CS & IT ENGINEERING

Computer Organisation and Architecture

Machine Instructions and Addressing Mode

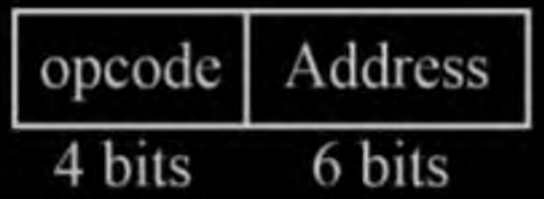
DPP Discussion

By-Vijay Agarwal sir

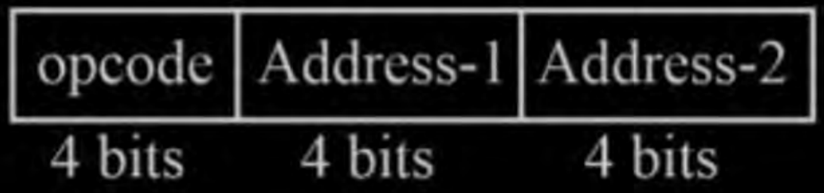
[MCQ]

Which of the following is primitive instruction in the CPU?

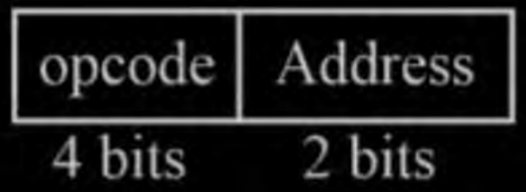
A



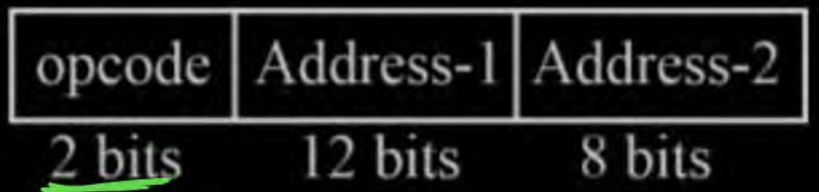
B



C



D



Ans (D)

Primitive Instruction: Lowest [Smallest] bit in opcode.

Smallest opecode bit

Derived Instⁿ: More Opcode bits.

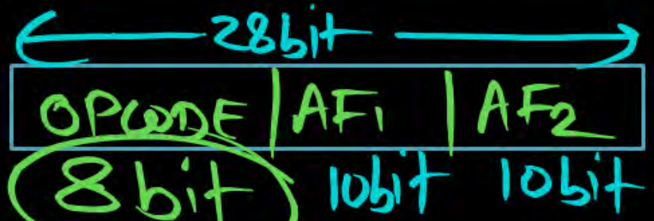
More (or) further Derived Instⁿ: More & More bit in opcode.

[NAT]

Consider a hypothetical system which support only 2 address instructions. If size of the instruction is 28 bit and size of each address is 10 bits then, the maximum number of instructions the system can support____.

Instruction Size = 28 bits

Address field = 10 bit



Opcode = 28 − (10 + 10)

= 8bit

n bit opcode can perform 2^n operation

Total Number of operation/Instr $= 2^8$

$= 256$

Ans (256)

[NAT]

Consider a digital computer which support 64 3-address instruction. If the size of each address is 14 bits, then the instruction is of 6 bytes.

64 3AI

↓↓

64 operation Performed

Ans (6)

3AF / 3AI (Address Instruction is given

AF = 14 bits

64 operation $\Rightarrow$ Opcode $= 2^n = 64 \Rightarrow 2^n = 2^6$

$n = 6$ bit

AF = 14 bit

Instruction Length $= 6 + 14 + 14 + 14$

$= 48$ bits $\Rightarrow \frac{48}{8}$ Byte

$= 6$ Byte Ans

[MCQ]

The correct sequence in Fetch-Execute cycle is:

A Decode, Fetch, Execute

B Fetch, Execute, Decode

C Fetch, Decode, Execute

D None of the above

Ans (C).

Instruction Cycle: The Process Required for each Instruction execution is Called Instruction cycle.

1. Fetch cycle
2. Execute cycle
   - Decode
   - Execute.

Steps in Instruction Cycle.

Fetch cycle
1. IAC [Instruction Address Calculation]
2. IF (Instruction Fetch) [Mem to CPU [IR]]

Execute cycle
- Decode
  3. Decoding (Analysis of the Instruction (what opcode, How many operands, where operands are available etc.))
- Execute
  4. OAC [Operand Address Calculation]
  5. OF [Operand Fetch] (Addressing Modes Required)
  6. Data Processing
  7. Result Storage

[MCQ]

Consider a system which supports 2 - address instructions only. The system has 2^P K bytes of memory. If there are 'q' distinct instructions supported by system then, what is the size of instruction?

A (q + p) bits

B $[\log_2 q + 2 (P + 10)]$ bits

C (logq + 2P) bits

D log (p + q) bits

Ans (B).

2AF | 2AI

Memory Size = 2^p k Byte $\Rightarrow$ $2^p \times 2^{10}$ Byte $\Rightarrow$ 2^{p+10} Byte

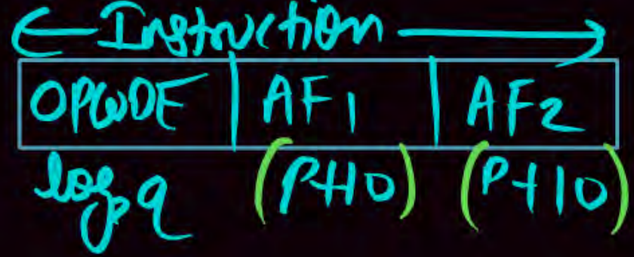
Total No. of Distinct operation 'q'

Instruction Size ?

Memory Size = 2^{p+10} Byte then $\boxed{AF = p+10}$

$\log_2(2^{p+10})$ or $2^n = 2^{p+10}$

opcode bit = $\lceil \log_2 \#operation \rceil \Rightarrow \lceil \log_2 q \rceil$



Instruction Size = $\log_2 q + (p+10) + (p+10)$

$= \log_2 q + 2(p+10)$ Ans

[MCQ]

Consider a system which support 2 - address instructions add 1-address instruction both. Suppose, the system has 6 bits instruction and 2 - bits addresses. If there are three, 2 - address in the system then, maximum and minimum how many 1 - address instruction the system can support?

A 15, 1

B 4, 1

C 16, 0

Ans(B)

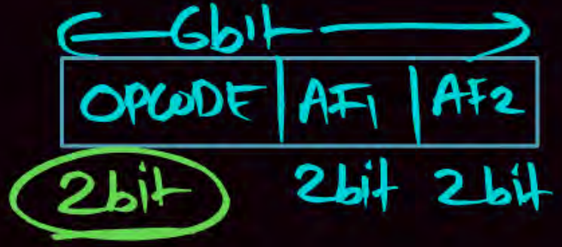
D None

Instruction Size = 6 bit

Address field (AF) = 2 bit

Given 2AI = 3 then Asking about 1AI ?

Step1



Opcode = 6 − (2+2) = 2bit

Step2

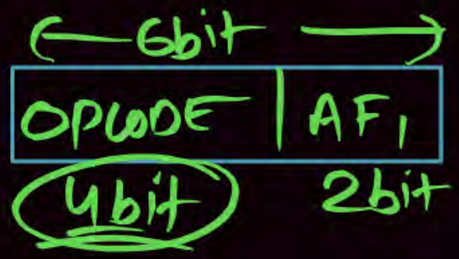
Total # operation in 2AI = 2^2 = 4 operation

GIVEN 2AI = 3

Number of Free opcode After Allocating 2AI = 4 − 3 = 1

Expand opcode Technique, fixed length Instruction

1AI



Total # operation in 1AI = Free opcode X 2 (Increment bit in opcode)

$4-2$

$\Rightarrow 1 \times 2 \Rightarrow 1 \times 2^2$

1AI = 4 operation

Minimum operation in 1AI = 1

Maximum operation in 1AI ≤ 4

$\therefore$

[MCQ]

Consider a PC - relative mode type branch instruction, which takes branch on addresses 680 in memory. The instruction has offset value 420. What is the address of this instruction in memory. If each instruction is stored in memory on 2 - locations?

(Branch) Target Address = Current PC Value + Relative Value (OFFSET)

Instn Starting Address = 258

A 258

B 280

C 282

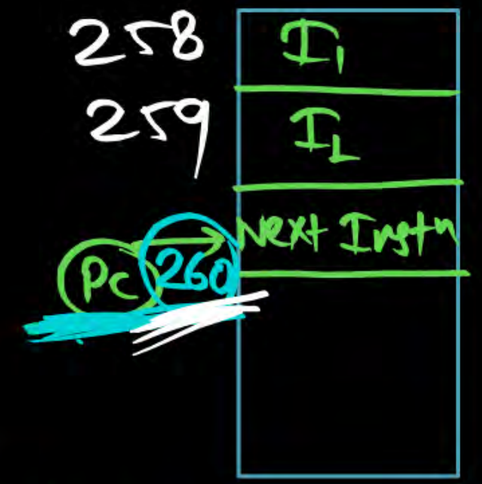
D 260

Ans(A)

680 = PC + 420

PC = 680 − 420

PC = 260



PC [Program Counter Denote Next Instr Starting Address.

[MCQ]

Consider a 6 - word instruction which is of the following type:

The first operand (destination) uses register indirect mode and second operand uses indirect mode. Assume each operand of size 2 - words, each address is of 2 word and main memory takes 20 ns for each word access. What is the different between fetch cycle of instruction time and execution cycle of instruction time?

[Note : Register access time is negligible.]

A 260 ns

B 20 ns

Ans (C).

C 40 ns

D 120 ns

OPCODE	MODE1	MODE2	AF1	AF2

Instruction Size = 6 Word

A.F = 2 Word

operand Size = 2 word.

Memory Access for each word = 20 nsec

Fetch cycle

Total Time taken in Fetch cycle = 6 X 20 ns = 120 nsec

Execute cycle

Source 1 : Registel Indirect → (1 Reg Ref + 1 Mem Ref)

⇒ 1 Mem Ref [2 word] (∵ AF = 2 Word)

⇒ 2 X 20 = 40 nsec.

∴

1 mem Ref for EA
1 mem Ref for Data (Read).

Source 2 Memory Indirect : 2 Mem Ref. & each mem Ref = 20

⇒ A.F = 2 Word ⇒ 2 × (2 × 20) = 80 nsec.

Destination (Source 1) : Register Indirect : (1 Reg + 1 mem)

⇒ 2 Word × 20 = 40 nsec

Total time in Execution cycle = 40 + 80 + 40 = 160 nsec.

Difference = 160 − 120 = 40 nsec Ans

[MCQ]

65000

Consider a computer with 34 bits instruction and 14 bits addresses. If there are 60 2 - address instruction and 65000 1 - address instruction then, how many maximum 0 - address instruction can be formulated?

A 65536 K

B 8576 K

C 16384 K

D none

Ans (B)

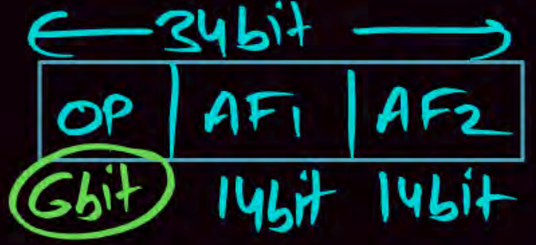
Instruction Size = 34 bit

AF = 14 bit

2AI | Operation = 60

1AI | operation = 65000

Expand Opcode Technique.

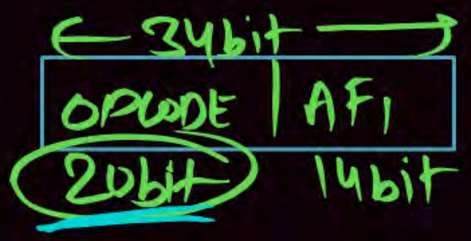


OPCODE = 34 − (14 + 14) = 6bit

Total # operation in 2AI = 2^6 = 64 operation | Instruction.

GIVEN 2AI = 60

Free opcode After Allocating 1AI = 64 − 60 = 4



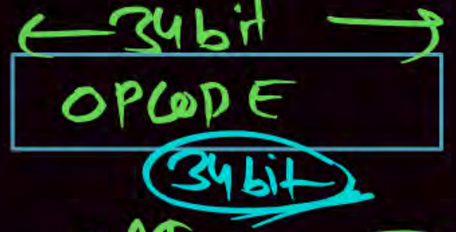
Increment bit in opcode

= Free opcode X 2

In 1 AF Total # operation = $4 \times 2^{20-6}$ ⇒ 4×2^{16} = 65536 operation

Given 1AI = 65000

Free opcode After Allocating 1AI = 65536 − 65000 = 536



34 − 20

OAI = 536×2

⇒ 536×2^{14} ⇒ $536 \times 2^4 \times 2^{10}$ (k)

⇒ 536 X 16 X k

= 8576 k.

[MCQ]

If an opcode is of 4 bit then, how many maximum minimum type of instruction supported by CPU?

n bit opcode can perform 2^n operation.

A 15, 1

B 15, 0

C 16, 1

D 16, 0

OPCODE Size = 4 bit

Total Number of operation = 2^4 = 16 operation

Maximum = 16

Minimum = 1

Ans [C].


THANK YOU GW
SOLDIERS !