



SNS COLLEGE OF ENGINEERING



Kurumbapalayam(Po), Coimbatore - 641 107

Accredited by NAAC-UGC with 'A' Grade

Approved by AICTE, Recognized by UGC & Affiliated to Anna University, Chennai

Department of Information Technology

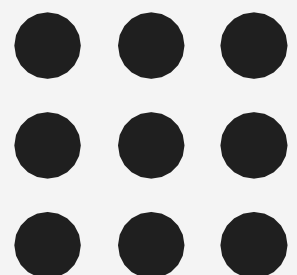
**Course Name - 19IT301 Computer Organization and
Aechitecture**

II Year / III Semester

Unit 1 - Basic Structures of Computers

Topic : Addressing Modes

19IT301 / UNIT 1/ Addressing Modes/K.Sangeetha/ECE/SNSCE



Addressing Modes

➤ **Implied**

AC is implied in “ADD M[AR]” in “One-Address” instr.

TOS is implied in “ADD” in “Zero-Address” instr.

➤ **Immediate**

The use of a constant in “MOV R1, 5”, i.e. $R1 \leftarrow 5$

➤ **Register**

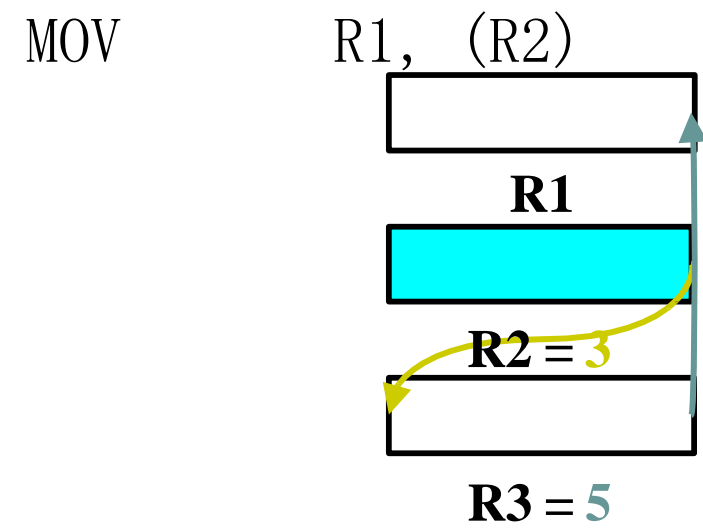
Indicate which register holds the operand





- **Register Indirect**

- Indicate the register that holds the number of the register that holds the operand



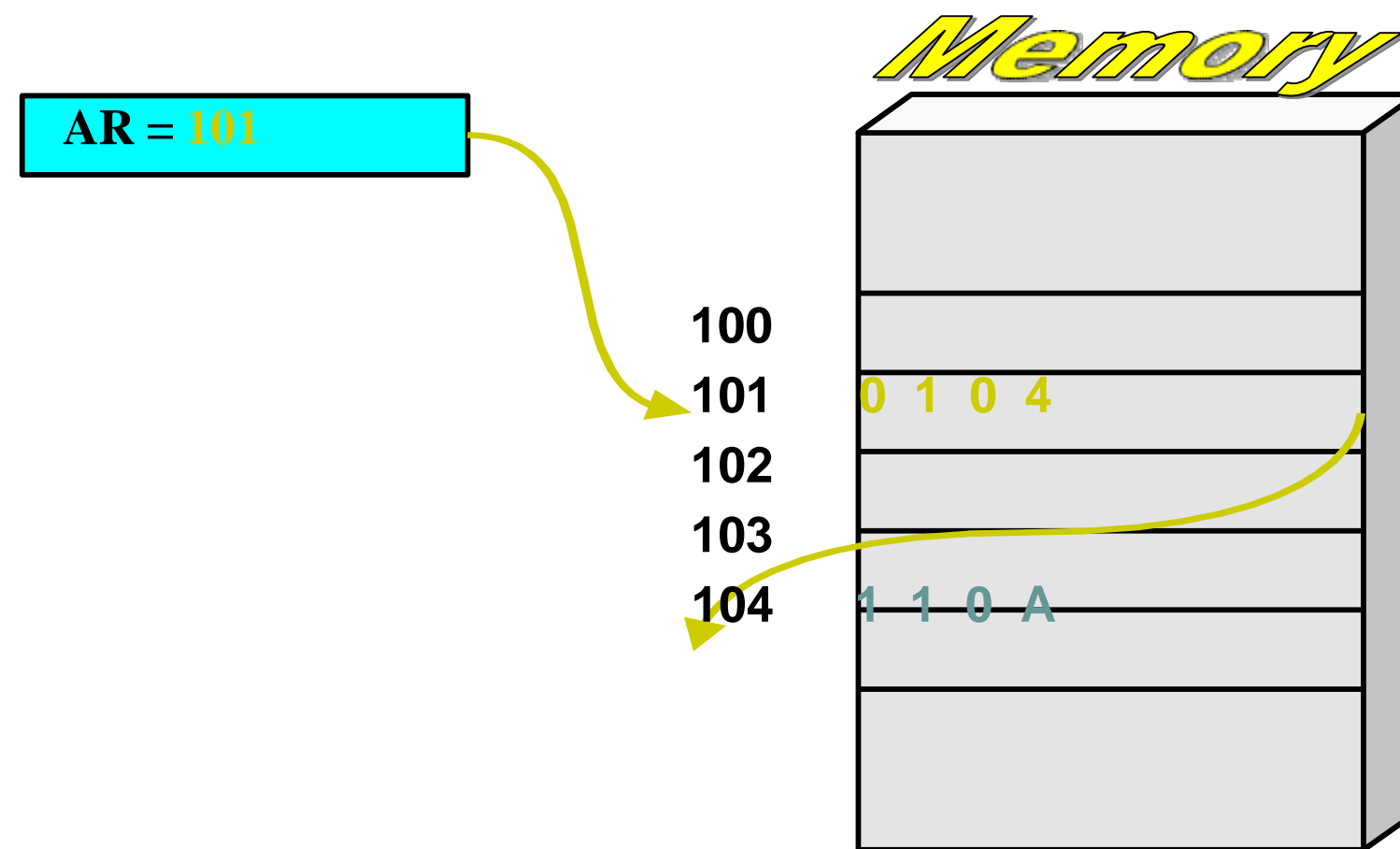
- **Autoincrement / Autodecrement**

Access & update in 1 instruction. **DirectAddress**

Use the given address to access a memory location

Indirect Address

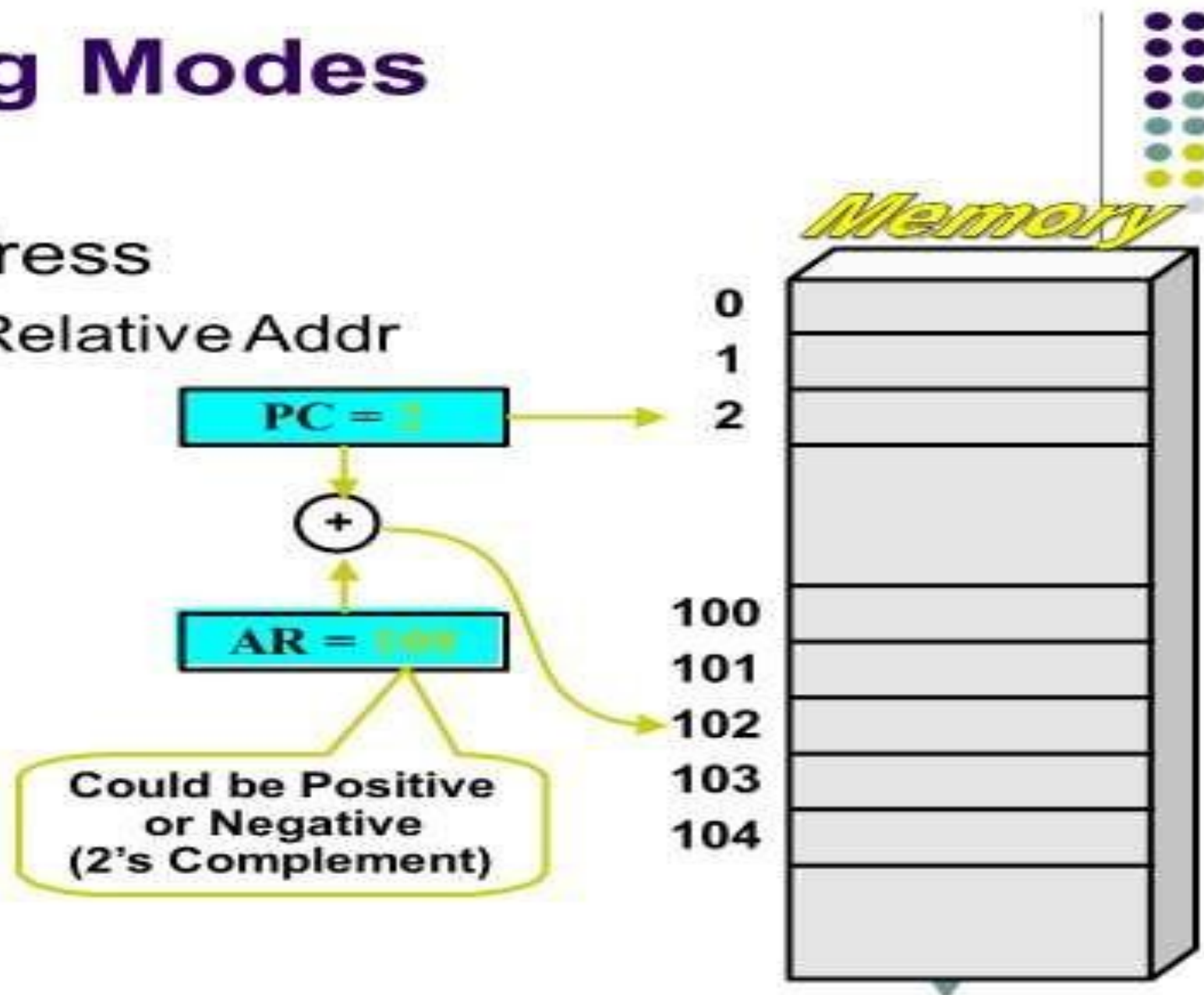
- IndirectAddress: Indicate the memory location that holds the address of the memory location that holds the data



Relative Address

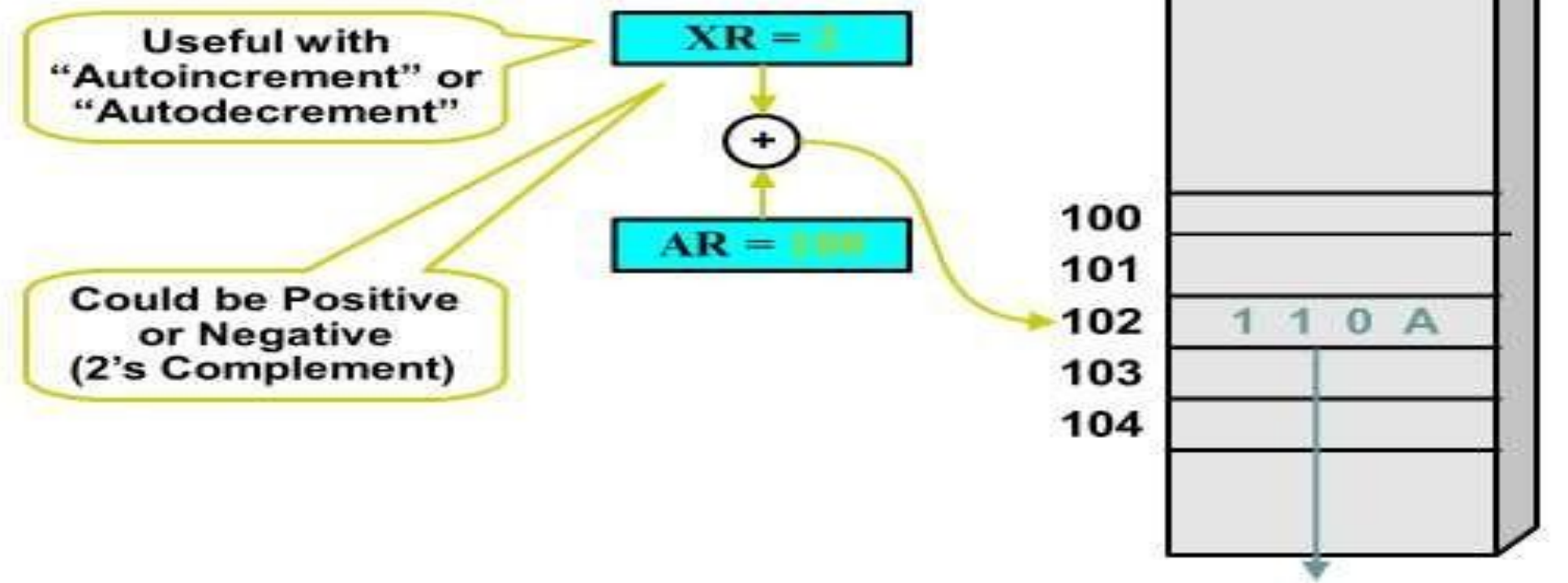
Addressing Modes

- Relative Address
 - $EA = PC + \text{Relative Addr}$



Addressing Modes

- Indexed
 - $EA = \text{Index Register} + \text{Relative Addr}$

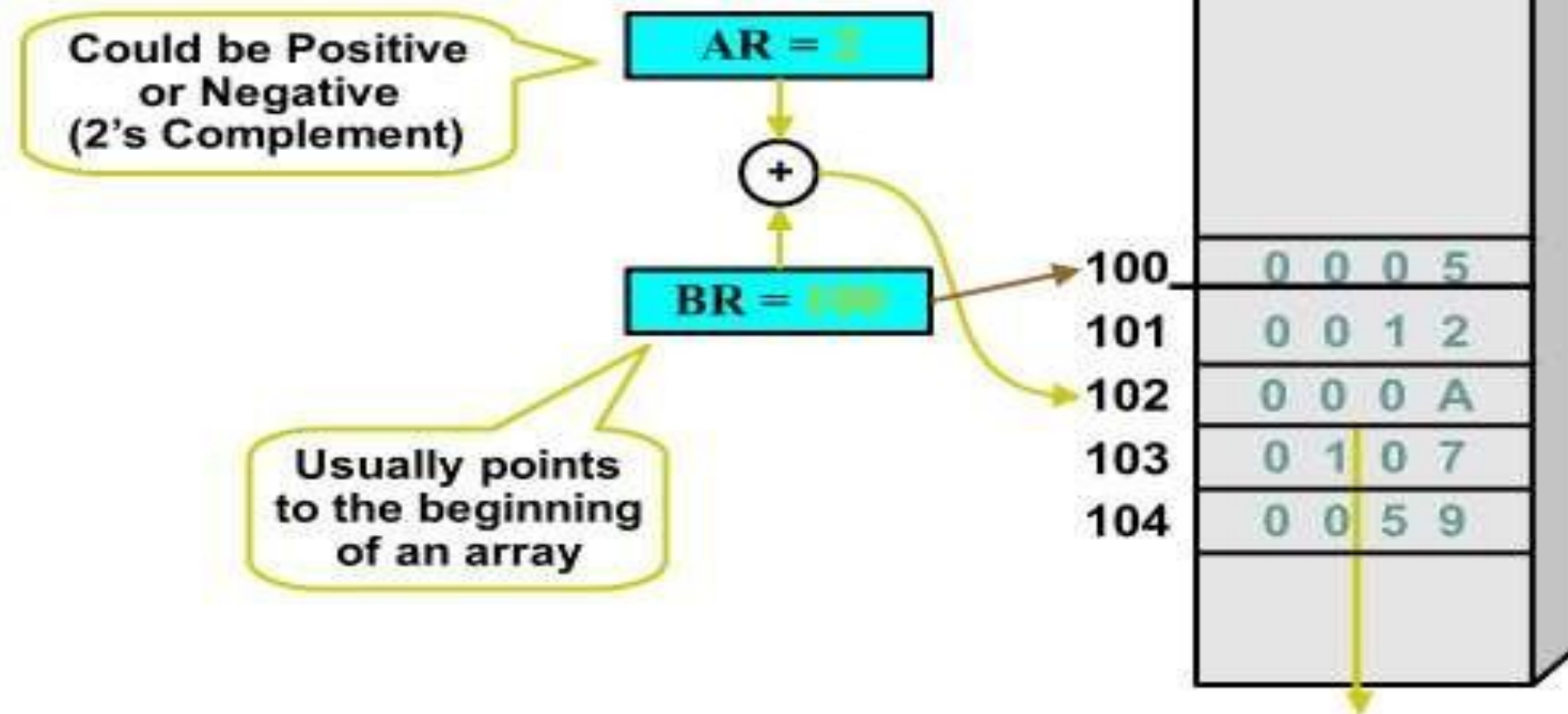


Base register

Addressing Modes

Base Register

- $EA = \text{Base Register} + \text{Relative Addr}$



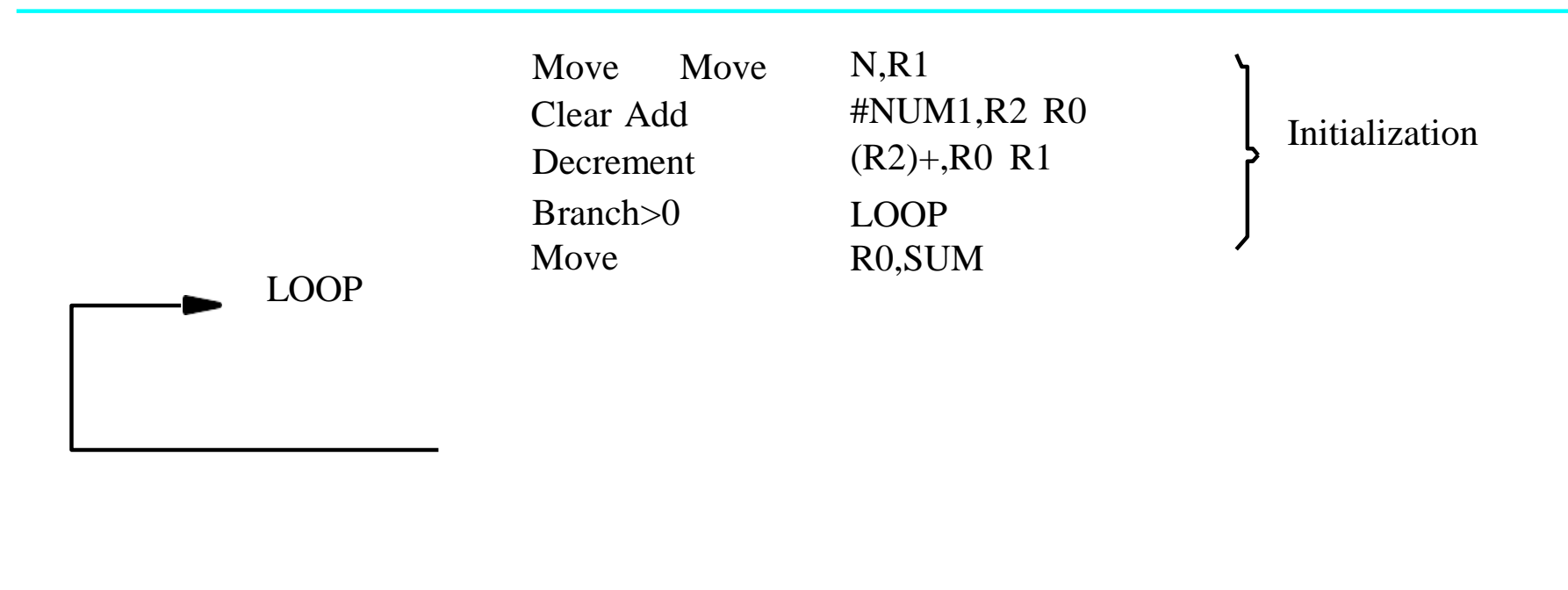
Address mode types

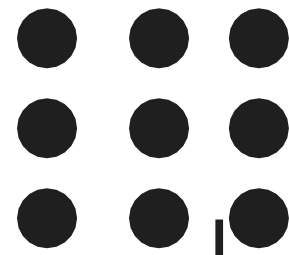
- The different ways in which the location of an operand is specified in an instruction are referred to as addressing modes.

Name	Assembler syntax	Addressing function
Immediate	#Value	Operand = Value
Register	R_i	$EA = R_i$
Absolute(Direct)	LOC	$EA = LOC$
Indirect	(R_i)	$EA = [R_i]$
	(LOC)	$EA = [LOC]$
Index	$X(R_i)$	$EA = [R_i] + X$
Basewith index	(R_i, R_j)	$EA = [R_i] + [R_j]$
Basewith index and offset	$X(R_i, R_j)$	$EA = [R_i] + [R_j] + X$
Relative	$X(PC)$	$EA = [PC] + X$
Autoincrement	$(R_i)+$	$EA = [R_i]$; Increment R_i
Autodecrement	$-(R_i)$	Decrement R_i ; $EA = [R_i]$

Additional Modes

- Autoincrement mode – the effective address of the operand is the contents of a register specified in the instruction. After accessing the operand, the contents of this register are automatically incremented to point to the next item in a list.
- $(R_i)+$. The increment is 1 for byte-sized operands, 2 for 16-bit operands, and 4 for 32-bit operands.
- Autodecrement mode: $-(R_i)$ – decrement first





THANK YOU