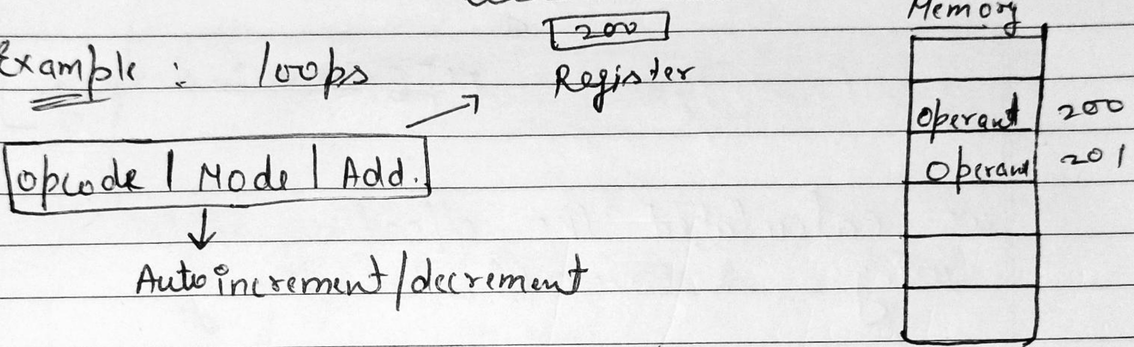
Questions on addressing modes Lecture 11

* Computable Modes :-

7* Auto increment / Autodcrement Mode

varient of register indicrect mode, in which the Content of register is automatically increment or decrement to access the squential data. → Table of data

Example : loops



Register will auto inerement the add. to access the sequential data.
for excessing in rev we will use auto derement.

Auto inc -> Post increment]→ old/Some particular microprocessor's architecture.
Auto dec. => Pre drement

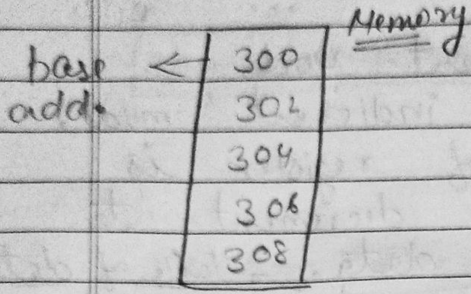
~~This~~

8* Indexed or Indexd Register mode

→ used for access the element of an array

* Add. of operand is effective address

int a[5]   loc(A[i]) = Base + w*i



i = 3, w = size of each element

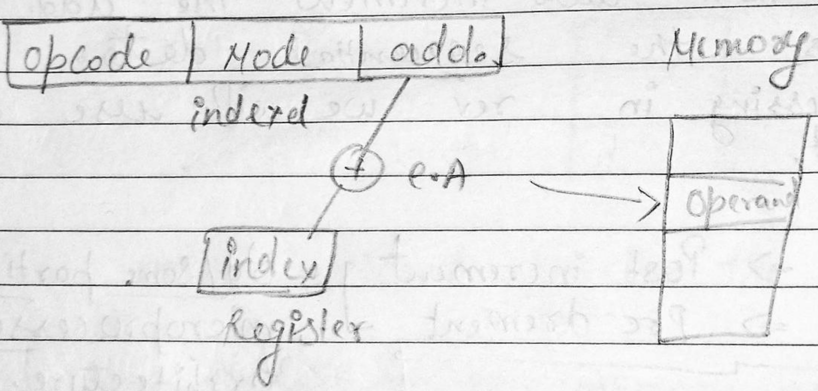
Compiler generates instruction

⇓

index Reg. = w*i

M[Base + indexReg.]

we calculated the effective add. using = Base add. + index Reg



effective add = { Add. part of inst. } + { index Reg value }

↓ base add.

↓ index

* One memory access required to get the operand.

* This add. mode does not support rellocation because of rellocation then the base ~~ad~~ add. should be update in instruction

~~implementation of index reg~~

index Reg

→ Special purpose reg

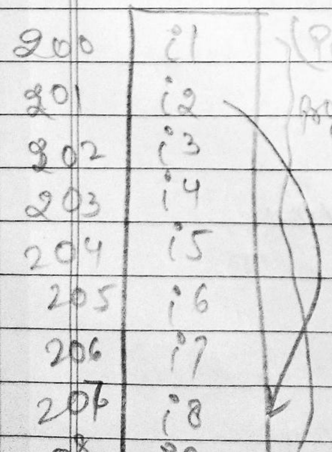
| opcode | mode | add. |

→ g.p.R

| opcode | index reg identifier | mode | Add. |

*9 · PC - Relative mode
→ used for branch instruction

Memory



Assume, CPU is executing i2 instruction.
(PC) Program Counter = 202
Program

i2 is branch instruction

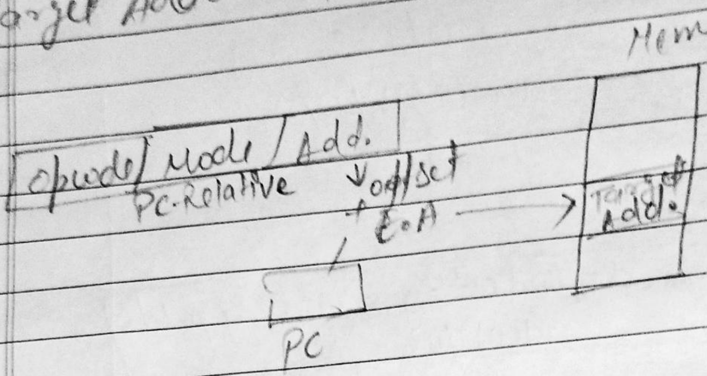
True
Branch taken
Target instruction executed
PC = 207
(Target add.)

false
Next in the sequence instruc
i3 executes
(Branch not take
No change in PC

i8 is target instruction
↓
its add. is known as offset
target add.

Target add. = P.C + no. of memory loc to skip
4

Target Add. = PC + 5 => 202 + 5 = 207



Pc Relative mode

effective add = PC value + add. part value of instruction
PC

-> before fetch of i2 — 201
After fetch of i2 — 202

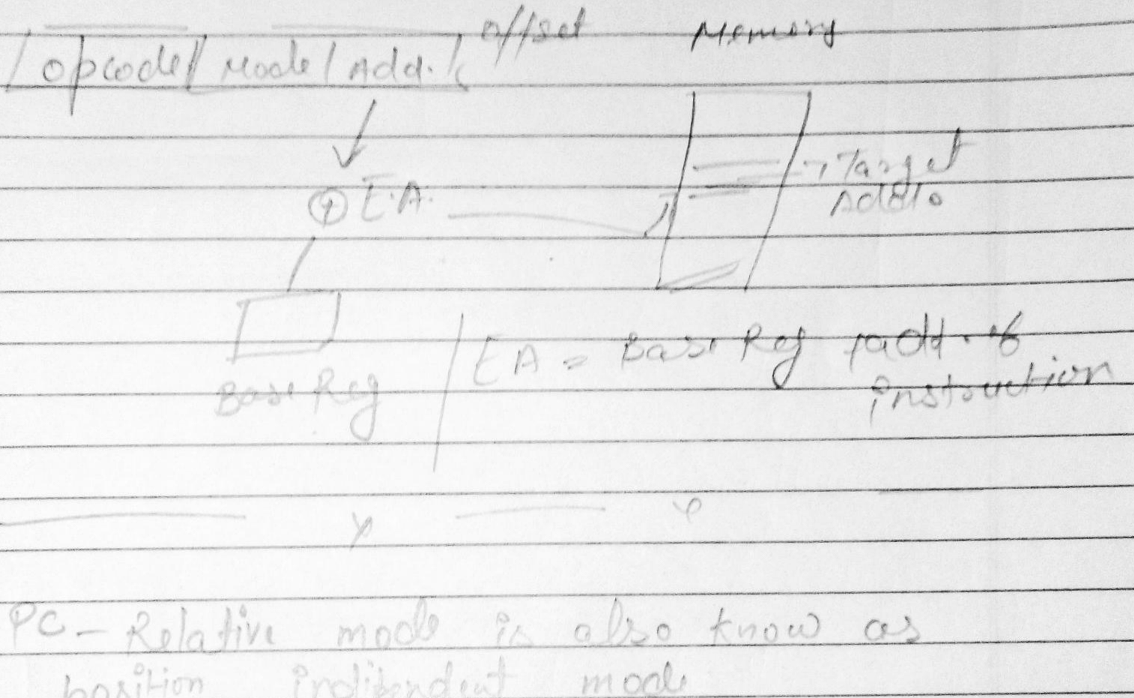
Decode => It's a branch instruction
E.A calculation => E.A = 202 + 5 = 207

execution => CPU check the condition and updates the pc if condition is true

* for, forward jumpin offset +ve
for, backword " -ve

* Intra-Segment branching

* Base Register Mode
used for inter-segment branching



EA = Base Reg + add. of instruction

PC-Relative mode is also know as position Indipendent mode

PC Relative, Base Register mode => Supports Rellocation

Both are used for Branching instruction
Rest are used to obtain operand or data