Slides for lecture 4

CE/CZ 3001:
Advanced Computer Architecture

(Module 2: Instruction Set Architecture Design)

Dr Smitha K. G.
School of Computer Science
And Engineering

Pre- Video Summary

- ARMv8 ISA: A design example named as LEGv8
  - Instruction format
    - R-format, D-format, I-format, B format and CB fomat
  - Addressing modes
    - Register addressing
    - Immediate addressing
    - Base or displacement addressing
    - PC-relative addressing

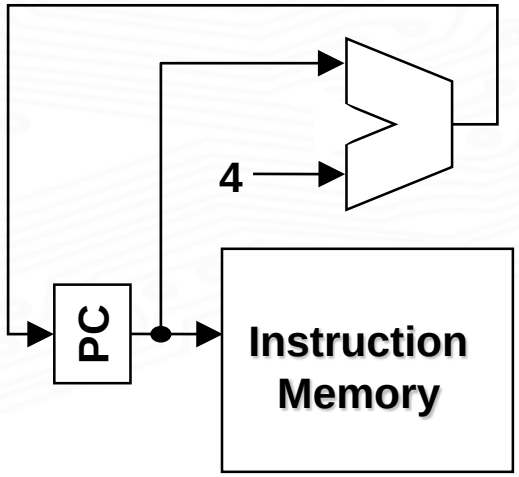
Classification of LEGv8 Instruction based on Instruction format -Register (R) type

- All data values (operands/result) are located in registers.
- **Rn** and **Rm** : specify the first and the second source registers
- **Rd:** specifies the destination register
- **shamt** stands for shift amount: specifies the number of bit positions to be shifted (used in shift instructions.)
- **Opcode**: Operation Code: specifies the type of instruction.

Register (R) type – Register Addressing

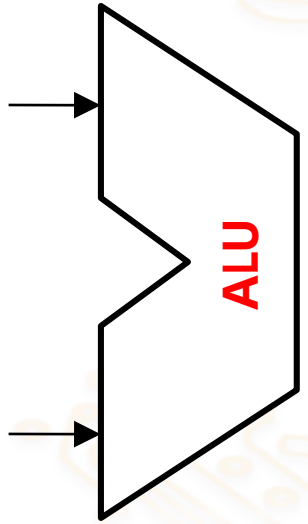
- Instruction structure: <operation> <Rd>, <Rn>, <Rm>
- Eg: ADD X9, X20, X21 meaning: [X9]← [X20]+[X21]

opcode	21_{ten}	shamt	20_{ten}	9_{ten}
	Rm		Rn	Rd

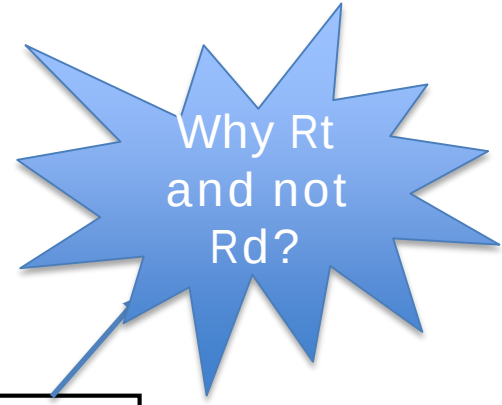


Control unit

0	…….
9	65
19	…….
20	50
21	15
22	
	…….
31	



LEGv8 Instruction format – Data transfer (D) type



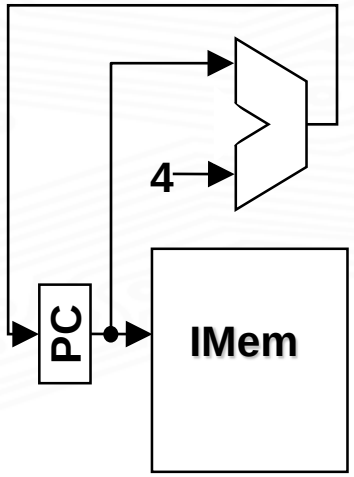
opcode	address	op2	Rn	Rt
11 bits	9 bits	2 bits	5 bits	5 bits

- Load/Store instructions,
- **Rn**: specify the base registers
- "address": constant **offset** from contents of base register **(+/- 32 doublewords)**
- **Rt**: specifies the destination register (load) or source (store) register number
  - ALU calculates the address (address+[Rn])
  - Data at memory location (address+[Rn]) is read
- The read data is saved in the destination register (Rt).

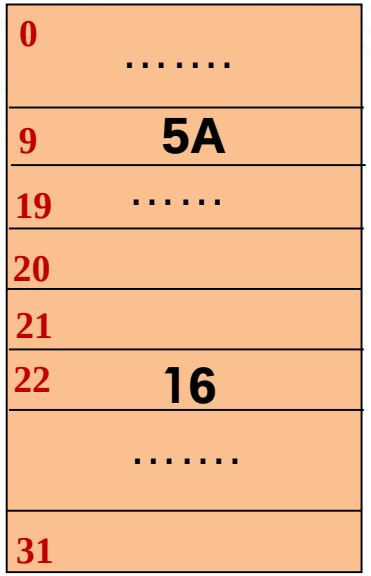
Base/ Displacement Addressing (load word)

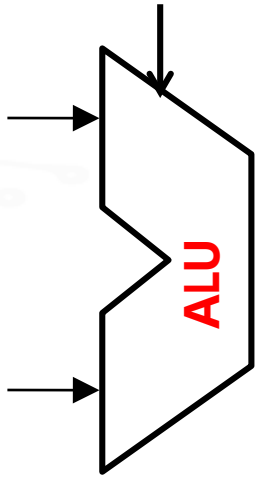
1986	64	0	22	9
Opcode 11 bits	Address 9 bits	Op2 2 bits	Rn 5 bits	Rt 5 bits

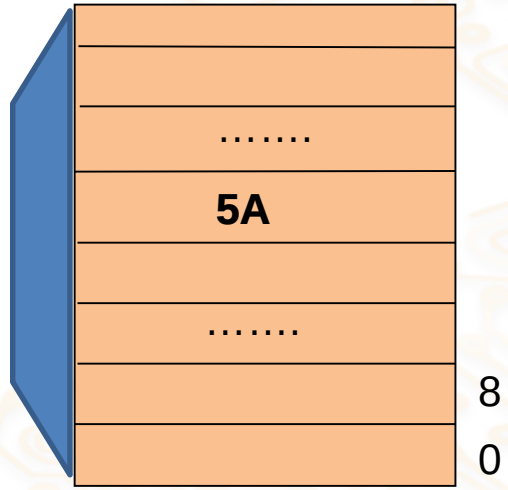
LDUR X9, [X22, #64]:
meaning
[X9] ← mem[[R22] + 64]



Control unit



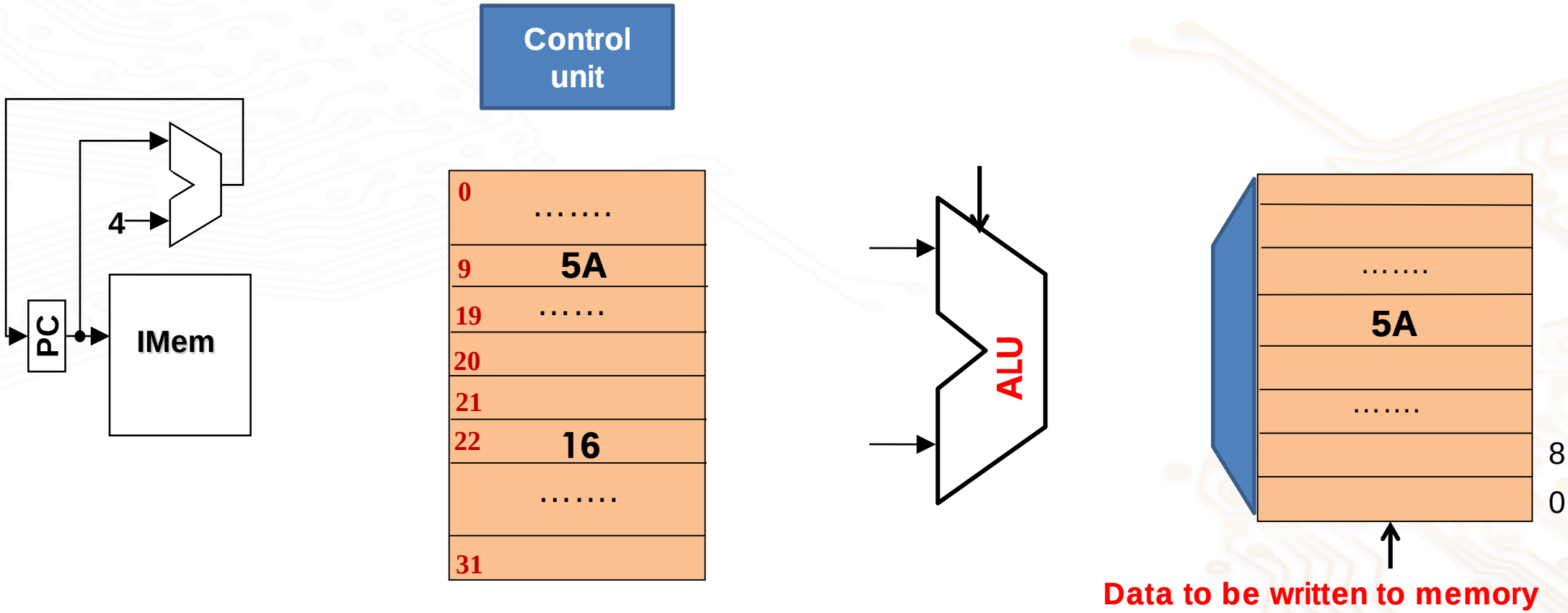




Base/displacement Addressing (store word)

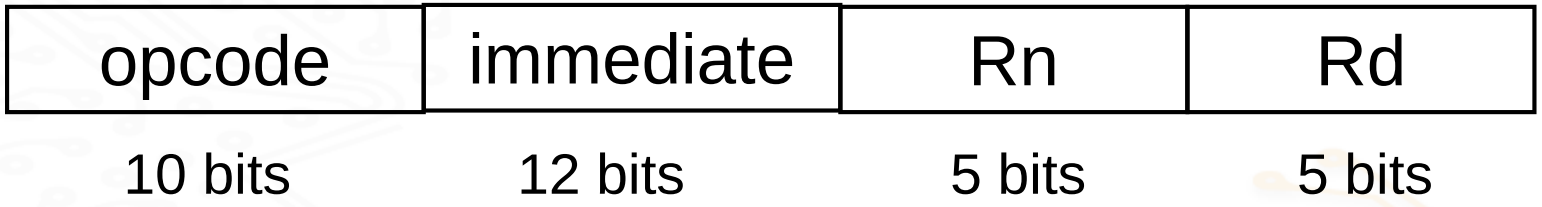
1984	64	0	22	9
Opcode 11 bits	Address 9 bits	Op2 2 bits	Rn 5 bits	Rt 5 bits

STUR X9, [X22, #64]:
meaning
[X9] → mem[[R22] + 64]

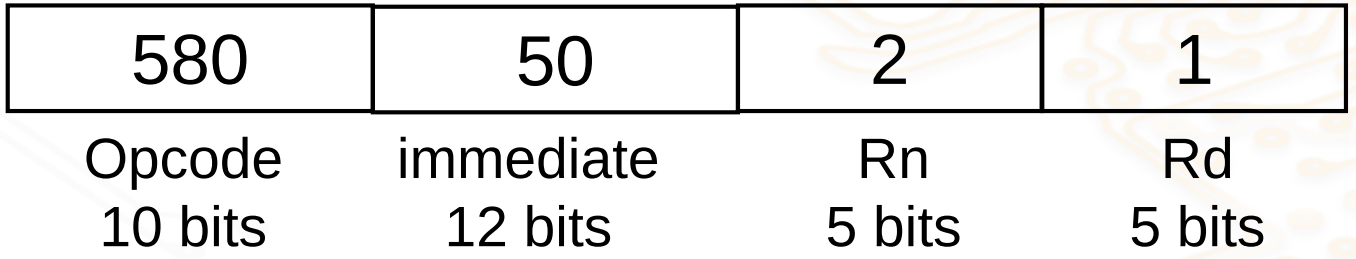


- Question : what is the max address and min address possible for base displacement?

LEGv8 Instruction format – Immediate(I) type



- Rn: source register
- Rd: destination register

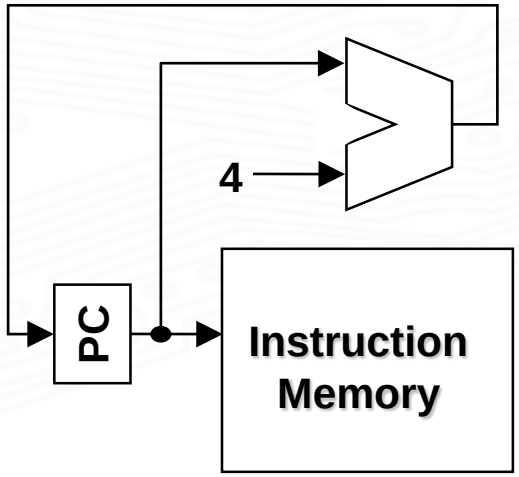


- Immediate field is zero-extended
- Opcode is reduced to 10 bits to have more range for immediate field.
  - Example: ADDI X1, X2, #50 ([X1] ← [X2]+ 50)

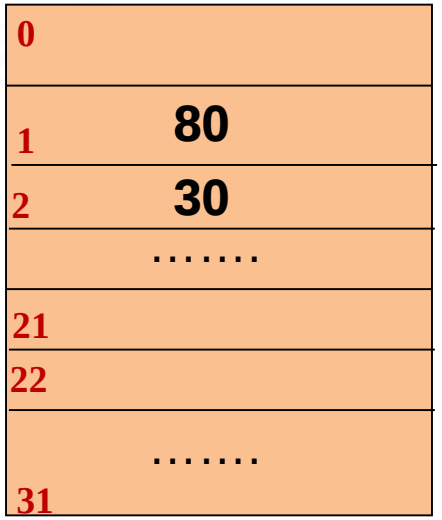
Immediate (I) type – Immediate Addressing

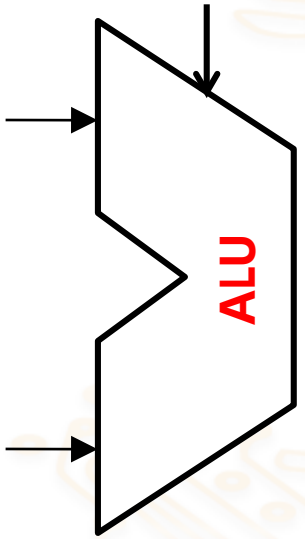
- Example: **ADDI X1, X2, #50:**
  meaning [X1] ←[X2] + 50

580	50	2	1
Opcode 10 bits	immediate 12 bits	Rn 5 bits	Rd 5 bits



Control unit





LEGv8 Instruction format – conditional branch (CB) type

- Based on a condition
  - CBZ (if register value is zero, then branch)
  - CBNZ (if register value is nonzero, then branch)

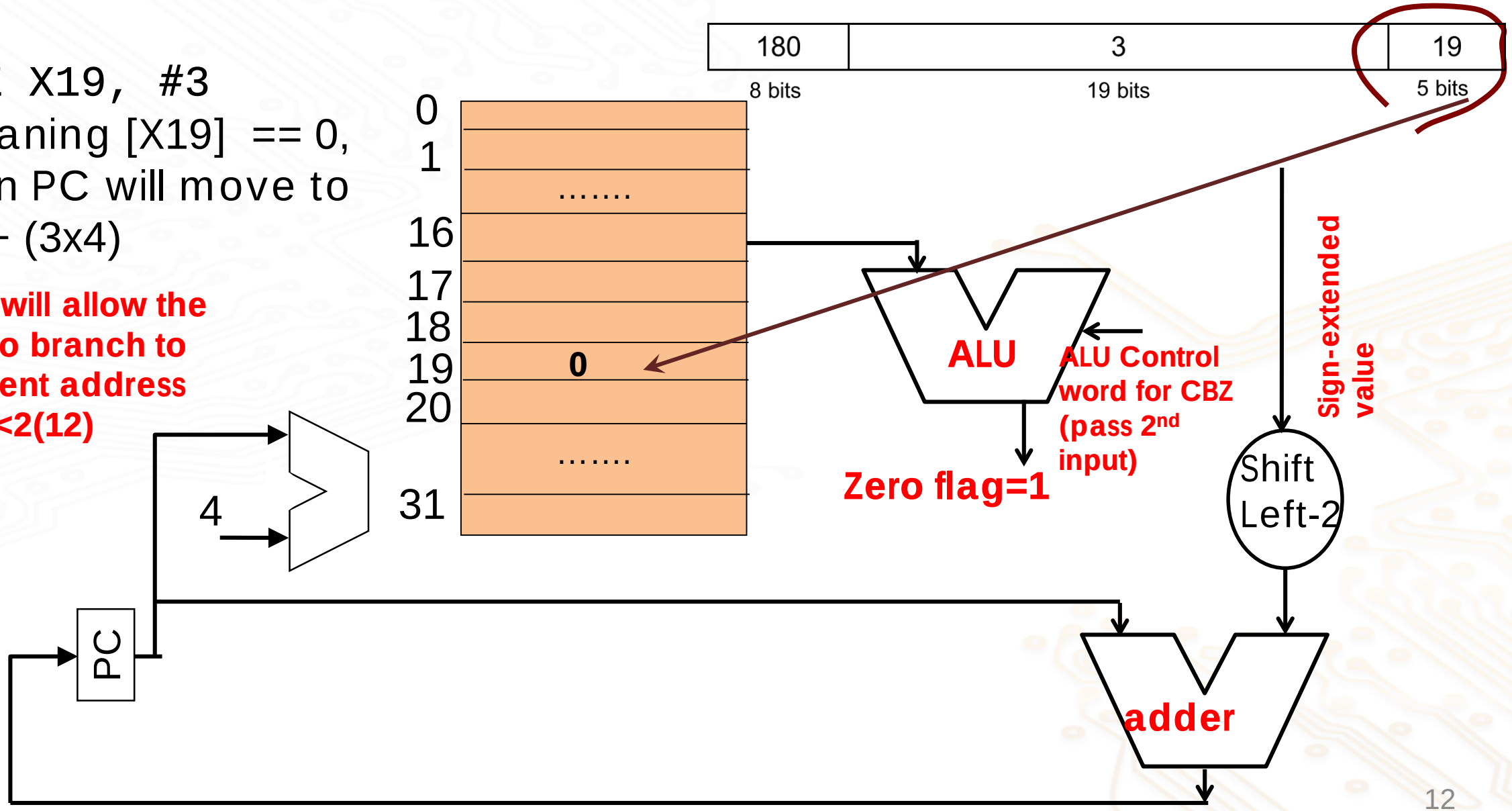
- Example: CBZ X19, Exit // go to Exit if X19 == 0

- Addressing mode is (Program Counter) PC-relative
  - Address = PC + address (from instruction)

PC relative addressing (Part 1/2)

CBZ X19, #3
meaning [X19] == 0,
then PC will move to
PC+ (3x4)

This will allow the PC to branch to current address +3<<2(12)



PC relative addressing (Part 2/2)

```
SUB X9,X22,X23
CBNZ X9,Else
ADD X19,X20,X21
B Exit
Else:SUB X9,X22,x23
Exit: …
```

`CBNZ X9,#3` **meaning [X9] ≠ 0, then PC will move to PC+ (3x4)**

PC Address	Instruction mem
0x0000000000000000	`SUB X9,X22,X23`
0x0000000000000004	`CBNZ X9,#3`
0x0000000000000008	`ADD X19,X20,X21`
0x000000000000000C	`B Exit`
0x0000000000000010	`SUB X9,X22,x23`
0x0000000000000014	……
……	……
0xFFFFFFFFFFFFFFF4	……
0xFFFFFFFFFFFFFFF8	……
0xFFFFFFFFFFFFFFFC	……

Else:

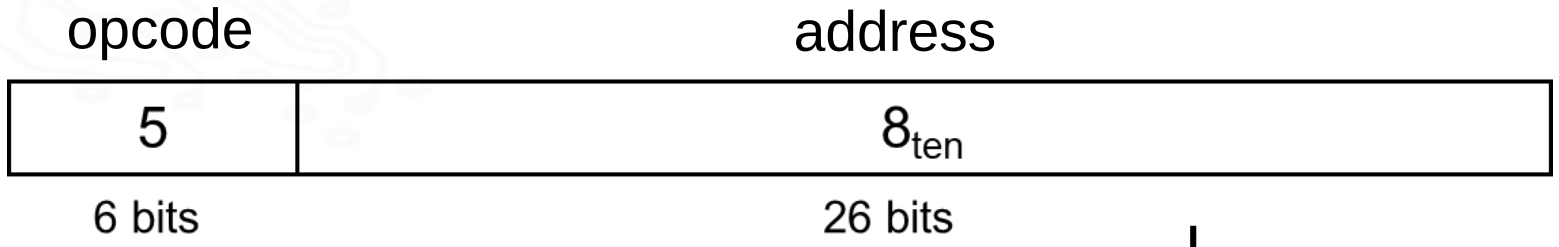
Exit:

Step 1 → [X9] ≠ 0, then nPC=PC+(3x4)

Skip instructions

newPC (nPC) points to this location

PC relative addressing mode:- Unconditional branch (B) type




```
PC                  Instruction
0X0000000000000004  B #8
```

then PC will move to PC+ (8x4)=4 + 32= $36_{ten}(24_{hex})$

14

- Find the address to which CBZ and B branches to?

Address	Instruction in mnemonic
0x....00000	EOR X2,X3,X4
0x....00004	**CBZ X2, #4**
0x....00008	ADD X5,X1,X0
0x....0000C	SUB X6,X7,X8
0x....00010	ADD X31,X31,X31
0x....00014	ADD X0,X31,X31
0x....00018	**B #-3**
0x....0001C	**……**

Addressing mode summary