Computing System Design

Virendra Singh

Professor

Computer Architecture and Dependable Systems Lab

Department of Electrical Engineering

Indian Institute of Technology Bombay

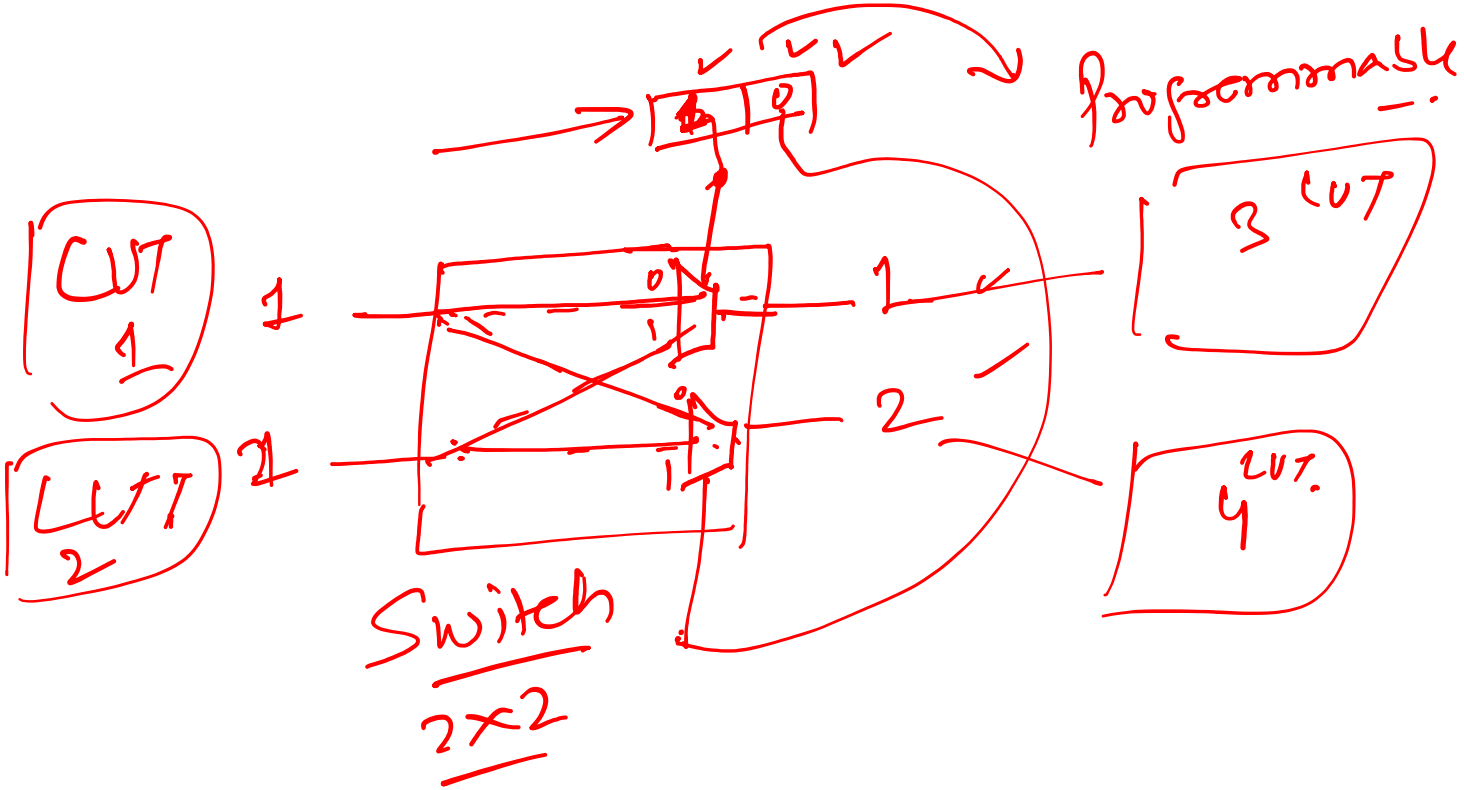
http://www.ee.iitb.ac.in/~viren/

E-mail: viren@ee.iitb.ac.in

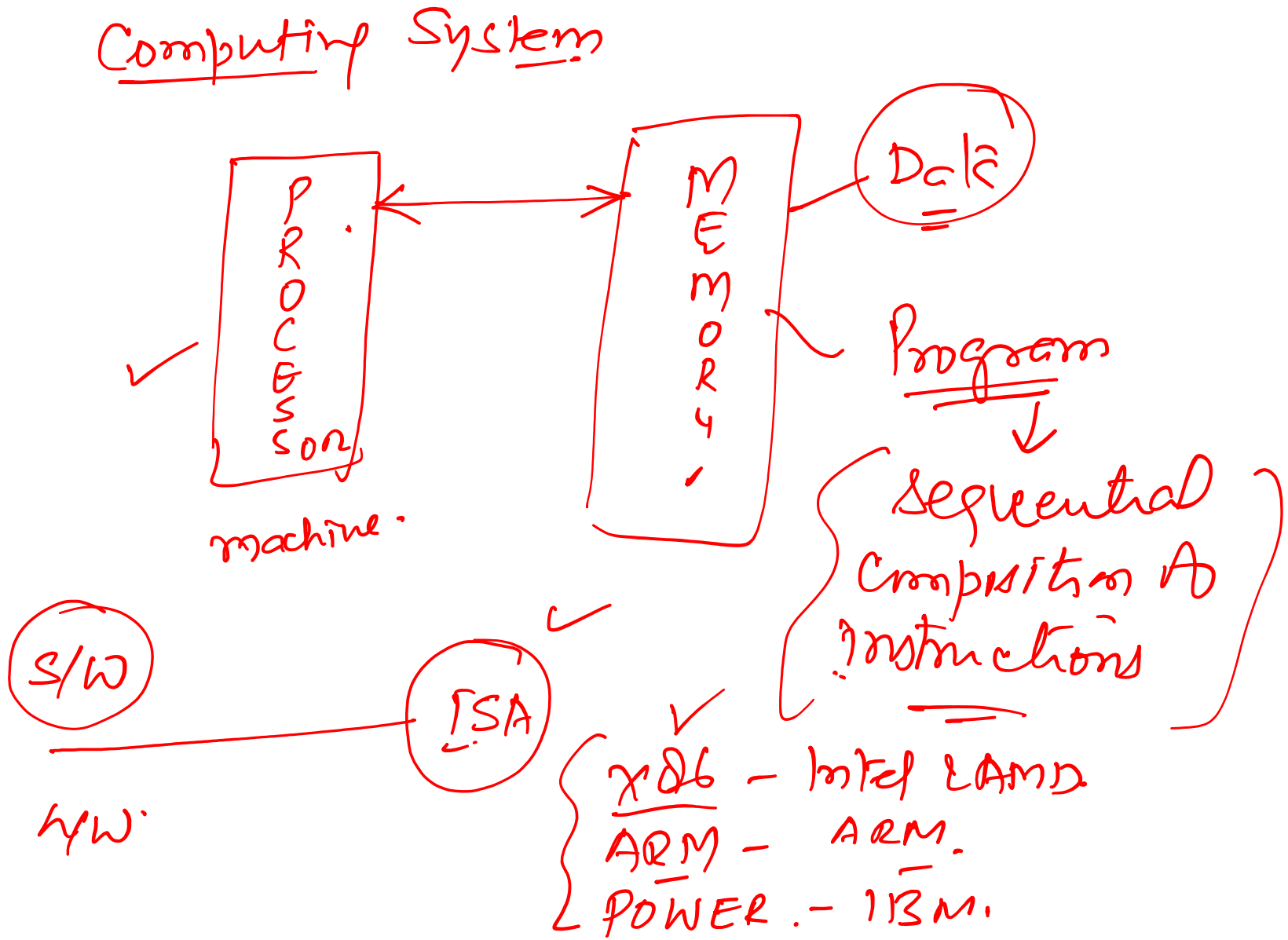
CS-226: Digital Logic Design

Lecture 29: 15 April 2021

CADSL



FPGA


Computing System
PROCESSOR
machine.
MEMORY
Data
Program
Sequential Composition of Instructions
S/W
ISA
H/W.
x86 - Intel, AMD
ARM - ARM.
POWER - IBM.

Logical operations / Memory communication /

Control flow change (branches)

Arithmetic

fn( )

if (i == j) then { }
else { }

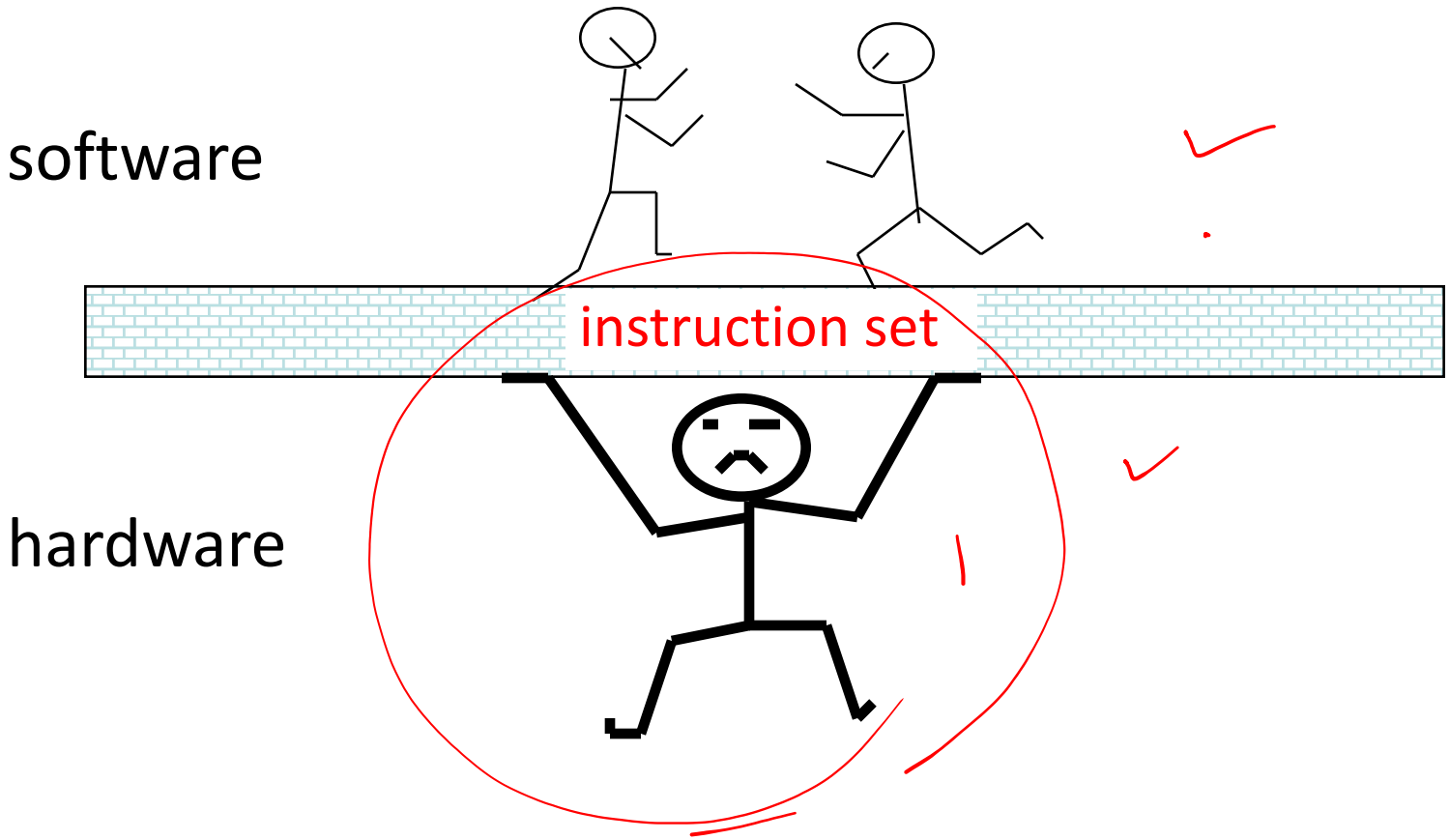
Generic Circuit

Instruction Set Architecture

- Instruction set architecture is the structure of a computer that a machine language programmer must understand to write a correct (timing independent) program for that machine.

- The instruction set architecture is also the machine description that a hardware designer must understand to design a correct implementation of the computer.

Instruction Set Architecture (ISA)



Instruction

a = b + c + d + e

$t_1 = b + c$

$t_2 = t_1 + d$

$t_3 = t_2 + e$

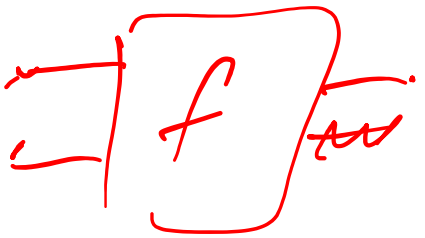
Instruction

- logical / arithmetic
- memory Comm
- control flow.

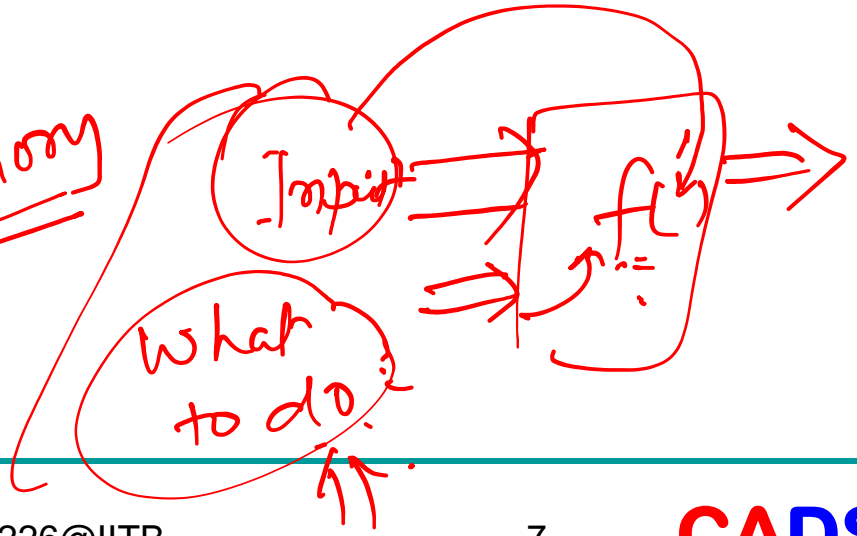
FORMAT

1. what operation
2. who are operands
3. where to place the result

$a = b + c$



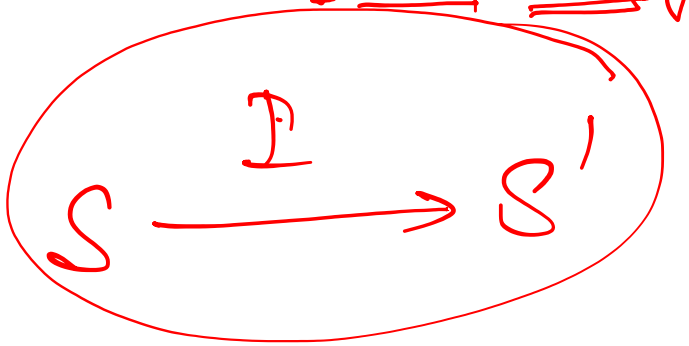
Memory

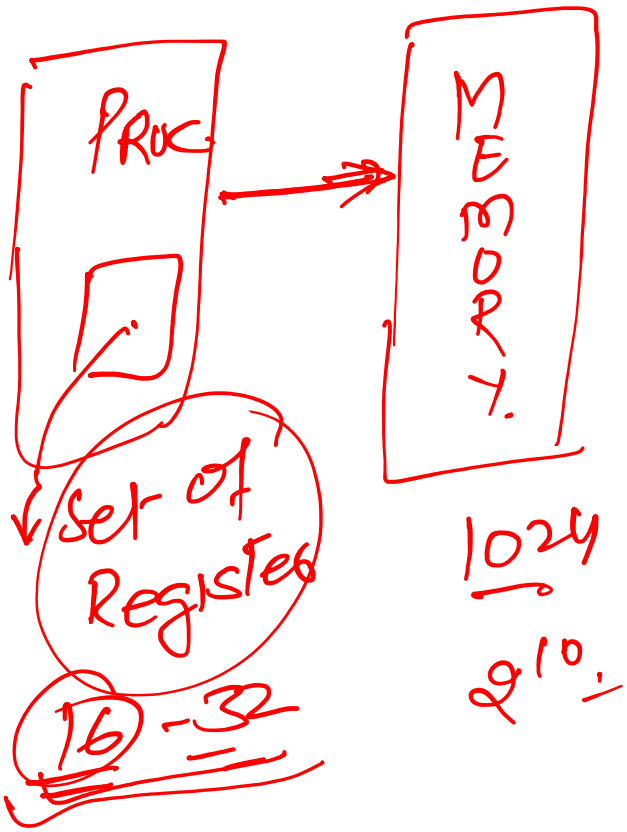


REGISTER MEMORY

Instruction

State
- content of Memory
- Content of Registers





Instruction

Addressing modes

a = b+1

Reg =

a = 5;

OPERATION	OPERAND1	OPERAND2	DEST

Memory — 200

Register — R5

a = b+c
a = b-c
a = b&c
a = ~b

1. ADD / SUB / AND / NOT

2. LW (Read from memory)

load R1, @ → Memory location 500

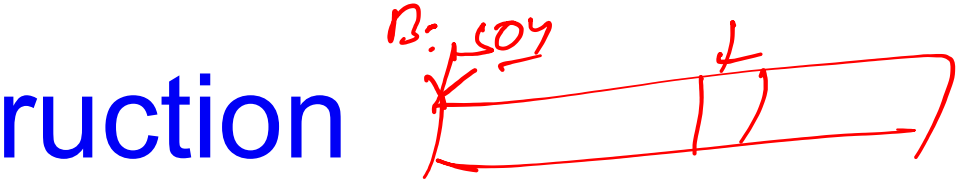
R1 = a;

3. SW (Write to memory)

sw R1, a

a = R1

Instruction



A(0)

A(5)

520

Base + offset

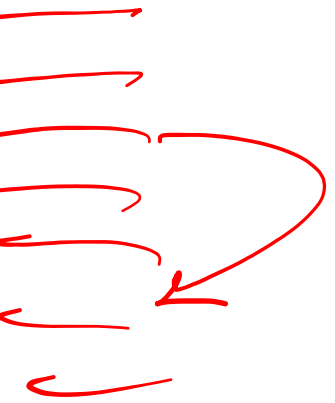
Base address

(3) Jump location

BZ R_1, R_2 loc.

if $R_1 == R_2$ then move to loc

otherwise Sep. next

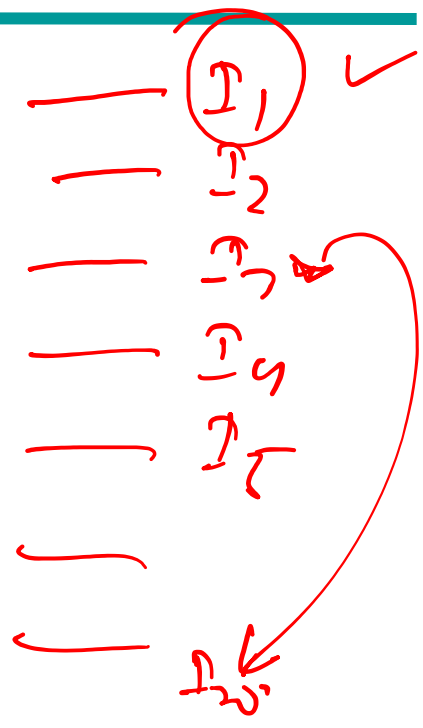


Instruction

Program counter / instruction pointer (IP)
(PC)

tracks where we are

(keeps pointer to the next instruction)



Instructions → logical / Arithmetic
→ Memory Read / Write
→ Control flow

ADD / SUB / AND / NOT / LW / SW / J / BEQ.

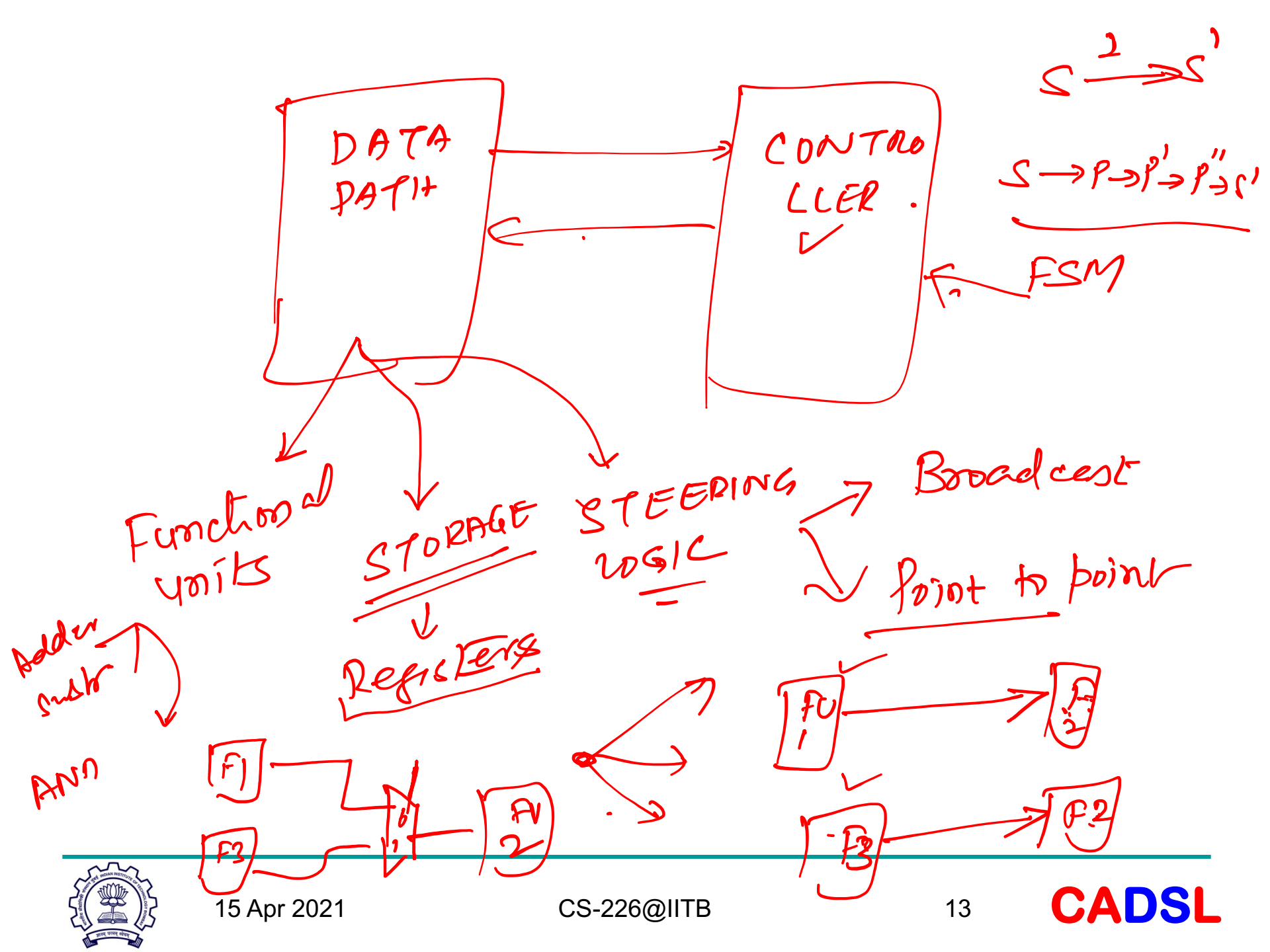
FORMAT

Generic hardware (Processor)

what should be the components.

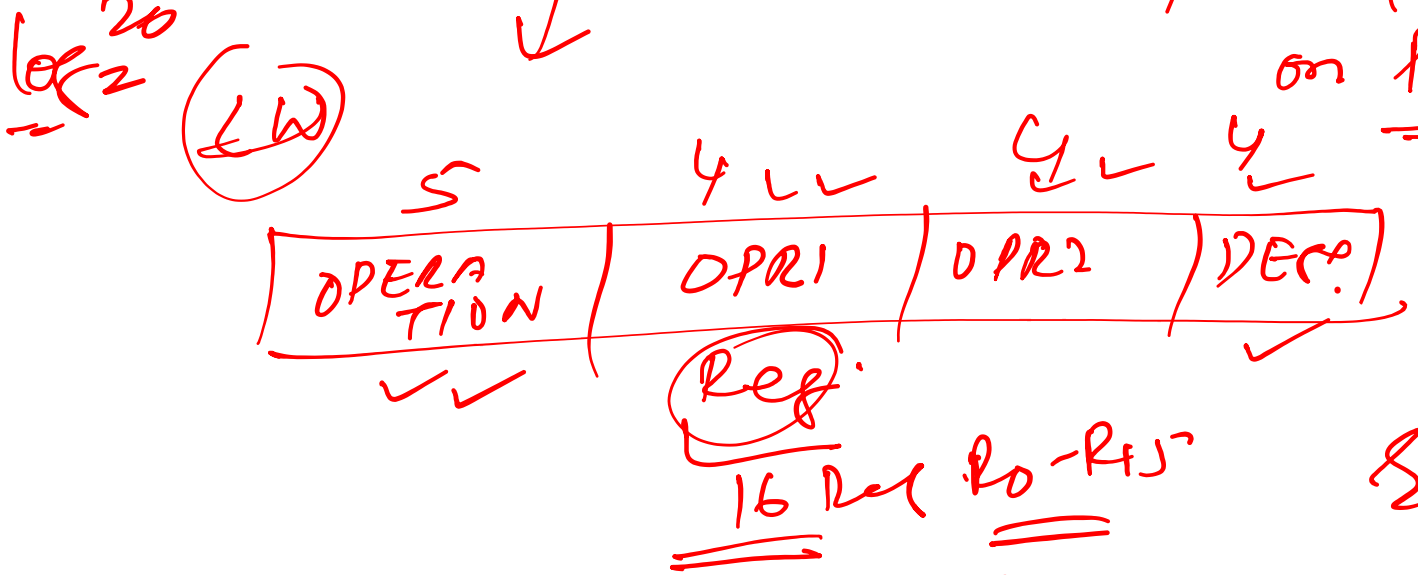
① Resources to perform the job] DATAPATH

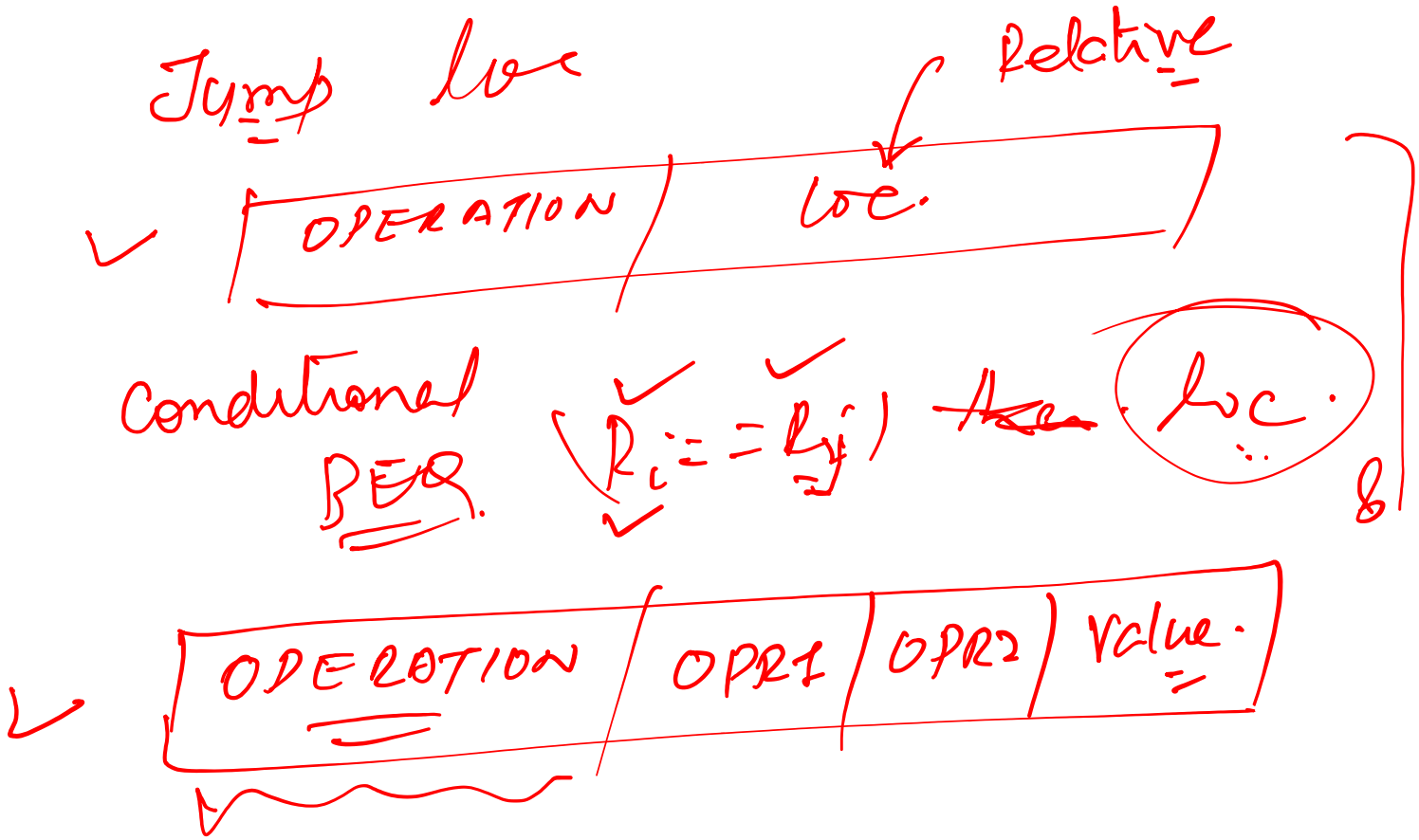
② Mechanism to instruct → CONTROLLER

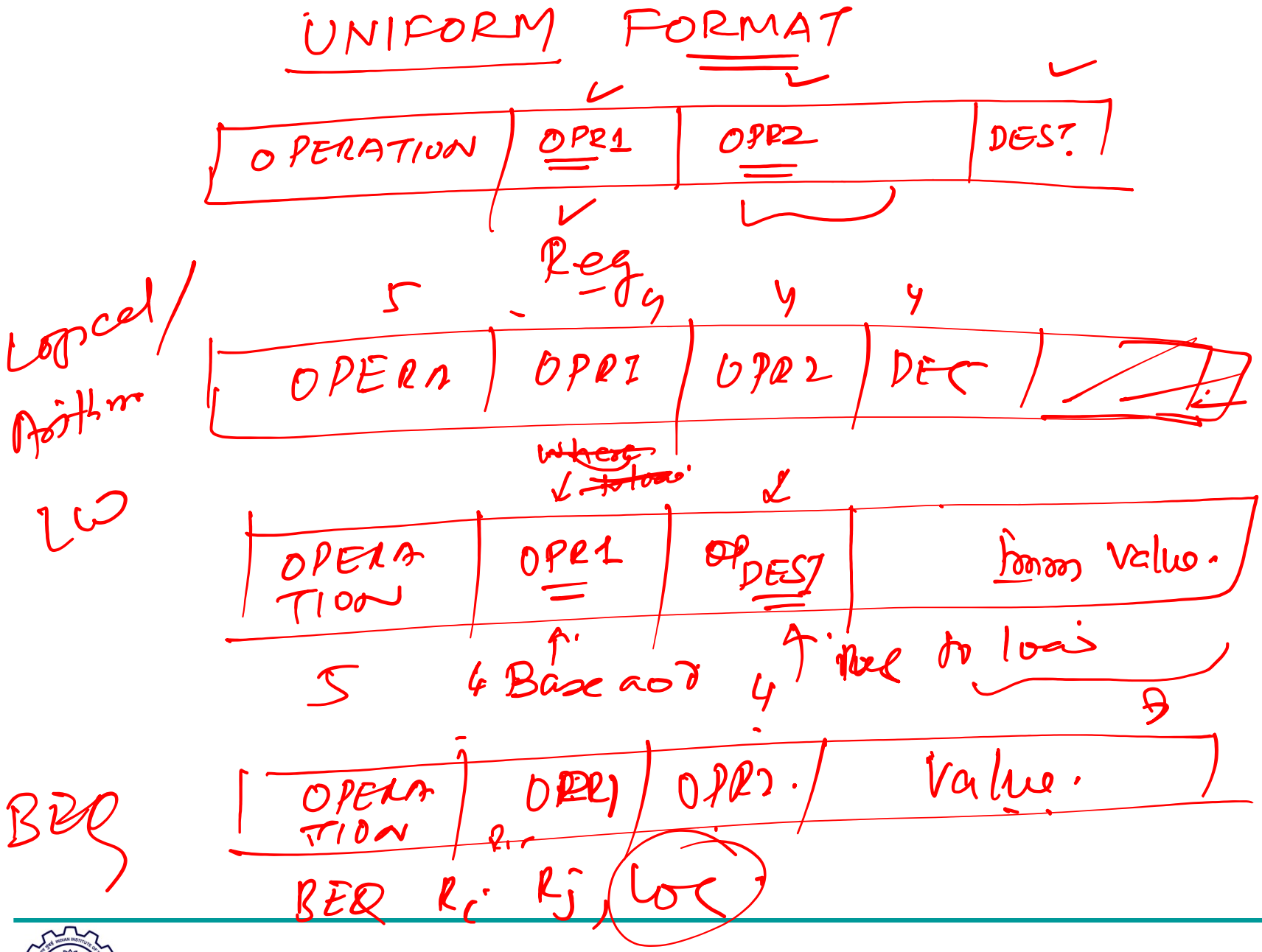

DATA PATH
CONTROLLER
FSM
Functional units
STORAGE
Registers
STEERING LOGIC
Broadcast
Point to point
Adder
sub
AND
F1
F3
FU 1
FU 2
F2
F3

ADD / AND / SUB

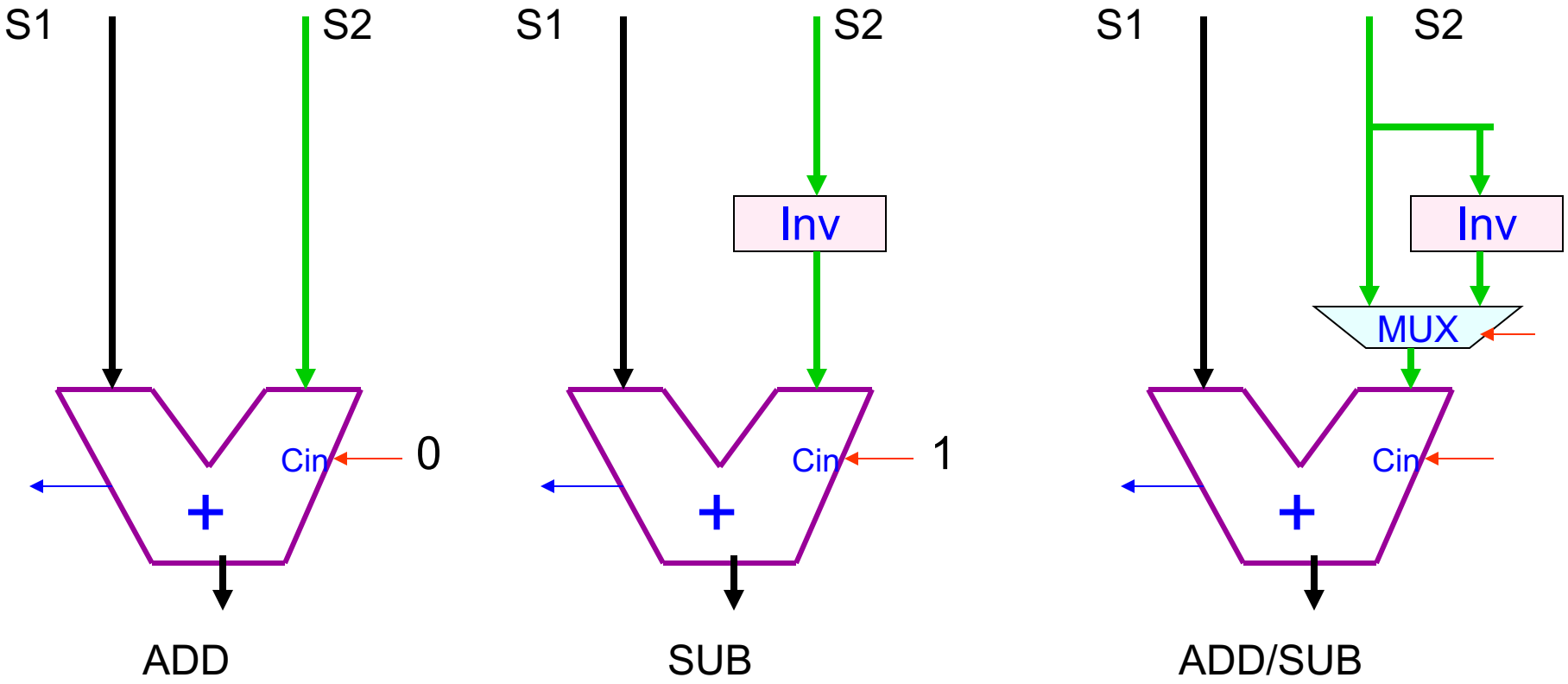
← all operands are in a set of Registers on Processor.



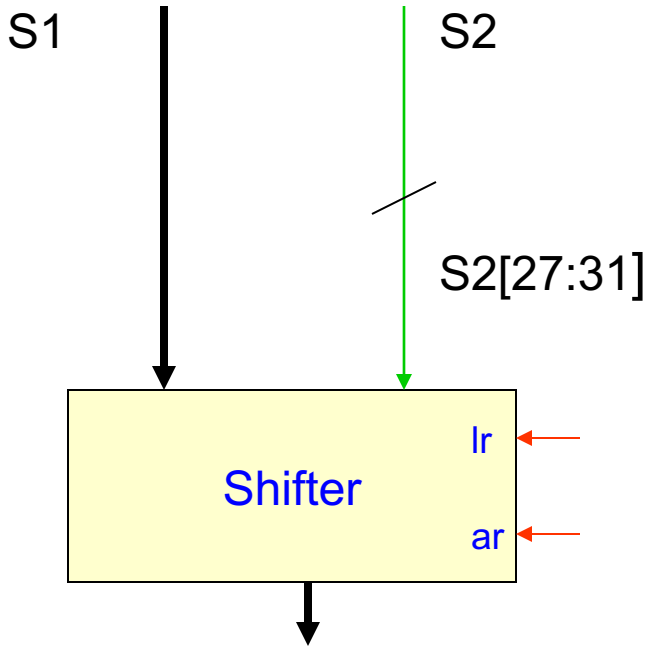

Jump loc
Relative
OPERATION
loc.
Conditional
BEQ.
(Ri == Rj)
loc.
OPERATION
OPR1
OPR2
Value.


UNIFORM FORMAT
OPERATION
OPR1
OPR2
DEST
Reg
5
4
4
4
Logical/
Arithm
OPERA
OPR1
OPR2
DEC
LW
OPERA TION
OPR1
DEST
Immm Value.
5
4 Base add
4
Ref to load
BEQ
OPERA TION
OPR1
OPR2.
Value.
BEQ Ri Rj, Loc

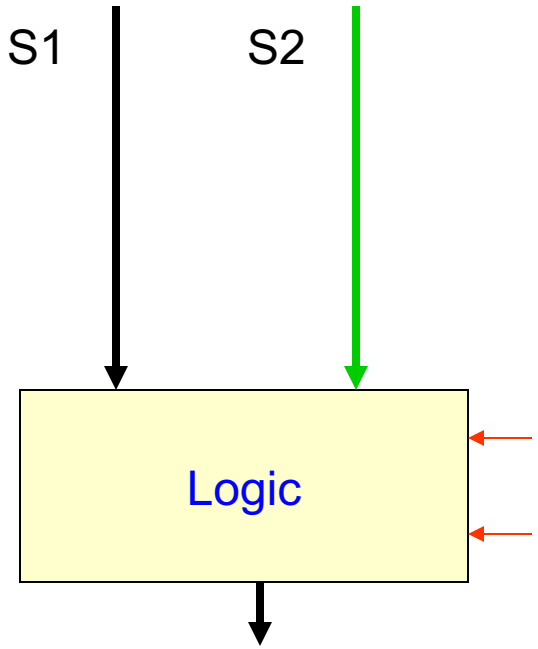
Arithmetic Logic Unit

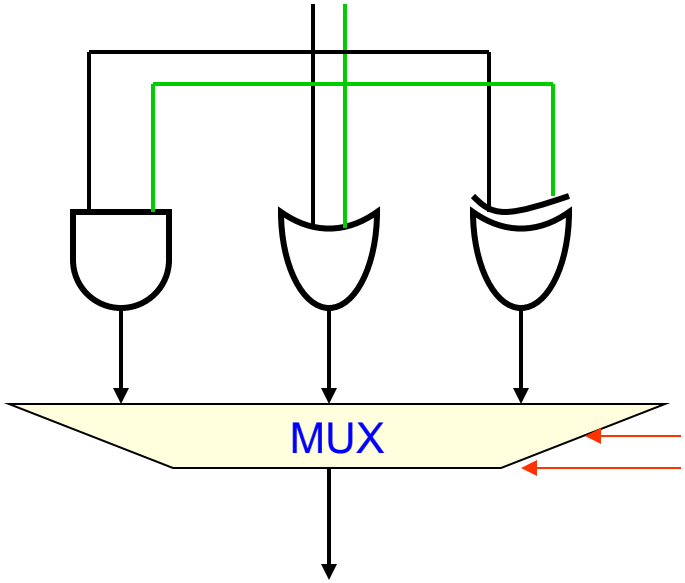


Arithmetic Logic Unit

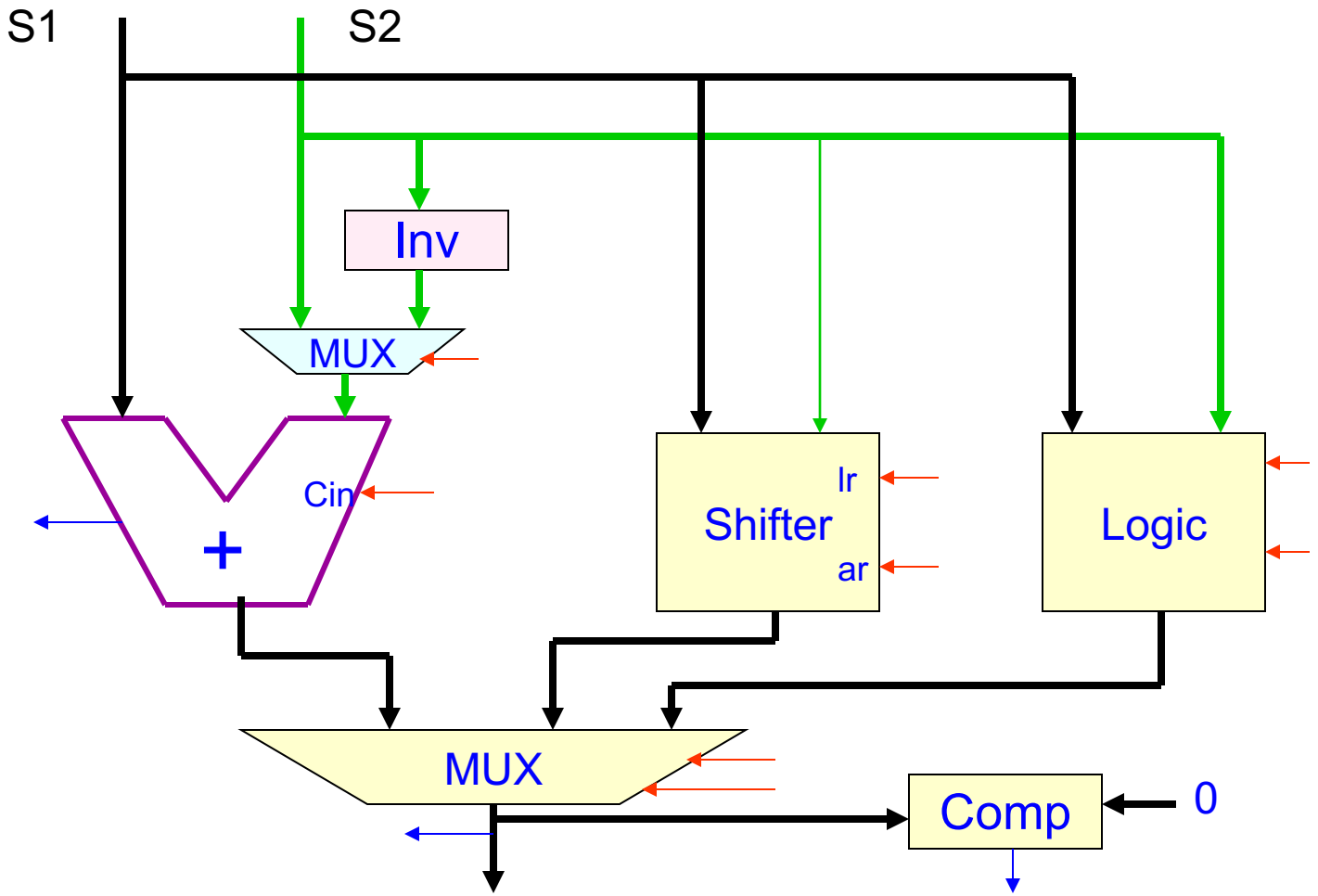


Arithmetic Logic Unit

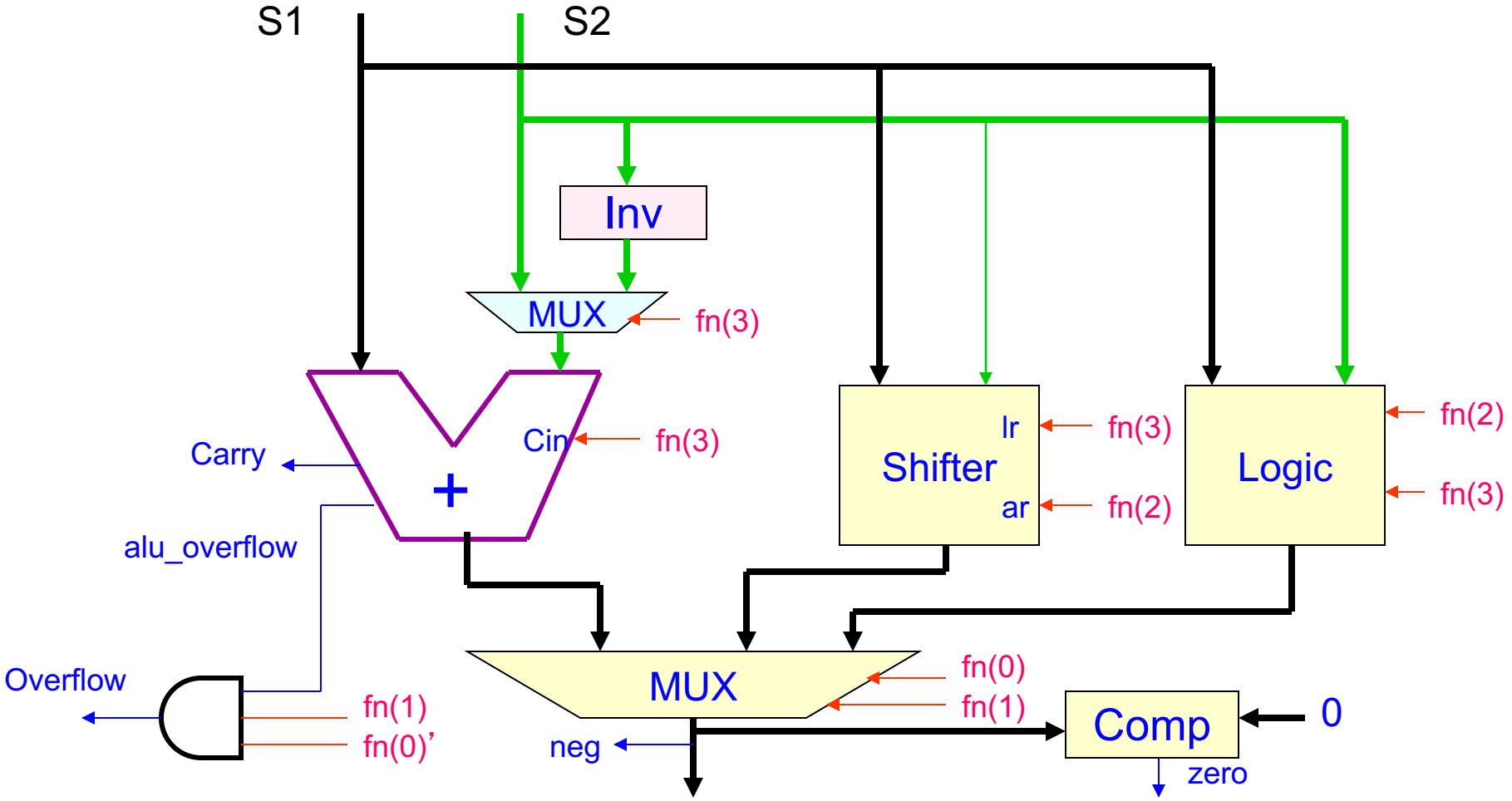




Arithmetic Logic Unit



Arithmetic Logic Unit



Thank You