19bit	13bit

1 Instruction length(size) = 7+6+6+19+13 = 51 bits

1 Instruction Size = 7 Byte

Program having = 300 Instn

Program Size = 7×300 = 2100 Byte Ans

Q.2

Consider a processor which contain the following pin structure

$AD_0 - AD_{23}$, $A_{24} - A_{39}$

Processor contain 250 register instruction is designed with 4 fields i.e OPCODE, register address, memory address and 16 bit immediate field.

Processor support 180 instruction (operation). A program contain 400 instruction then how much space is required for the program

(i) In Byte?

(ii) Words?

Mem AF : $A_0 - A_{39}$

Mem AF = 40 bit

Word length = $AD_0 - AD_{23}$

= 24 bit

1 Word = 24 bit

Size

3 Byte

250 Register ⇒ Reg AF = 8 bit

Immediate field = 16 bit

180 Instⁿ / operation ⇒ opcode = 8 bit

OPCODE	Reg AF	Memory AF	Immediate field
8 bit	8 bit	40 bit	16 bit

1 Word = 24 bit

1 Instruction Size (Length) = 8 + 8 + 40 + 16 = 72 bits

1 Instruction Size = 9 Byte

1W = 3 Byte

9B / 3B = 3 Words

9B = 3 Words

Program Size (in Byte)

Program Contain 400 Instn.

Program Size = 400 × 9

= 3600 Byte Ans

Program Size in Words

3 Byte = 1 Words So 9 Byte = 3 Words

1 Instn Size = 3 Words

Prog Size = 400 × 3

= 1200 Words. Ans

AF
- Register AF
- Memory AF

Immediate field : Constant.

Note:

Immediate field is n bit

Unsigned Range = $(0 \text{ to } 2^n - 1)$

Signed Range = $-(2^{n-1})$ to $+(2^{n-1} - 1)$

Example

If immediate field is 4 bit

Then unsigned range = $(0 \text{ to } 2^4 - 1) \Rightarrow 0 \text{ to } 15$

Signed Range = $-(2^{4-1})$ to $+(2^{4-1} - 1)$

$-(2^{4-1})$ to $+(2^{4-1} - 1)$

-8 to $+7$

Q.3

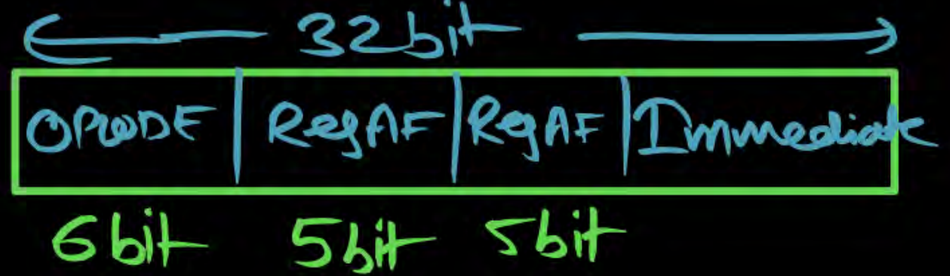
A machine has a 32-bit architecture, with 1-word long instructions. It has 64 registers, each of which is 32 bits long. It needs to support 45 instruction, which have an immediate operand in addition to two register operands. Assuming that the immediate operand is an unsigned integer the maximum value of the immediate operand is _____. **[GATE-2014 (Set-1)]**

$2^n - 1$

Q.4

A processor has 40 distinct instructions and 24 general purpose registers. A 32-bit instruction word has an opcode, two register operands and an immediate operand. The number of bits available for the immediate operand field is 16 bit

[GATE-2016 (Set-2)]



24 Register : ⇒ Reg AF = 5 bit

40 operation ⇒ opcode = 6bit

2^n

Immediate = 32 − (6 + 5 + 5)

= 32 − 16

= 16 Ans

Ans (16)

Q.5

Consider a processor with 64 registers and an instruction set of size twelve. Each instruction has five distinct fields, namely, opcode, two source register identifiers, one destination register identifier, and a twelve-bit immediate value. Each instruction must be stored in memory in a byte-aligned fashion. If a program has 100 instructions, the amount of memory (in bytes) consumed by the program text is _____.

[GATE-2016 (Set-2): 2Marks]

THANK YOU!