

Lecture - 5 (Addressing Modes)

24 January 2024 21:47

Instruction Formats -

① 1 address

LOAD X



② 2-address

ADD X, Y



③ Register - memory

ADD R1, X



④ Register - register

ADD R1, R2, R3



Example :-

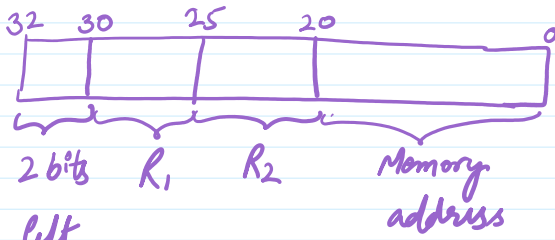
Instruction Set Architecture of a 32-bit instruction

(Just an example not real)

Let us assume,

- It is a fixed size instruction
- 5 bits for 1 register.
(32 registers)

ADD R1, LOCA

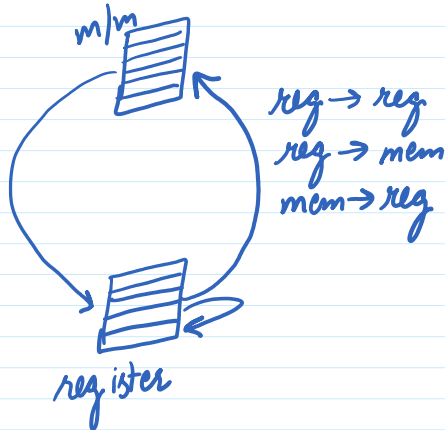


left

only 4 possible operations

Types of instructions

Data Transfer Instructions



LOAD
STORE
EXCHANGE
INPUT
OUTPUT
PUSH
POP

Program Control Instructions

→ decision
if, for, while

To modify the program
counters
changes flow of execution

BRANCH
CALL
TEST
JUMP
SKIP
RETURN.

Data Manipulation Instructions



Arithmetic	Logical	Shift
INCREMENT	CLEAR	LEFT SHIFT
DECREMENT	COMPLEMENT	RIGHT SHIFT
ADD	AND	ROTATE RIGHT
SUBTRACT	OR	ROTATE LEFT
MULTIPLY	XOR	
DIVIDE		
ADD WITH CARRY		
SUBTRACT WITH BORROW		
NEGATE		

ADDRESSING MODES :-

Why ?

To specify the mechanism by which location of **OPERAND** is specified in an instruction.

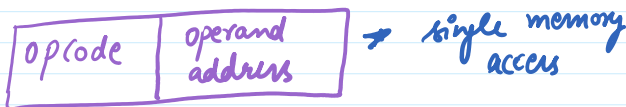
Various Addressing Modes :-

① Immediate Addressing

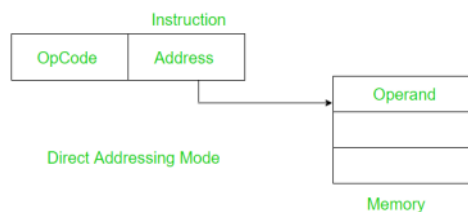


ADD #25

② Direct Addressing



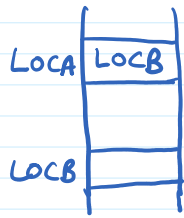
ADD R1, 20A6H



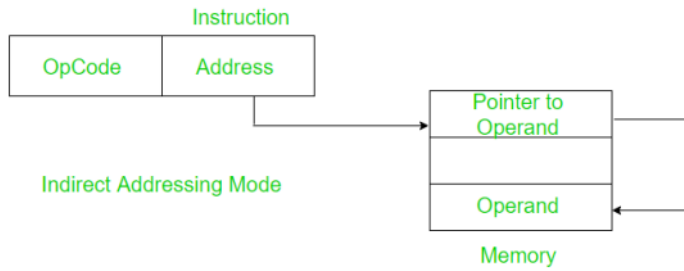
“Limited Address Space”

③ Indirect Addressing

ADD R1, LocA



- * Two memory accesses
- * Slower but can access large address space.

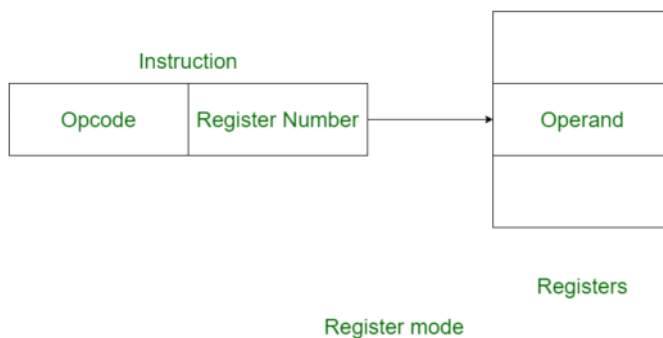


“Large address space”

④ Register Addressing

ADD R1, R2, R3

- * Operands are stored in register bank.
- * No memory access, fast execution.

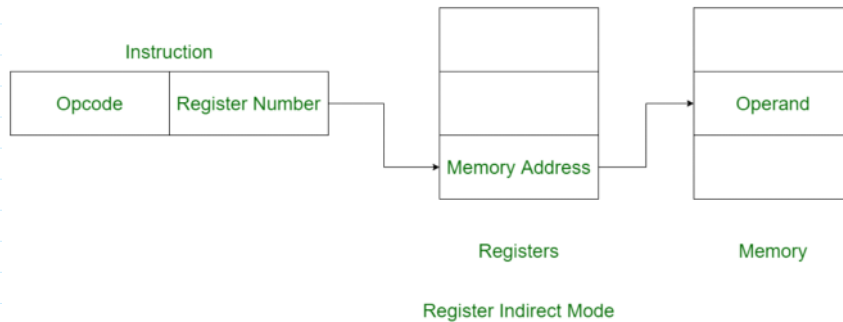


⑤ Register Indirect Addressing

ADD R1, mem(R5)

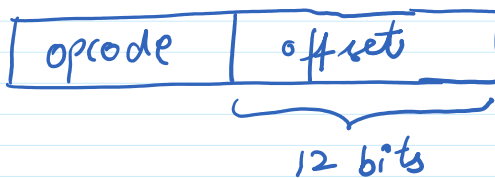
- * Can access large address space.

* One Less memory access as compared to indirect addressing.

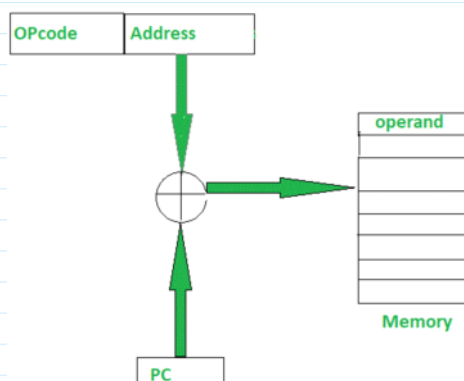


⑥ Relative Addressing (Program Counter Relative)

The instruction specifies an offset of displacement, which is added to the program counter to get the effective address of the operand.



What is the range of relative addressing?

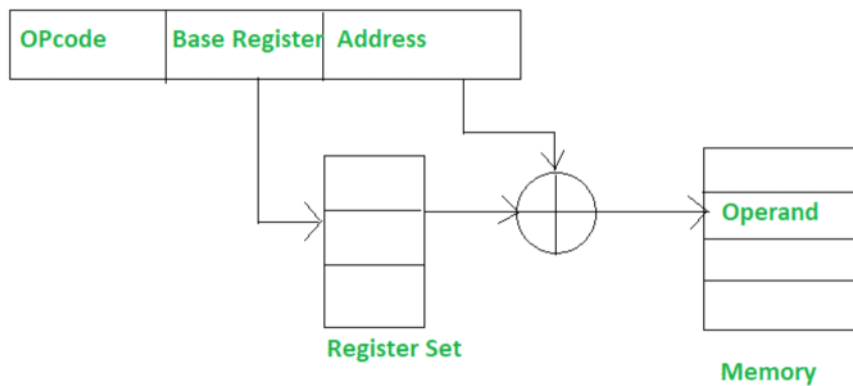


⑦ Indexed Addressing

* Register is used as index.

LOAD R1, 1050(R3)
Mem [1050+R3]

* Can be used to sequentially access the elements of an array.



⑧ Stack Addressing

* Operand is on the top of the stack.

* ADD
PUSH X
POP X

⑩ Autoincrement and Auto decrement

The register holding the operand address is automatically incremented or decremented

a++
a--