

RISC Design: DLX – HFC Method

Virendra Singh

Associate 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

EE-309: Microprocessors



Lecture

CADSL

Overview of DLX

- ❖ simple instructions, all 32 bits wide
- ❖ very structured, no unnecessary baggage
- ❖ only three instruction formats

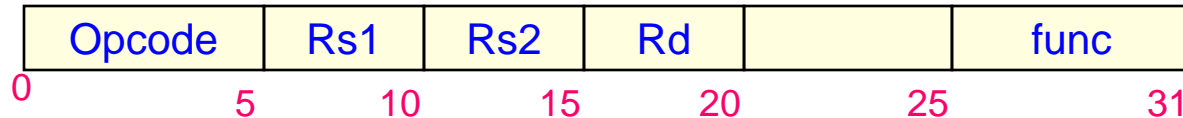
R	op	rs1	rs2	rd	funct
I	op	rs1	rd	16 bit address	
J	op	26 bit address			

- ❖ rely on compiler to achieve performance



Instruction Set

Register-Register Instructions



Arithmetic and Logical Instruction

- ADD Rd, Rs1, Rs2 $\text{Regs}[\text{Rd}] \leftarrow \text{Reg}[\text{Rs1}] + \text{Reg}[\text{Rs2}]$
- SUB Rd, Rs1, Rs2 $\text{Regs}[\text{Rd}] \leftarrow \text{Reg}[\text{Rs1}] - \text{Reg}[\text{Rs2}]$
- AND Rd, Rs1, Rs2 $\text{Regs}[\text{Rd}] \leftarrow \text{Reg}[\text{Rs1}] \text{ and } \text{Reg}[\text{Rs2}]$
- OR Rd, Rs1, Rs2 $\text{Regs}[\text{Rd}] \leftarrow \text{Reg}[\text{Rs1}] \text{ or } \text{Reg}[\text{Rs2}]$
- XOR Rd, Rs1, Rs2 $\text{Regs}[\text{Rd}] \leftarrow \text{Reg}[\text{Rs1}] \text{ xor } \text{Reg}[\text{Rs2}]$
- SUB Rd, Rs1, Rs2 $\text{Regs}[\text{Rd}] \leftarrow \text{Reg}[\text{Rs1}] - \text{Reg}[\text{Rs2}]$



DLX Instruction Set

ADD Rd, Rs1, Rs2	$Rd \leftarrow Rs1 + Rs2$ (overflow – exception)	R	000_000 000_100
SUB Rd, Rs1, Rs2	$Rd \leftarrow Rs1 - Rs2$ (overflow – exception)	R	000_000 000_110
AND Rd, Rs1, Rs2	$Rd \leftarrow Rs1 \text{ and } Rs2$	R	000_000/ 001_000
OR Rd, Rs1, Rs2	$Rd \leftarrow Rs1 \text{ or } Rs2$	R	000_000/ 001_001
XOR Rd, Rs1, Rs2	$Rd \leftarrow Rs1 \text{ xor } Rs2$	R	000_000/ 001_010
SLL Rd, Rs1, Rs2	$Rd \leftarrow Rs1 << Rs2$ (logical) (5 lsb of Rs2 are significant)	R	000_000 001_100
SRL Rd, Rs1, Rs2	$Rd \leftarrow Rs1 >> Rs2$ (logical) (5 lsb of Rs2 are significant)	R	000_000 001_110
SRA Rd, Rs1, Rs2	$Rd \leftarrow Rs1 >> Rs2$ (arithmetic) (5 lsb of Rs2 are significant)	R	000_000 001_111



DLX Instruction Set

ADDI Rd, Rs1, Imm	$Rd \leftarrow Rs1 + Imm$ (sign extended) (overflow – exception)	I	010_100
SUBI Rd, Rs1, Imm	$Rd \leftarrow Rs1 - Imm$ (sign extended) (overflow – exception)	I	010_110
ANDI Rd, Rs1, Imm	$Rd \leftarrow Rs1 \text{ and } Imm$ (zero extended)	I	011_000
ORI Rd, Rs1, Imm	$Rd \leftarrow Rs1 \text{ or } Imm$ (zero extended)	I	011_001
XORI Rd, Rs1, Imm	$Rd \leftarrow Rs1 \text{ xor } Imm$ (zero extended)	I	011_010
SLLI Rd, Rs1, Imm	$Rd \leftarrow Rs1 \ll Imm$ (logical) (5 lsb of Imm are significant)	I	011_100
SRLI Rd, Rs1, Imm	$Rd \leftarrow Rs1 \gg Imm$ (logical) (5 lsb of Imm are significant)	I	011_110
SRAI Rd, Rs1, Imm	$Rd \leftarrow Rs1 \ggg Imm$ (arithmetic) (5 lsb of Imm are significant)	I	011_111



DLX Instruction Set

LHI Rd, Imm	$Rd(0:15) \leftarrow Imm$ $Rd(16:32) \leftarrow \text{hex}0000$ (Imm: 16 bit immediate)	I	011_011
NOP	Do nothing	R	000_000 000_000



DLX Instruction Set

SEQ Rd, Rs1, Rs2	Rs1 = Rs2: Rd \leftarrow hex0000_0001 else: Rd \leftarrow hex0000_0000	R	000_000 010_000
SNE Rd, Rs1, Rs2	Rs1 \neq Rs2: Rd \leftarrow hex0000_0001 else: Rd \leftarrow hex0000_0000	R	000_000 010_010
SLT Rd, Rs1, Rs2	Rs1 < Rs2: Rd \leftarrow hex0000_0001 else: Rd \leftarrow hex0000_0000	R	000_000 010_100
SLE Rd, Rs1, Rs2	Rs1 \leq Rs2: Rd \leftarrow hex0000_0001 else: Rd \leftarrow hex0000_0000	R	000_000 010_110
SGT Rd, Rs1, Rs2	Rs1 > Rs2: Rd \leftarrow hex0000_0001 else: Rd \leftarrow hex0000_0000	R	000_000 011_000
SGE Rd, Rs1, Rs2	Rs1 \geq Rs2: Rd \leftarrow hex0000_0001 else: Rd \leftarrow hex0000_0000	R	000_000 011_010



DLX Instruction Set

SEQI Rd, Rs1, Imm	Rs1 = Imm : Rd \leftarrow hex0000_0001 else: Rd \leftarrow hex0000_0000 (Imm: Sign extended 16 bit immediate)	I	100_000
SNEI Rd, Rs1, Imm	Rs1 \neq Imm : Rd \leftarrow hex0000_0001 else: Rd \leftarrow hex0000_0000	I	100_010
SLTI Rd, Rs1, Imm	Rs1 < Imm : Rd \leftarrow hex0000_0001 else: Rd \leftarrow hex0000_0000	I	100_100
SLEI Rd, Rs1, Imm	Rs1 \leq Imm : Rd \leftarrow hex0000_0001 else: Rd \leftarrow hex0000_0000	I	100_110
SGTI Rd, Rs1, Imm	Rs1 > Imm : Rd \leftarrow hex0000_0001 else: Rd \leftarrow hex0000_0000	I	101_000
SGEI Rd, Rs1, Imm	Rs1 \geq Imm : Rd \leftarrow hex0000_0001 else: Rd \leftarrow hex0000_0000	I	101_010



DLX Instruction Set

BEQZ Rs, Label	Rs = 0: $PC \leftarrow PC + 4 + \text{Label}$ Rs \neq 0: $PC \leftarrow PC + 4$ (Label: Sign extended 16 bit immediate)	I	010_000
BNEZ Rs, Label	Rs \neq 0: $PC \leftarrow PC + 4 + \text{Label}$ Rs = 0: $PC \leftarrow PC + 4$	I	010_001
J Label	$PC \leftarrow PC + 4 + \text{sign_extd}(\text{imm26})$	J	001_100
JAL Label	R31 $\leftarrow PC + 4$ $PC \leftarrow PC + 4 + \text{sign_extd}(\text{imm26})$	J	001_100
JAL Label	R31 $\leftarrow PC + 4$ $PC \leftarrow PC + 4 + \text{sign_extd}(\text{imm26})$	J	001_101
JR Rs	$PC \leftarrow Rs$	I	001_110
JALR Rs	R31 $\leftarrow PC + 4$ $PC \leftarrow Rs$	I	001_111

DLX Instruction Set

LW Rd, Rs2 (Rs1)	$Rd \leftarrow M(Rs1 + Rs2)$ (word aligned address)	R	000_000 100_000
SW Rs2(Rs1), Rd	$M(Rs1 + Rs2) \leftarrow Rd$	R	000_000 101_000
LH Rd, Rs2 (Rs1)	$Rd(16:31) \leftarrow M(Rs1 + Rs2)$ (Rd sign extended to 32 bit)	R	000_000 100_001
SH Rs2(Rs1), Rd	$M(Rs1 + Rs2) \leftarrow Rd(16:31)$	R	000_000 101_001
LB Rd, Rs2 (Rs1)	$Rd(24:31) \leftarrow M(Rs1 + Rs2)$ (Rd sign extended to 32 bit)	R	000_000 101_010
SB Rs2(Rs1), Rd	$M(Rs1 + Rs2) \leftarrow Rd(24:31)$	R	000_000 101_010

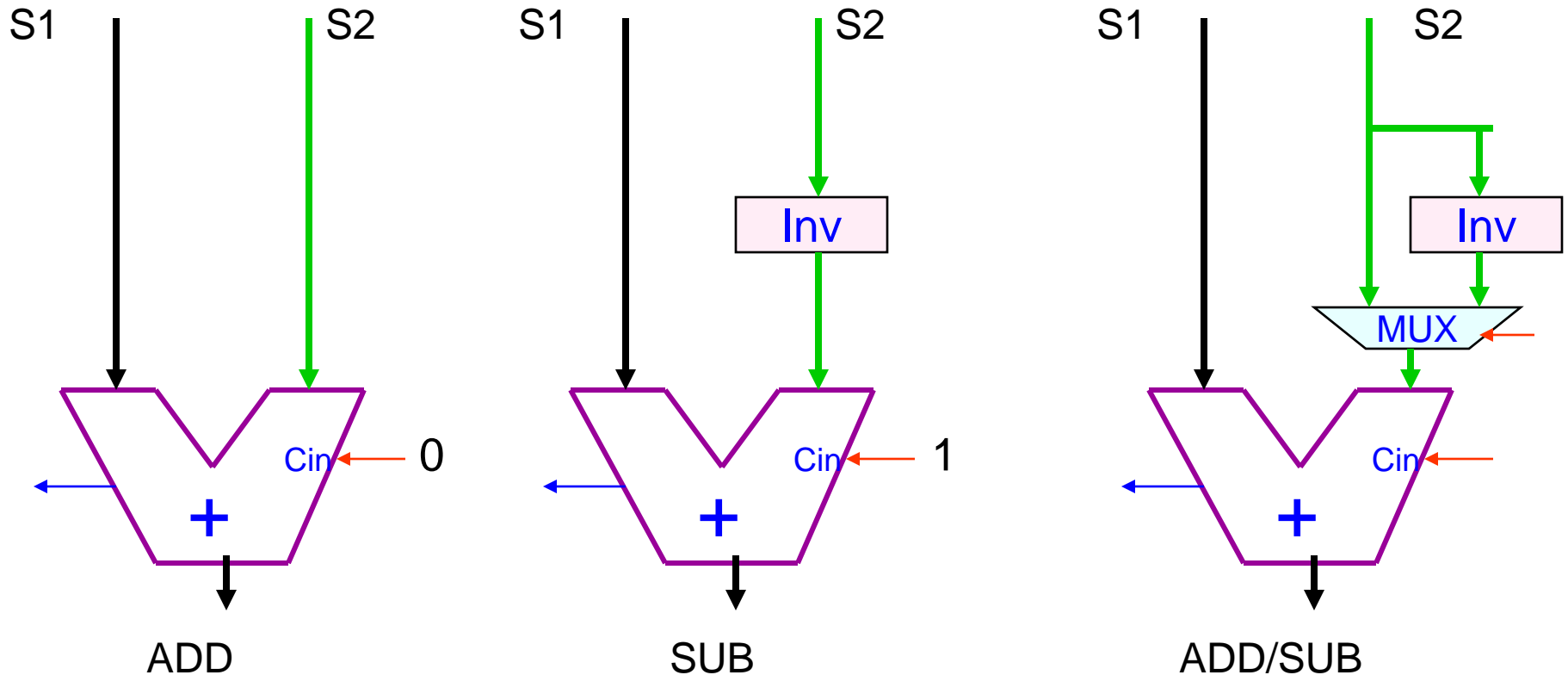


DLX Instruction Set

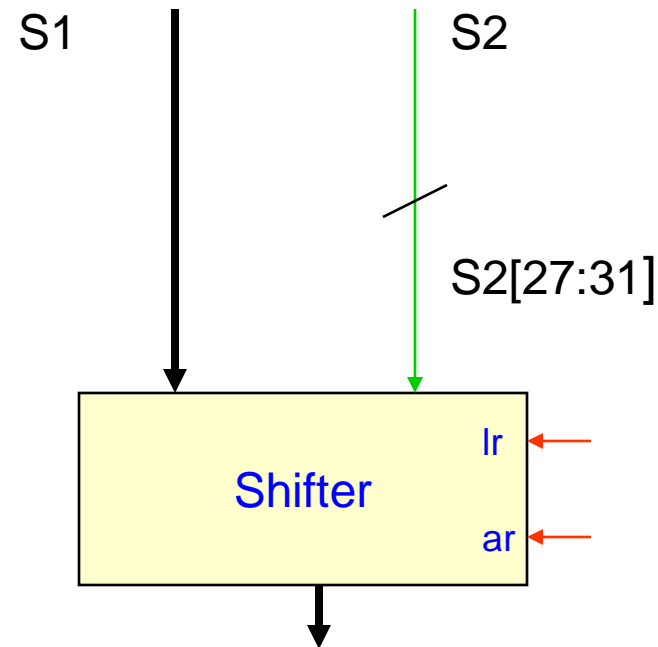
LWI Rd, Imm (Rs)	$Rd \leftarrow M(Rs + Imm)$ (Imm: sign extended 16 bit) (word aligned address)	I	000_100
SWI Imm(Rs), Rd	$M(Rs + Imm) \leftarrow Rd$	I	001_000
LHI Rd, Imm (Rs)	$Rd(16:31) \leftarrow M(Rs + Imm)$ (Rd sign extended to 32 bit)	I	000_101
SHI Imm(Rs), Rd	$M(Rs1 + Rs2) \leftarrow Rd(16:31)$	I	001_001
LBI Rd, Imm (Rs)	$Rd(24:31) \leftarrow M(Rs + Imm)$ (Rd sign extended to 32 bit)	I	000_110
SBI Imm(Rs), Rd	$M(Rs + Imm) \leftarrow Rd(24:31)$	I	001_010



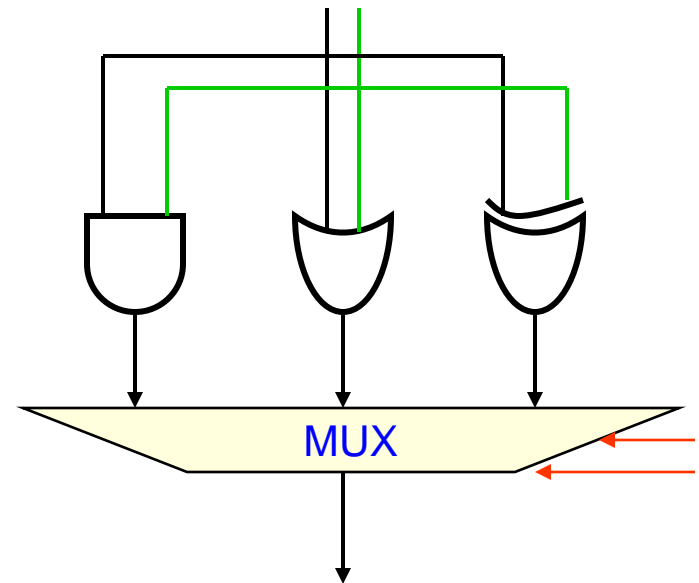
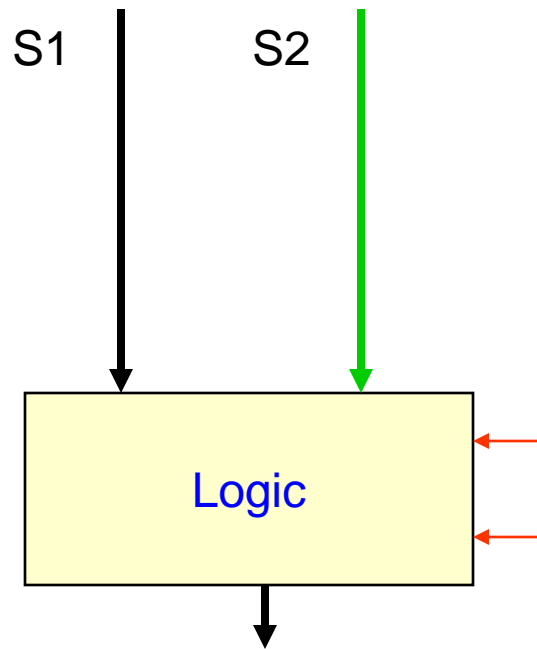
Arithmetic Logic Unit



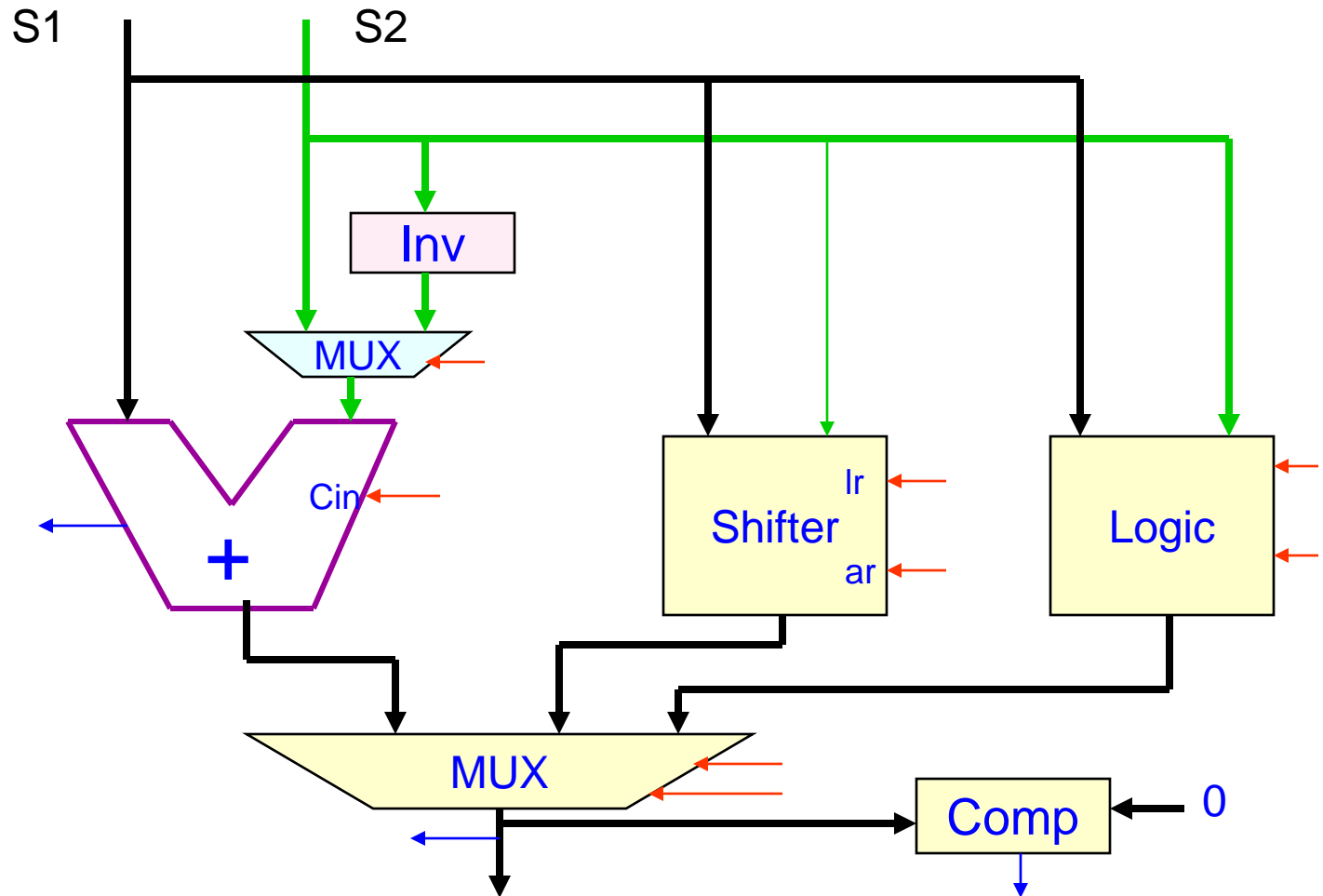
Arithmetic Logic Unit



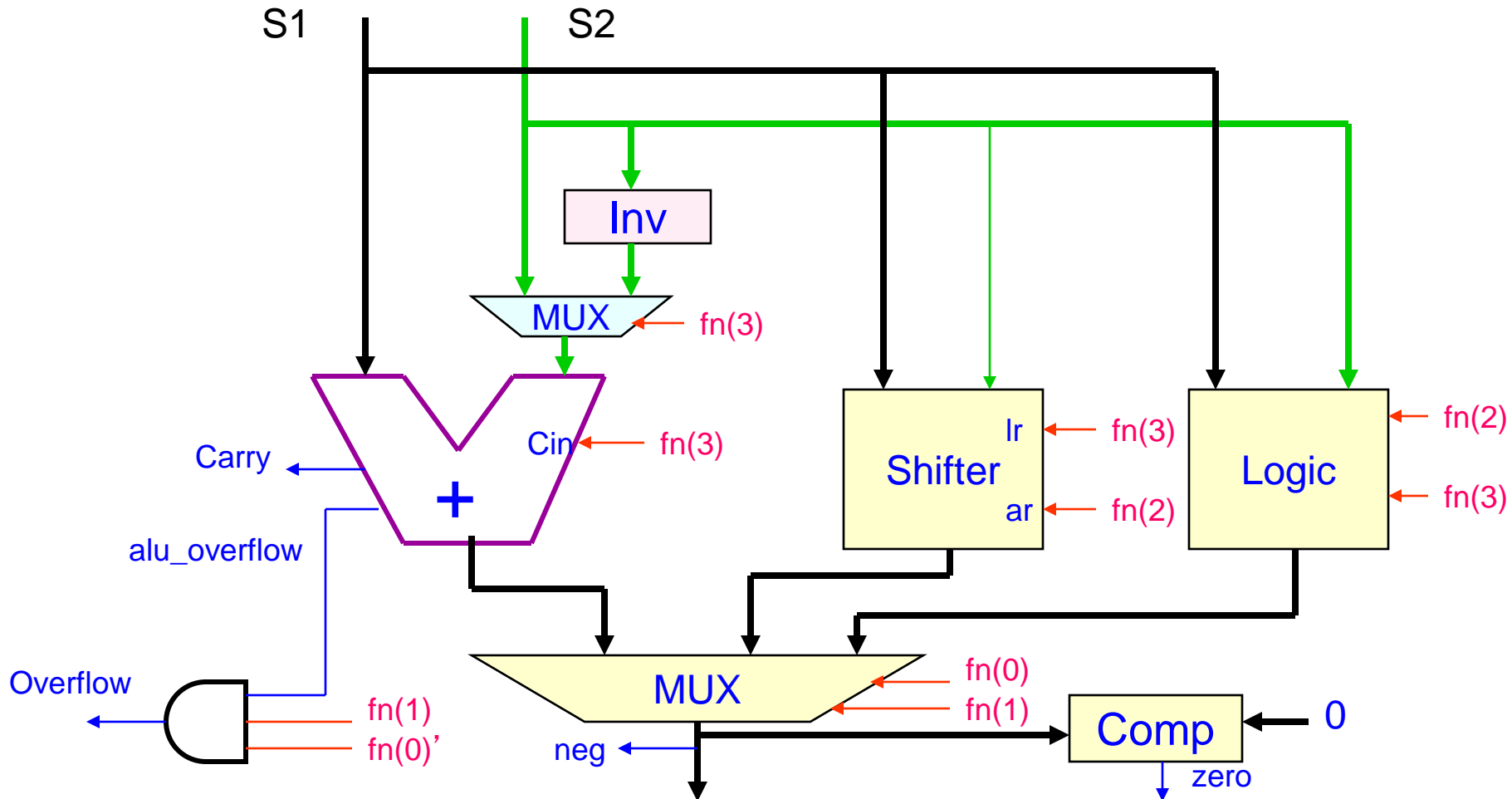
Arithmetic Logic Unit



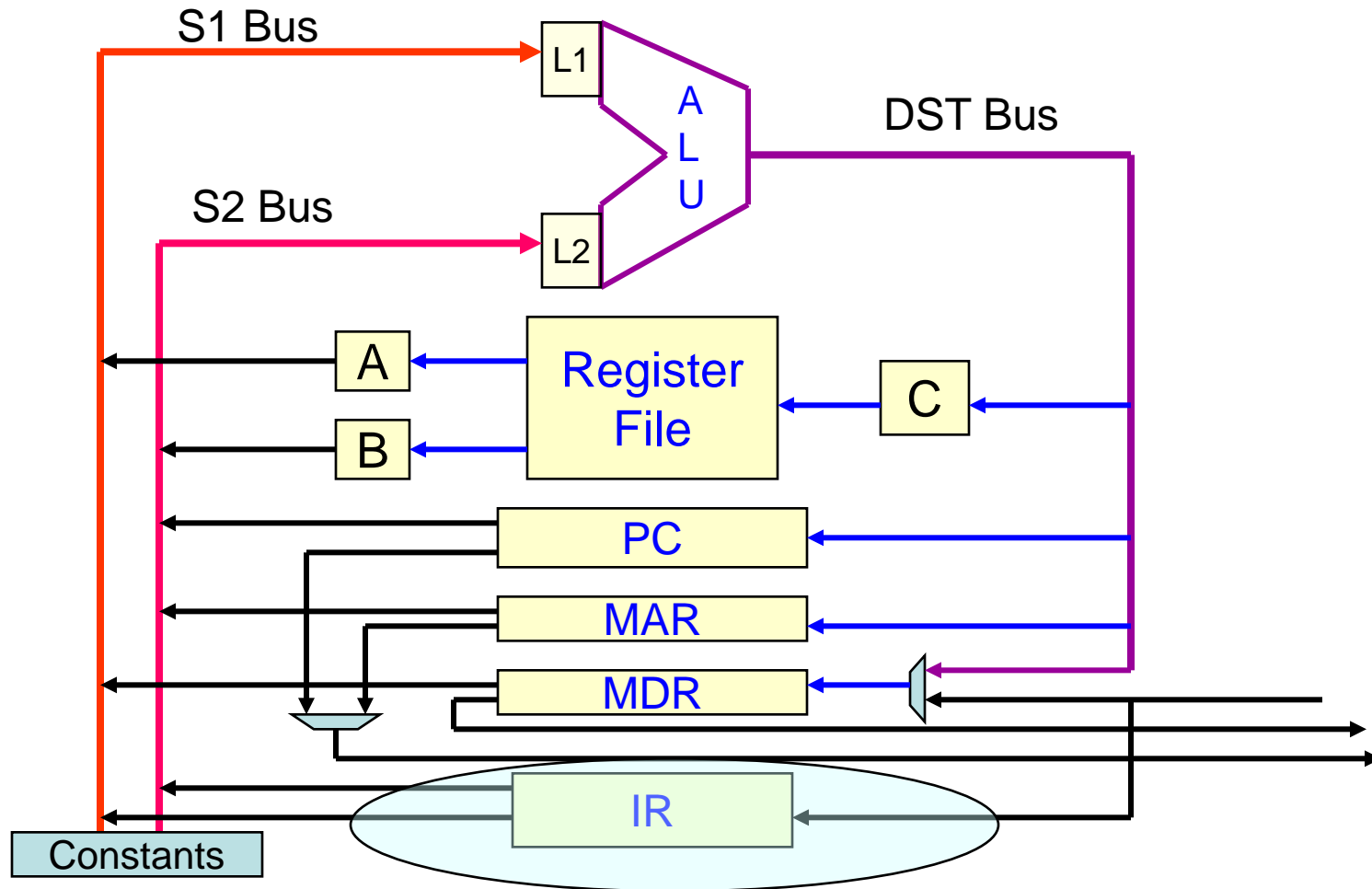
Arithmetic Logic Unit



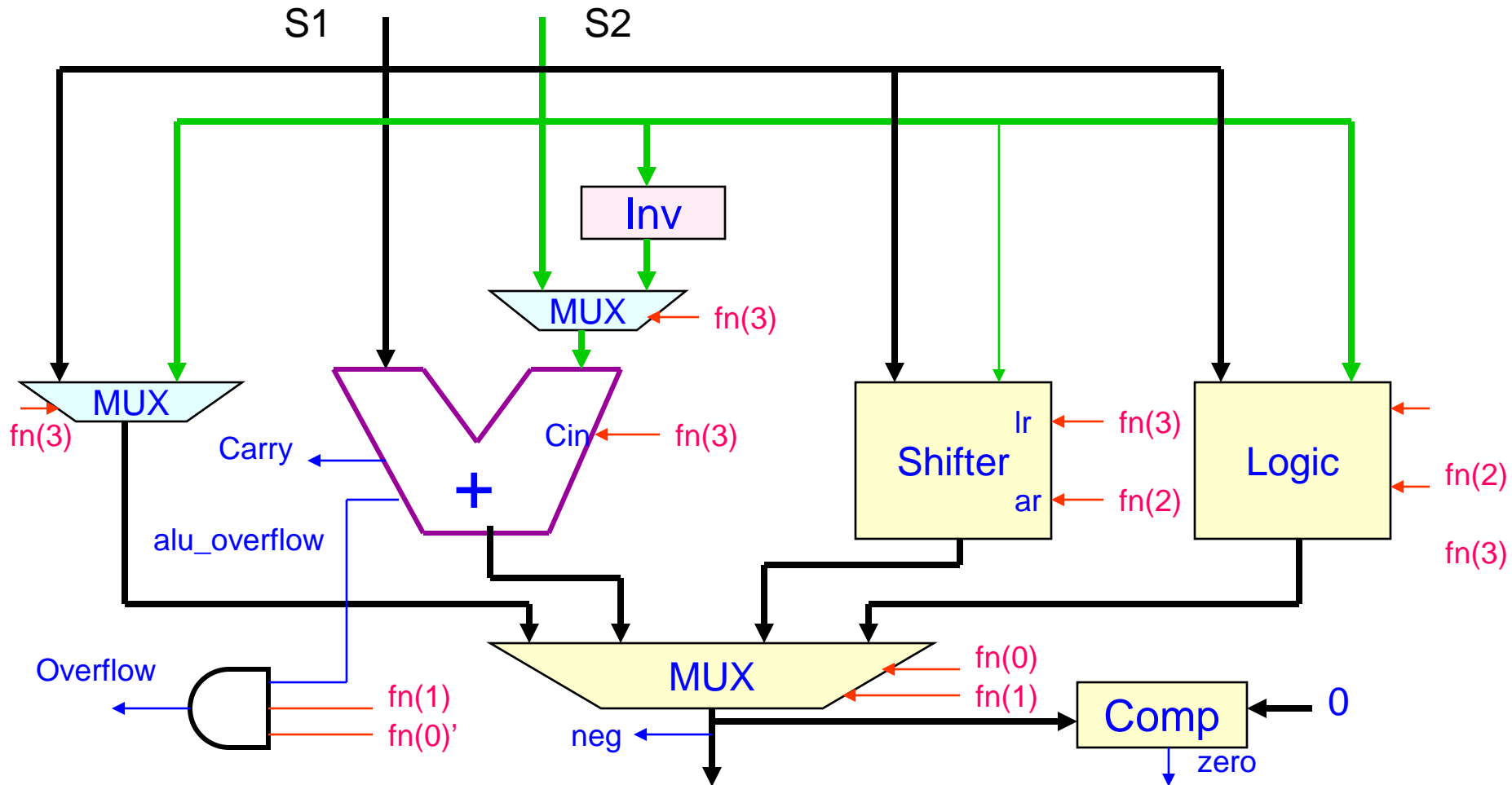
Arithmetic Logic Unit



Datapath of DLX



Arithmetic Logic Unit



Hardware Flowchart

ADD Rd, Rs1, Rs2

$rf \rightarrow a$ $rf \rightarrow b$	$pc \rightarrow eab$ $edb \rightarrow ir$
$a \rightarrow s1 \rightarrow alu$ $b \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow c$	$pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$
$c \rightarrow rf$	

Hardware Flowchart

LW Rd, Rs2(Rs1)

$rf \rightarrow a$ $rf \rightarrow b$	$pc \rightarrow eab$ $edb \rightarrow ir$
$a \rightarrow s1 \rightarrow alu$ $b \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow mar$	$pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$
$mar \rightarrow eab$ $edb \rightarrow mdr$	
$mdr \rightarrow s1 \rightarrow alu$ $alu \rightarrow c$	
$c \rightarrow rf$	



Hardware Flowchart

SW Rd, Rs2(Rs1)

rf \rightarrow a rf \rightarrow b	pc \rightarrow eab edb \rightarrow ir
a \rightarrow s1 \rightarrow alu b \rightarrow s2 \rightarrow alu alu \rightarrow dst \rightarrow mar	pc \rightarrow s2 \rightarrow alu +4 \rightarrow s1 \rightarrow alu alu \rightarrow dst \rightarrow pc
b \rightarrow s1 \rightarrow alu alu \rightarrow dst \rightarrow mdr	
mar \rightarrow eab mdr \rightarrow edb	



Hardware Flowchart

JAL Label

	$pc \rightarrow eab$ $edb \rightarrow ir$
$pc \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow c$	$pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$
$ir \rightarrow s1 \rightarrow alu$ $pc \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow mar$ $pc \rightarrow rf$	
$mar \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$	

Level 1 Hardware Flowchart

ADD Rd, Rs1, Rs2

$rf \rightarrow a$ $rf \rightarrow b$	$pc \rightarrow eab$ $edb \rightarrow ir$
$a \rightarrow s1 \rightarrow alu$ $b \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow c$	$pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$
$c \rightarrow rf$	

ADD.I Rd, Rs1, Imm

$rf \rightarrow a$	$pc \rightarrow eab$ $edb \rightarrow ir$
$a \rightarrow s1 \rightarrow alu$ $ir \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow c$	$pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$
$c \rightarrow rf$	



Level 1 Hardware Flowchart

SUB Rd, Rs1, Rs2

$rf \rightarrow a$ $rf \rightarrow b$	$pc \rightarrow eab$ $edb \rightarrow ir$
$a \rightarrow s1 \rightarrow alu$ $b \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow c$	$pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$
$c \rightarrow rf$	

SUB.I Rd, Rs1, Imm

$rf \rightarrow a$	$pc \rightarrow eab$ $edb \rightarrow ir$
$a \rightarrow s1 \rightarrow alu$ $ir \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow c$	$pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$
$c \rightarrow rf$	



Level 1 Hardware Flowchart

OR Rd, Rs1, Rs2

$rf \rightarrow a$ $rf \rightarrow b$	$pc \rightarrow eab$ $edb \rightarrow ir$
$a \rightarrow s1 \rightarrow alu$ $b \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow c$	$pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$
$c \rightarrow rf$	

OR.I Rd, Rs1, Imm

$rf \rightarrow a$	$pc \rightarrow eab$ $edb \rightarrow ir$
$a \rightarrow s1 \rightarrow alu$ $ir \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow c$	$pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$
$c \rightarrow rf$	



Level 1 Hardware Flowchart

AND Rd, Rs1, Rs2

rf \rightarrow a rf \rightarrow b	pc \rightarrow eab edb \rightarrow ir
a \rightarrow s1 \rightarrow alu b \rightarrow s2 \rightarrow alu alu \rightarrow dst \rightarrow c	pc \rightarrow s2 \rightarrow alu +4 \rightarrow s1 \rightarrow alu alu \rightarrow dst \rightarrow pc
c \rightarrow rf	

AND.I Rd, Rs1, Imm

rf \rightarrow a	pc \rightarrow eab edb \rightarrow ir
a \rightarrow s1 \rightarrow alu ir \rightarrow s2 \rightarrow alu alu \rightarrow dst \rightarrow c	pc \rightarrow s2 \rightarrow alu +4 \rightarrow s1 \rightarrow alu alu \rightarrow dst \rightarrow pc
c \rightarrow rf	



Level 1 Hardware Flowchart

XOR Rd, Rs1, Rs2

rf \rightarrow a rf \rightarrow b	pc \rightarrow eab edb \rightarrow ir
a \rightarrow s1 \rightarrow alu b \rightarrow s2 \rightarrow alu alu \rightarrow dst \rightarrow c	pc \rightarrow s2 \rightarrow alu +4 \rightarrow s1 \rightarrow alu alu \rightarrow dst \rightarrow pc
c \rightarrow rf	

XOR.I Rd, Rs1, Imm

rf \rightarrow a	pc \rightarrow eab edb \rightarrow ir
a \rightarrow s1 \rightarrow alu ir \rightarrow s2 \rightarrow alu alu \rightarrow dst \rightarrow c	pc \rightarrow s2 \rightarrow alu +4 \rightarrow s1 \rightarrow alu alu \rightarrow dst \rightarrow pc
c \rightarrow rf	



Level 1 Hardware Flowchart

SLL Rd, Rs1, Rs2

rf \rightarrow a rf \rightarrow b	pc \rightarrow eab edb \rightarrow ir
a \rightarrow s1 \rightarrow alu b \rightarrow s2 \rightarrow alu alu \rightarrow dst \rightarrow c	pc \rightarrow s2 \rightarrow alu +4 \rightarrow s1 \rightarrow alu alu \rightarrow dst \rightarrow pc
c \rightarrow rf	

SLL.I Rd, Rs1, Imm

rf \rightarrow a	pc \rightarrow eab edb \rightarrow ir
a \rightarrow s1 \rightarrow alu ir \rightarrow s2 \rightarrow alu alu \rightarrow dst \rightarrow c	pc \rightarrow s2 \rightarrow alu +4 \rightarrow s1 \rightarrow alu alu \rightarrow dst \rightarrow pc
c \rightarrow rf	



Level 1 Hardware Flowchart

SRL Rd, Rs1, Rs2

rf \rightarrow a rf \rightarrow b	pc \rightarrow eab edb \rightarrow ir
a \rightarrow s1 \rightarrow alu b \rightarrow s2 \rightarrow alu alu \rightarrow dst \rightarrow c	pc \rightarrow s2 \rightarrow alu +4 \rightarrow s1 \rightarrow alu alu \rightarrow dst \rightarrow pc
c \rightarrow rf	

SRL.I Rd, Rs1, Imm

rf \rightarrow a	pc \rightarrow eab edb \rightarrow ir
a \rightarrow s1 \rightarrow alu ir \rightarrow s2 \rightarrow alu alu \rightarrow dst \rightarrow c	pc \rightarrow s2 \rightarrow alu +4 \rightarrow s1 \rightarrow alu alu \rightarrow dst \rightarrow pc
c \rightarrow rf	



Level 1 Hardware Flowchart

SRA Rd, Rs1, Rs2

rf \rightarrow a rf \rightarrow b	pc \rightarrow eab edb \rightarrow ir
a \rightarrow s1 \rightarrow alu b \rightarrow s2 \rightarrow alu alu \rightarrow dst \rightarrow c	pc \rightarrow s2 \rightarrow alu +4 \rightarrow s1 \rightarrow alu alu \rightarrow dst \rightarrow pc
c \rightarrow rf	

SRA.I Rd, Rs1, Imm

rf \rightarrow a	pc \rightarrow eab edb \rightarrow ir
a \rightarrow s1 \rightarrow alu ir \rightarrow s2 \rightarrow alu alu \rightarrow dst \rightarrow c	pc \rightarrow s2 \rightarrow alu +4 \rightarrow s1 \rightarrow alu alu \rightarrow dst \rightarrow pc
c \rightarrow rf	

Level 1 Hardware Flowchart

LHI Rd, Imm

Rd(0:15) <- Imm; Rd(16:31) <- hex0000

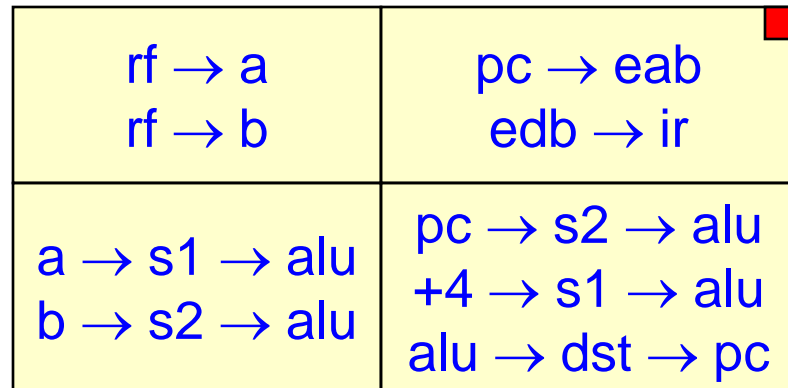
	$pc \rightarrow eab$ $edb \rightarrow ir$
$ir \rightarrow s1 \rightarrow alu$ $16 \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow c$	$pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$
$c \rightarrow rf$	

NOP

	$pc \rightarrow eab$ $edb \rightarrow ir$
	$pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$

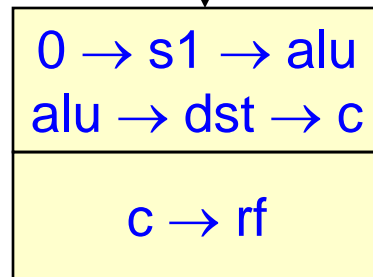
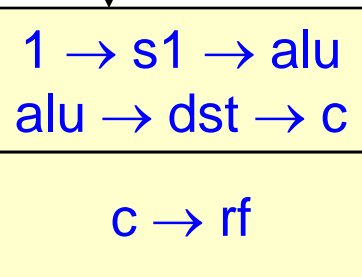
Level 1 Hardware Flowchart

SEQ Rd, Rs1, Rs2

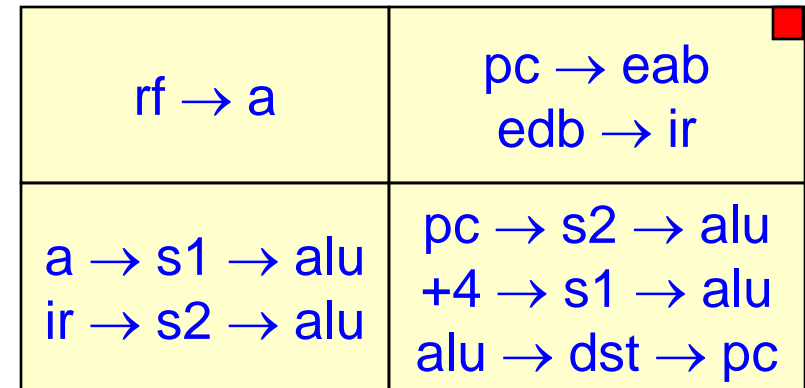


Z=1

Z=0

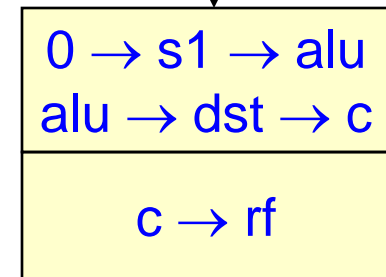
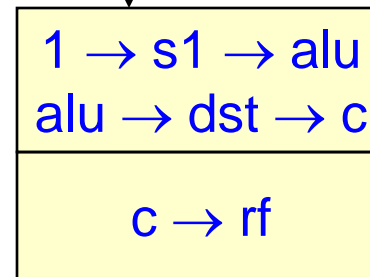


SEQ.I Rd, Rs1, Imm



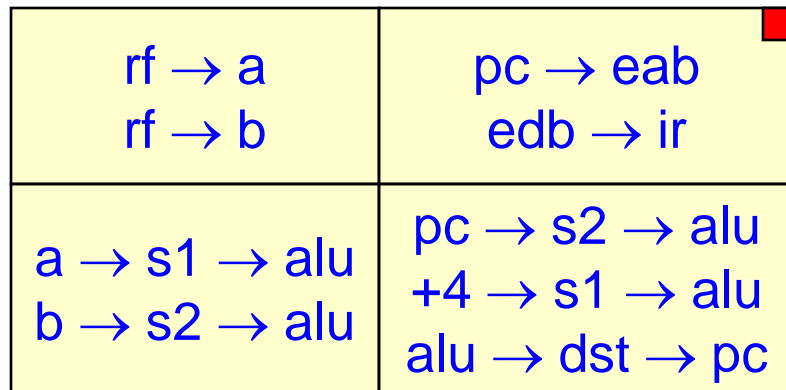
Z=1

Z=0



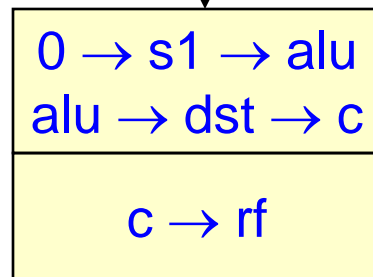
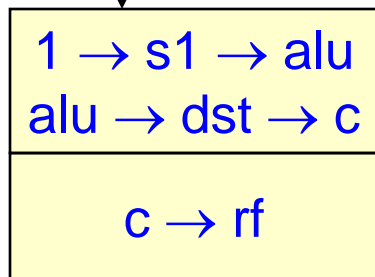
Level 1 Hardware Flowchart

SNE Rd, Rs1, Rs2

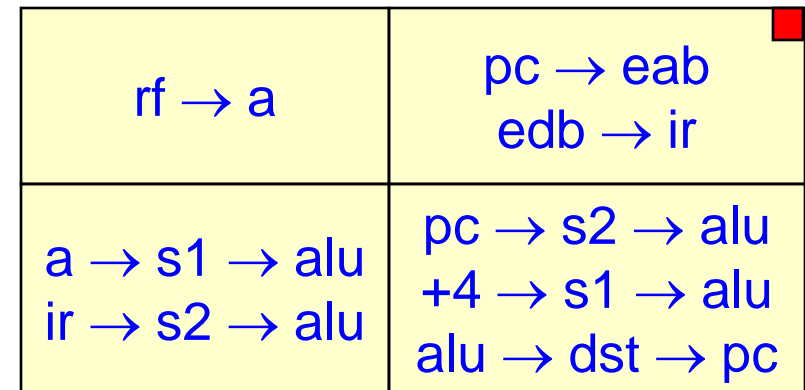


Z=0

Z=1

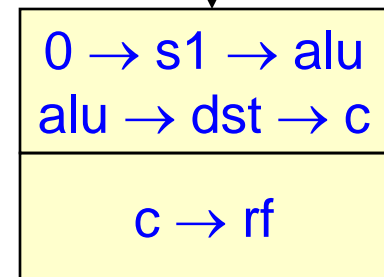
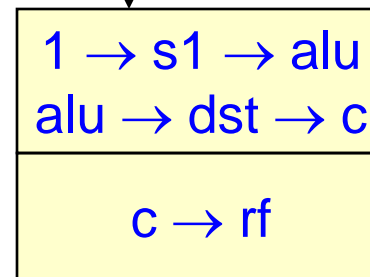


SNE.I Rd, Rs1, Imm



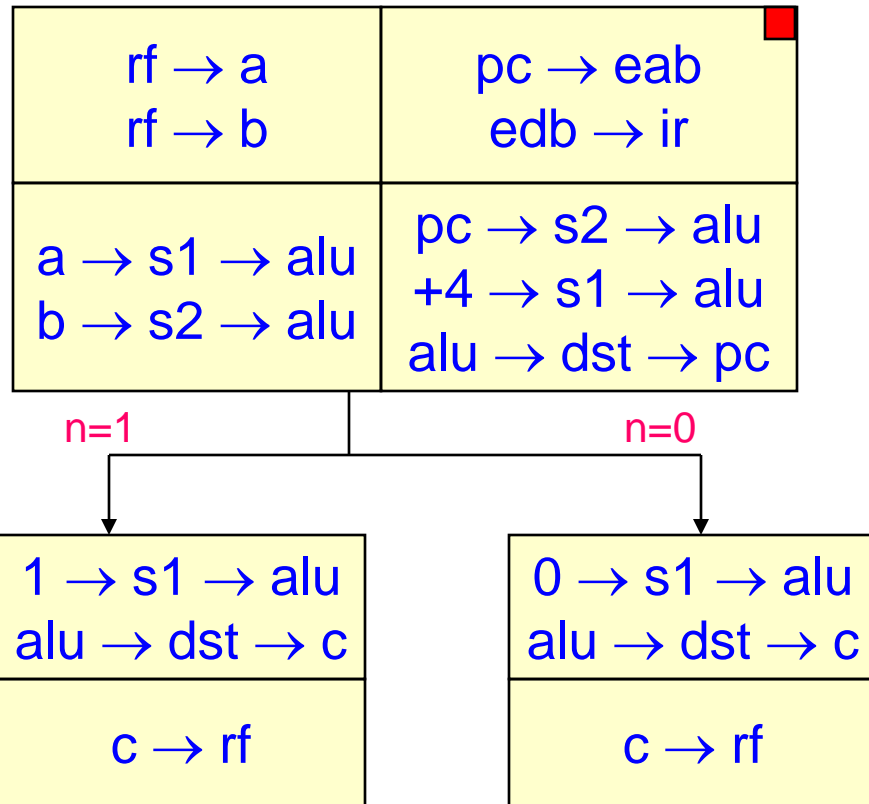
Z=0

Z=1

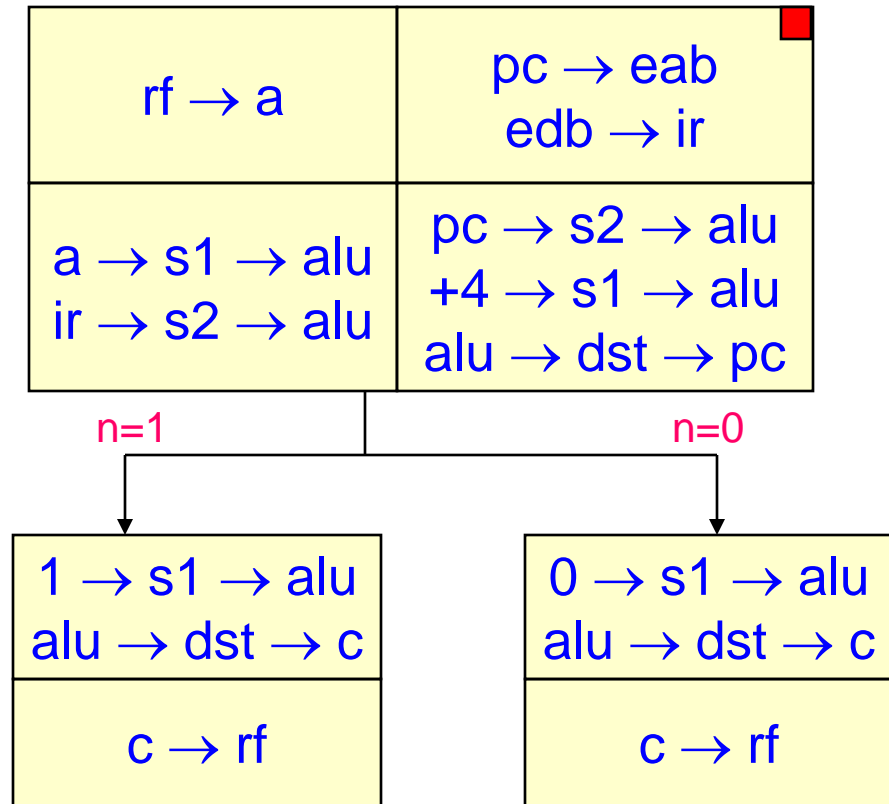


Level 1 Hardware Flowchart

SLT Rd, Rs1, Rs2

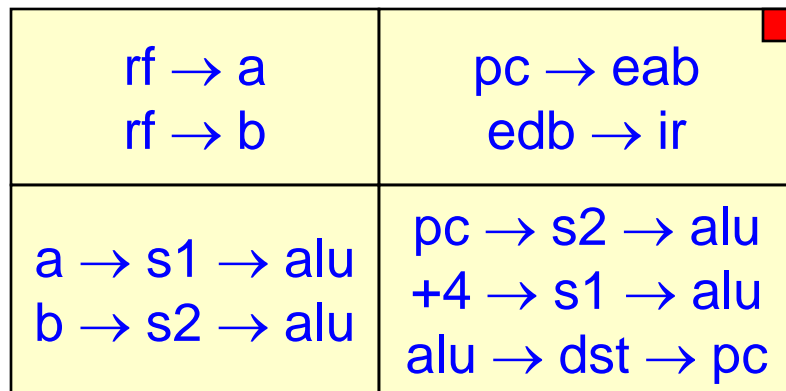


SLT.I Rd, Rs1, Imm



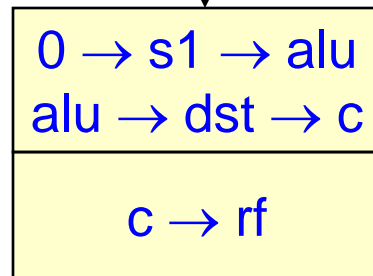
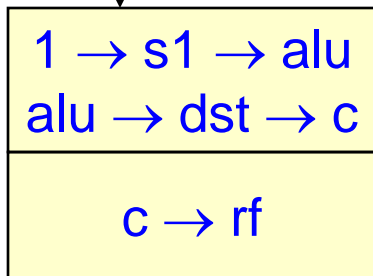
Level 1 Hardware Flowchart

SLE Rd, Rs1, Rs2

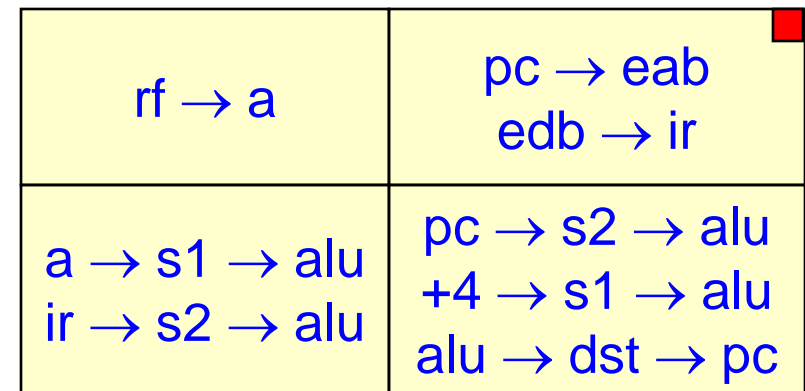


n=1 or z = 1

n=0 and z = 0

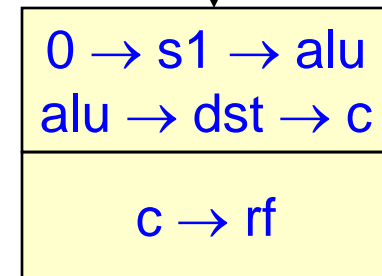
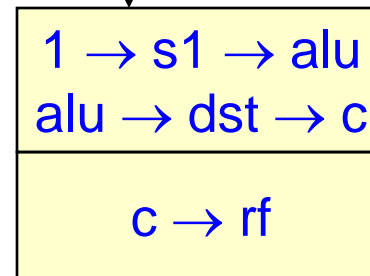


SLE.I Rd, Rs1, Imm



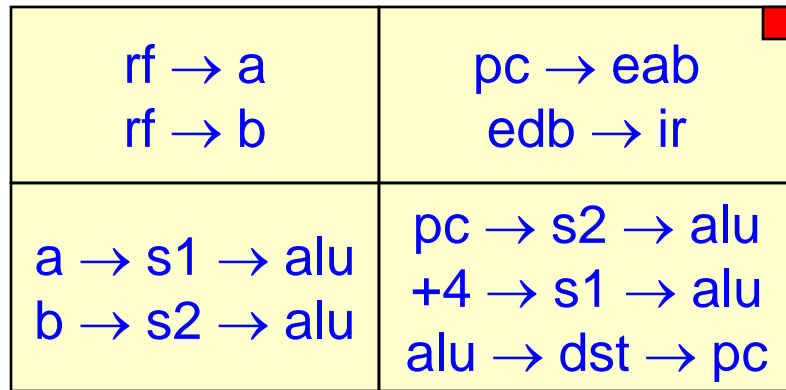
n=1 or z = 1

n=0 and z = 0



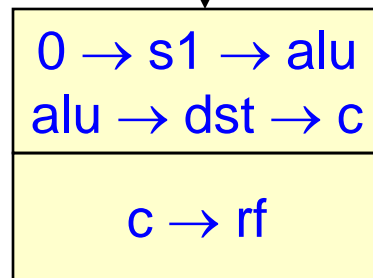
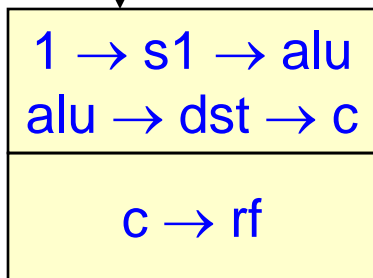
Level 1 Hardware Flowchart

SGT Rd, Rs1, Rs2

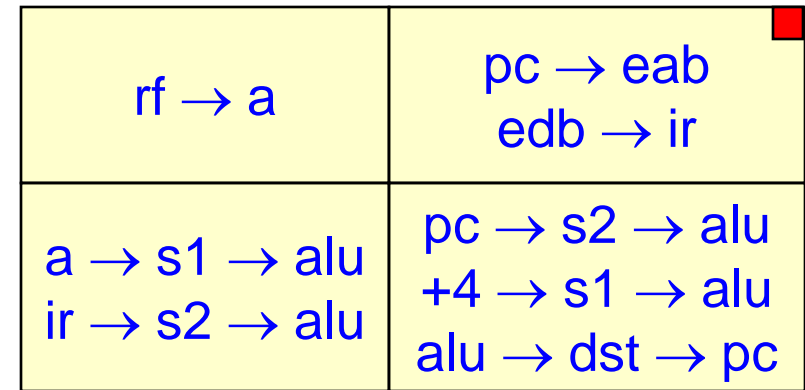


n=0 and z = 0

n=1 or z = 1

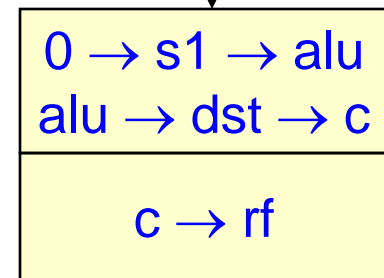
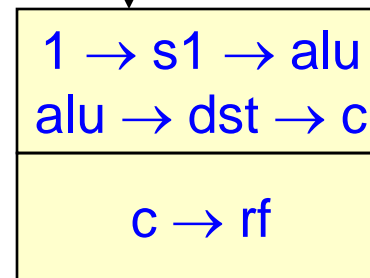


SGT.I Rd, Rs1, Imm



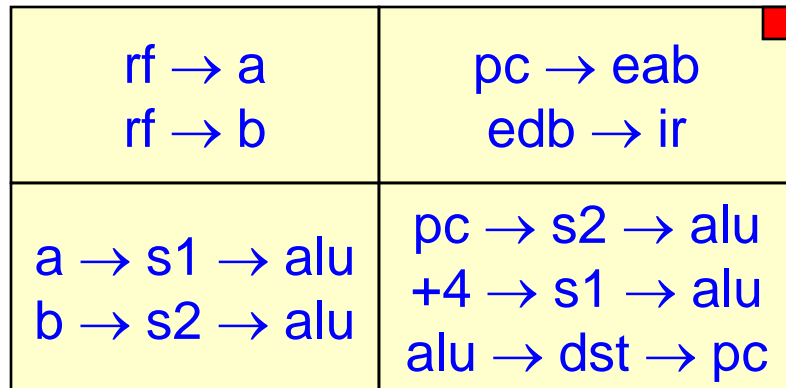
n=0 and z = 0

n=1 or z = 1



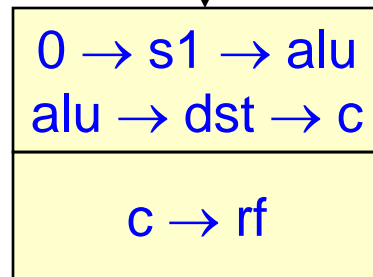
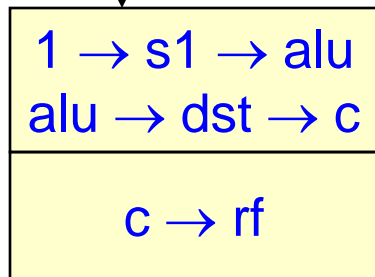
Level 1 Hardware Flowchart

SGE Rd, Rs1, Rs2

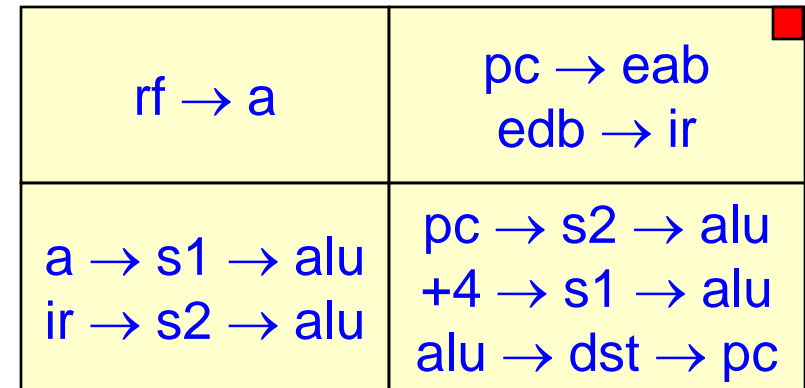


n=0

n=1

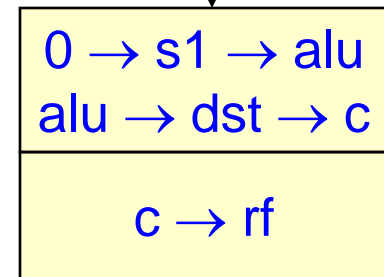
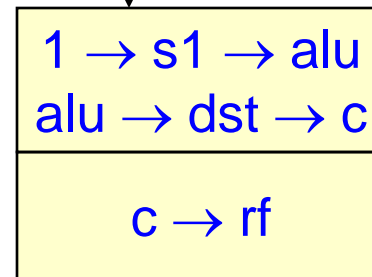


SGE.I Rd, Rs1, Imm



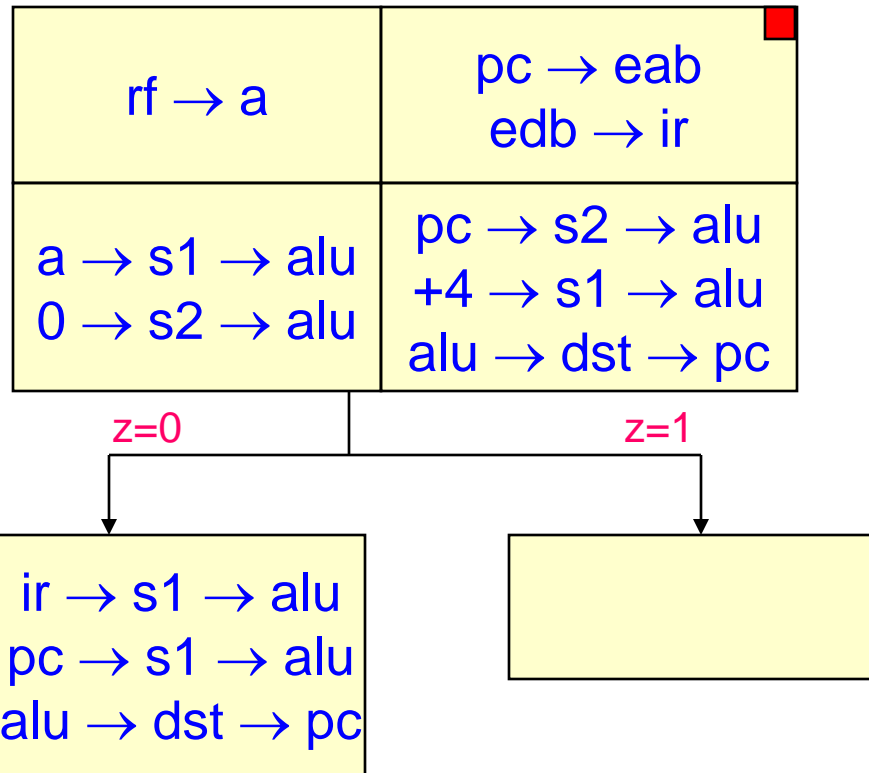
n=0

n=1

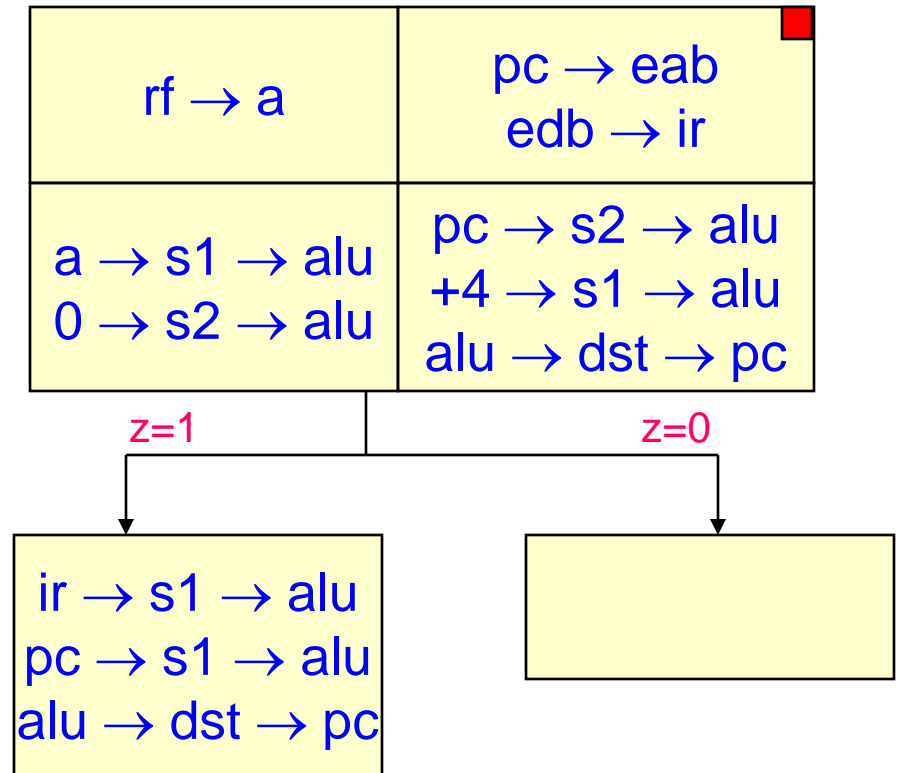


Level 1 Hardware Flowchart

BEQZ Rs, Label



BNEZ Rs, Label



Level 1 Hardware Flowchart

J Label

	pc \rightarrow eab edb \rightarrow ir
ir \rightarrow s1 \rightarrow alu pc \rightarrow s2 \rightarrow alu alu \rightarrow dst \rightarrow mar	pc \rightarrow s2 \rightarrow alu +4 \rightarrow s1 \rightarrow alu alu \rightarrow dst \rightarrow pc
mar \rightarrow s2 \rightarrow alu alu \rightarrow dst \rightarrow pc	

JR Rs

rf \rightarrow a	pc \rightarrow eab edb \rightarrow ir
a \rightarrow s1 \rightarrow alu alu \rightarrow dst \rightarrow mar	pc \rightarrow s2 \rightarrow alu +4 \rightarrow s1 \rightarrow alu alu \rightarrow dst \rightarrow pc
mar \rightarrow s2 \rightarrow alu alu \rightarrow dst \rightarrow pc	



Level 1 Hardware Flowchart

JAL Label

	pc \rightarrow eab edb \rightarrow ir
pc \rightarrow s2 \rightarrow alu alu \rightarrow dst \rightarrow c	pc \rightarrow s2 \rightarrow alu +4 \rightarrow s1 \rightarrow alu alu \rightarrow dst \rightarrow pc
ir \rightarrow s1 \rightarrow alu pc \rightarrow s2 \rightarrow alu alu \rightarrow dst \rightarrow mar c \rightarrow rf	
mar \rightarrow s2 \rightarrow alu alu \rightarrow dst \rightarrow pc	

JALR Rs

rf \rightarrow a	pc \rightarrow eab edb \rightarrow ir
pc \rightarrow s2 \rightarrow alu alu \rightarrow dst \rightarrow c	pc \rightarrow s2 \rightarrow alu +4 \rightarrow s1 \rightarrow alu alu \rightarrow dst \rightarrow pc
a \rightarrow s1 \rightarrow alu alu \rightarrow dst \rightarrow mar c \rightarrow rf	
mar \rightarrow s2 \rightarrow alu alu \rightarrow dst \rightarrow pc	



Level 1 Hardware Flowchart

LW Rd, Rs2(Rs1)

rf → a rf → b	pc → eab edb → ir
a → s1 → alu b → s2 → alu alu → dst → mar	pc → s2 → alu +4 → s1 → alu alu → dst → pc
mar → eab edb → mdr	
mdr → s1 → alu alu → dst → c	
c → rf	

LW.I Rd, Imm(Rs1)

rf → a	pc → eab edb → ir
a → s1 → alu ir → s2 → alu alu → dst → mar	pc → s2 → alu +4 → s1 → alu alu → dst → pc
mar → eab edb → mdr	
mdr → s1 → alu alu → dst → c	
c → rf	



Level 1 Hardware Flowchart

SW Rd, Rs2(Rs1)

rf \rightarrow a rf \rightarrow b	pc \rightarrow eab edb \rightarrow ir
a \rightarrow s1 \rightarrow alu b \rightarrow s2 \rightarrow alu alu \rightarrow dst \rightarrow mar rf \rightarrow b	pc \rightarrow s2 \rightarrow alu +4 \rightarrow s1 \rightarrow alu alu \rightarrow dst \rightarrow pc
b \rightarrow s2 \rightarrow alu alu \rightarrow dst \rightarrow mdr	
mar \rightarrow eab mdr \rightarrow edb	

SW.I Rd, Imm(Rs1)

rf \rightarrow a	pc \rightarrow eab edb \rightarrow ir
a \rightarrow s1 \rightarrow alu ir \rightarrow s2 \rightarrow alu alu \rightarrow dst \rightarrow mar rf \rightarrow b	pc \rightarrow s2 \rightarrow alu +4 \rightarrow s1 \rightarrow alu alu \rightarrow dst \rightarrow pc
b \rightarrow s2 \rightarrow alu alu \rightarrow dst \rightarrow mdr	
mar \rightarrow eab mdr \rightarrow edb	



Level 1 Hardware Flowchart

LH Rd, Rs2(Rs1)

rf \rightarrow a rf \rightarrow b	pc \rightarrow eab edb \rightarrow ir
a \rightarrow s1 \rightarrow alu b \rightarrow s2 \rightarrow alu alu \rightarrow dst \rightarrow mar	pc \rightarrow s2 \rightarrow alu +4 \rightarrow s1 \rightarrow alu alu \rightarrow dst \rightarrow pc
mar \rightarrow eab edb \rightarrow mdr	
mdr \rightarrow s1 \rightarrow alu 16 \rightarrow s2 \rightarrow alu alu \rightarrow dst \rightarrow c	
c \rightarrow rf	

LH.I Rd, Imm(Rs1)

rf \rightarrow a	pc \rightarrow eab edb \rightarrow ir
a \rightarrow s1 \rightarrow alu ir \rightarrow s2 \rightarrow alu alu \rightarrow dst \rightarrow mar	pc \rightarrow s2 \rightarrow alu +4 \rightarrow s1 \rightarrow alu alu \rightarrow dst \rightarrow pc
mar \rightarrow eab edb \rightarrow mdr	
mdr \rightarrow s1 \rightarrow alu 16 \rightarrow s1 \rightarrow alu alu \rightarrow dst \rightarrow c	
c \rightarrow rf	



Level 1 Hardware Flowchart

SH Rd, Rs2(Rs1)

rf → a rf → b	pc → eab edb → ir
a → s1 → alu b → s2 → alu alu → dst → mar rf → b	pc → s2 → alu +4 → s1 → alu alu → dst → pc
b → s2 → alu alu → dst → mdr	
mar → eab mdr → edb	

SH.I Rd, Imm(Rs1)

rf → a	pc → eab edb → ir
a → s1 → alu ir → s2 → alu alu → dst → mar rf → b	pc → s2 → alu +4 → s1 → alu alu → dst → pc
b → s2 → alu alu → dst → mdr	
mar → eab mdr → edb	



Level 1 Hardware Flowchart

LB Rd, Rs2(Rs1)

rf → a rf → b	pc → eab edb → ir
a → s1 → alu b → s2 → alu alu → dst → mar	pc → s2 → alu +4 → s1 → alu alu → dst → pc
mar → eab edb → mdr	
mdr → s1 → alu 24 → s2 → alu alu → dst → c	
c → rf	

LB.I Rd, Imm(Rs1)

rf → a	pc → eab edb → ir
a → s1 → alu ir → s2 → alu alu → dst → mar	pc → s2 → alu +4 → s1 → alu alu → dst → pc
mar → eab edb → mdr	
mdr → s1 → alu 24 → s1 → alu alu → dst → c	
c → rf	



Level 1 Hardware Flowchart

SB Rd, Rs2(Rs1)

rf \rightarrow a rf \rightarrow b	pc \rightarrow eab edb \rightarrow ir
a \rightarrow s1 \rightarrow alu b \rightarrow s2 \rightarrow alu alu \rightarrow dst \rightarrow mar rf \rightarrow b	pc \rightarrow s2 \rightarrow alu +4 \rightarrow s1 \rightarrow alu alu \rightarrow dst \rightarrow pc
b \rightarrow s2 \rightarrow alu alu \rightarrow dst \rightarrow mdr	
mar \rightarrow eab mdr \rightarrow edb	

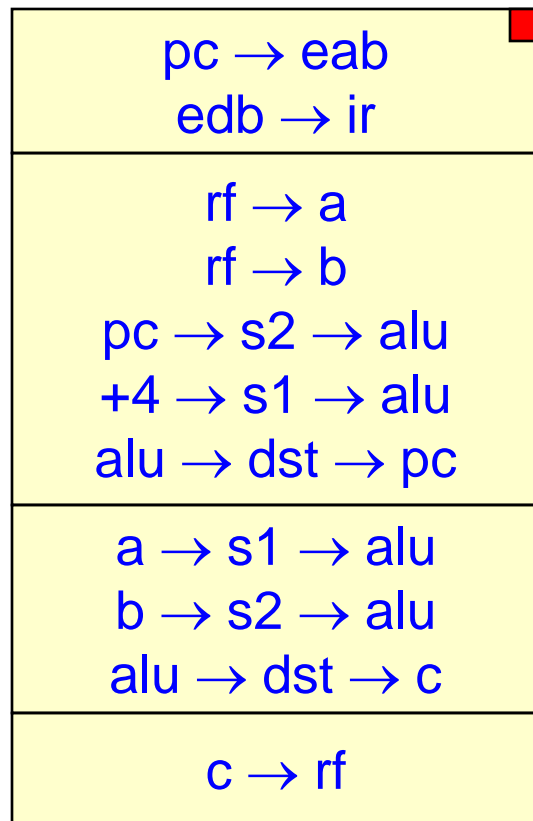
SB.I Rd, Imm(Rs1)

rf \rightarrow a	pc \rightarrow eab edb \rightarrow ir
a \rightarrow s1 \rightarrow alu ir \rightarrow s2 \rightarrow alu alu \rightarrow dst \rightarrow mar rf \rightarrow b	pc \rightarrow s2 \rightarrow alu +4 \rightarrow s1 \rightarrow alu alu \rightarrow dst \rightarrow pc
b \rightarrow s2 \rightarrow alu alu \rightarrow dst \rightarrow mdr	
mar \rightarrow eab mdr \rightarrow edb	

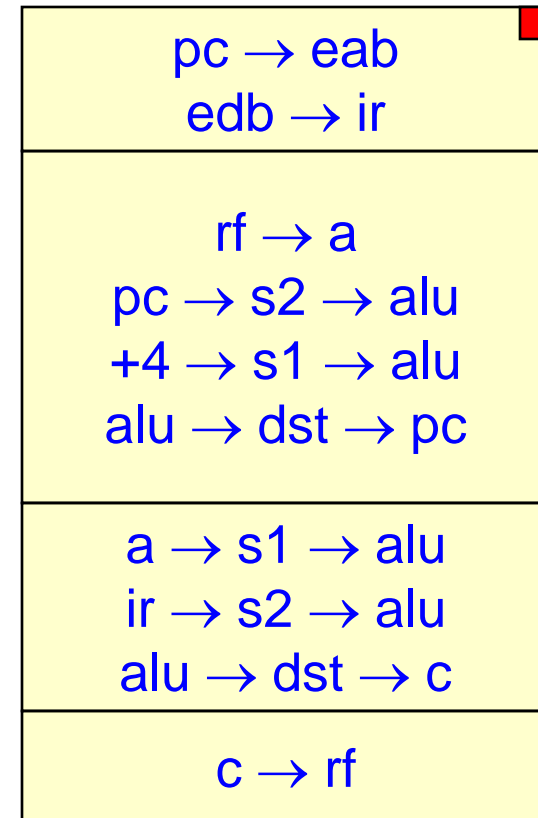


Merged Level 1 Hardware Flowchart

ADD Rd, Rs1, Rs2

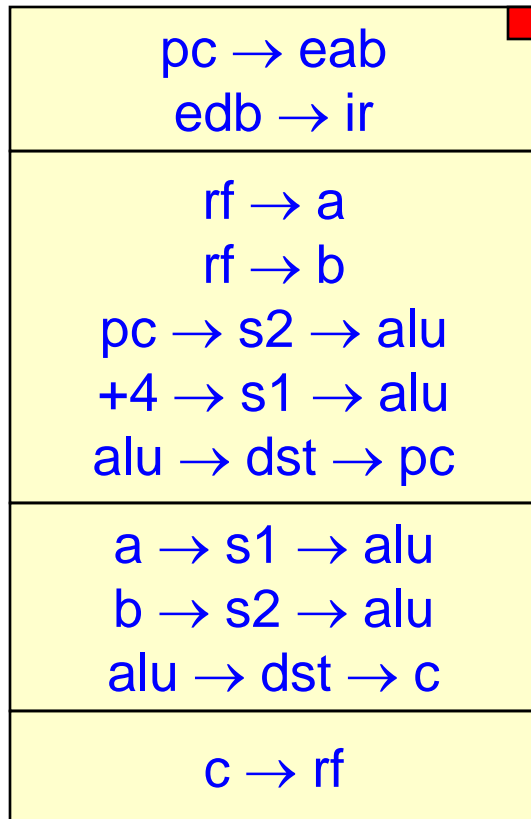


ADD.I Rd, Rs1, Imm

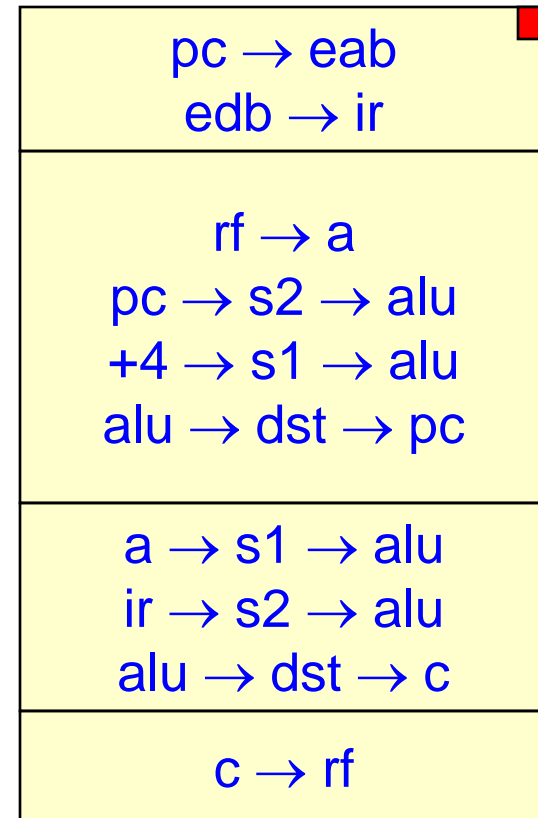


Merged Level 1 Hardware Flowchart

SUB Rd, Rs1, Rs2

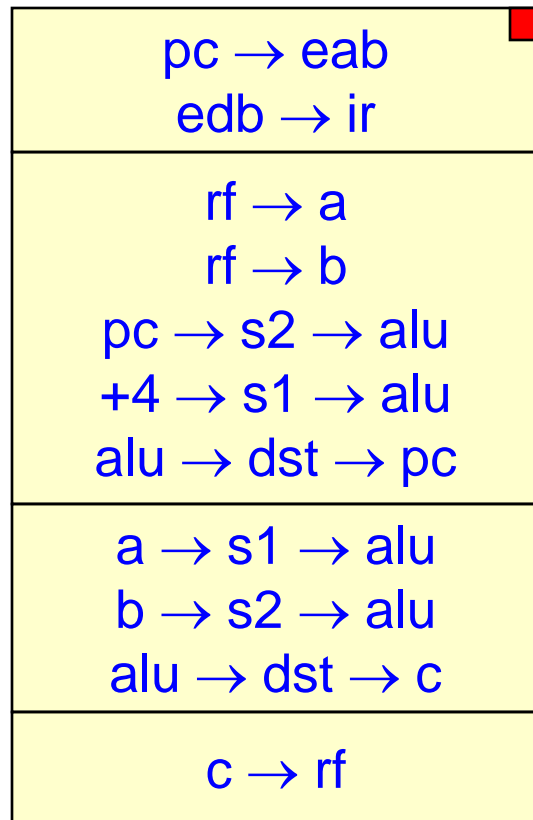


SUB.I Rd, Rs1, Imm

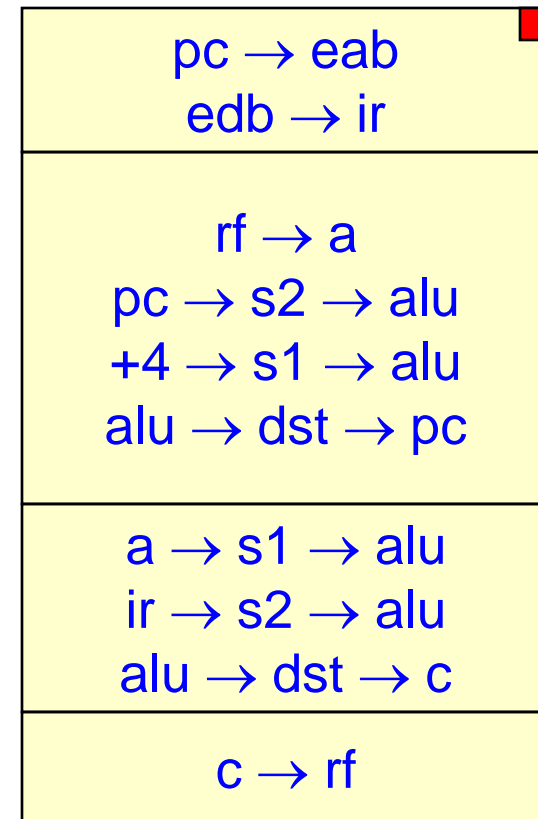


Merged Level 1 Hardware Flowchart

AND Rd, Rs1, Rs2

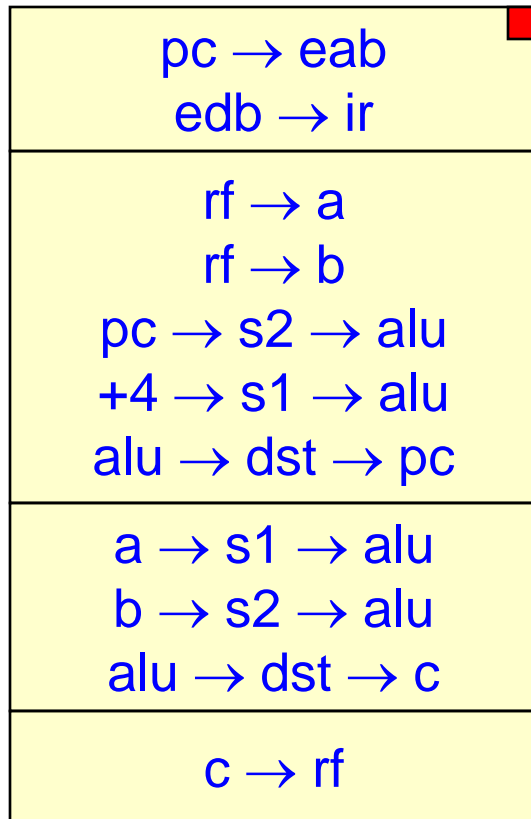


AND.I Rd, Rs1, Imm

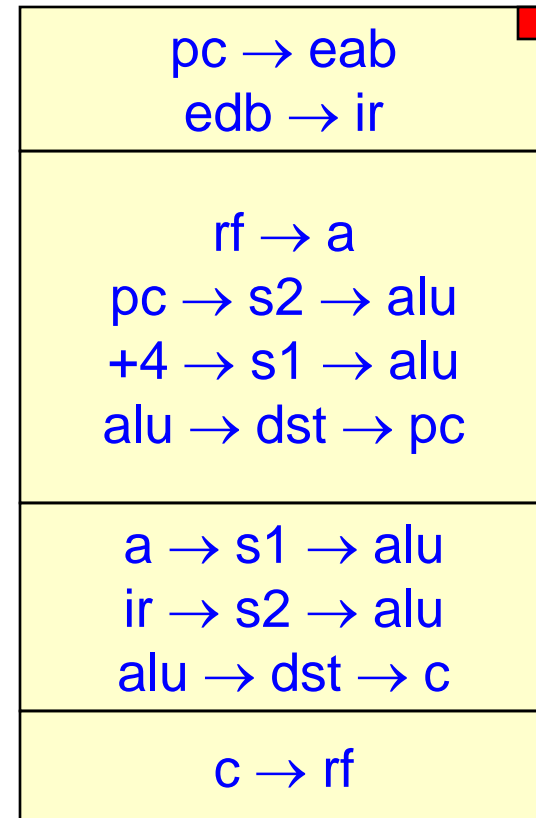


Merged Level 1 Hardware Flowchart

OR Rd, Rs1, Rs2

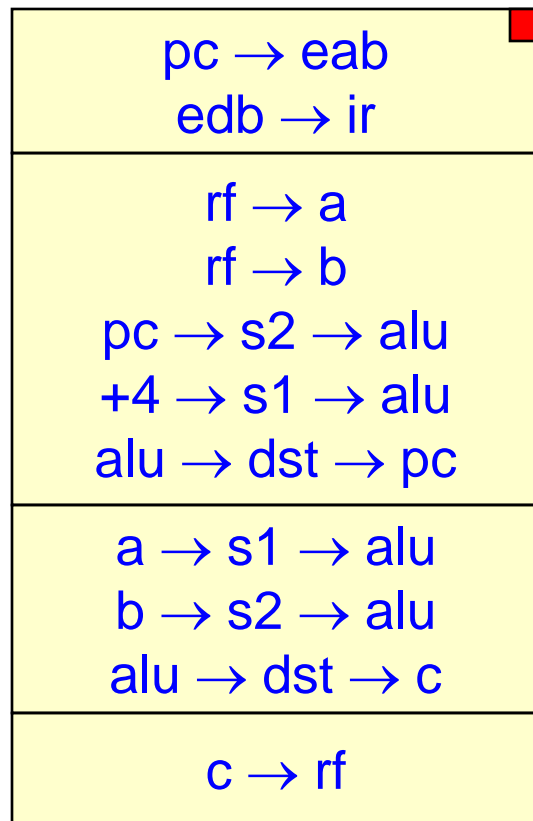


OR.I Rd, Rs1, Imm

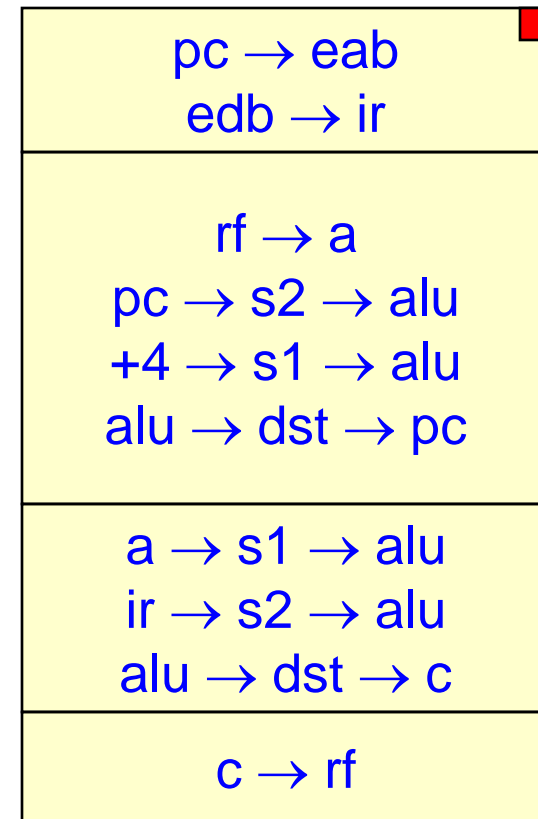


Merged Level 1 Hardware Flowchart

XOR Rd, Rs1, Rs2

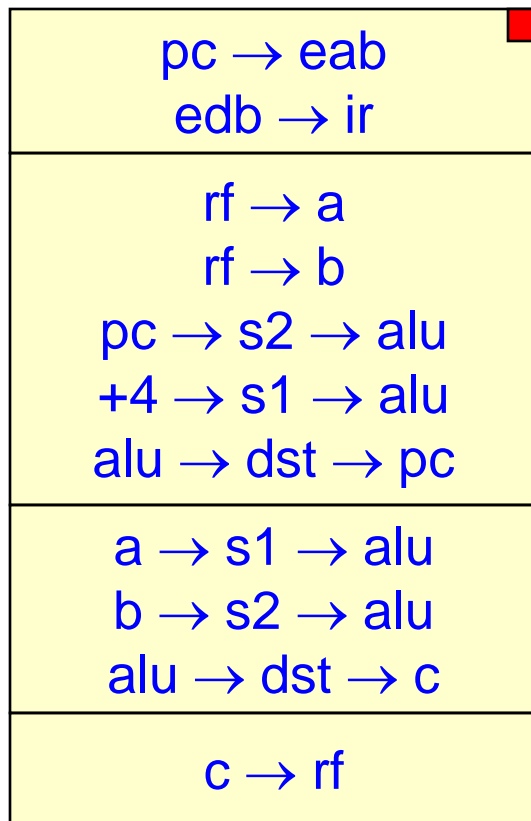


XOR.I Rd, Rs1, Imm

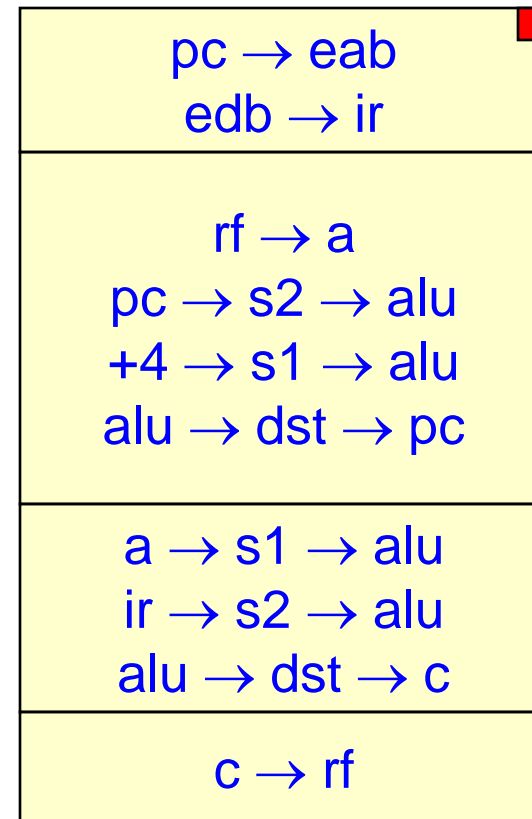


Merged Level 1 Hardware Flowchart

SLL Rd, Rs1, Rs2

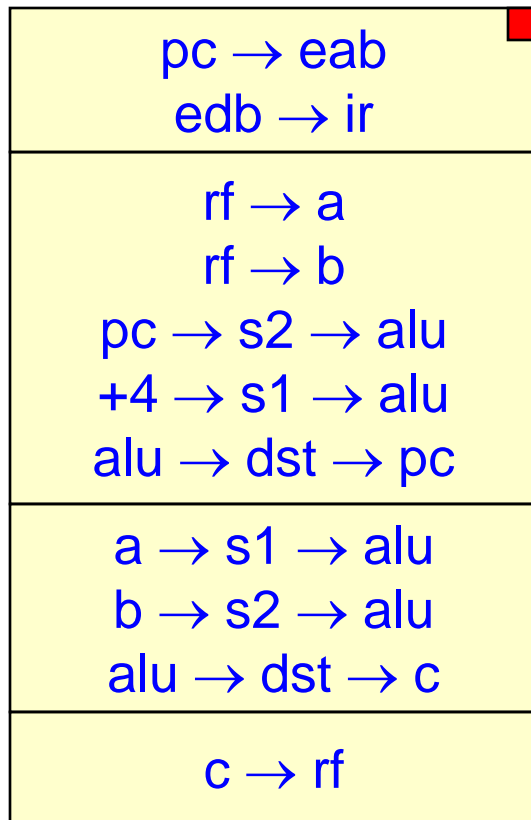


SLL.I Rd, Rs1, Imm

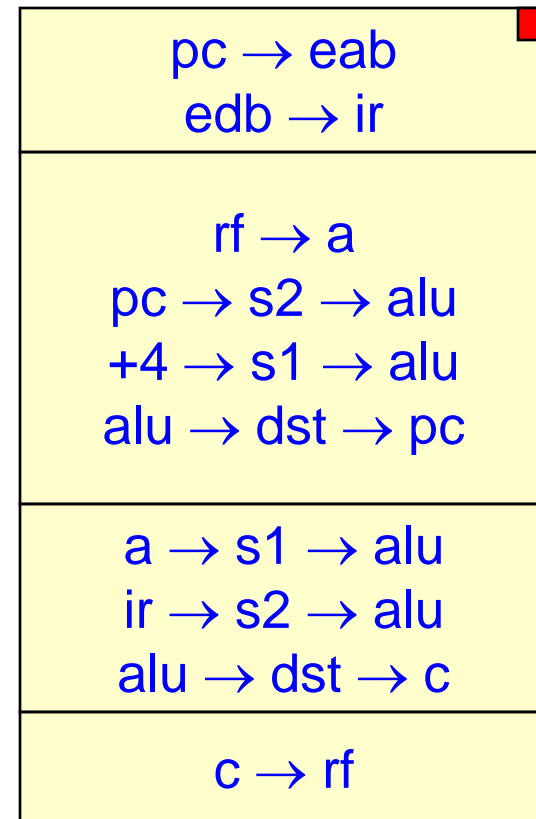


Merged Level 1 Hardware Flowchart

SRL Rd, Rs1, Rs2

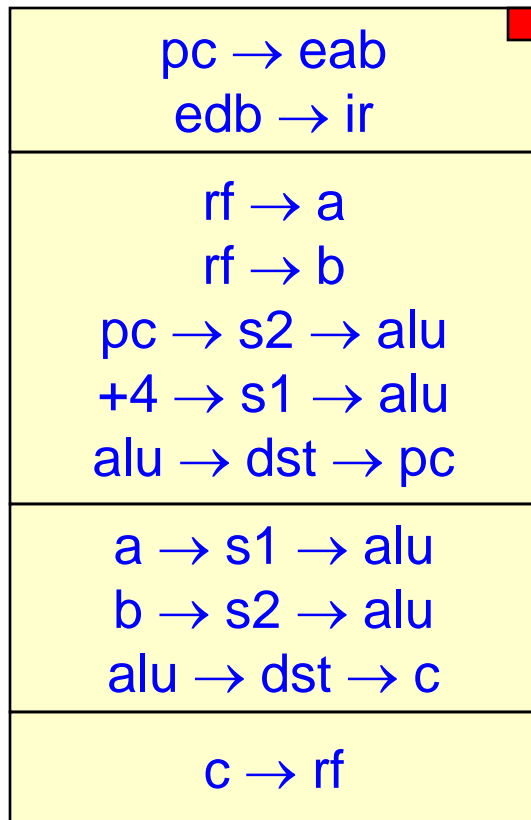


SRL.I Rd, Rs1, Imm

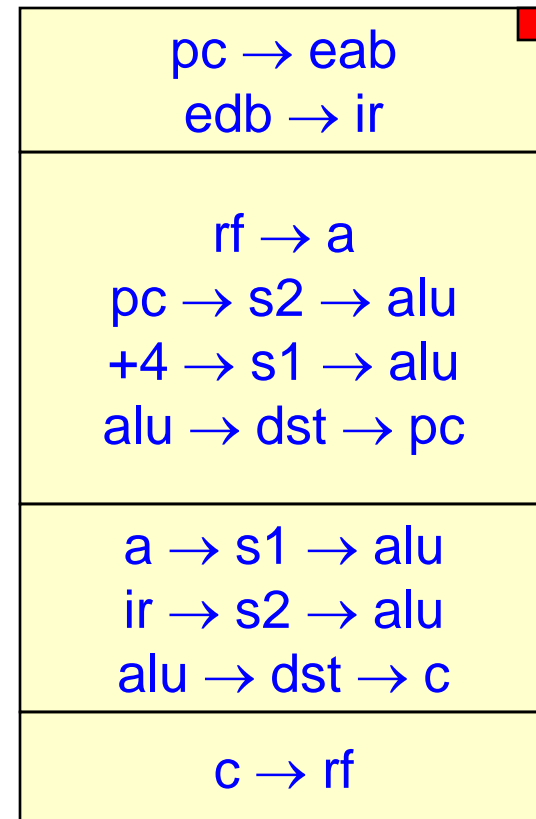


Merged Level 1 Hardware Flowchart

SRA Rd, Rs1, Rs2



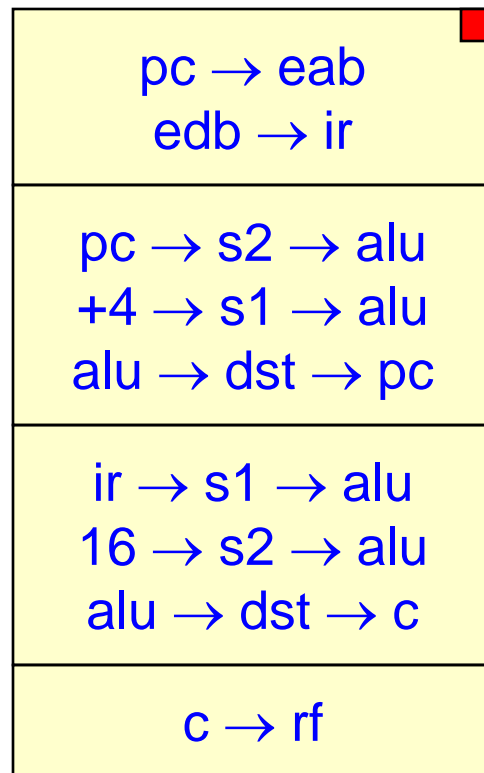
SRA.I Rd, Rs1, Imm



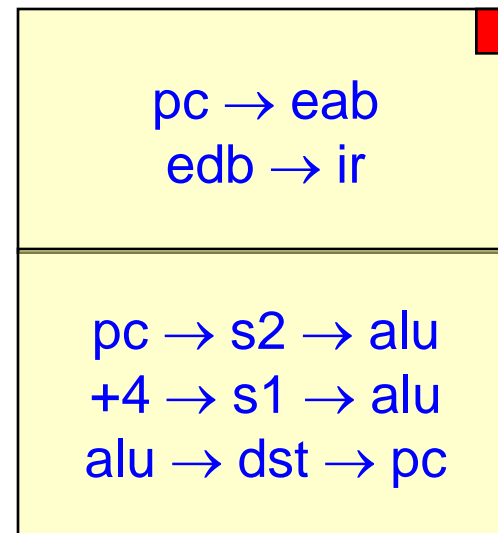
Merged Level 1 Hardware Flowchart

LHI Rd, Imm

Rd(0:15) <- Imm; Rd(16:31) <- hex0000

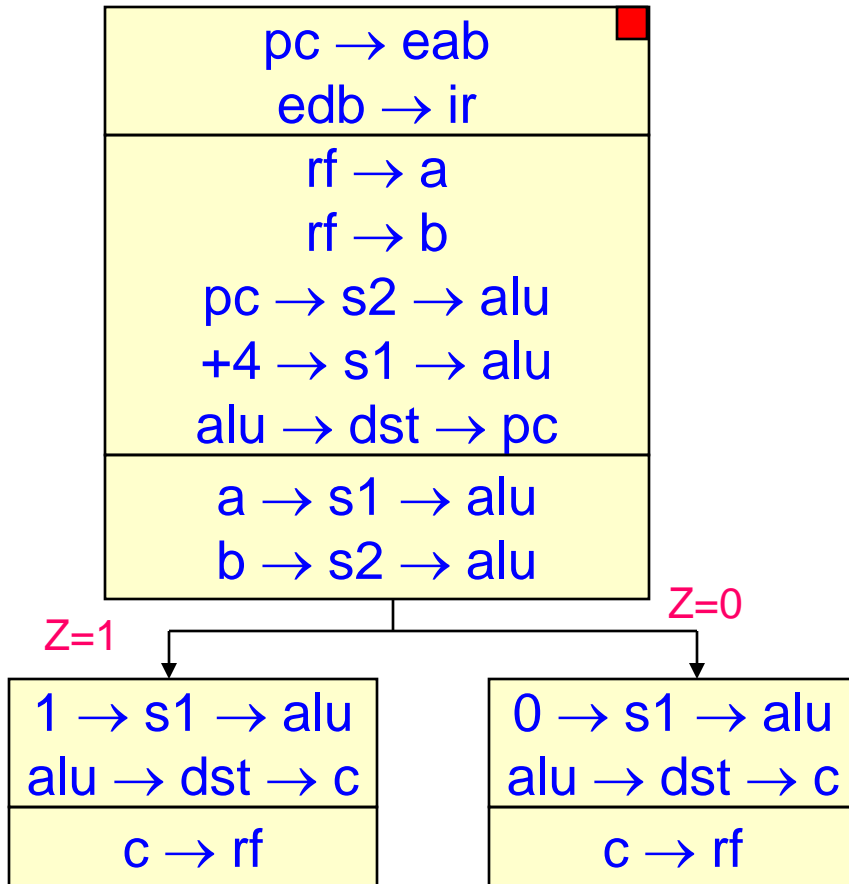


NOP

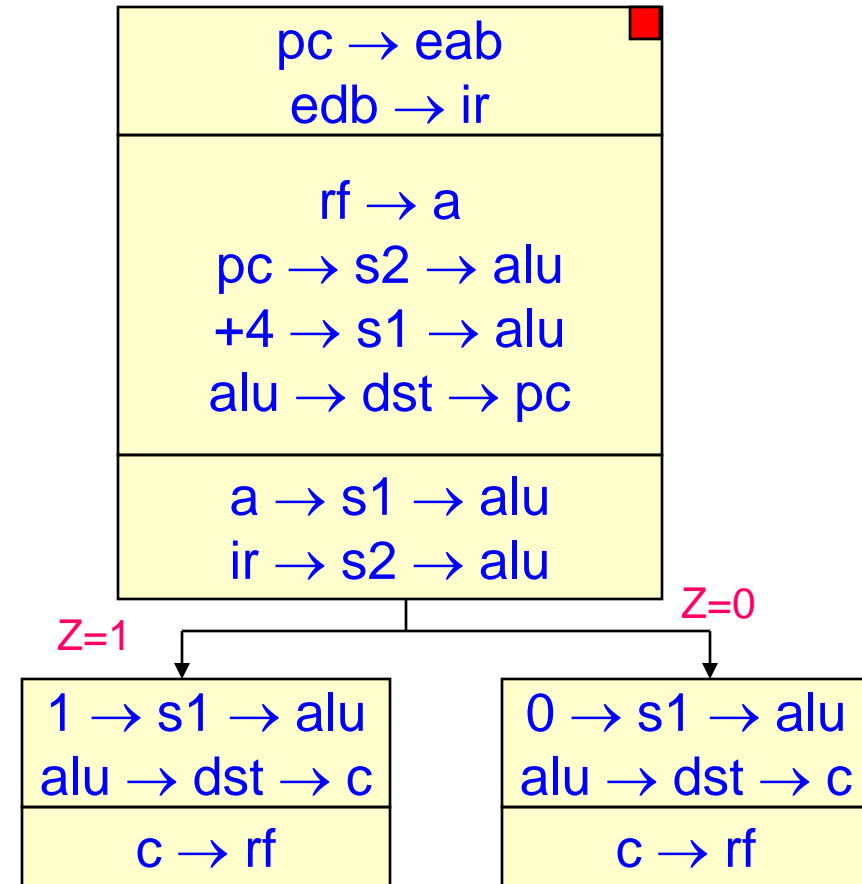


Merged Level 1 Hardware Flowchart

SEQ Rd, Rs1, Rs2

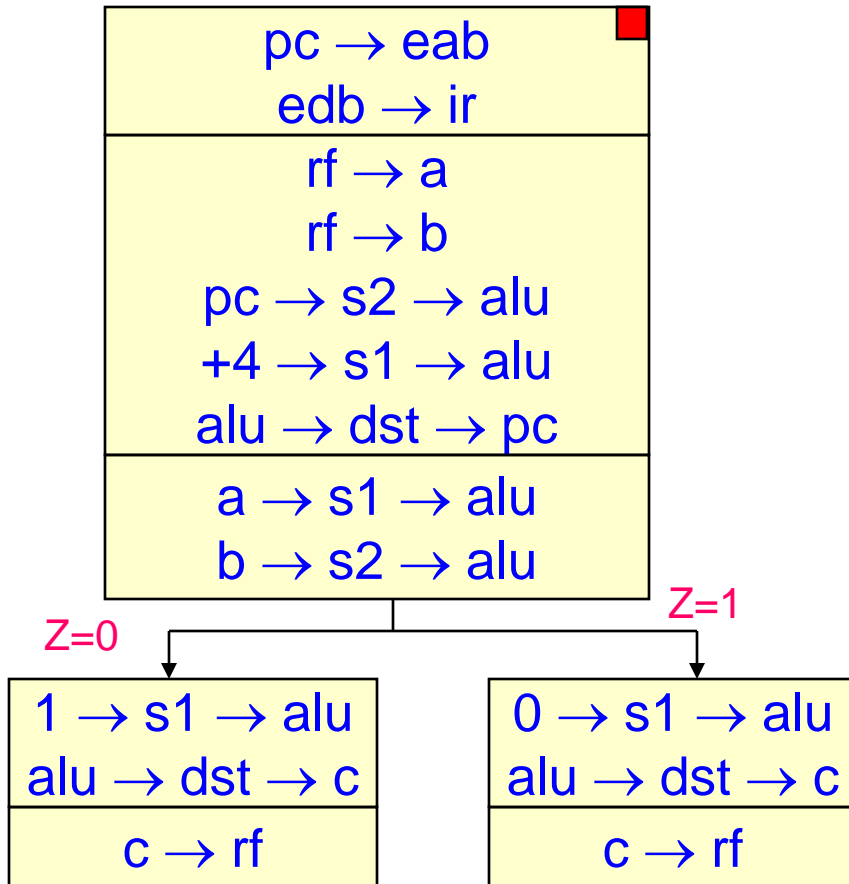


SEQ.I Rd, Rs1, Imm

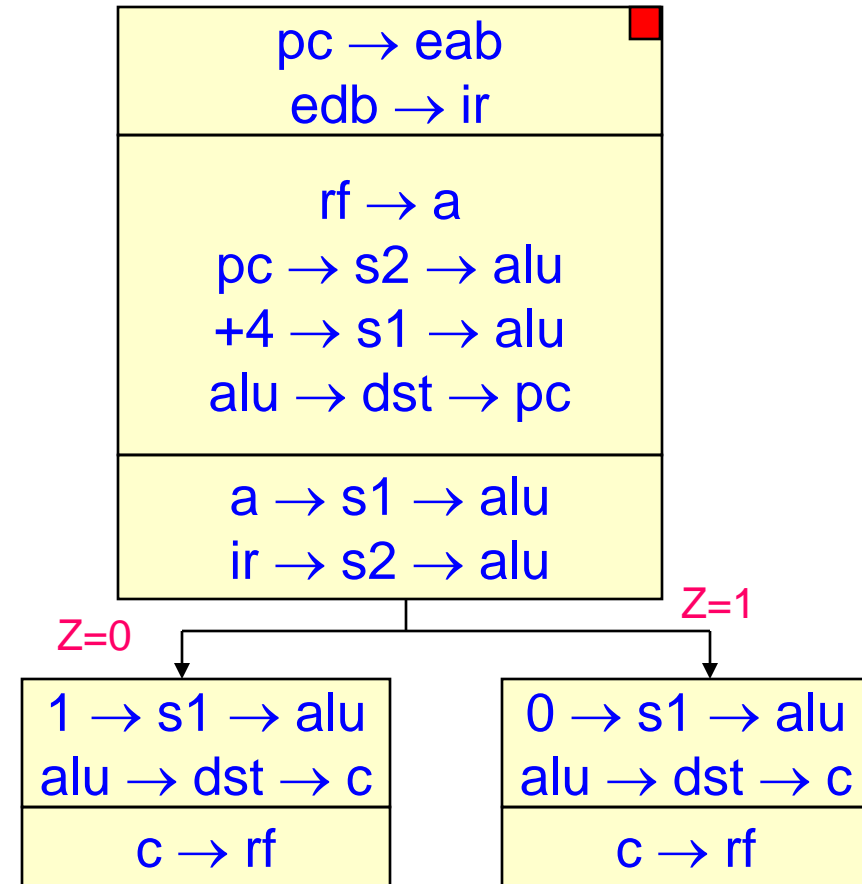


Merged Level 1 Hardware Flowchart

SNE Rd, Rs1, Rs2

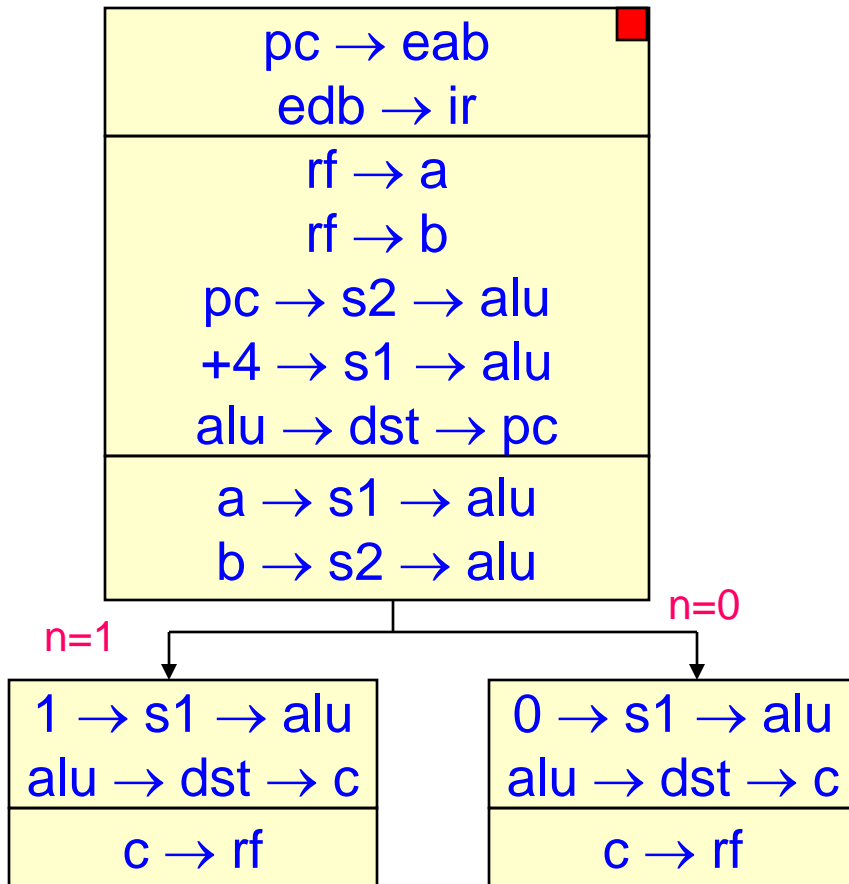


SNE.I Rd, Rs1, Imm

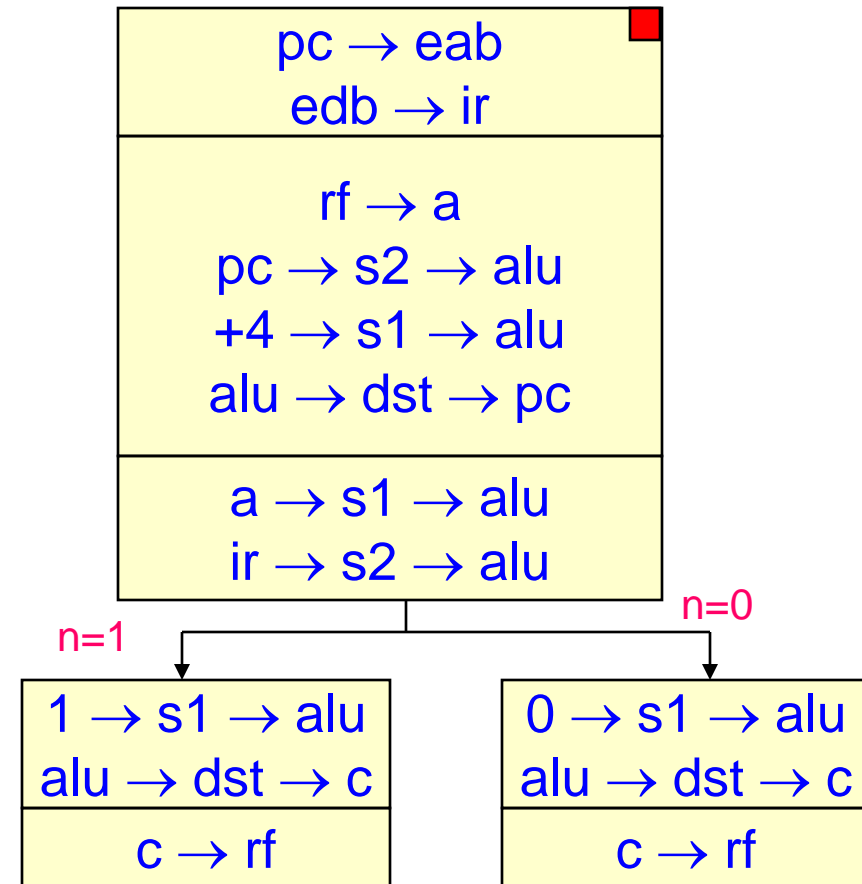


Merged Level 1 Hardware Flowchart

SLT Rd, Rs1, Rs2

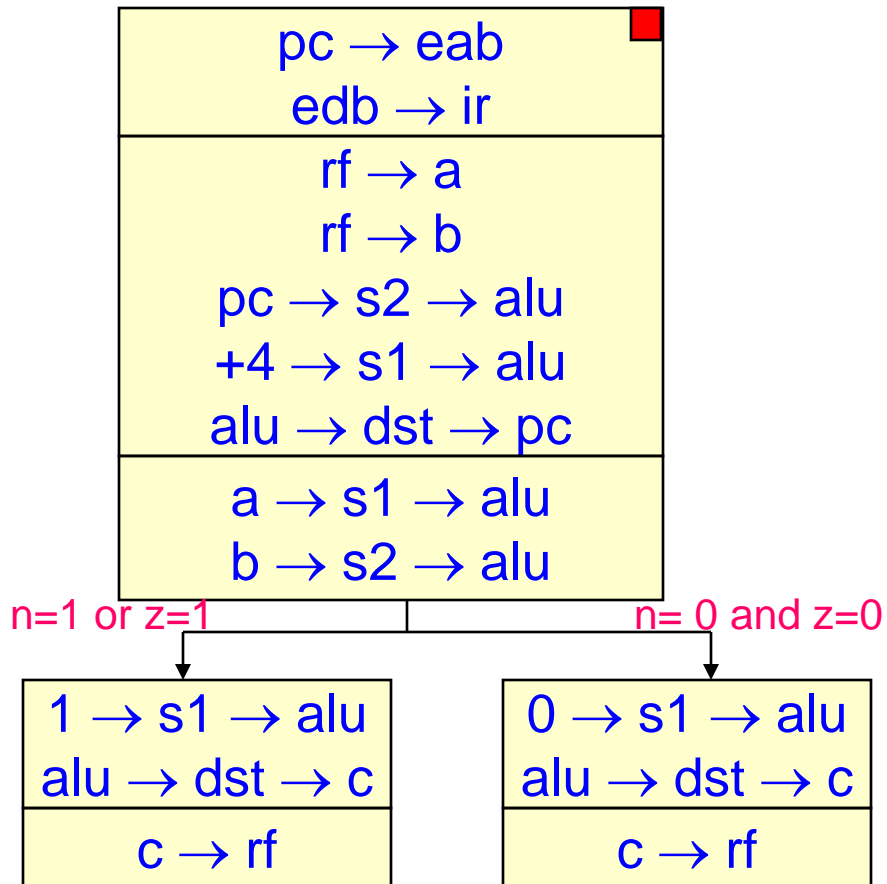


SLT.I Rd, Rs1, Imm

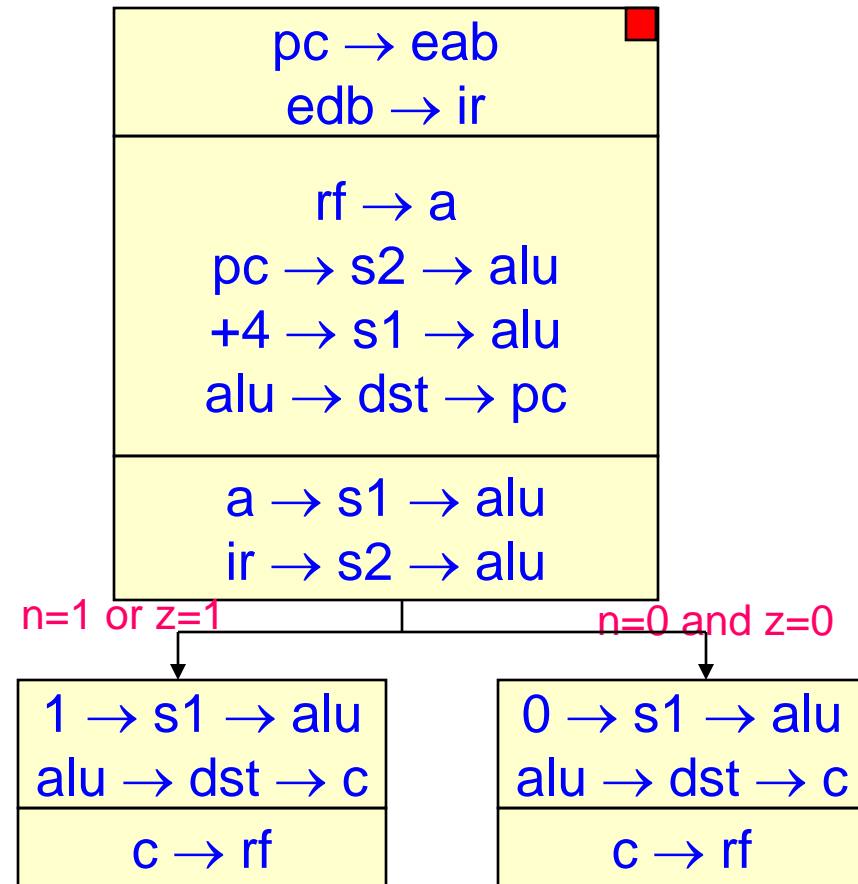


Merged Level 1 Hardware Flowchart

SLE Rd, Rs1, Rs2

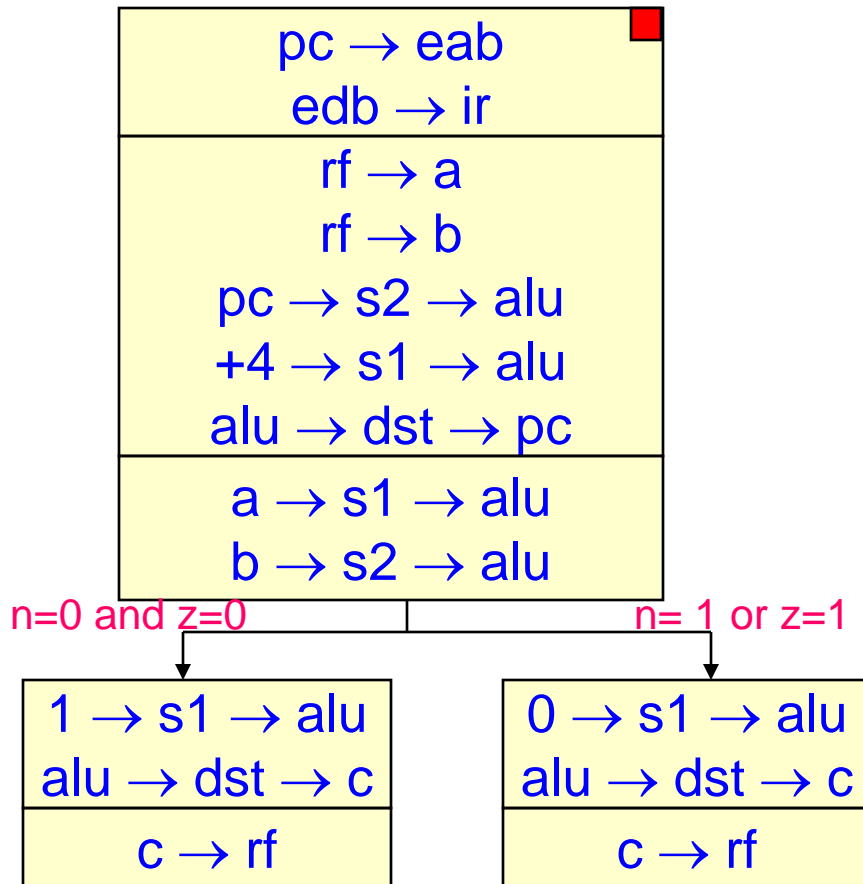


SLE.I Rd, Rs1, Imm

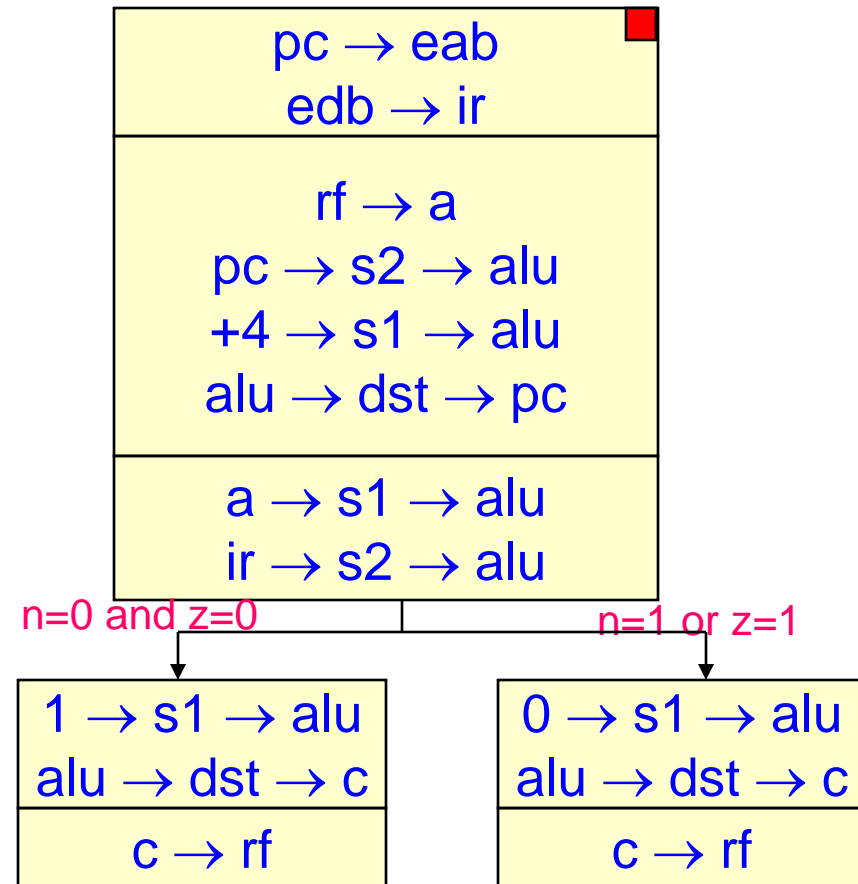


Merged Level 1 Hardware Flowchart

SGT Rd, Rs1, Rs2

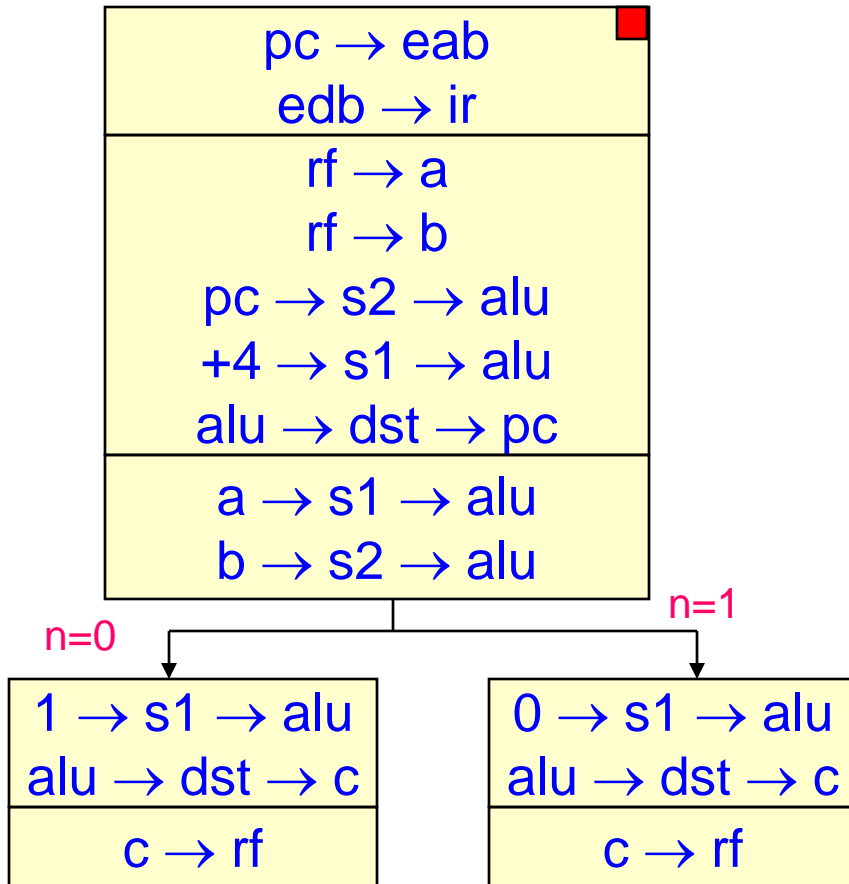


SGT.I Rd, Rs1, Imm

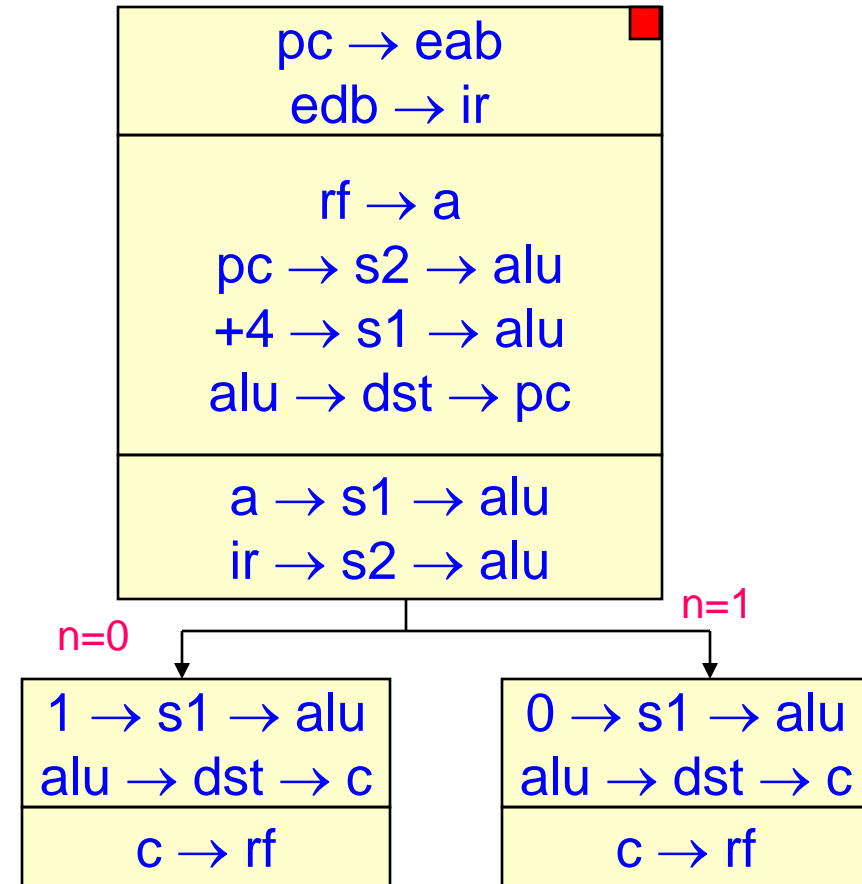


Merged Level 1 Hardware Flowchart

SGE Rd, Rs1, Rs2

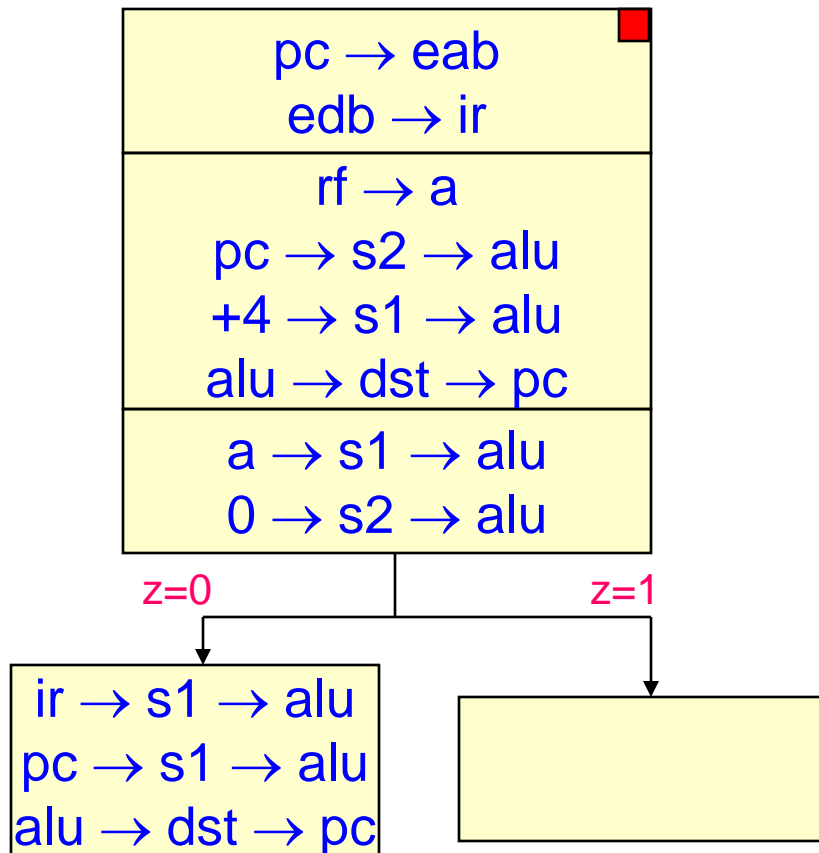


SGE.I Rd, Rs1, Imm

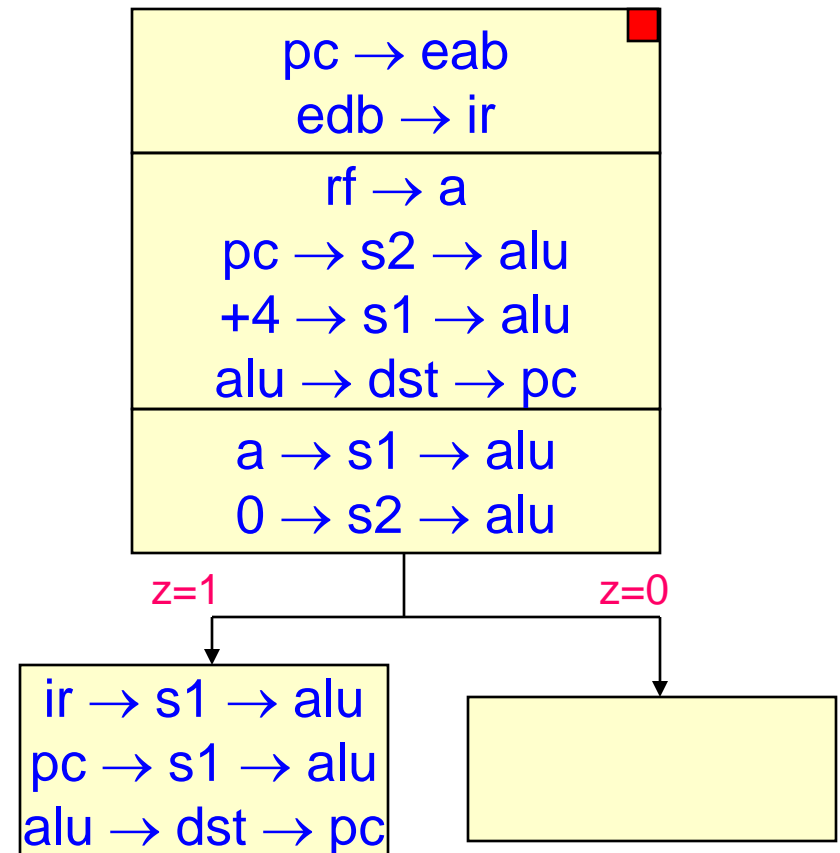


Merged Level 1 Hardware Flowchart

BEQZ Rs, Label

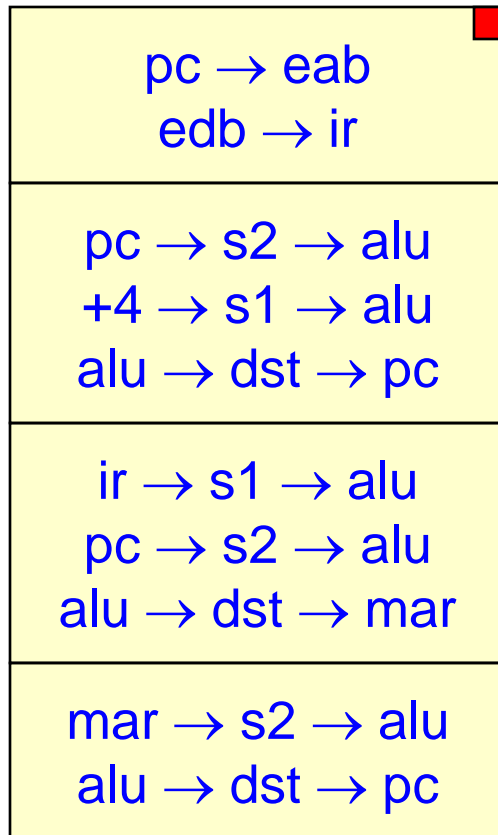


BNEZ Rs, Label

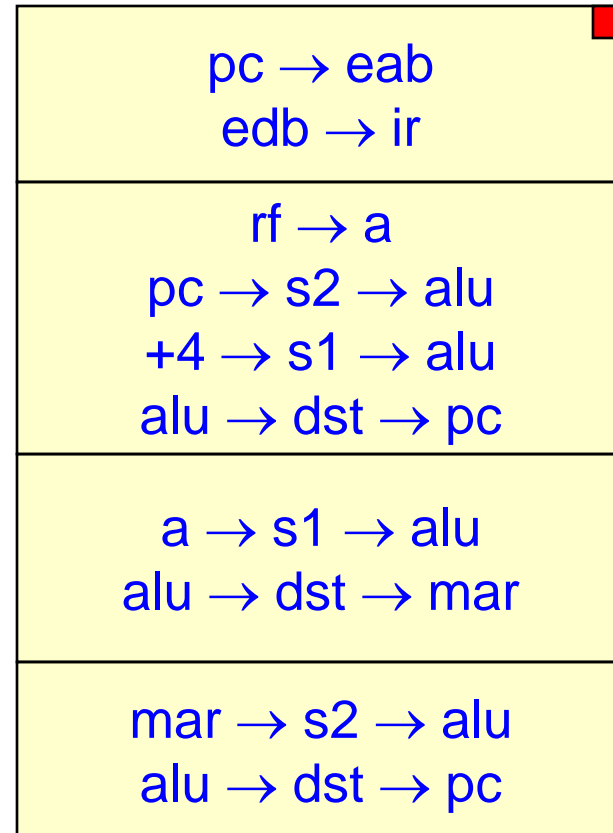


Merged Level 1 Hardware Flowchart

J Label



JR Rs



Merged Level 1 Hardware Flowchart

JAL Label

pc \rightarrow eab edb \rightarrow ir
pc \rightarrow s2 \rightarrow alu +4 \rightarrow s1 \rightarrow alu alu \rightarrow dst \rightarrow pc
pc \rightarrow s2 \rightarrow alu alu \rightarrow dst \rightarrow c
ir \rightarrow s1 \rightarrow alu pc \rightarrow s2 \rightarrow alu alu \rightarrow dst \rightarrow mar c \rightarrow rf
mar \rightarrow s2 \rightarrow alu alu \rightarrow dst \rightarrow pc

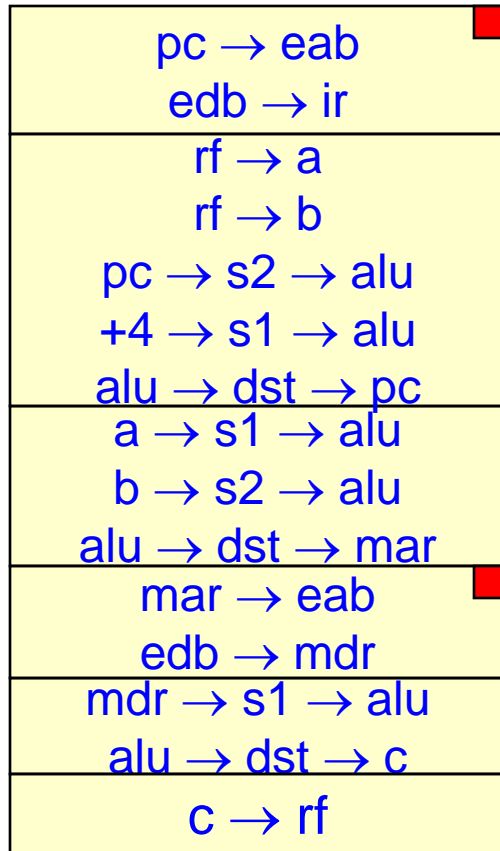
JALR Rs

pc \rightarrow eab edb \rightarrow ir
rf \rightarrow a pc \rightarrow s2 \rightarrow alu +4 \rightarrow s1 \rightarrow alu alu \rightarrow dst \rightarrow pc
pc \rightarrow s2 \rightarrow alu alu \rightarrow dst \rightarrow c
a \rightarrow s1 \rightarrow alu alu \rightarrow dst \rightarrow mar c \rightarrow rf
mar \rightarrow s2 \rightarrow alu alu \rightarrow dst \rightarrow pc

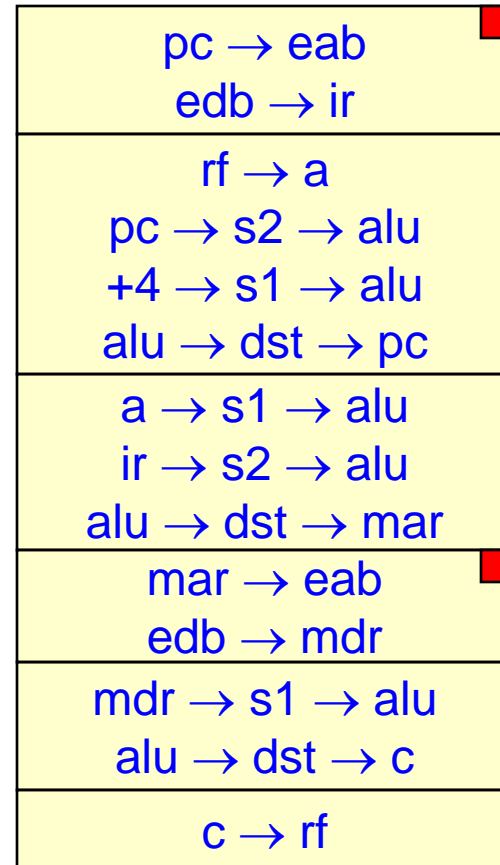


Merged Level 1 Hardware Flowchart

LW Rd, Rs2(Rs1)

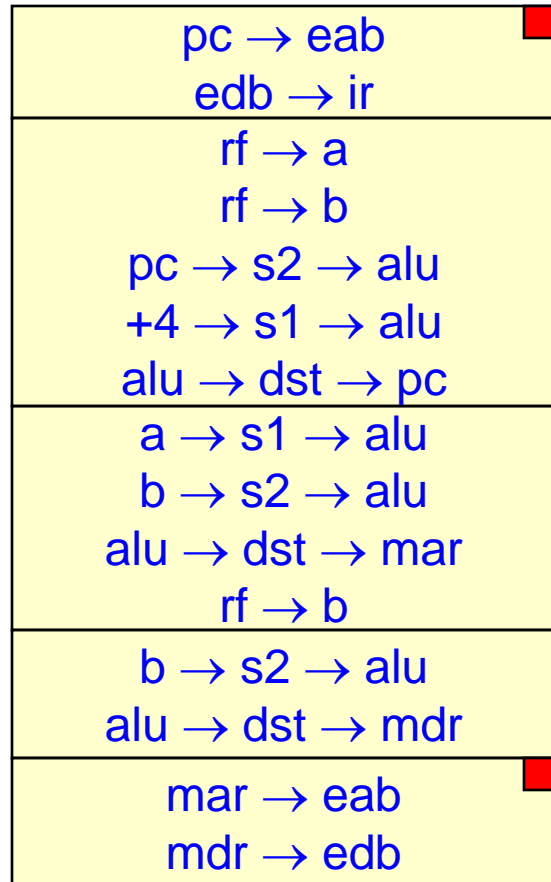


LW.I Rd, Imm(Rs1)

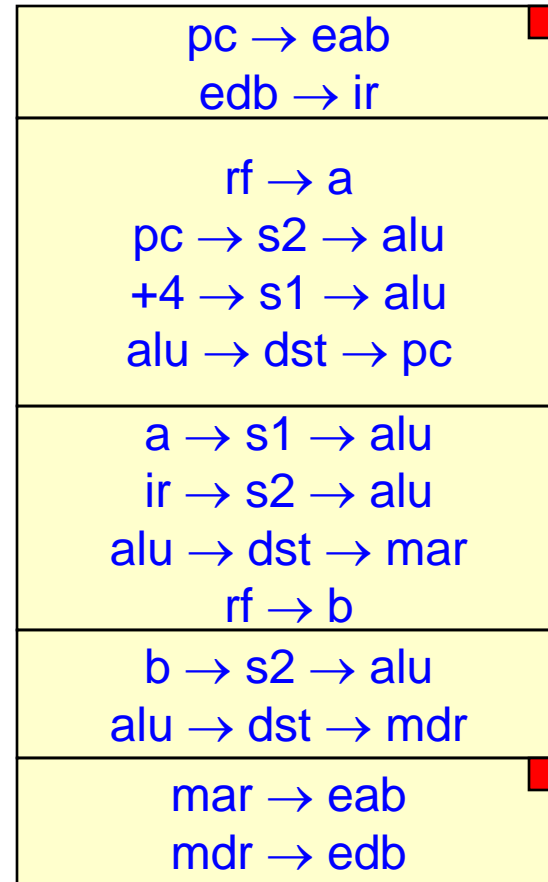


Merged Level 1 Hardware Flowchart

SW Rd, Rs2(Rs1)



SW.I Rd, Imm(Rs1)



Merged Level 1 Hardware Flowchart

LH Rd, Rs2(Rs1)

pc → eab edb → ir
rf → a rf → b pc → s2 → alu +4 → s1 → alu alu → dst → pc
a → s1 → alu b → s2 → alu alu → dst → mar
mar → eab edb → mdr
mdr → s1 → alu 16 → s2 → alu alu → dst → c
c → rf

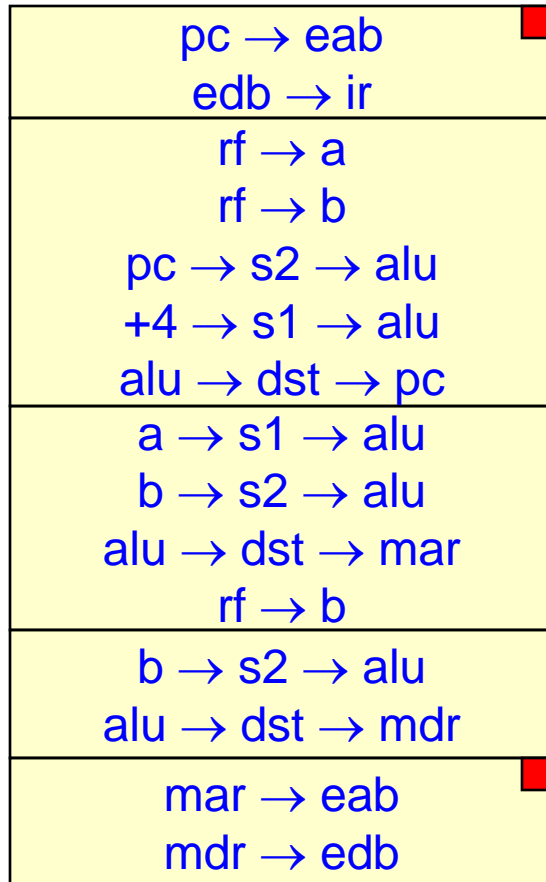
LW.I Rd, Imm(Rs1)

pc → eab edb → ir
rf → a pc → s2 → alu +4 → s1 → alu alu → dst → pc
a → s1 → alu ir → s2 → alu alu → dst → mar
mar → eab edb → mdr
mdr → s1 → alu 16 → s2 → alu alu → dst → c
c → rf

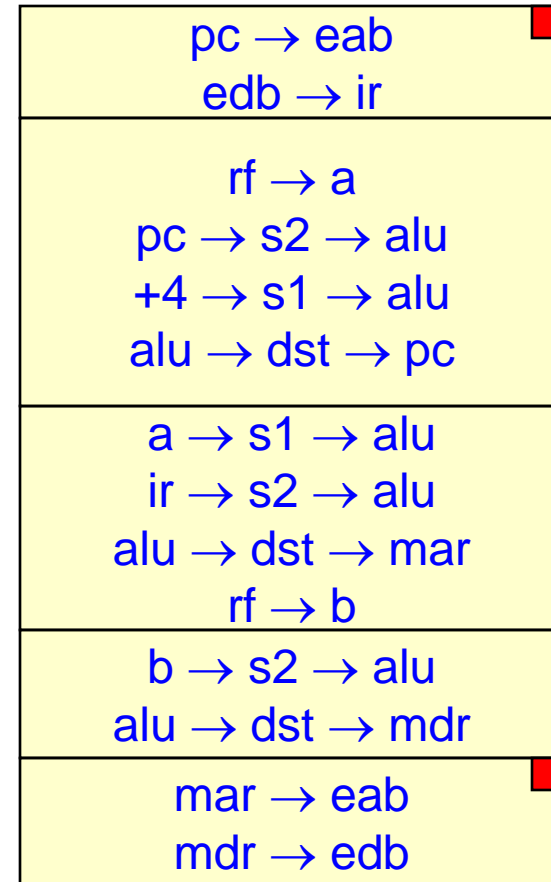


Merged Level 1 Hardware Flowchart

SH Rd, Rs2(Rs1)



SH.I Rd, Imm(Rs1)



Merged Level 1 Hardware Flowchart

LB Rd, Rs2(Rs1)

pc \rightarrow eab edb \rightarrow ir
rf \rightarrow a rf \rightarrow b pc \rightarrow s2 \rightarrow alu +4 \rightarrow s1 \rightarrow alu alu \rightarrow dst \rightarrow pc
a \rightarrow s1 \rightarrow alu b \rightarrow s2 \rightarrow alu alu \rightarrow dst \rightarrow mar
mar \rightarrow eab edb \rightarrow mdr
mdr \rightarrow s1 \rightarrow alu 24 \rightarrow s2 \rightarrow alu alu \rightarrow dst \rightarrow c
c \rightarrow rf

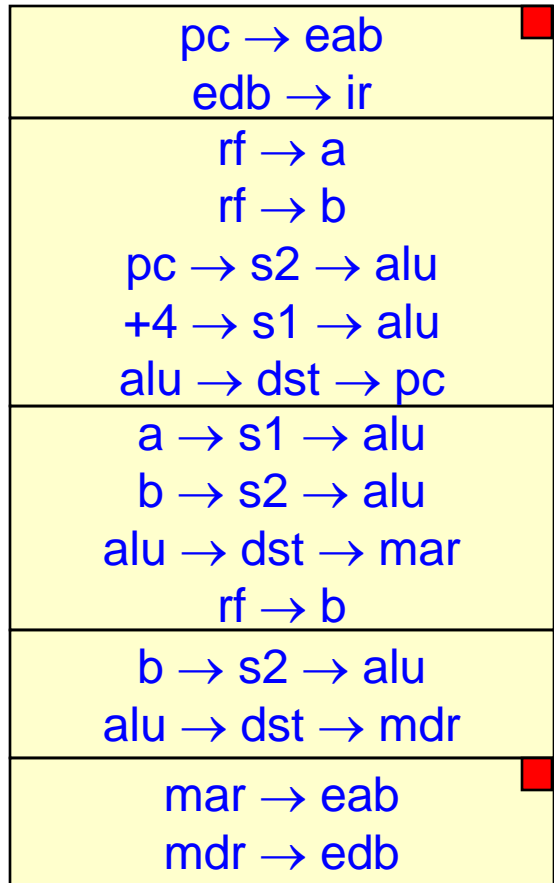
LB.I Rd, Imm(Rs1)

pc \rightarrow eab edb \rightarrow ir
rf \rightarrow a pc \rightarrow s2 \rightarrow alu +4 \rightarrow s1 \rightarrow alu alu \rightarrow dst \rightarrow pc
a \rightarrow s1 \rightarrow alu ir \rightarrow s2 \rightarrow alu alu \rightarrow dst \rightarrow mar
mar \rightarrow eab edb \rightarrow mdr
mdr \rightarrow s1 \rightarrow alu 24 \rightarrow s2 \rightarrow alu alu \rightarrow dst \rightarrow c
c \rightarrow rf

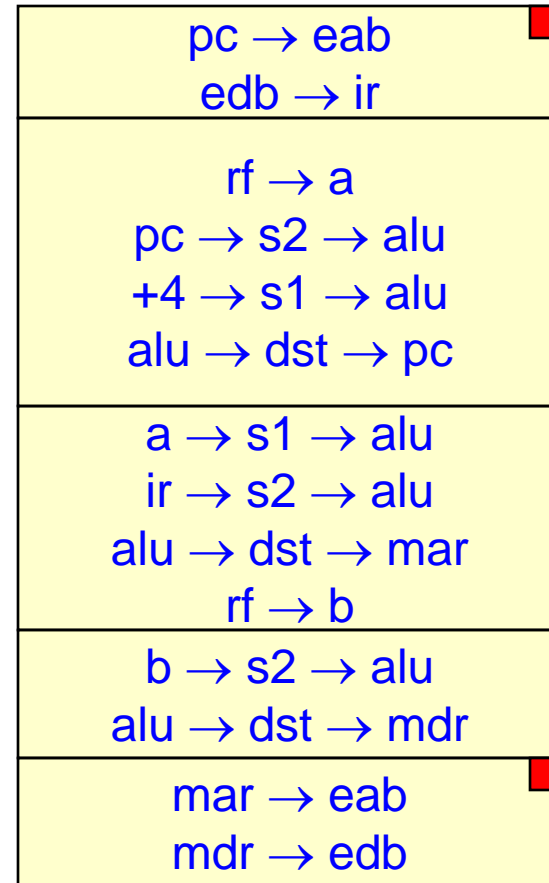


Merged Level 1 Hardware Flowchart

SB Rd, Rs2(Rs1)

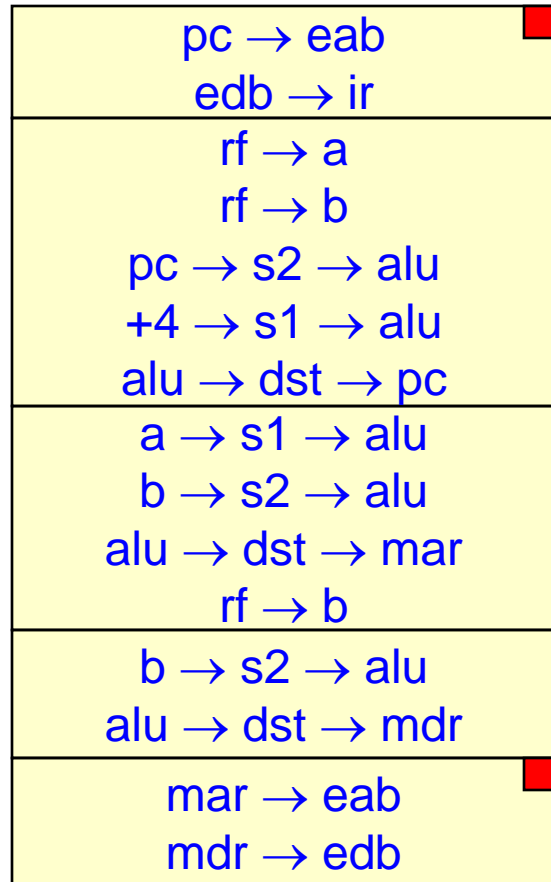


SB.I Rd, Imm(Rs1)

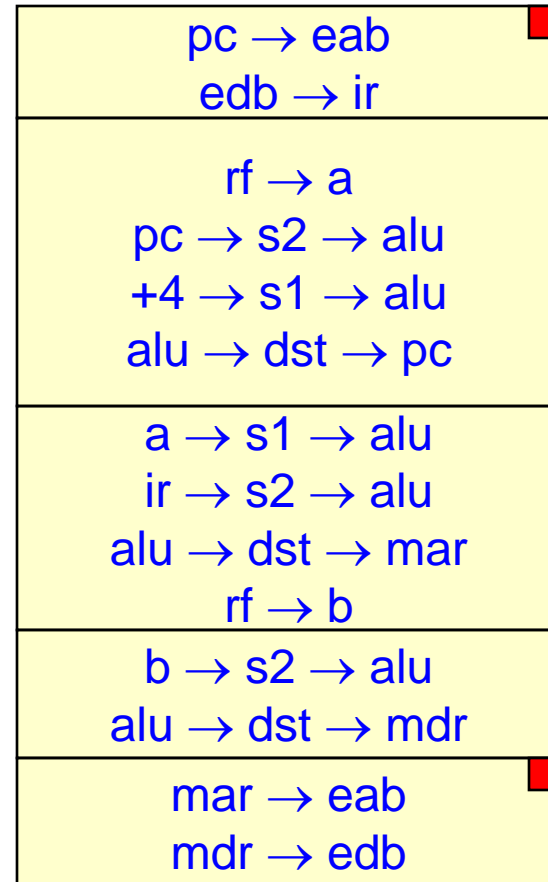


Merged Level 1 Hardware Flowchart

SB Rd, Rs2(Rs1)

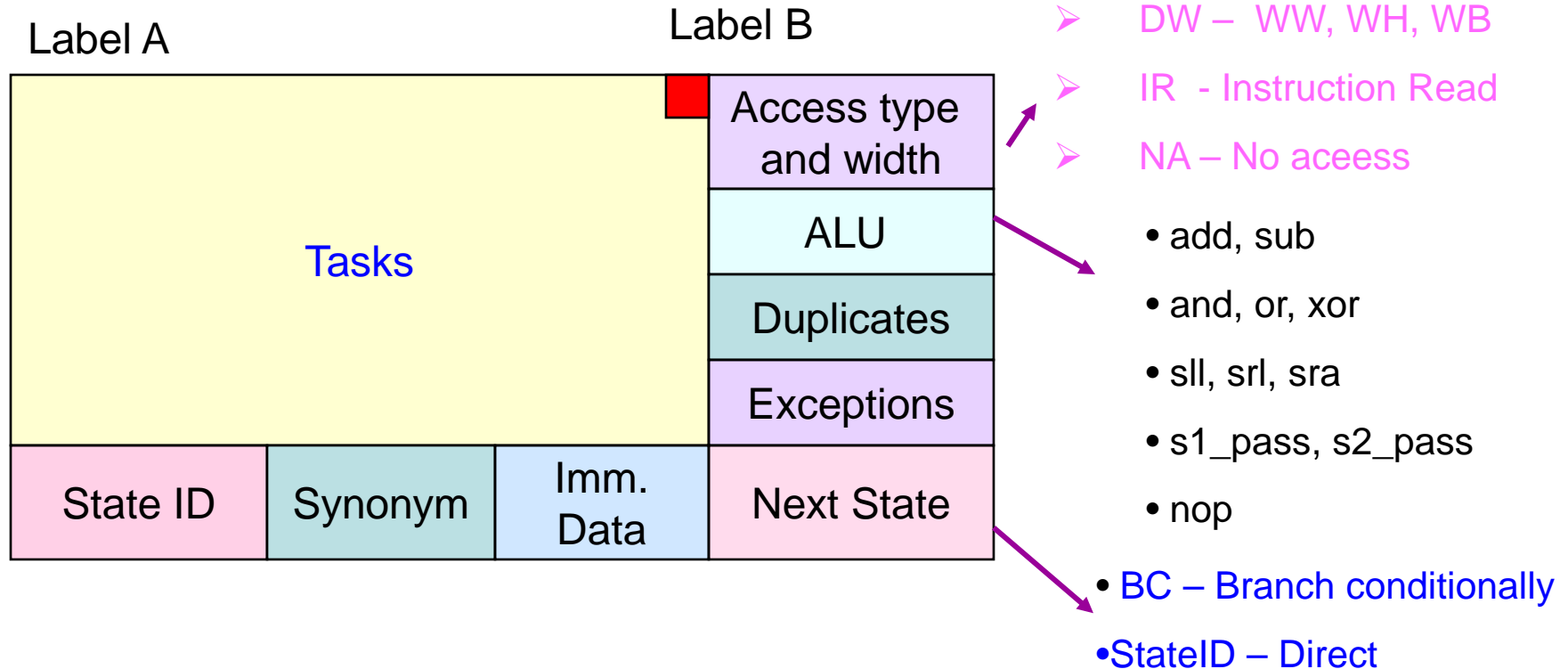


SB.I Rd, Imm(Rs1)



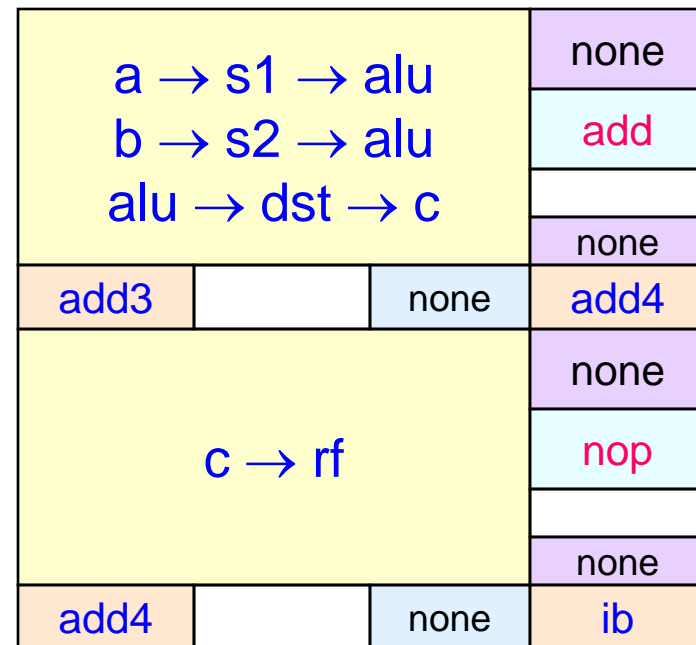
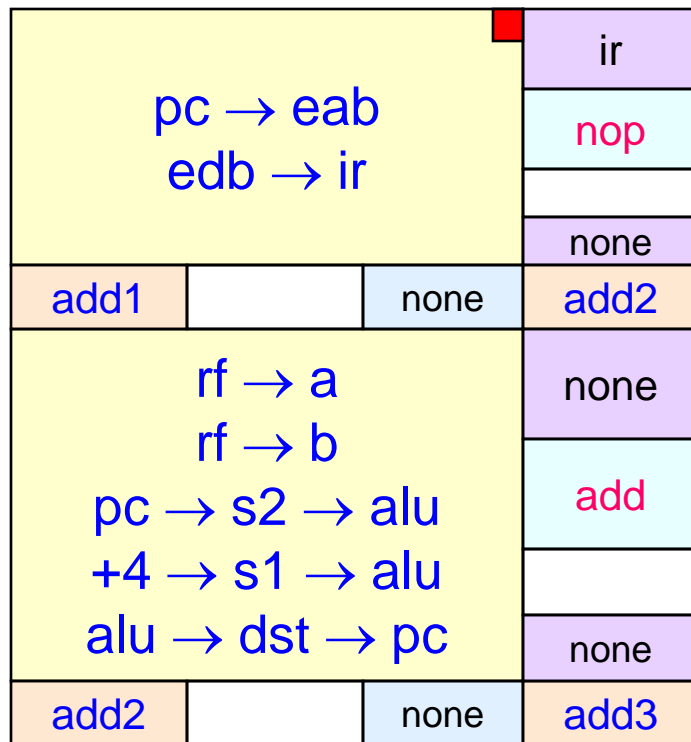
Level 2 Flowcharts

Format for Level 2 flowchart state



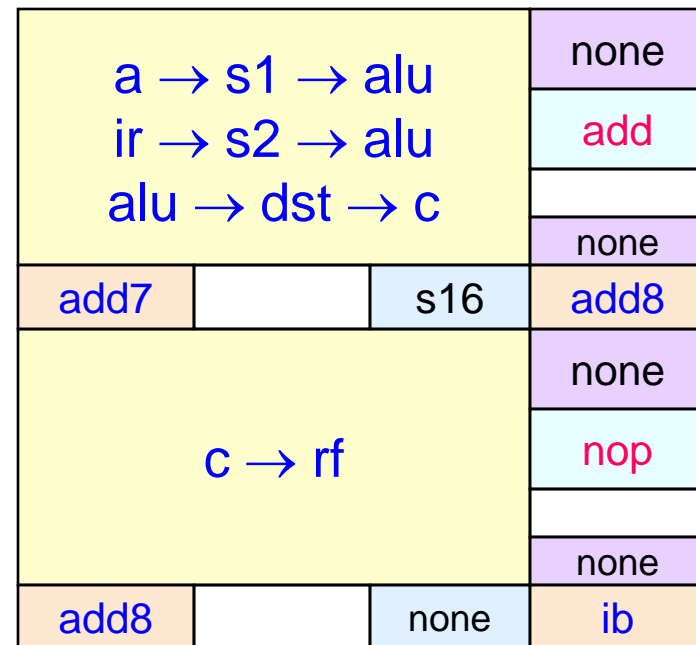
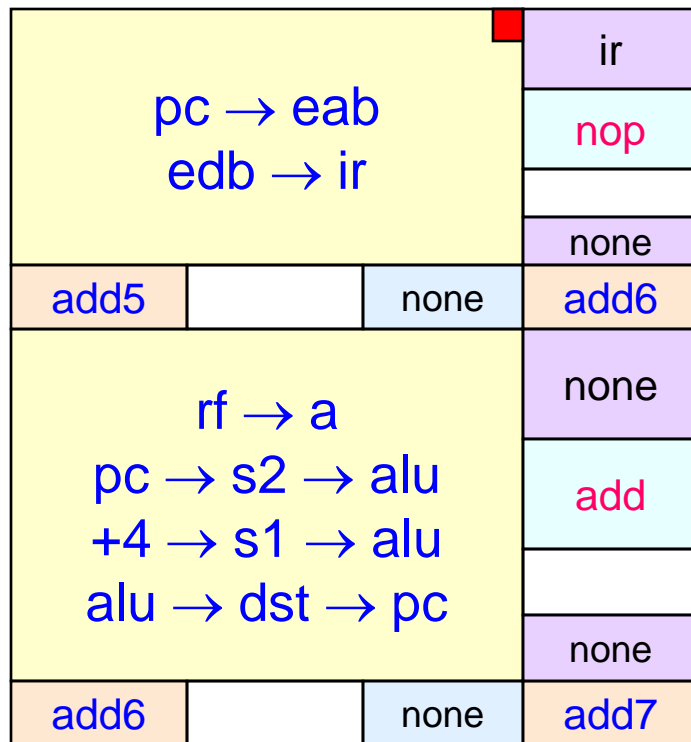
Level 2 Hardware Flowchart

ADD Rd, Rs1, Rs2



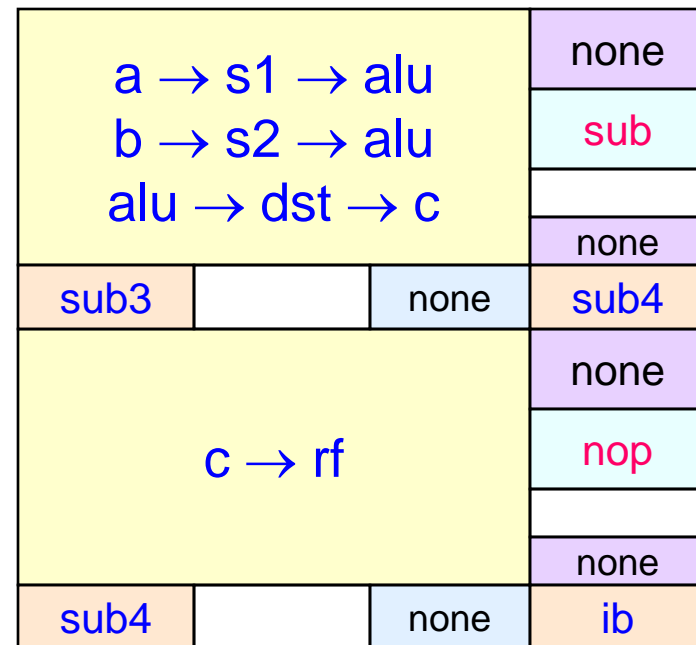
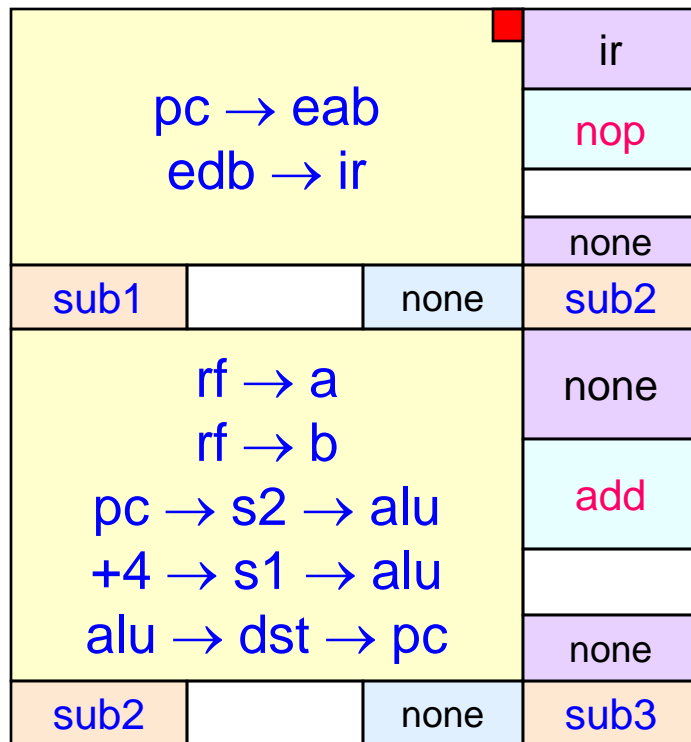
Level 2 Hardware Flowchart

ADD.I Rd, Rs1, Imm



Level 2 Hardware Flowchart

SUB Rd, Rs1, Rs2



Level 2 Hardware Flowchart

SUB.I Rd, Rs1, Imm

$pc \rightarrow eab$ $edb \rightarrow ir$			ir
			nop
			none
			none
sub5		none	sub6
$rf \rightarrow a$ $pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$			none
			add
			none
			none
sub6		none	sub7

$a \rightarrow s1 \rightarrow alu$ $ir \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow c$			none
			sub
			none
			none
sub7		s16	sub8
$c \rightarrow rf$			none
			nop
			none
			none
sub8		none	ib

Level 2 Hardware Flowchart

AND Rd, Rs1, Rs2

$pc \rightarrow eab$ $edb \rightarrow ir$			ir
			nop
			none
			none
and1		none	and2
$rf \rightarrow a$ $rf \rightarrow b$ $pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$			none
			add
			none
			none
and2		none	and3

$a \rightarrow s1 \rightarrow alu$ $b \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow c$			none
			and
			none
and3		none	and4
$c \rightarrow rf$			none
			nop
			none
and4		none	ib

Level 2 Hardware Flowchart

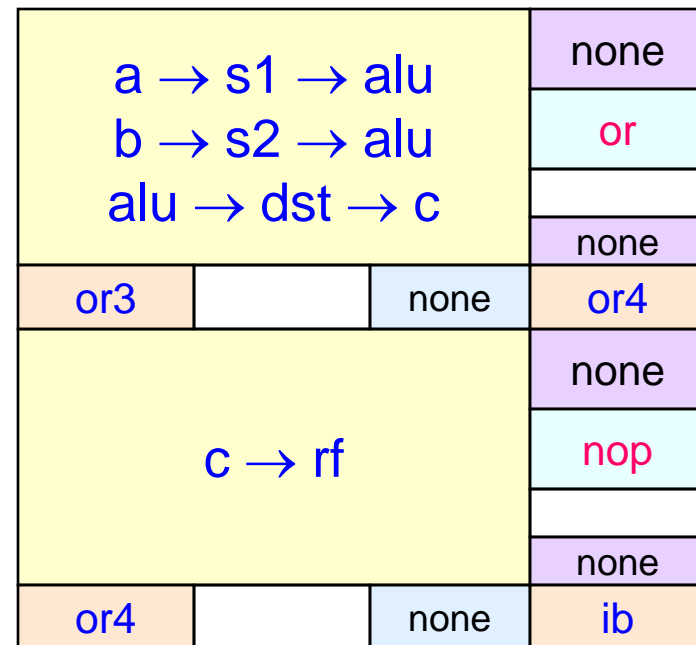
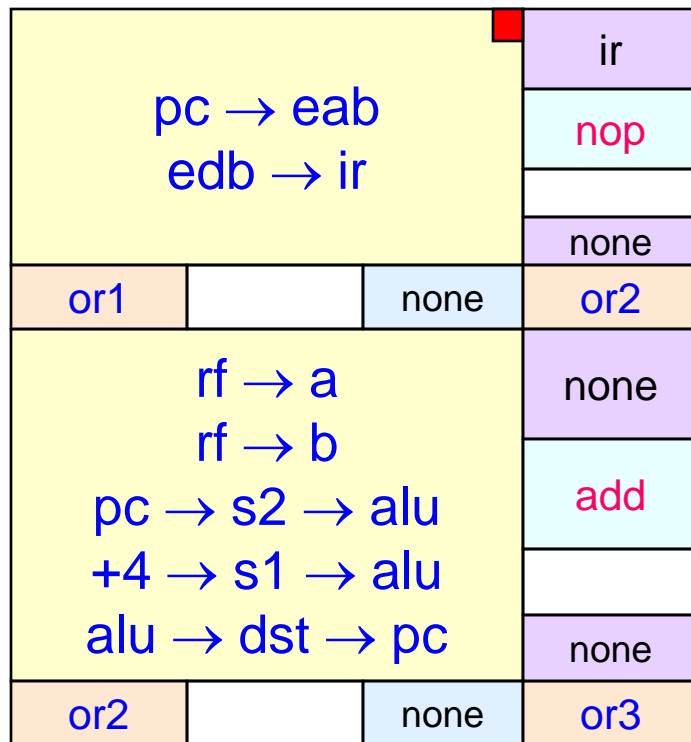
AND.I Rd, Rs1, Imm

$pc \rightarrow eab$ $edb \rightarrow ir$			ir
			nop
			none
			none
and5		none	and6
$rf \rightarrow a$ $pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$			none
			add
			none
			none
and6		none	and7

$a \rightarrow s1 \rightarrow alu$ $ir \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow c$			none
			and
			none
			none
and7		s16	and8
$c \rightarrow rf$			none
			nop
			none
			none
and8		none	ib

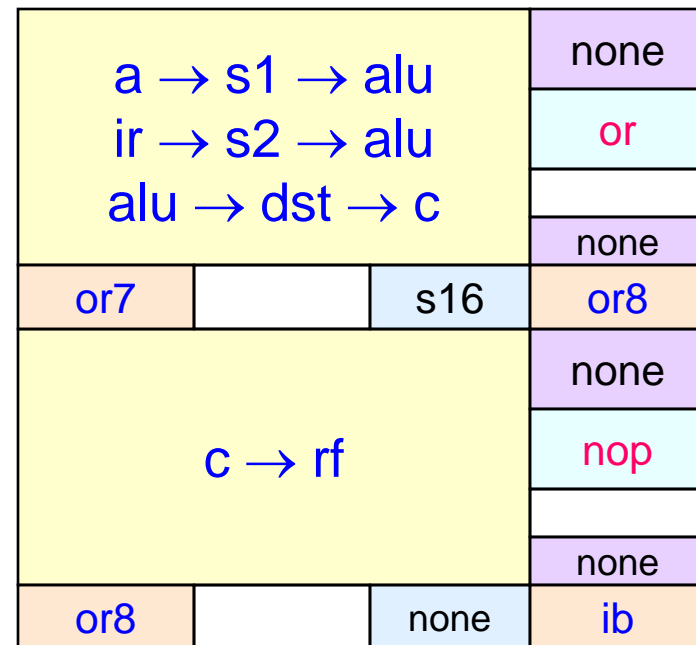
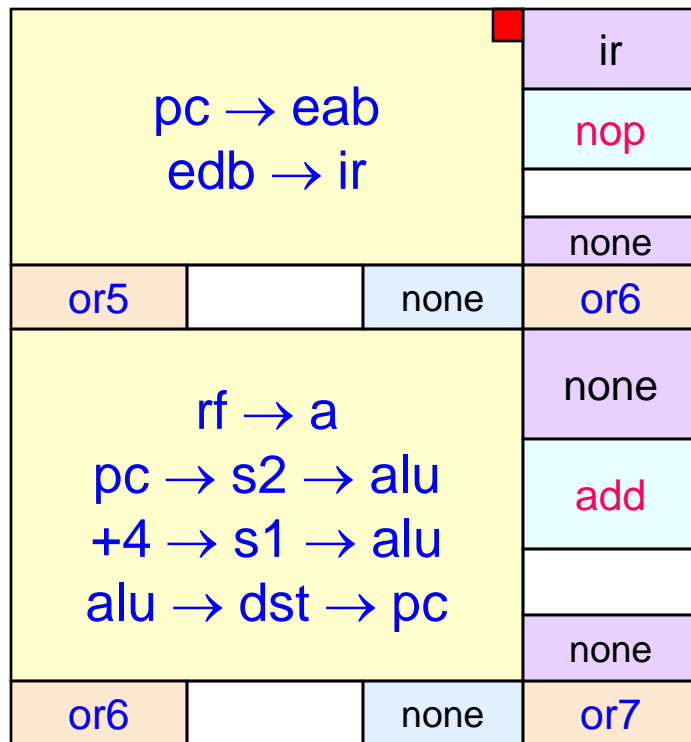
Level 2 Hardware Flowchart

OR Rd, Rs1, Rs2



Level 2 Hardware Flowchart

OR.I Rd, Rs1, Imm



Level 2 Hardware Flowchart

XOR Rd, Rs1, Rs2

$pc \rightarrow eab$ $edb \rightarrow ir$			ir
			nop
			none
			none
xor1		none	xor2
$rf \rightarrow a$ $rf \rightarrow b$ $pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$			none
			add
			none
			none
xor2		none	xor3

$a \rightarrow s1 \rightarrow alu$ $b \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow c$			none
			xor
			none
			none
xor3		none	xor4
$c \rightarrow rf$			none
			nop
			none
			none
xor4		none	ib



Level 2 Hardware Flowchart

SUB.I Rd, Rs1, Imm

$pc \rightarrow eab$ $edb \rightarrow ir$			ir
			nop
			none
			none
xor5		none	xor6
$rf \rightarrow a$ $pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$			none
			add
			none
			none
xor6		none	xor7

$a \rightarrow s1 \rightarrow alu$ $ir \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow c$			none
			xor
			none
			none
xor7		s16	xor8
$c \rightarrow rf$			none
			nop
			none
			none
xor8		none	ib

Level 2 Hardware Flowchart

SLL Rd, Rs1, Rs2

$pc \rightarrow eab$ $edb \rightarrow ir$			ir
			nop
			none
			none
sll1		none	sll2
$rf \rightarrow a$ $rf \rightarrow b$ $pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$			none
			add
			none
			none
sll2		none	sll3

$a \rightarrow s1 \rightarrow alu$ $b \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow c$			none
			sll
			none
			none
sll3		none	sll4
$c \rightarrow rf$			none
			nop
			none
			none
sll4		none	ib

Level 2 Hardware Flowchart

SLL.I Rd, Rs1, Imm

$pc \rightarrow eab$ $edb \rightarrow ir$			ir
			nop
			none
			none
sll5		none	sll6
$rf \rightarrow a$ $pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$			none
			add
			none
			none
sll6		none	sll7

$a \rightarrow s1 \rightarrow alu$ $ir \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow c$			none
			sll
			none
			none
sll7		s16	sll8
$c \rightarrow rf$			none
			nop
			none
			none
sll8		none	ib



Level 2 Hardware Flowchart

SRL Rd, Rs1, Rs2

$pc \rightarrow eab$ $edb \rightarrow ir$			ir
			nop
			none
			none
srl1		none	srl2
$rf \rightarrow a$ $rf \rightarrow b$ $pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$			none
			add
			none
			none
			none
srl2		none	srl3

$a \rightarrow s1 \rightarrow alu$ $b \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow c$			none
			srl
			none
			none
srl3		none	srl4
$c \rightarrow rf$			none
			nop
			none
			none
srl4		none	ib

Level 2 Hardware Flowchart

SRL.I Rd, Rs1, Imm

$pc \rightarrow eab$ $edb \rightarrow ir$			ir
			nop
			none
			none
srl5		none	srl6
$rf \rightarrow a$ $pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$			none
			add
			none
			none
srl6		none	srl7

$a \rightarrow s1 \rightarrow alu$ $ir \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow c$			none
			srl
			none
			none
srl7		s16	srl8
$c \rightarrow rf$			none
			nop
			none
			none
srl8		none	ib

Level 2 Hardware Flowchart

SRA Rd, Rs1, Rs2

$pc \rightarrow eab$ $edb \rightarrow ir$			ir
			nop
			none
			none
sra1		none	sra2
$rf \rightarrow a$ $rf \rightarrow b$ $pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$			none
			add
			none
			none
sra2		none	sra3

$a \rightarrow s1 \rightarrow alu$ $b \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow c$			none
			sra
			none
			none
sra3		none	sra4
$c \rightarrow rf$			none
			nop
			none
			none
sra4		none	ib

Level 2 Hardware Flowchart

SRA.I Rd, Rs1, Imm

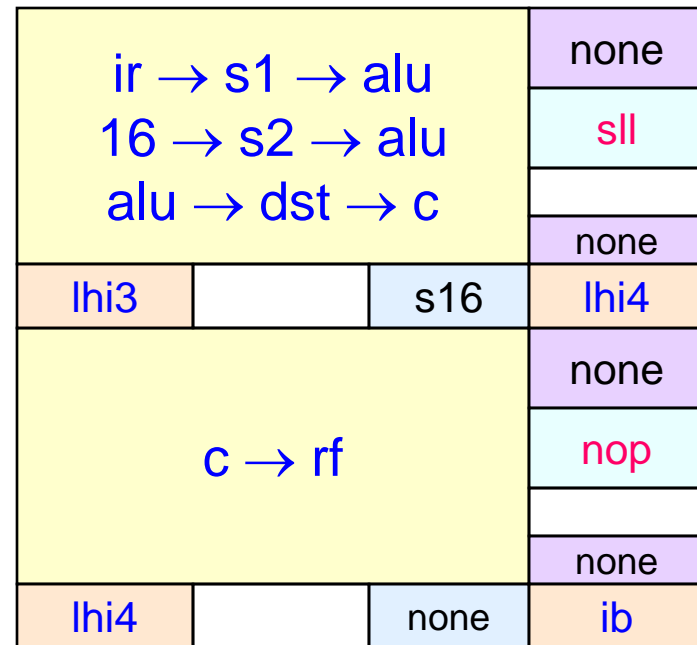
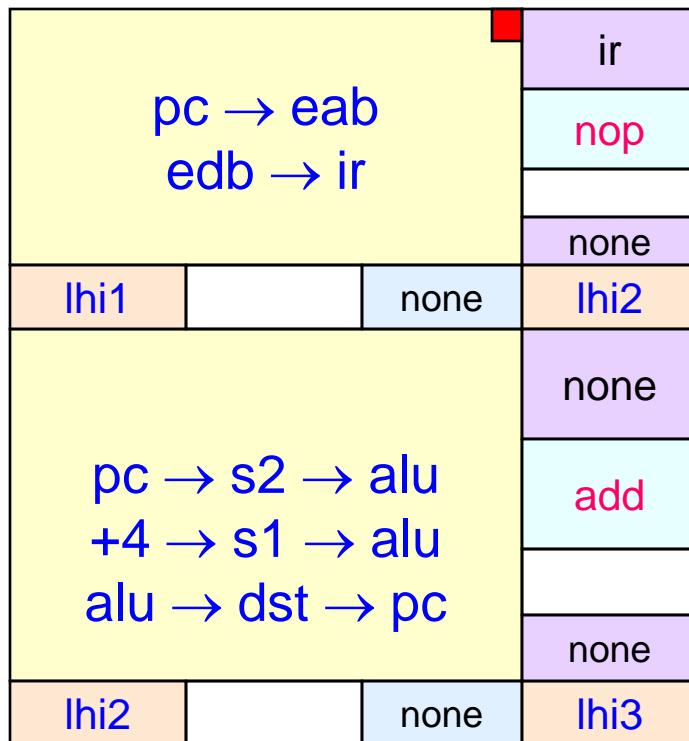
$pc \rightarrow eab$ $edb \rightarrow ir$			ir
			nop
			none
			none
sra5		none	sra6
$rf \rightarrow a$ $pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$			none
			add
			none
			none
sra6		none	sra7

$a \rightarrow s1 \rightarrow alu$ $ir \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow c$			none
			sra
			none
			none
sra7		s16	sra8
$c \rightarrow rf$			none
			nop
			none
			none
sra8		none	ib

Level 2 Hardware Flowchart

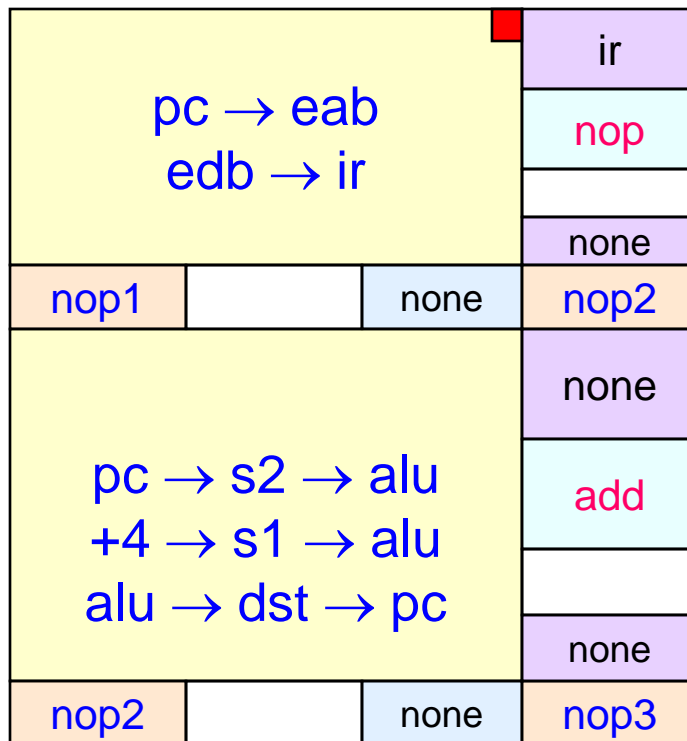
LHI Rd, Imm

Rd(0:15) <- Imm; Rd(16:31) <- hex0000



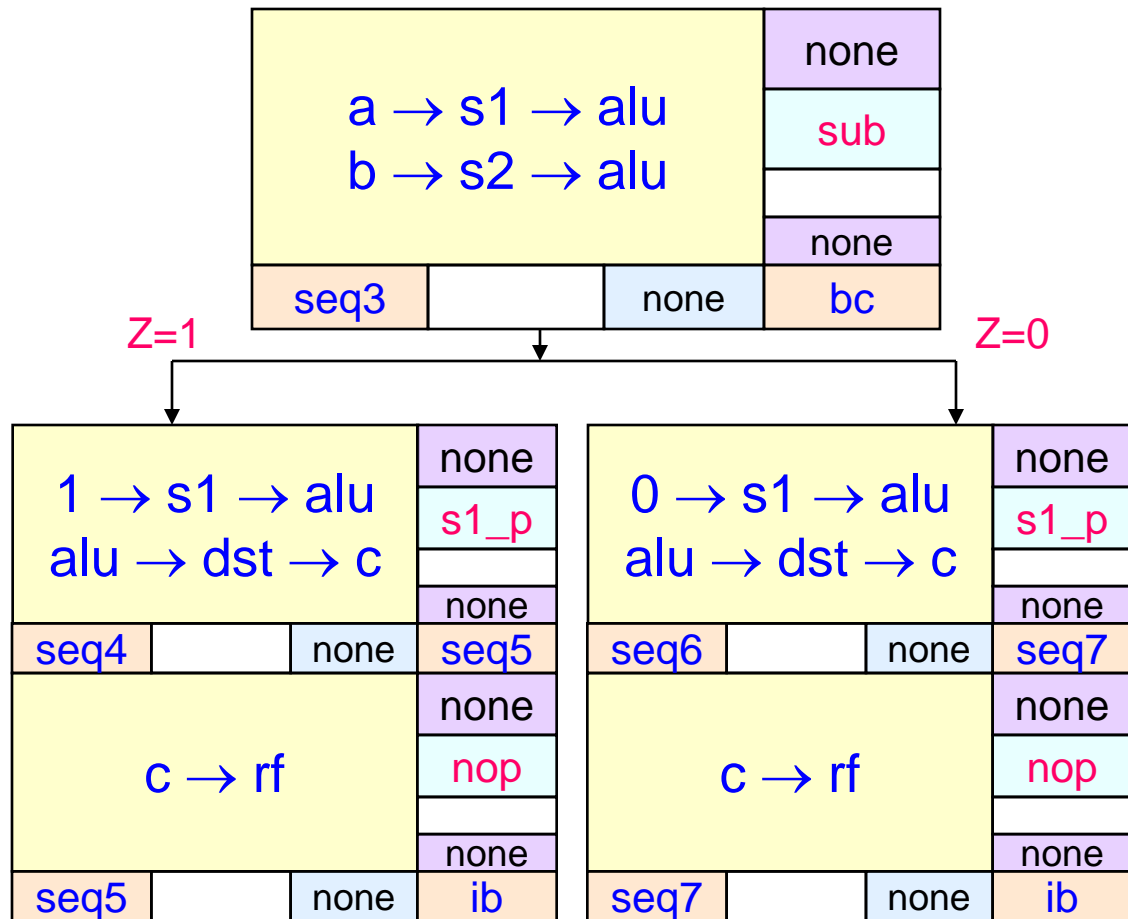
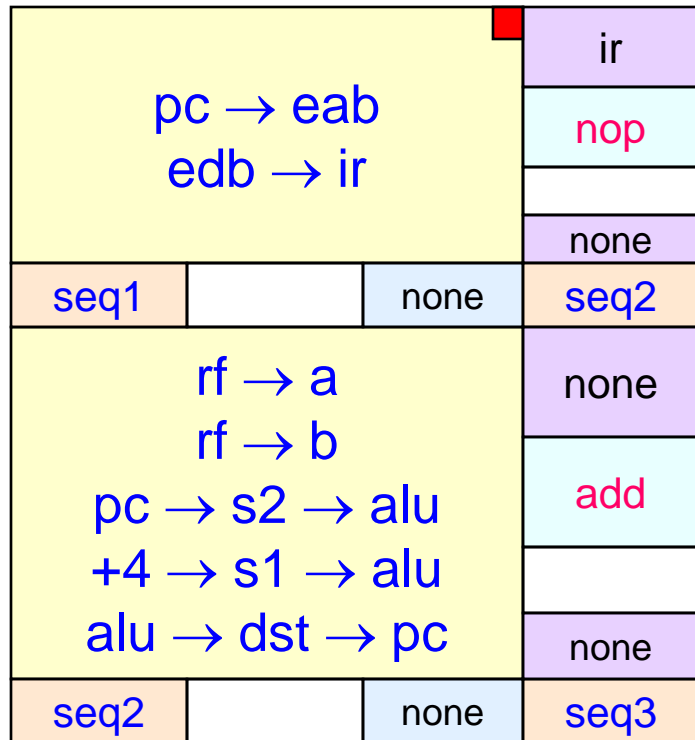
Level 2 Hardware Flowchart

NOP



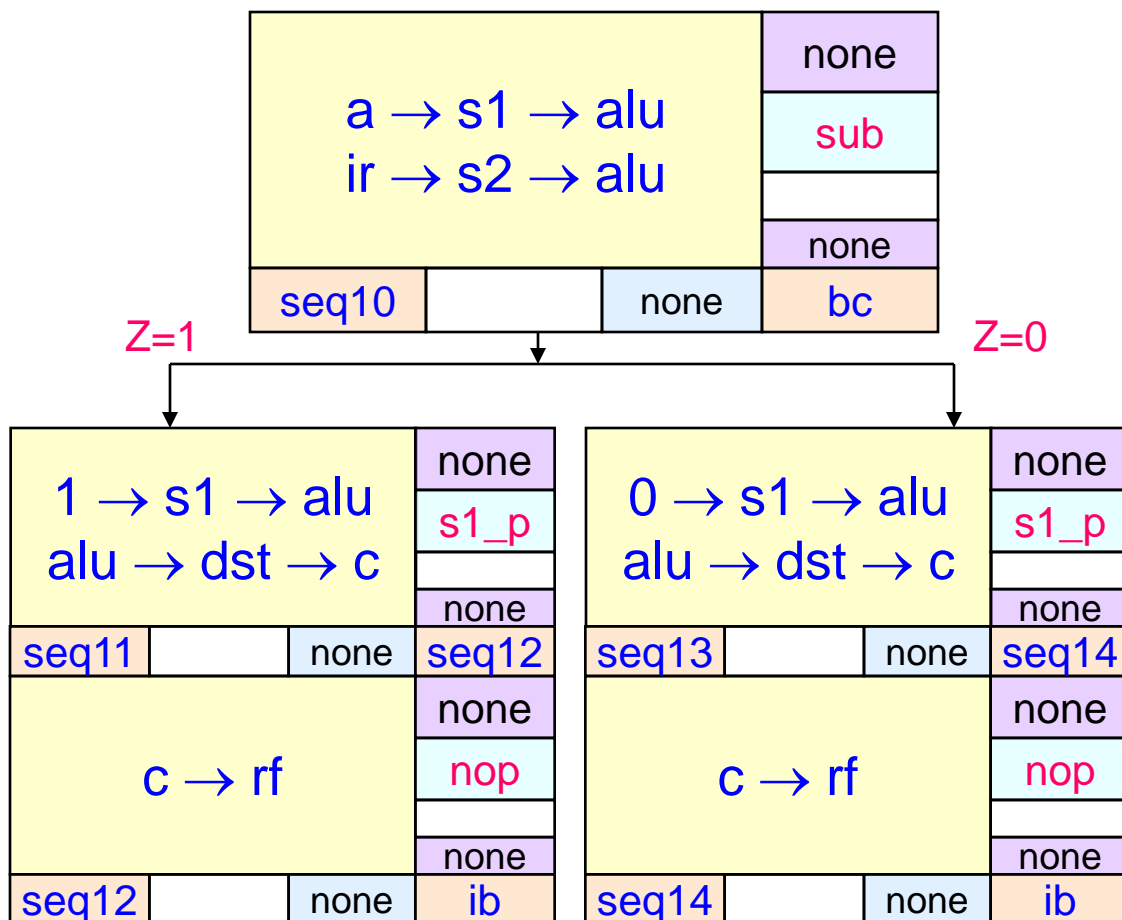
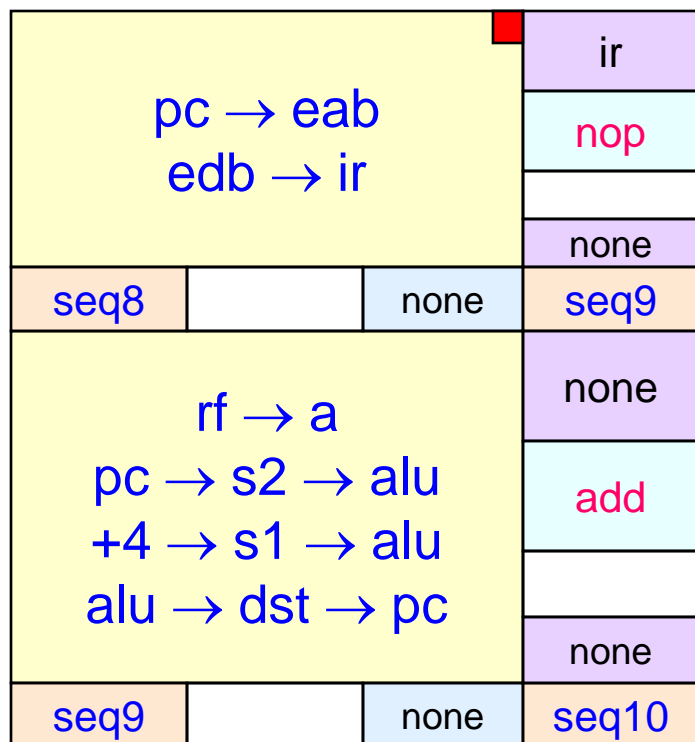
Level 2 Hardware Flowchart

SEQ Rd, Rs1, Rs2



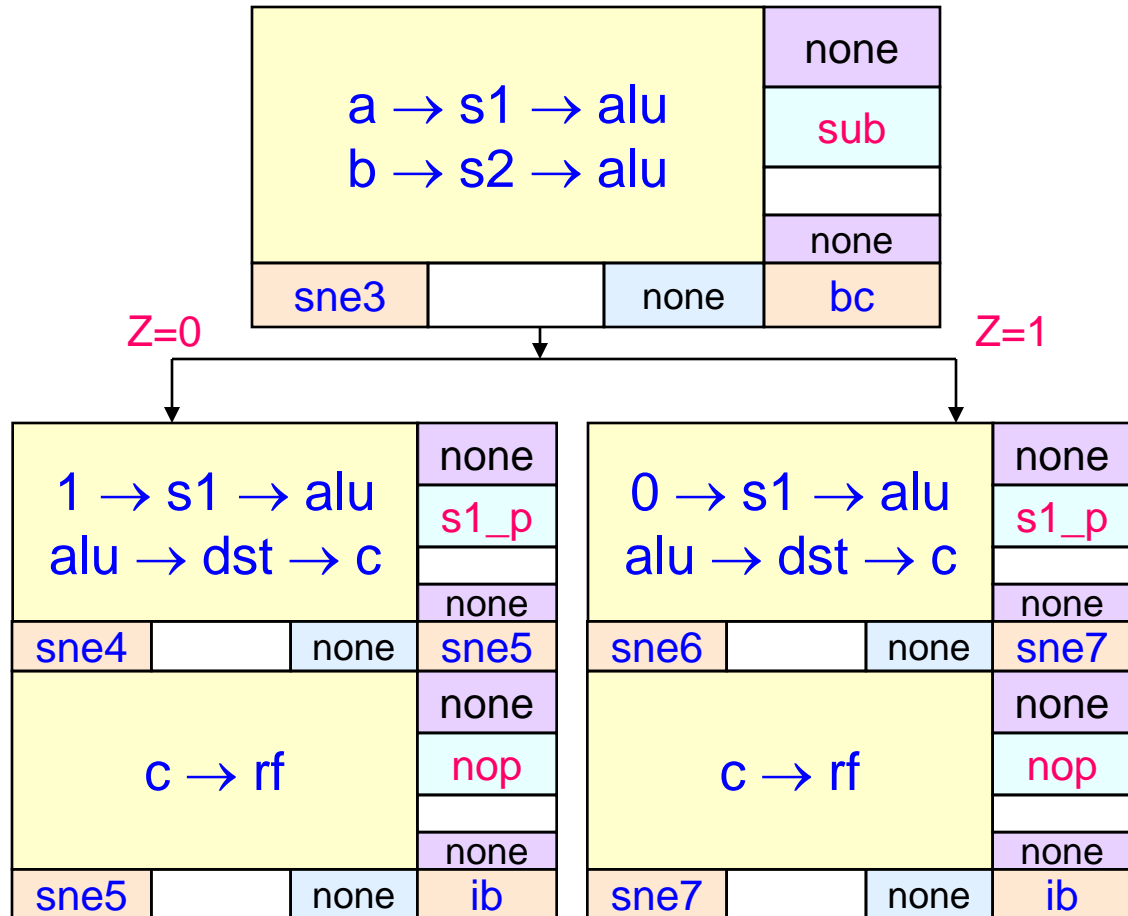
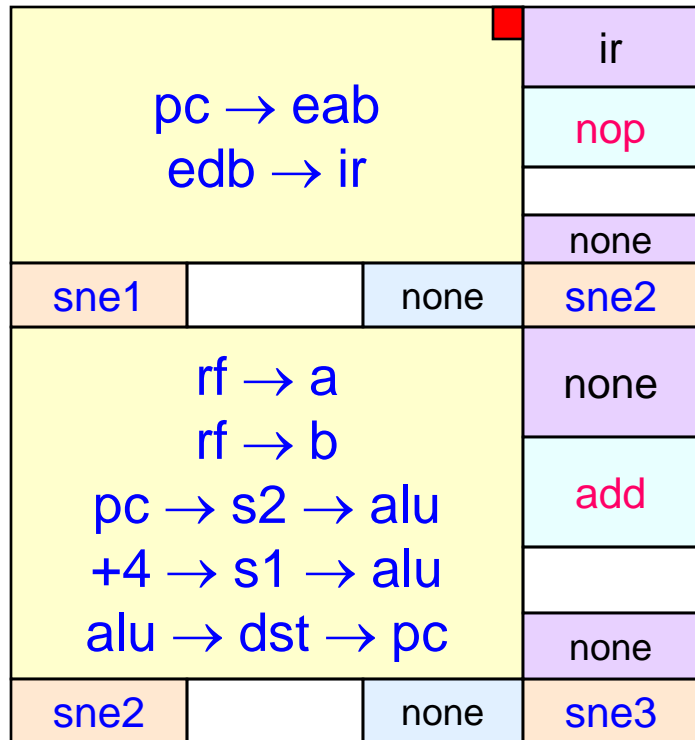
Level 2 Hardware Flowchart

SEQ.I Rd, Rs1, Imm



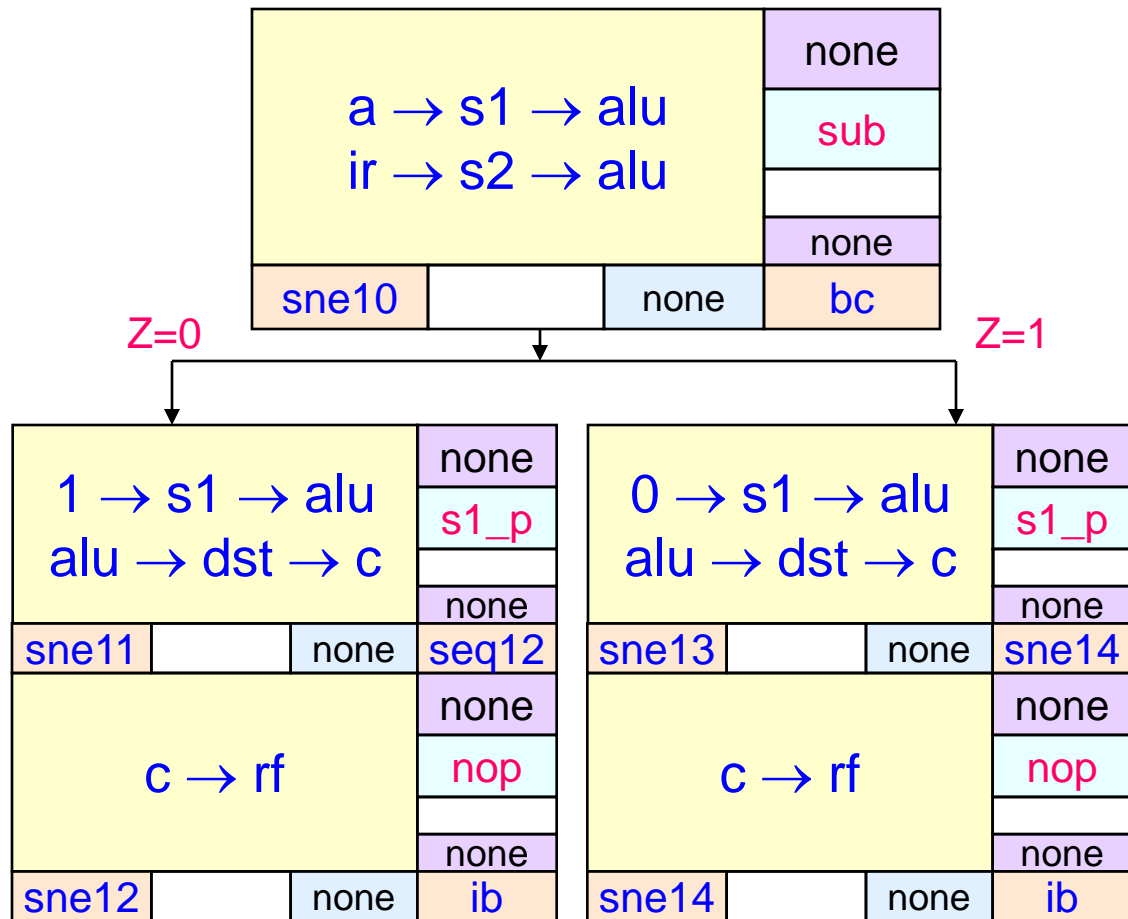
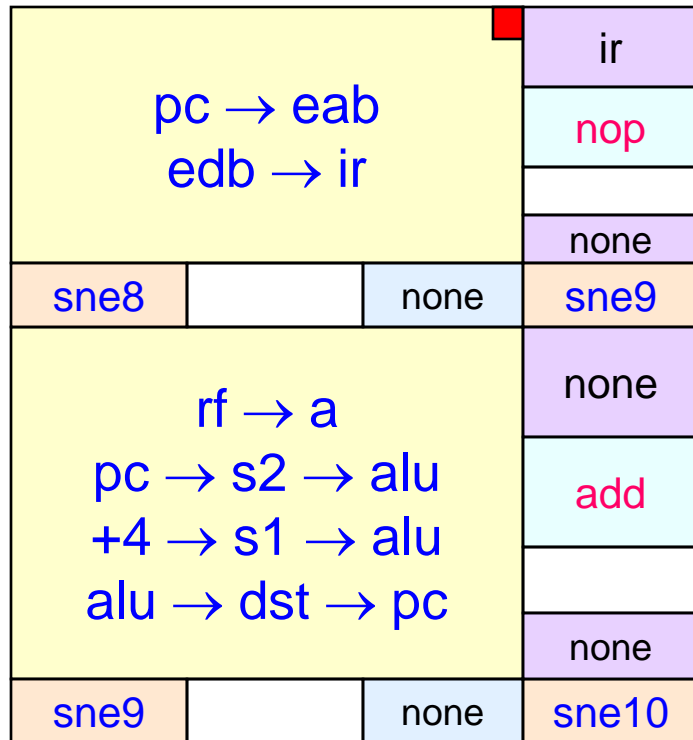
Level 2 Hardware Flowchart

SNE Rd, Rs1, Rs2



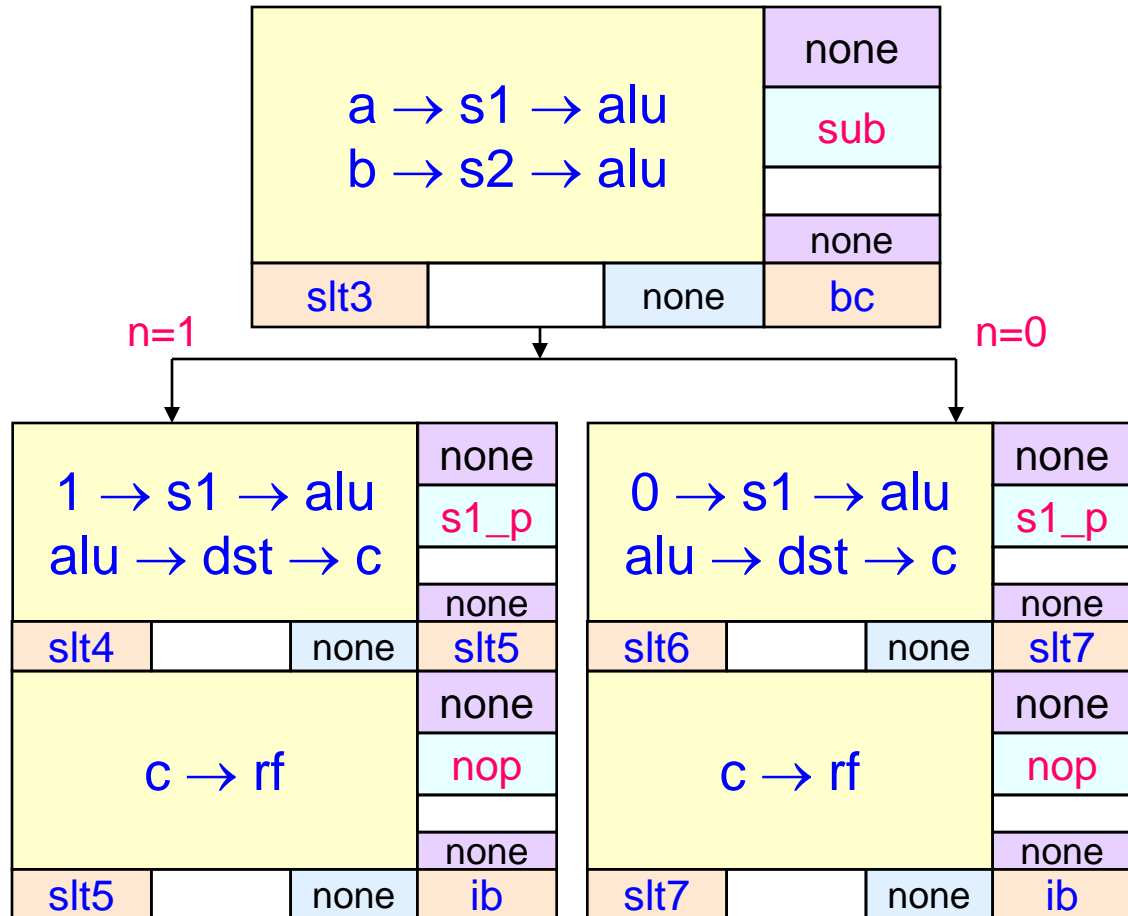
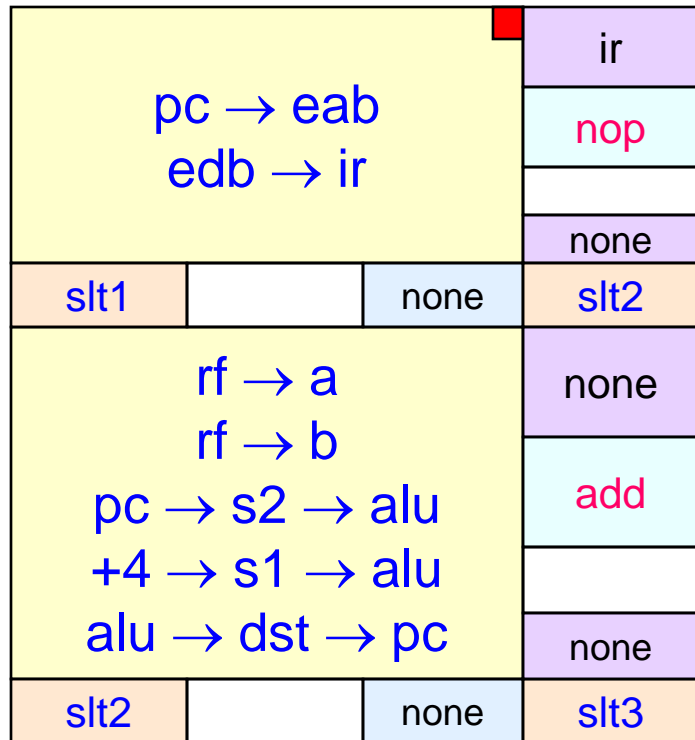
Level 2 Hardware Flowchart

SNE.I Rd, Rs1, Imm



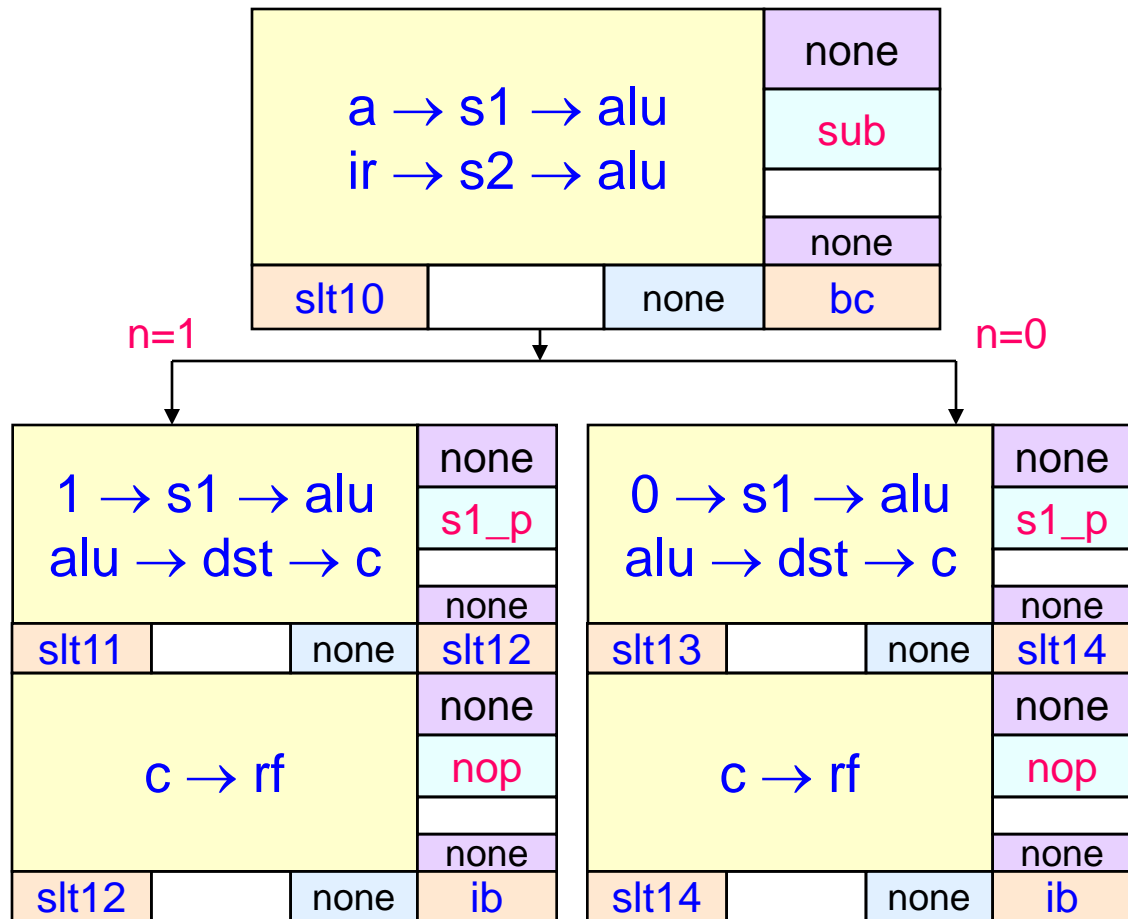
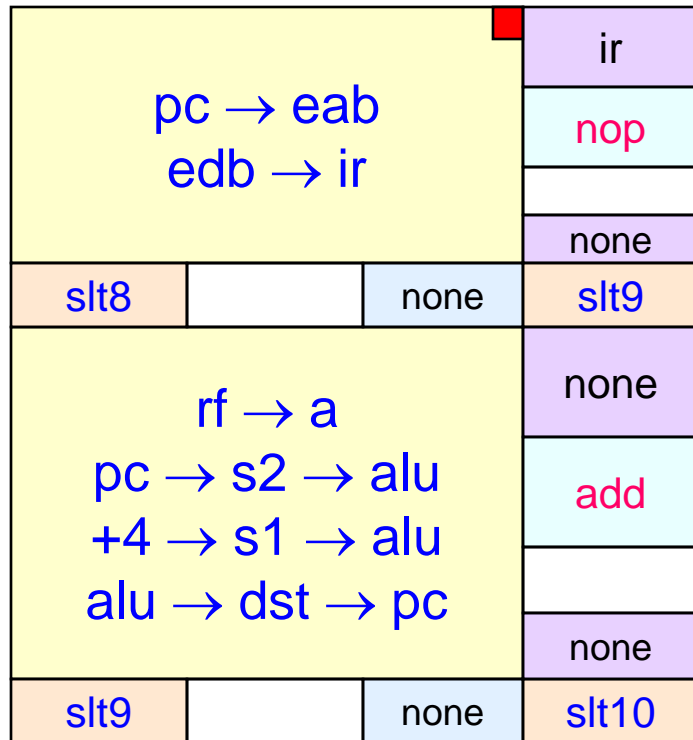
Level 2 Hardware Flowchart

SLT Rd, Rs1, Rs2



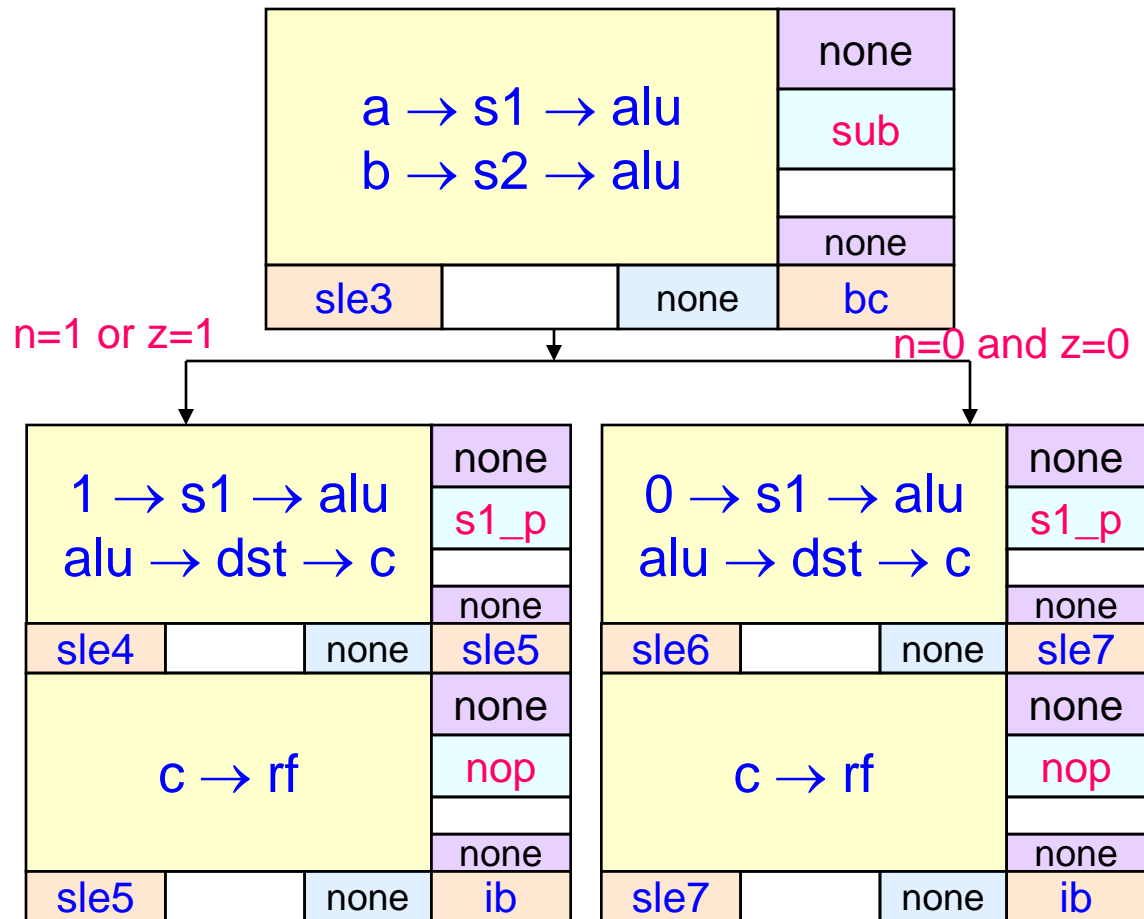
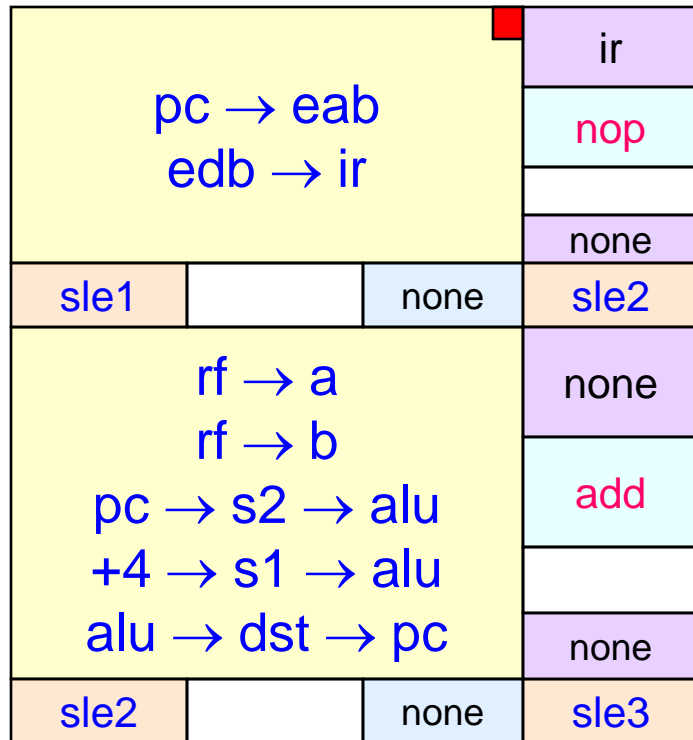
Level 2 Hardware Flowchart

SLT.I Rd, Rs1, Imm



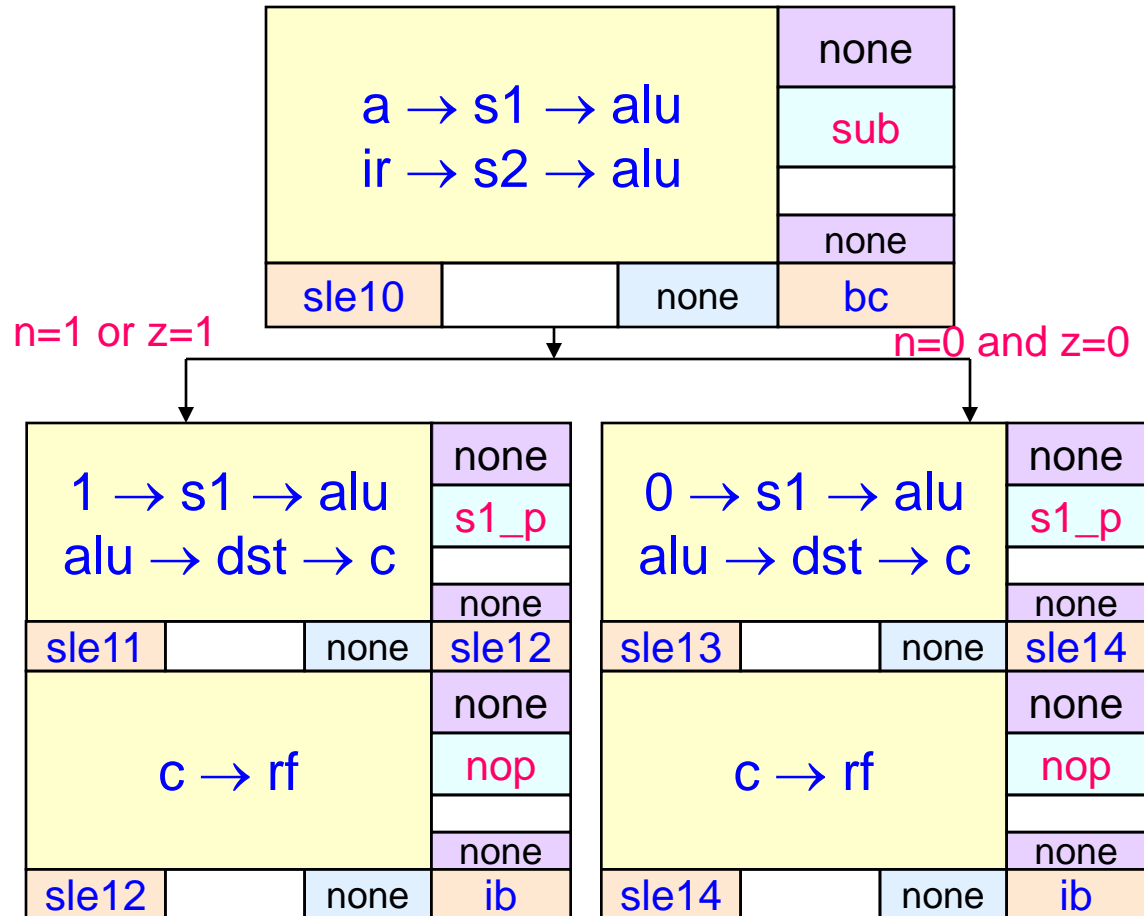
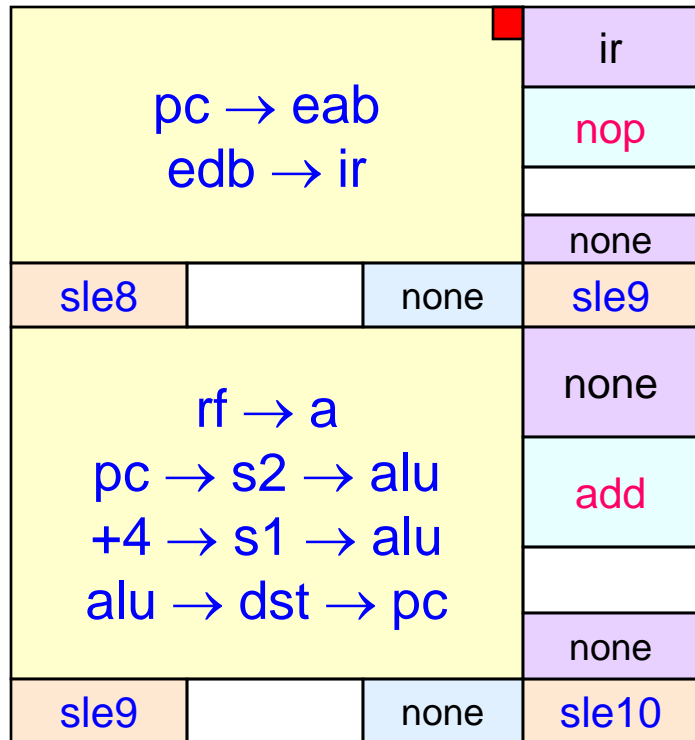
Level 2 Hardware Flowchart

SLE Rd, Rs1, Rs2



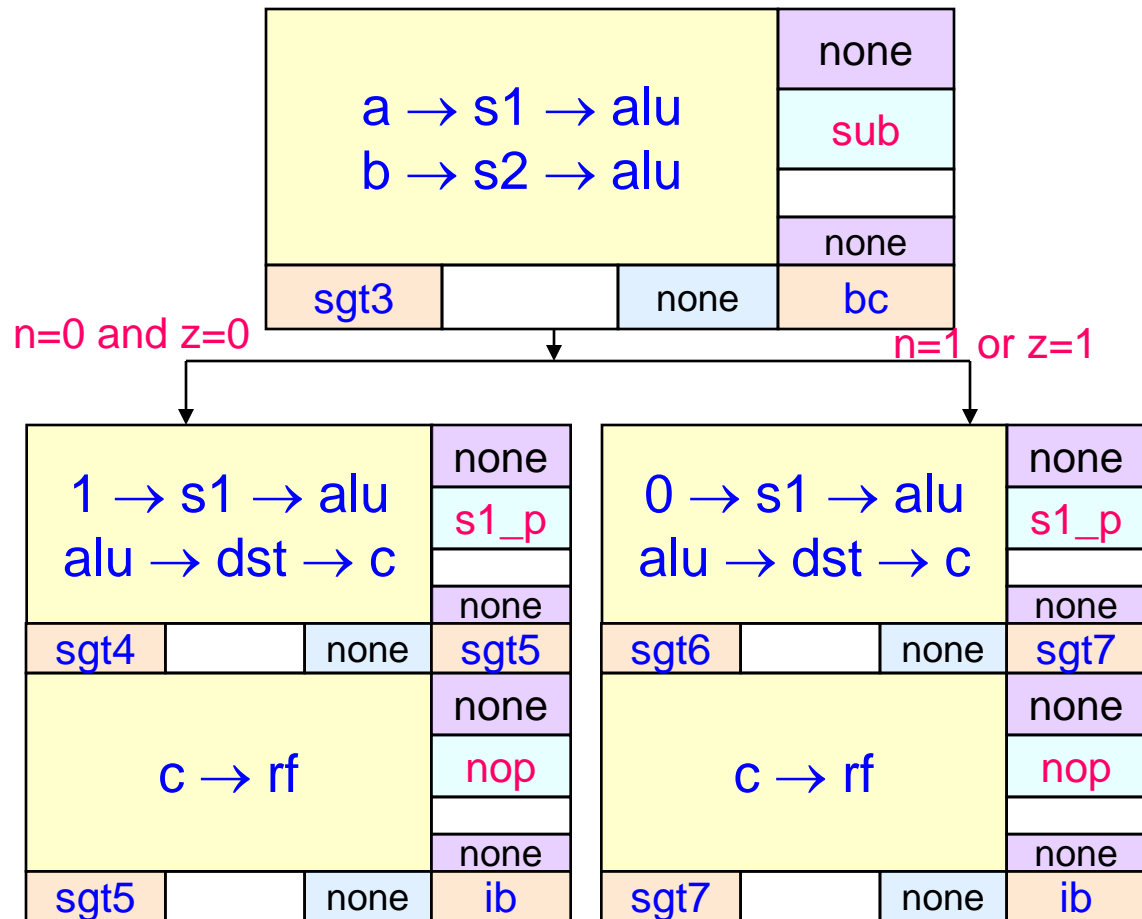
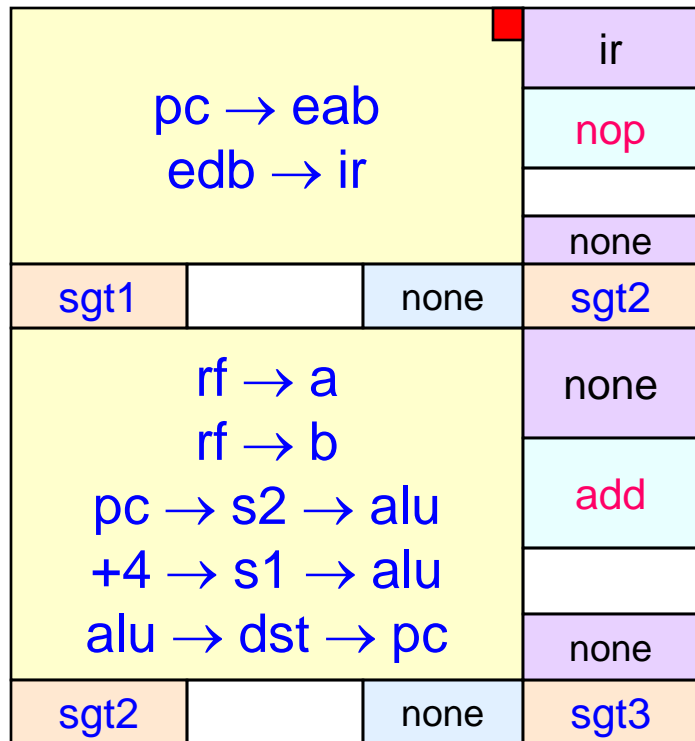
Level 2 Hardware Flowchart

SLE.I Rd, Rs1, Imm



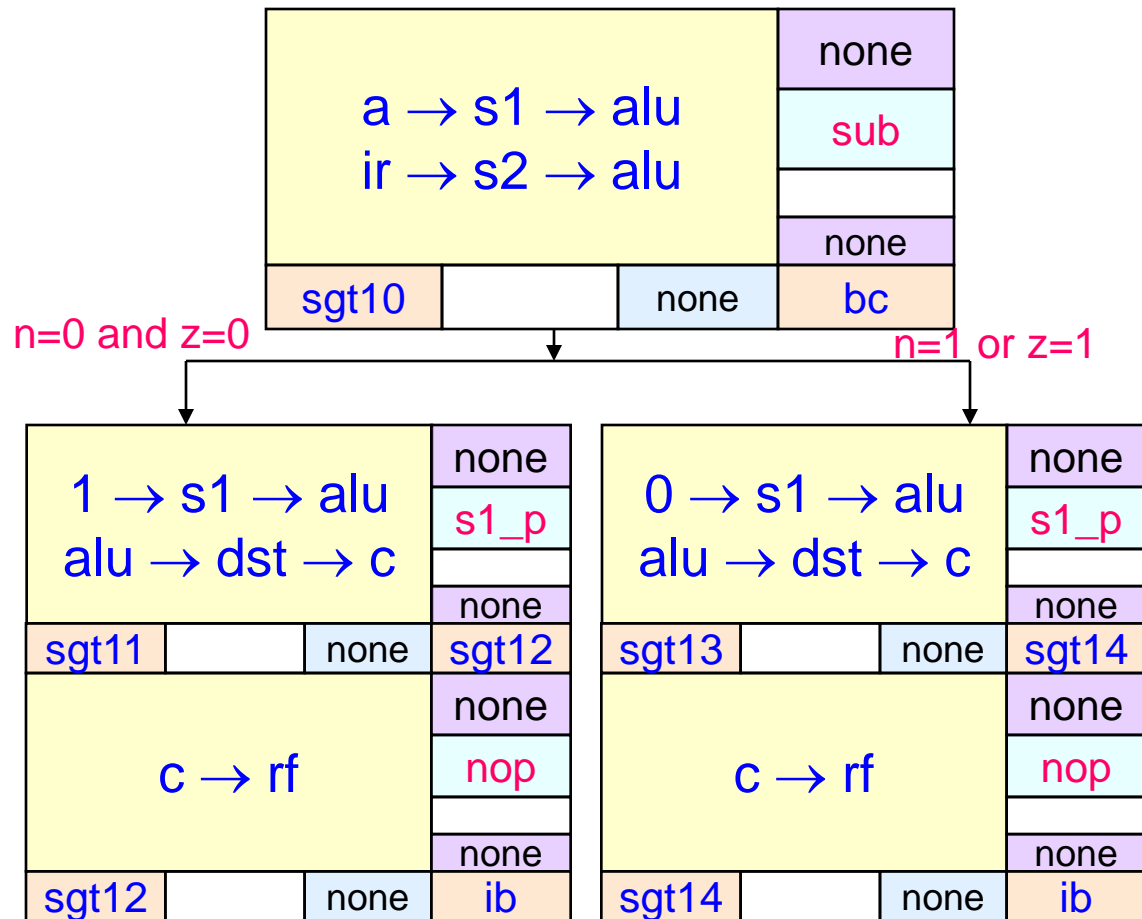
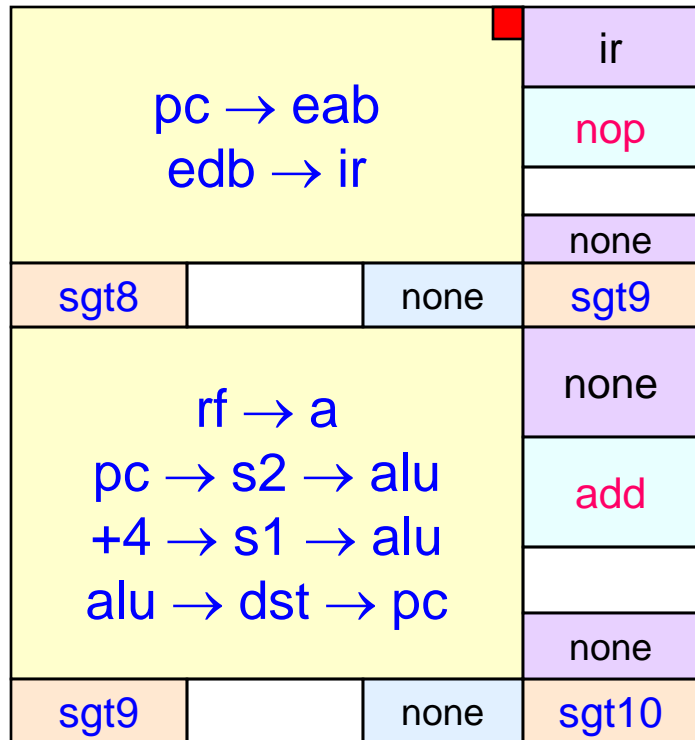
Level 2 Hardware Flowchart

SGT Rd, Rs1, Rs2



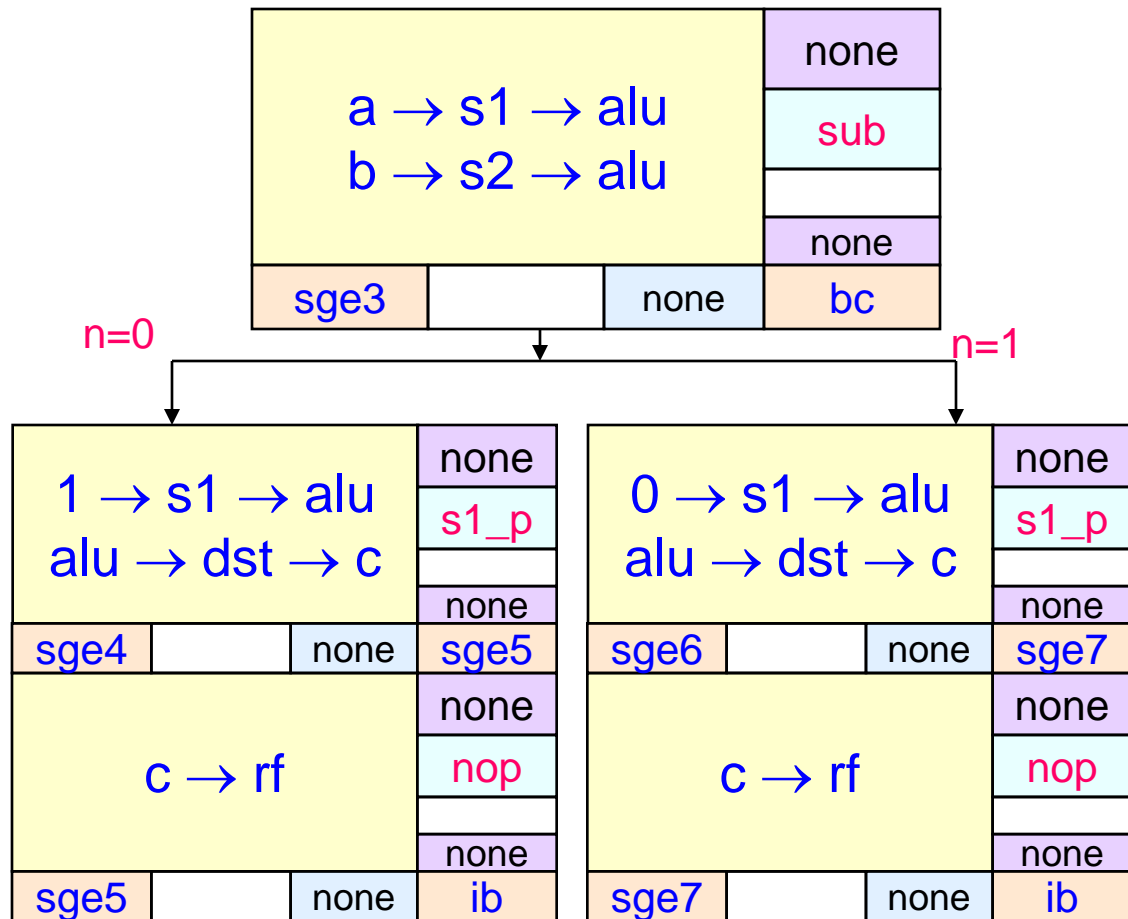
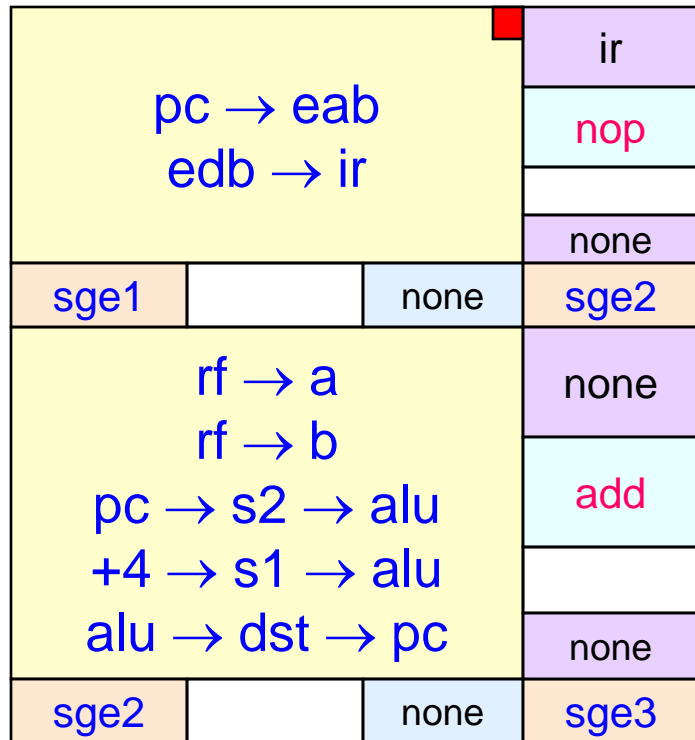
Level 2 Hardware Flowchart

SGT.I Rd, Rs1, Imm



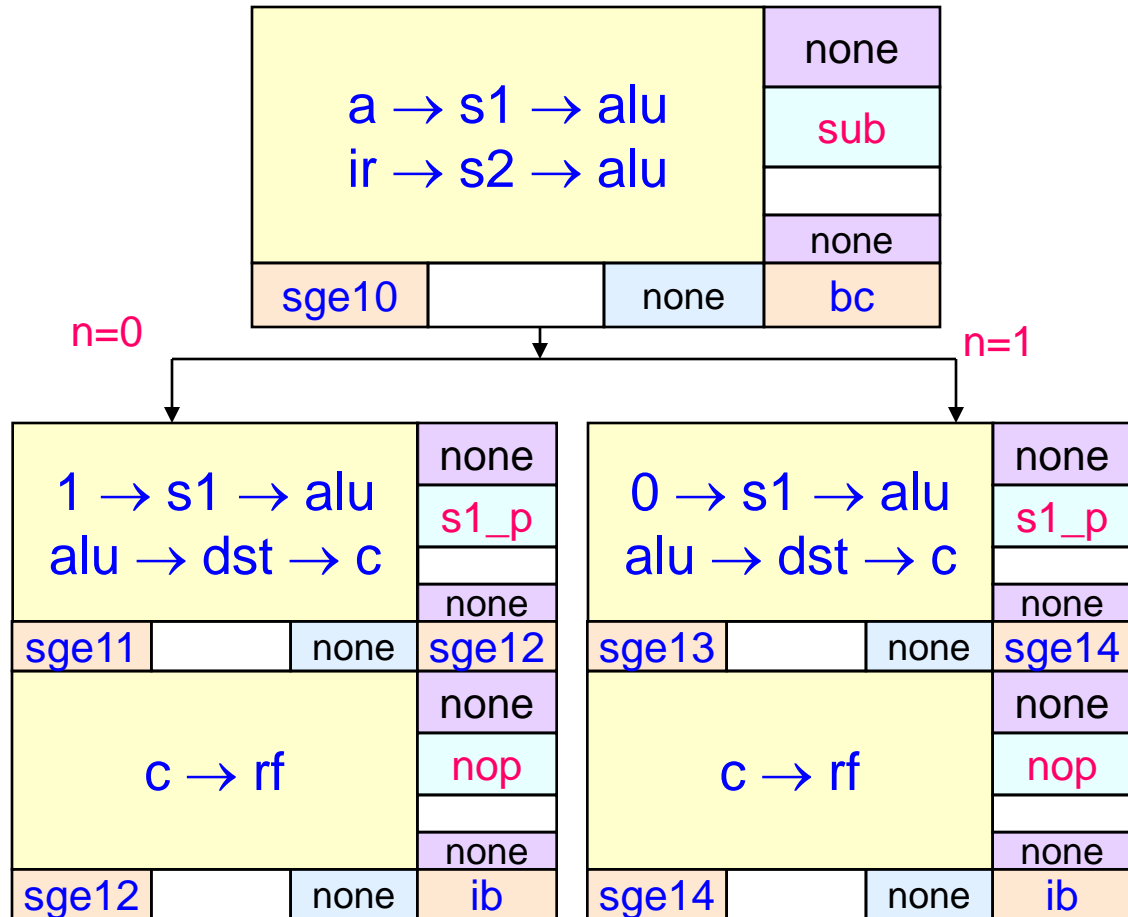
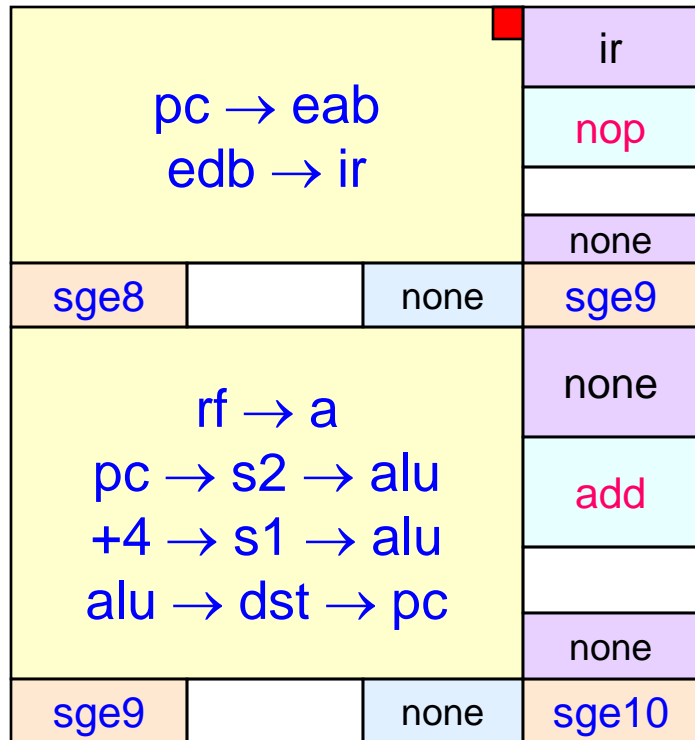
Level 2 Hardware Flowchart

SGE Rd, Rs1, Rs2



Level 2 Hardware Flowchart

SGE.I Rd, Rs1, Imm



Level 2 Hardware Flowchart

BEQZ Rs, Label

$pc \rightarrow eab$ $edb \rightarrow ir$			ir
			nop
			none
			none
beqz1		none	beqz2
$rf \rightarrow a$ $pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$			none
			add
			none
			none
beqz2		none	beqz3

$a \rightarrow s1 \rightarrow alu$ $0 \rightarrow s2 \rightarrow alu$			none
			sub
			none
			none
beqz3		none	bc

z=1

$ir \rightarrow s1 \rightarrow alu$ $pc \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow mar$			none
			add
			exc
			none
beqz4		s16	beqz5
$mar \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$			none
			S2_p
			none
			none
beqz5		none	ib



Level 2 Hardware Flowchart

BNEZ Rs, Label

$pc \rightarrow eab$ $edb \rightarrow ir$			ir
			nop
			none
			none
bnez1		none	bnez2
$rf \rightarrow a$ $pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$			none
			add
			none
			none
bnez2		none	bnez3

$a \rightarrow s1 \rightarrow alu$ $0 \rightarrow s2 \rightarrow alu$			none
			sub
			none
bnez3		none	bc

z=0

$ir \rightarrow s1 \rightarrow alu$ $pc \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow mar$			none
			add
			exc
			none
bnez4		s16	bnez5
$mar \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$			none
			S2_p
			none
			none
bnez5		none	ib



Level 2 Hardware Flowchart

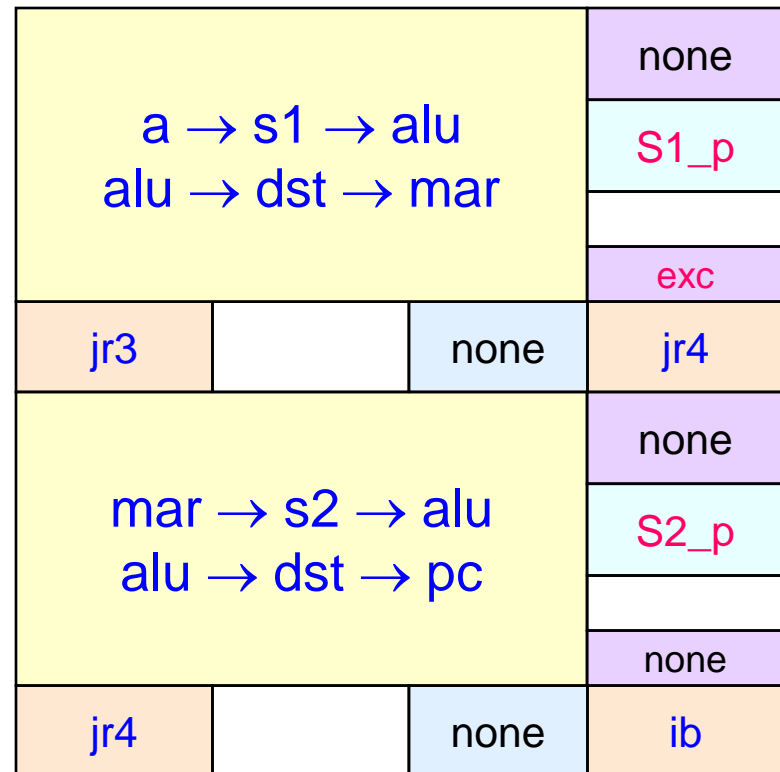
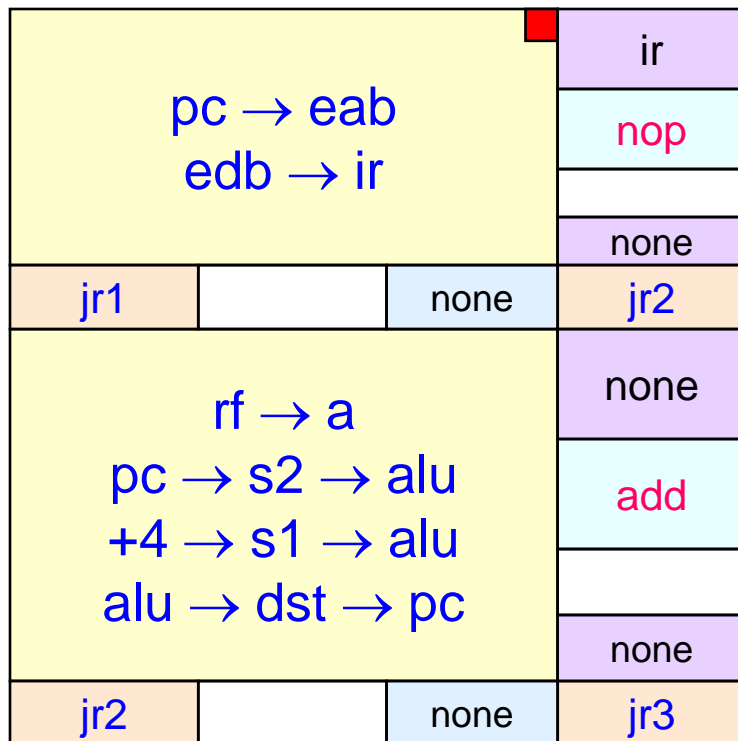
J Label

$pc \rightarrow eab$ $edb \rightarrow ir$			ir
			nop
			none
			none
j1		none	j2
$pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$			none
			add
			none
			none
j2		none	j3

$ir \rightarrow s1 \rightarrow alu$ $pc \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow mar$			none
			add
			exc
			exc
j3		s26	j4
$mar \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$			none
			S2_p
			none
			none
j4		none	ib

Level 2 Hardware Flowchart

JR Rs



Level 2 Hardware Flowchart

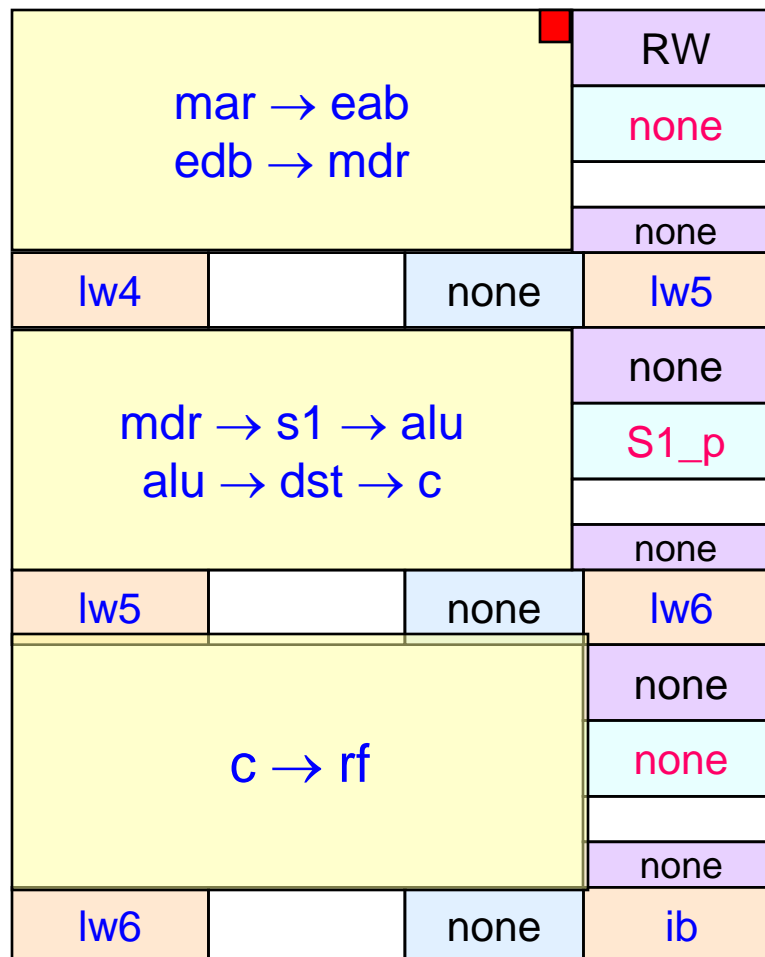
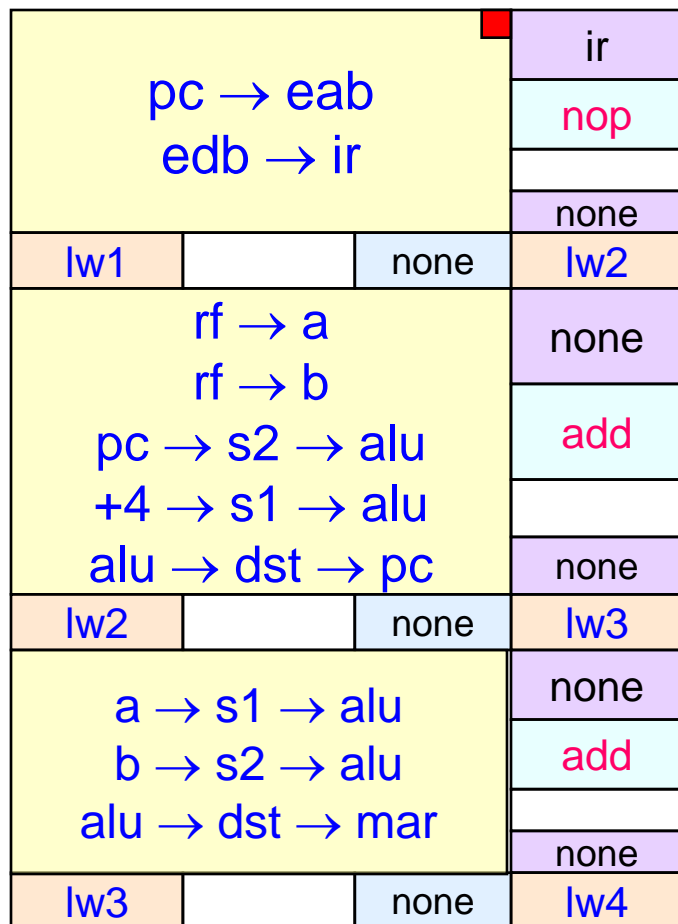
pc → eab edb → ir			ir
			nop
			none
jal1		none	jal2
pc → s2 → alu +4 → s1 → alu alu → dst → pc			none
			add
			none
jal2		none	jal3
pc → s2 → alu alu → dst → c			none
			S2_p
			none
jal3		none	jal4

JAL Label

ir → s1 → alu pc → s2 → alu alu → dst → mar c → rf			none
			add
			exc
jal4		s26	jal5
mar → s2 → alu alu → dst → pc			none
			S2_p
			none
jal5		none	ib

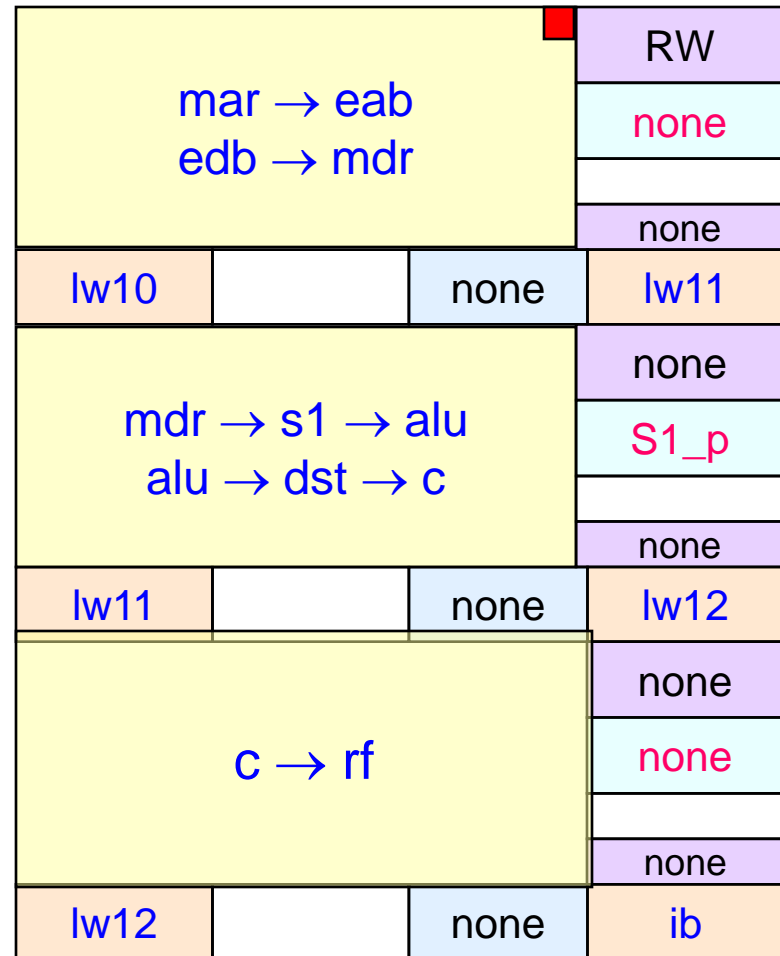
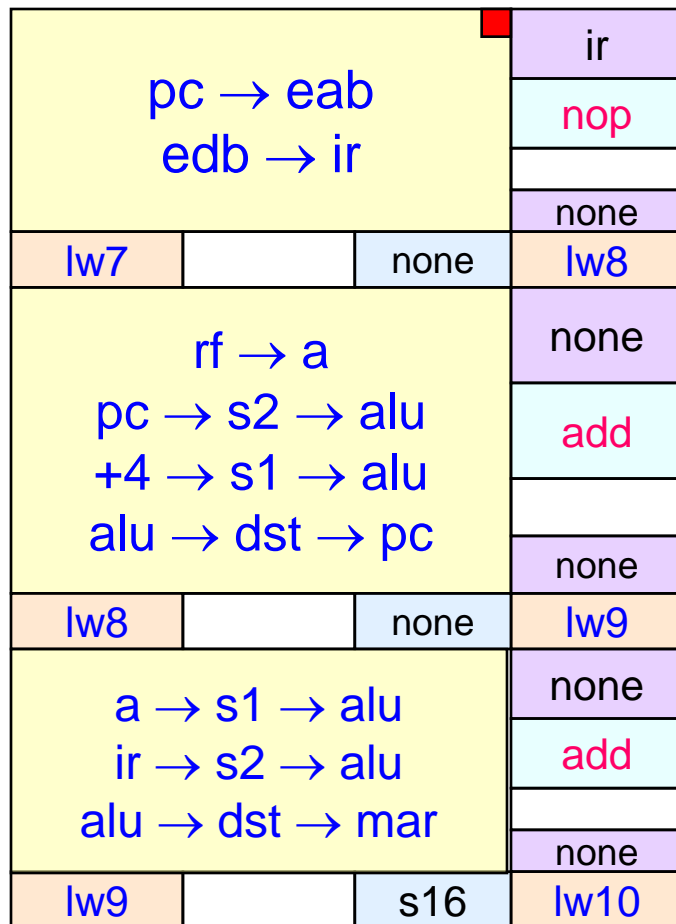
Level 2 Hardware Flowchart

LW Rd, Rs2(Rs1)



Level 2 Hardware Flowchart

LW.I Rd, Imm(Rs1)



Level 2 Hardware Flowchart

pc → eab edb → ir			ir
			nop
			none
sw1		none	sw2
rf → a rf → b pc → s2 → alu +4 → s1 → alu alu → dst → pc			none
			add
			none
			sw3
			none
a → s1 → alu b → s2 → alu alu → dst → mar rf → b			add
			none
			sw4
			none

SW Rd, Rs2(Rs1)

b → s2 → alu alu → dst → mdr			none
			S2_p
			none
sw4		none	sw5
mar → eab mdr → edb			WW
			none
			none
			ib
sw5		none	



Level 2 Hardware Flowchart

pc → eab edb → ir			ir
			nop
			none
sw6		none	sw7
rf → a pc → s2 → alu +4 → s1 → alu alu → dst → pc			none
			add
			none
			none
sw7		none	sw8
a → s1 → alu ir → s2 → alu alu → dst → mar rf → b			none
			add
			none
			none
sw8		s16	sw9

SW.I Rd, Imm(Rs1)

b → s2 → alu alu → dst → mdr			none
			S2_p
			none
sw9		none	sw10
mar → eab mdr → edb			WW
			none
			none
			none
sw10		none	ib



Level 2 Hardware Flowchart

LH Rd, Rs2(Rs1)

pc → eab edb → ir			ir
			nop
			none
lh1		none	lh2
rf → a rf → b pc → s2 → alu +4 → s1 → alu alu → dst → pc			none
			add
			none
			none
lh2		none	lh3
a → s1 → alu b → s2 → alu alu → dst → mar			none
			add
			none
lh3		none	lh4

mar → eab edb → mdr			RH
			none
			none
lh4		none	lh5
mdr → s1 → alu 16 → s2 → alu alu → dst → c			none
			sra
			none
lh5		none	lh6
c → rf			none
			none
			none
lh6		none	ib



Level 2 Hardware Flowchart

LH.I Rd, Imm(Rs1)

pc → eab edb → ir			ir
			nop
			none
lh7		none	lh8
rf → a pc → s2 → alu +4 → s1 → alu alu → dst → pc			none
			add
			none
			none
lh8		none	lh9
a → s1 → alu ir → s2 → alu alu → dst → mar			none
			add
			none
			none
lh9		s16	lh10

mar → eab edb → mdr			RW
			none
			none
lh10		none	lh11
mdr → s1 → alu 16 → s2 → alu alu → dst → c			none
			sra
			none
			none
lh11		none	lh12
c → rf			none
			none
			none
			none
lh12		none	ib



Level 2 Hardware Flowchart

pc → eab edb → ir			ir
			nop
			none
sh1		none	sh2
rf → a rf → b pc → s2 → alu +4 → s1 → alu alu → dst → pc			none
			add
			none
			none
sh2		none	sh3
a → s1 → alu b → s2 → alu alu → dst → mar rf → b			none
			add
			none
			none
sh3		none	sh4

SH Rd, Rs2(Rs1)

b → s2 → alu alu → dst → mdr			none
			S2_p
			none
sh4		none	sh5
mar → eab mdr → edb			WH
			none
			none
			none
sh5		none	ib

Level 2 Hardware Flowchart

pc → eab edb → ir			ir
			nop
			none
sh6		none	sh7
rf → a pc → s2 → alu +4 → s1 → alu alu → dst → pc			none
			add
			none
			none
sh7		none	sh8
a → s1 → alu ir → s2 → alu alu → dst → mar rf → b			none
			add
			none
			none
sh8		s16	sh9

SH.I Rd, Imm(Rs1)

b → s2 → alu alu → dst → mdr			none
			S2_p
			none
sh9		none	sh10
mar → eab mdr → edb			WH
			none
			none
			none
sh10		none	ib

Level 2 Hardware Flowchart

LB Rd, Rs2(Rs1)

pc → eab edb → ir			ir
			nop
			none
			none
lb1		none	lb2
rf → a rf → b pc → s2 → alu +4 → s1 → alu alu → dst → pc			none
			add
			none
			none
			none
lb2		none	lb3
a → s1 → alu b → s2 → alu alu → dst → mar			none
			add
			none
			none
lb3		none	lb4

mar → eab edb → mdr			RB
			none
			none
			none
lb4		none	lb5
mdr → s1 → alu 24 → s2 → alu alu → dst → c			none
			sra
			none
			none
lb5		none	lb6
c → rf			none
			none
			none
			none
lb6		none	ib



Level 2 Hardware Flowchart

LB.I Rd, Imm(Rs1)

pc → eab edb → ir			ir
			nop
			none
			none
lb7		none	lb8
rf → a pc → s2 → alu +4 → s1 → alu alu → dst → pc			none
			add
			none
			none
lb8		none	lb9
a → s1 → alu ir → s2 → alu alu → dst → mar			none
			add
			none
			none
lb9		s16	lb10

mar → eab edb → mdr			RB
			none
			none
			none
lb10		none	lb11
mdr → s1 → alu 24 → s2 → alu alu → dst → c			none
			sra
			none
			none
lb11		none	lb12
c → rf			none
			none
			none
			none
lb12		none	ib

Level 2 Hardware Flowchart

pc → eab edb → ir			ir
			nop
			none
sb1		none	sb2
rf → a rf → b pc → s2 → alu +4 → s1 → alu alu → dst → pc			none
			add
			none
			none
sb2		none	sb3
a → s1 → alu b → s2 → alu alu → dst → mar rf → b			none
			add
			none
			none
sb3		none	sb4

SB Rd, Rs2(Rs1)

b → s2 → alu alu → dst → mdr			none
			S2_p
			none
sb4		none	sb5
mar → eab mdr → edb			WB
			none
			none
			none
sb5		none	ib



Level 2 Hardware Flowchart

pc → eab edb → ir			ir
			nop
			none
sb6		none	sb7
rf → a pc → s2 → alu +4 → s1 → alu alu → dst → pc			none
			add
			none
			none
sb7		none	sb8
a → s1 → alu ir → s2 → alu alu → dst → mar rf → b			none
			add
			none
			none
sb8		s16	sb9

SB.I Rd, Imm(Rs1)

b → s2 → alu alu → dst → mdr			none
			S2_p
			none
sb9		none	sb10
mar → eab mdr → edb			WB
			none
			none
			none
sb10		none	ib

Modified Level 2 Hardware Flowchart

$rf \rightarrow a$ $pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$			none
			add
			none
			none
add6		none	add7

or

$pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$			none
			add
			none
			none
lhi2		none	lhi3

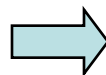


$rf \rightarrow a$ $rf \rightarrow b$ $pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$			none
			add
			none
			none
add2		none	add3

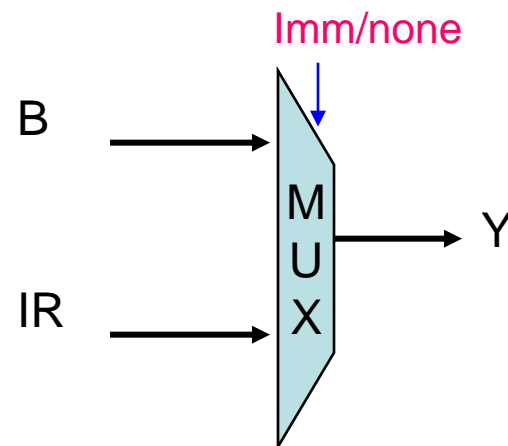
Modified Level 2 Hardware Flowchart

$a \rightarrow s1 \rightarrow \text{alu}$ $b \rightarrow s2 \rightarrow \text{alu}$ $\text{alu} \rightarrow \text{dst} \rightarrow c$			none
			add
			none
add3		none	add4

$a \rightarrow s1 \rightarrow \text{alu}$ $\text{ir} \rightarrow s2 \rightarrow \text{alu}$ $\text{alu} \rightarrow \text{dst} \rightarrow c$			none
			add
			none
add7		s16	add8



$a \rightarrow s1 \rightarrow \text{alu}$ $y \rightarrow s2 \rightarrow \text{alu}$ $\text{alu} \rightarrow \text{dst} \rightarrow c$			none
			add
			none
add7		s16/none	add8



Modified Level 2 Hardware Flowchart

ADD Rd, Rs1, Rs2

$pc \rightarrow eab$ $edb \rightarrow ir$			ir
			nop
			none
			none
add1		none	add2
$rf \rightarrow a$ $rf \rightarrow b$ $pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$			none
			add
			none
			none
add2		none	add3

$a \rightarrow s1 \rightarrow alu$ $y \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow c$			none
			add
			none
add3		none	add4
$c \rightarrow rf$			none
			nop
			none
add4		none	ib

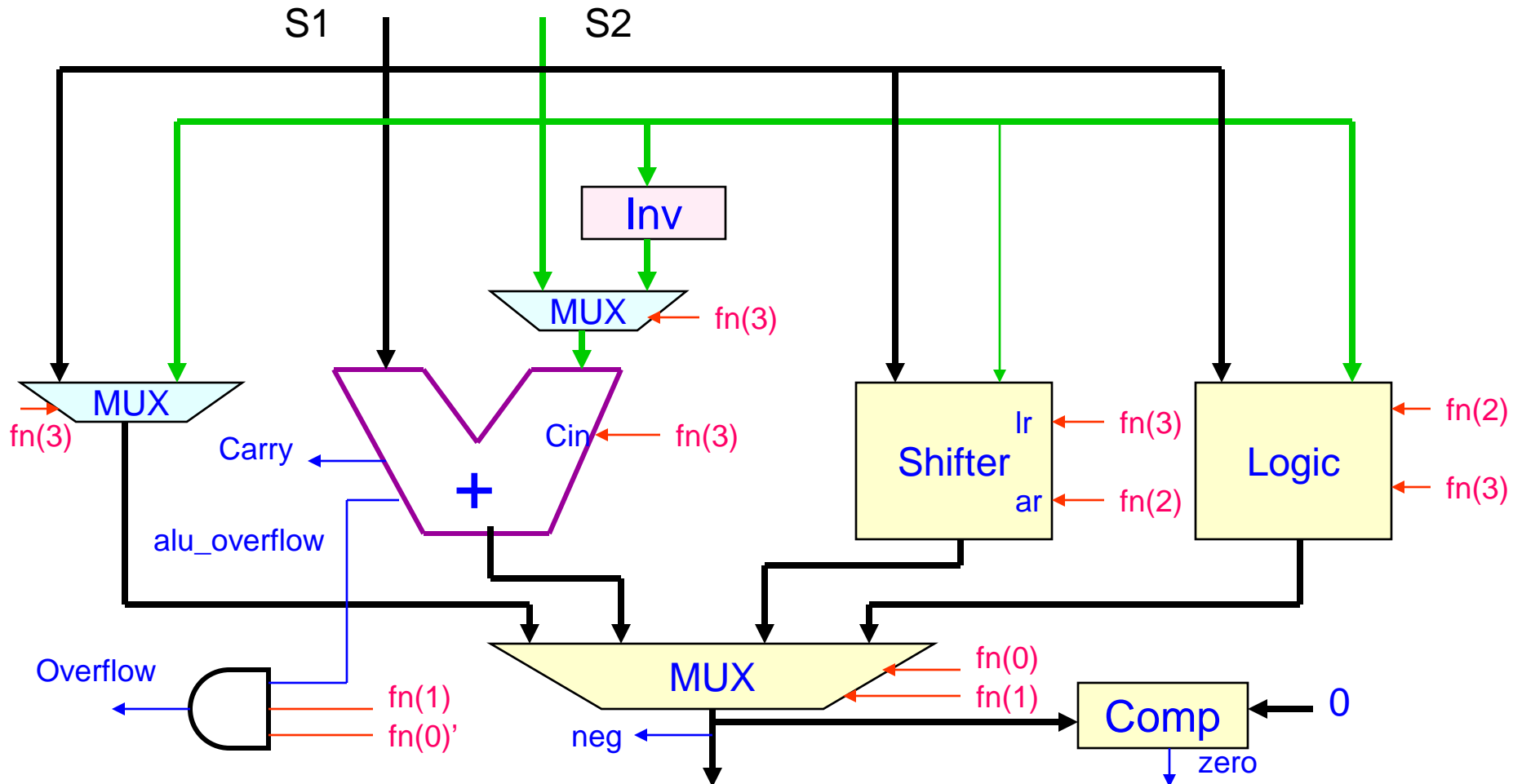
Modified Level 2 Hardware Flowchart

ADD.I Rd, Rs1, Imm

$pc \rightarrow eab$ $edb \rightarrow ir$			ir
			nop
			none
add5		none	add6
$rf \rightarrow a$ $rf \rightarrow b$ $pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$			none
			add
			none
			none
add6		none	add7

$a \rightarrow s1 \rightarrow alu$ $y \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow c$			none
			add
			none
add7		s16	add8
$c \rightarrow rf$			none
			nop
			none
add8		none	ib

Arithmetic Logic Unit



Level 2 Hardware Flowchart: Merging

➤ Merge identical states

add1 = add5 = sub1 = sub5 = and1 = and 5 = or1 = Or5 = xor1
= xor5 = sll1 = sll5 = srl 1 = srl5 = sra1 = sra5 = lhi1 = nop1
= seq1 = seq8 = sne1 = sne8 = sgt1 = sge8 = slt1 = slt 8 =
sle1 = sle8 = beqz1 = bnez1 = j1 = jr1 = jal1= jalr1 = lw1 =
lw7 = sw1 = sw6 = lh1 = lh7 = sh1 = sh6 = lb1 = lb7 = sb1 =
sb6 \equiv fetch



Level 2 Hardware Flowchart: Merging

add2 = add6 = sub2 = sub6 = and2 = and6 = or2 = or6 =
xor2 = xor6 = sll2 = sll6 = srl2 = srl6 = sra2 = sra6 =
lhi2 = nop2 = seq2 = seq9 = sne2 = sne9 = sgt2 =
sge9 = slt2 = slt 9 = sle2 = sle9 = beqz2 = bnez9 = j2
= jr2 = jal2 = jalr2 = lw2 = lw8 = sw2 = sw7 = lh2 =
lh8 = sh2 = sh7 = lb2 = lb8 = sb2 = sb7 Ξ dec



Merged Level 2 Hardware Flowchart

ADD Rd, Rs1, Rs2

$pc \rightarrow eab$ $edb \rightarrow ir$			ir
			nop
			none
			none
fetch		none	dec
$rf \rightarrow a$ $rf \rightarrow b$ $pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$			none
			add
			none
			none
			none
dec		none	add_1

$a \rightarrow s1 \rightarrow alu$ $y \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow c$			none
			add
			none
			none
add_1		none	W_B
$c \rightarrow rf$			none
			nop
			none
			none
W_B		none	fetch

Merged Level 2 Hardware Flowchart

ADD.I Rd, Rs1, Imm

$pc \rightarrow eab$ $edb \rightarrow ir$			ir
			nop
			none
			none
fetch		none	dec
$rf \rightarrow a$ $rf \rightarrow b$ $pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$			none
			add
			none
			none
dec		none	add_1

$a \rightarrow s1 \rightarrow alu$ $y \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow c$			none
			add
			none
add_1		s16	W_B
$c \rightarrow rf$			none
			nop
			none
W_B		none	fetch

Merged Level 2 Hardware Flowchart

SUB Rd, Rs1, Rs2

$pc \rightarrow eab$ $edb \rightarrow ir$			ir
			nop
			none
			none
fetch		none	dec
$rf \rightarrow a$ $rf \rightarrow b$ $pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$			none
			add
			none
			none
			none
dec		none	sub_1

$a \rightarrow s1 \rightarrow alu$ $y \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow c$			none
			sub
			none
			none
sub_1		none	W_B
$c \rightarrow rf$			none
			nop
			none
			none
W_B		none	fetch

Merged Level 2 Hardware Flowchart

SUB.I Rd, Rs1, Imm

$pc \rightarrow eab$ $edb \rightarrow ir$			ir
			nop
			none
			none
fetch		none	dec
$rf \rightarrow a$ $rf \rightarrow b$ $pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$			none
			add
			none
			none
dec		none	sub_1

$a \rightarrow s1 \rightarrow alu$ $y \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow c$			none
			sub
			none
			none
sub_1		s16	W_B
$c \rightarrow rf$			none
			nop
			none
			none
W_B		none	fetch

Merged Level 2 Hardware Flowchart

AND Rd, Rs1, Rs2

$pc \rightarrow eab$ $edb \rightarrow ir$			ir
			nop
			none
			none
fetch		none	dec
$rf \rightarrow a$ $rf \rightarrow b$ $pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$			none
			add
			none
			none
dec		none	and_1

$a \rightarrow s1 \rightarrow alu$ $y \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow c$			none
			and
			none
			none
and_1		none	W_B
$c \rightarrow rf$			none
			nop
			none
			none
W_B		none	fetch

Merged Level 2 Hardware Flowchart

AND.I Rd, Rs1, Imm

$pc \rightarrow eab$ $edb \rightarrow ir$			ir
			nop
			none
			none
fetch		none	dec
$rf \rightarrow a$ $rf \rightarrow b$ $pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$			none
			add
			none
			none
			none
dec		none	and_1

$a \rightarrow s1 \rightarrow alu$ $y \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow c$			none
			and
			none
			none
and_1		s16	W_B
$c \rightarrow rf$			none
			nop
			none
			none
W_B		none	fetch

Merged Level 2 Hardware Flowchart

OR Rd, Rs1, Rs2

$pc \rightarrow eab$ $edb \rightarrow ir$			ir
			nop
			none
			none
fetch		none	dec
$rf \rightarrow a$ $rf \rightarrow b$ $pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$			none
			add
			none
			none
dec		none	or_1

$a \rightarrow s1 \rightarrow alu$ $y \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow c$			none
			or
			none
			none
or_1		none	W_B
$c \rightarrow rf$			none
			nop
			none
			none
W_B		none	fetch

Merged Level 2 Hardware Flowchart

OR.I Rd, Rs1, Imm

$pc \rightarrow eab$ $edb \rightarrow ir$			ir
			nop
			none
			none
fetch		none	dec
$rf \rightarrow a$ $rf \rightarrow b$ $pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$			none
			add
			none
			none
dec		none	or_1

$a \rightarrow s1 \rightarrow alu$ $y \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow c$			none
			or
			none
			none
or_1		s16	W_B
$c \rightarrow rf$			none
			nop
			none
			none
W_B		none	fetch

Merged Level 2 Hardware Flowchart

XOR Rd, Rs1, Rs2

$pc \rightarrow eab$ $edb \rightarrow ir$			ir
			nop
			none
			none
fetch		none	dec
$rf \rightarrow a$ $rf \rightarrow b$ $pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$			none
			add
			none
			none
dec		none	xor_1

$a \rightarrow s1 \rightarrow alu$ $y \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow c$			none
			xor
			none
			none
xor_1		none	W_B
$c \rightarrow rf$			none
			nop
			none
			none
W_B		none	fetch

Merged Level 2 Hardware Flowchart

XOR.I Rd, Rs1, Imm

$pc \rightarrow eab$ $edb \rightarrow ir$			ir
			nop
			none
			none
fetch		none	dec
$rf \rightarrow a$ $rf \rightarrow b$ $pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$			none
			add
			none
			none
dec		none	xor_1

$a \rightarrow s1 \rightarrow alu$ $y \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow c$			none
			xor
			none
			none
xor_1		s16	W_B
$c \rightarrow rf$			none
			nop
			none
			none
W_B		none	fetch

Merged Level 2 Hardware Flowchart

SLL Rd, Rs1, Rs2

$pc \rightarrow eab$ $edb \rightarrow ir$			ir
			nop
			none
			none
fetch		none	dec
$rf \rightarrow a$ $rf \rightarrow b$ $pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$			none
			add
			none
			none
dec		none	Sll_1

$a \rightarrow s1 \rightarrow alu$ $y \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow c$			none
			sll
			none
			none
Sll_1		none	W_B
$c \rightarrow rf$			none
			nop
			none
			none
W_B		none	fetch



Merged Level 2 Hardware Flowchart

SLL.I Rd, Rs1, Imm

$pc \rightarrow eab$ $edb \rightarrow ir$			ir
			nop
			none
			none
fetch		none	dec
$rf \rightarrow a$ $rf \rightarrow b$ $pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$			none
			add
			none
			none
dec		none	Sll_1

$a \rightarrow s1 \rightarrow alu$ $y \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow c$			none
			sll
			none
			none
Sll_1		s16	W_B
$c \rightarrow rf$			none
			nop
			none
			none
W_B		none	fetch

Merged Level 2 Hardware Flowchart

SRL Rd, Rs1, Rs2

$pc \rightarrow eab$ $edb \rightarrow ir$			ir
			nop
			none
			none
fetch		none	dec
$rf \rightarrow a$ $rf \rightarrow b$ $pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$			none
			add
			none
			none
dec		none	srl_1

$a \rightarrow s1 \rightarrow alu$ $y \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow c$			none
			srl
			none
			none
srl_1		none	W_B
$c \rightarrow rf$			none
			nop
			none
			none
W_B		none	fetch

Merged Level 2 Hardware Flowchart

SRL.I Rd, Rs1, Imm

$pc \rightarrow eab$ $edb \rightarrow ir$			ir
			nop
			none
			none
fetch		none	dec
$rf \rightarrow a$ $rf \rightarrow b$ $pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$			none
			add
			none
			none
dec		none	srl_1

$a \rightarrow s1 \rightarrow alu$ $y \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow c$			none
			srl
			none
			none
srl_1		s16	W_B
$c \rightarrow rf$			none
			nop
			none
			none
W_B		none	fetch

Merged Level 2 Hardware Flowchart

SRA Rd, Rs1, Rs2

$pc \rightarrow eab$ $edb \rightarrow ir$			ir
			nop
			none
			none
fetch		none	dec
$rf \rightarrow a$ $rf \rightarrow b$ $pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$			none
			add
			none
			none
dec		none	sra_1

$a \rightarrow s1 \rightarrow alu$ $y \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow c$			none
			sra
			none
			none
sra_1		none	W_B
$c \rightarrow rf$			none
			nop
			none
			none
W_B		none	fetch



Merged Level 2 Hardware Flowchart

SRA.I Rd, Rs1, Imm

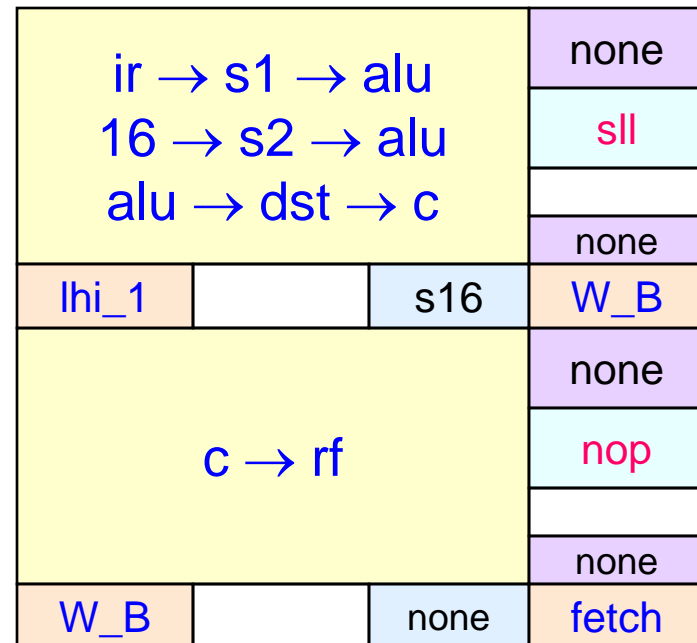
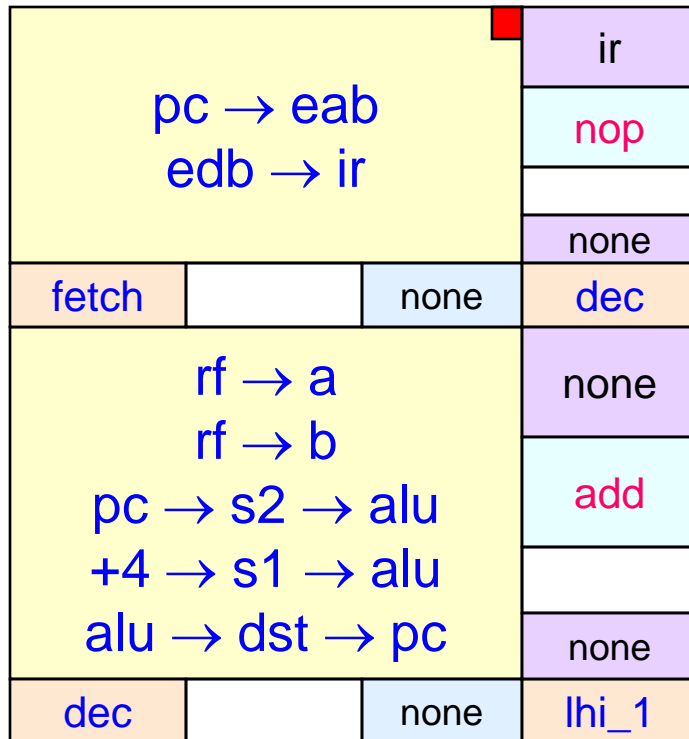
$pc \rightarrow eab$ $edb \rightarrow ir$			ir
			nop
			none
			none
fetch		none	dec
$rf \rightarrow a$ $rf \rightarrow b$ $pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$			none
			add
			none
			none
dec		none	sra_1

$a \rightarrow s1 \rightarrow alu$ $y \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow c$			none
			sra
			none
			none
sra_1		s16	W_B
$c \rightarrow rf$			none
			nop
			none
			none
W_B		none	fetch

Merged Level 2 Hardware Flowchart

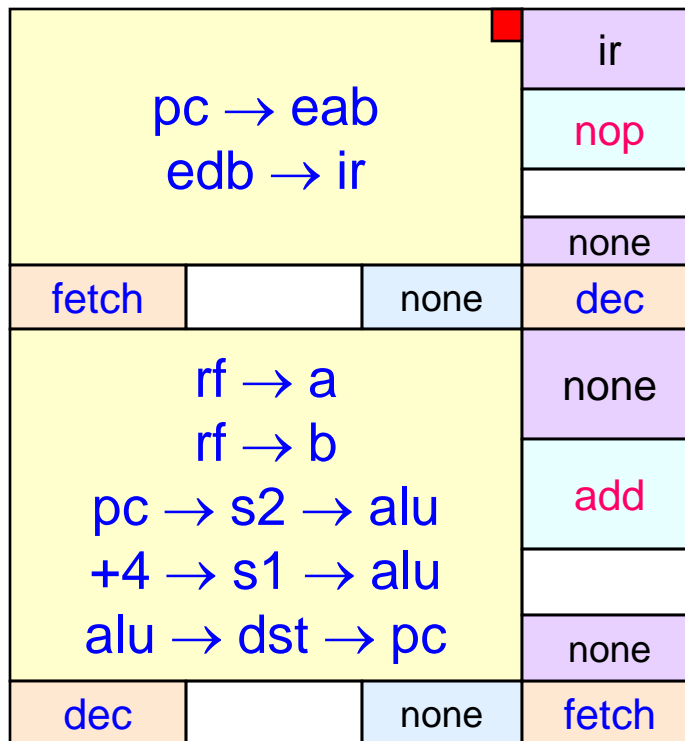
LHI Rd, Imm

Rd(0:15) <- Imm; Rd(16:31) <- hex0000



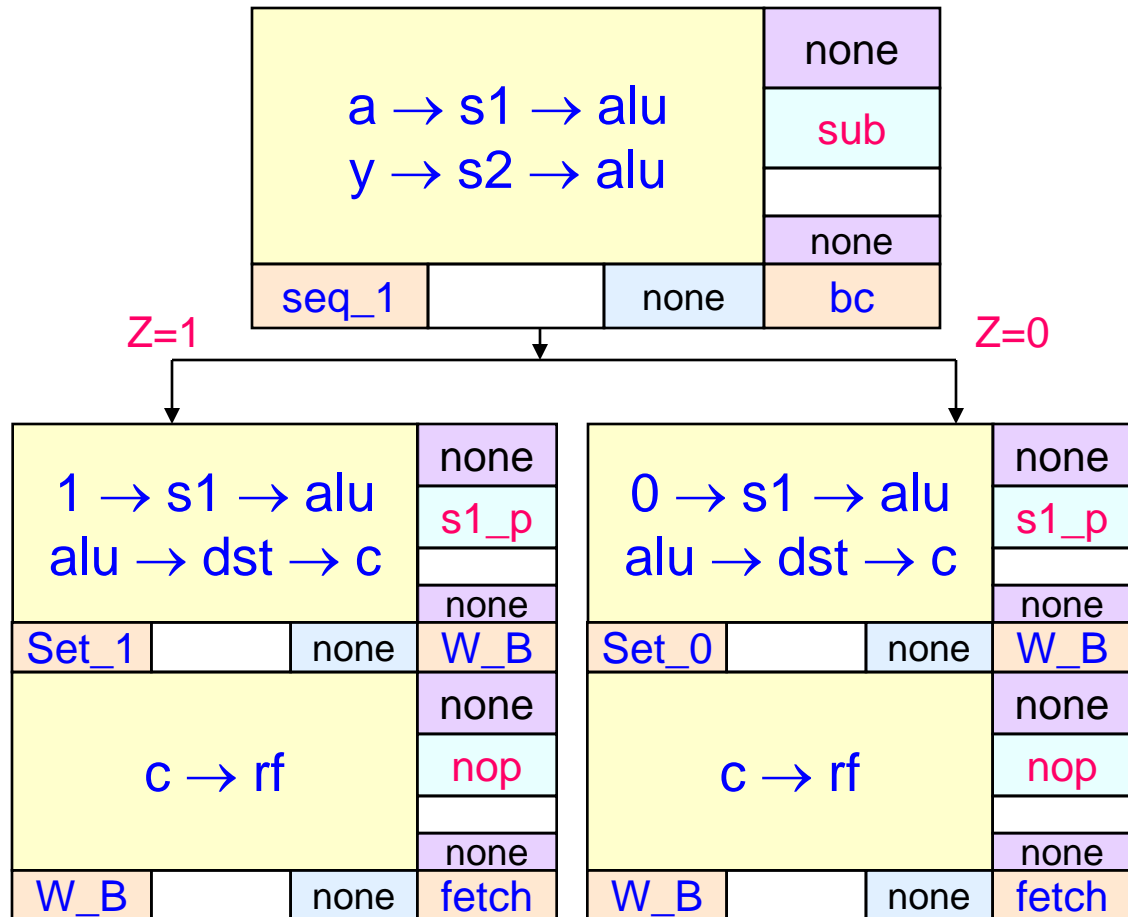
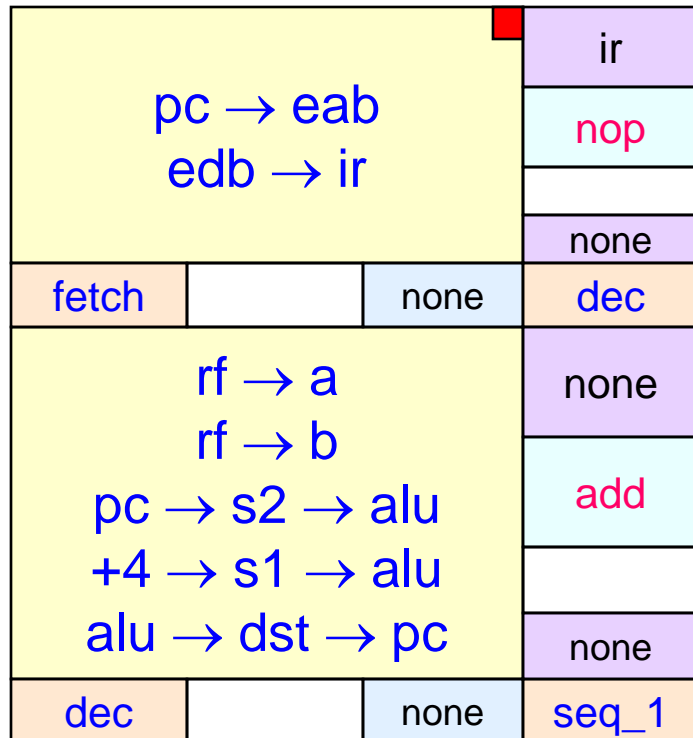
Merged Level 2 Hardware Flowchart

NOP



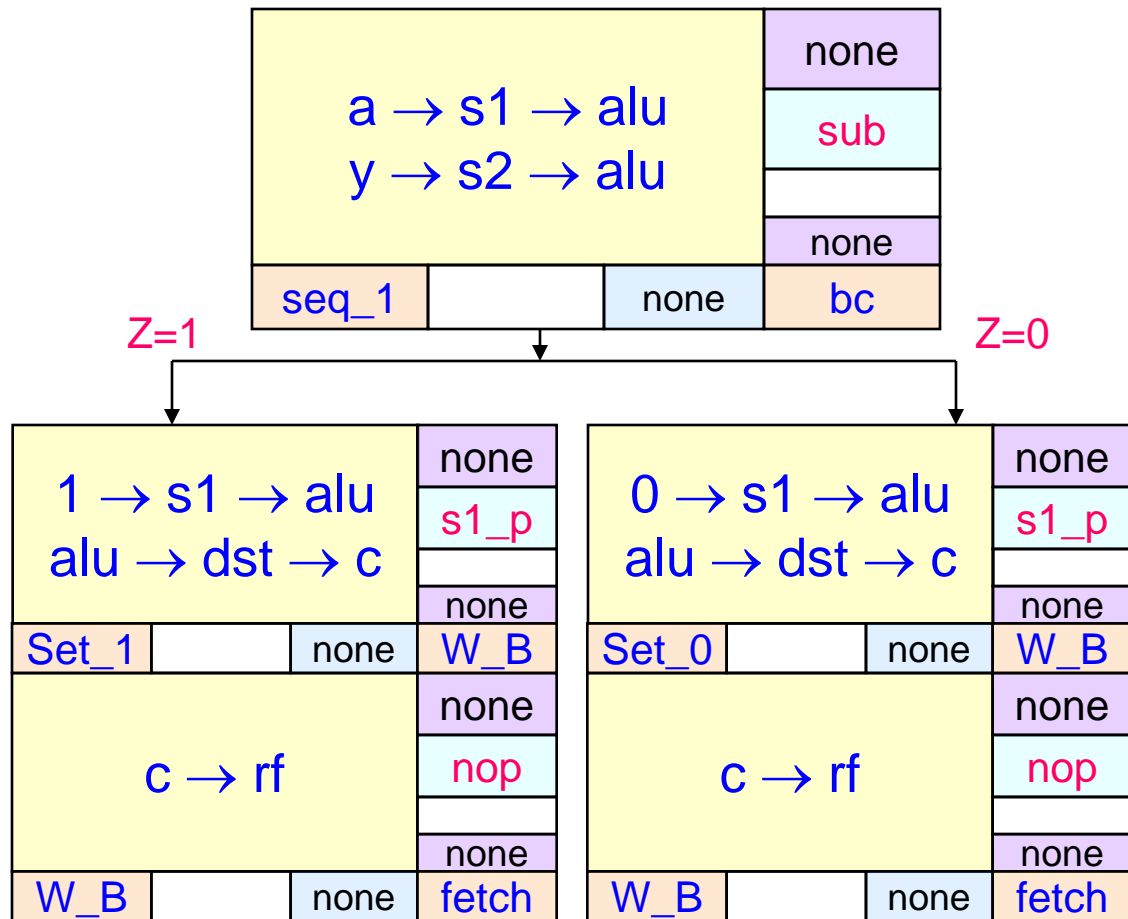
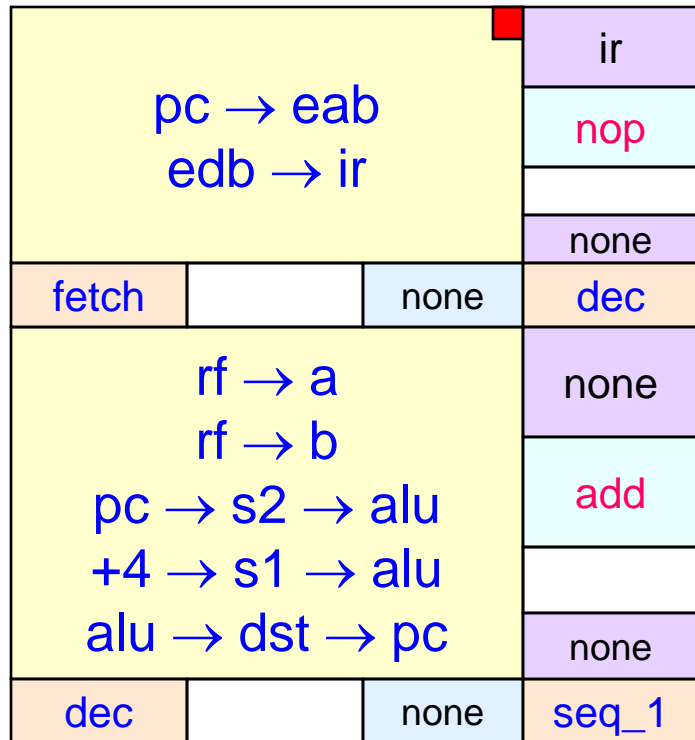
Merged Level 2 Hardware Flowchart

SEQ Rd, Rs1, Rs2



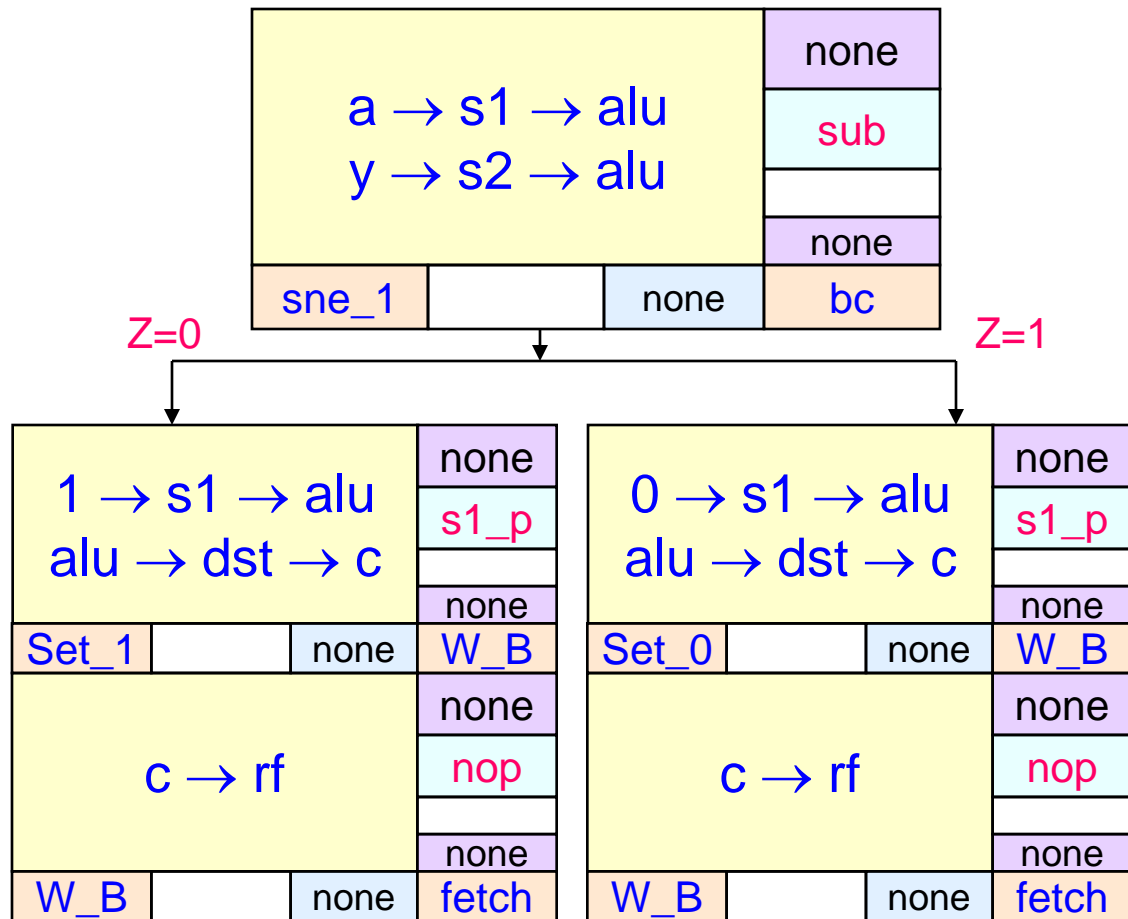
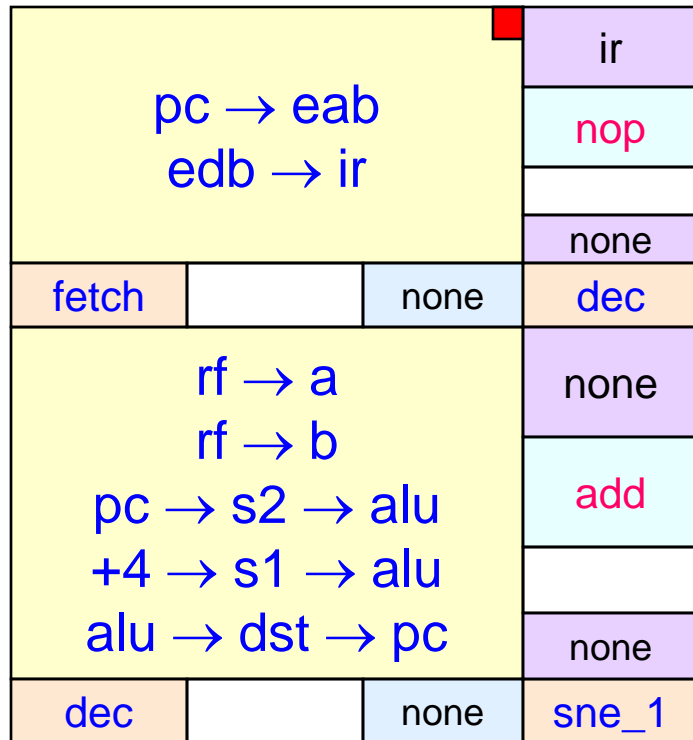
Merged Level 2 Hardware Flowchart

SEQ.I Rd, Rs1, Imm



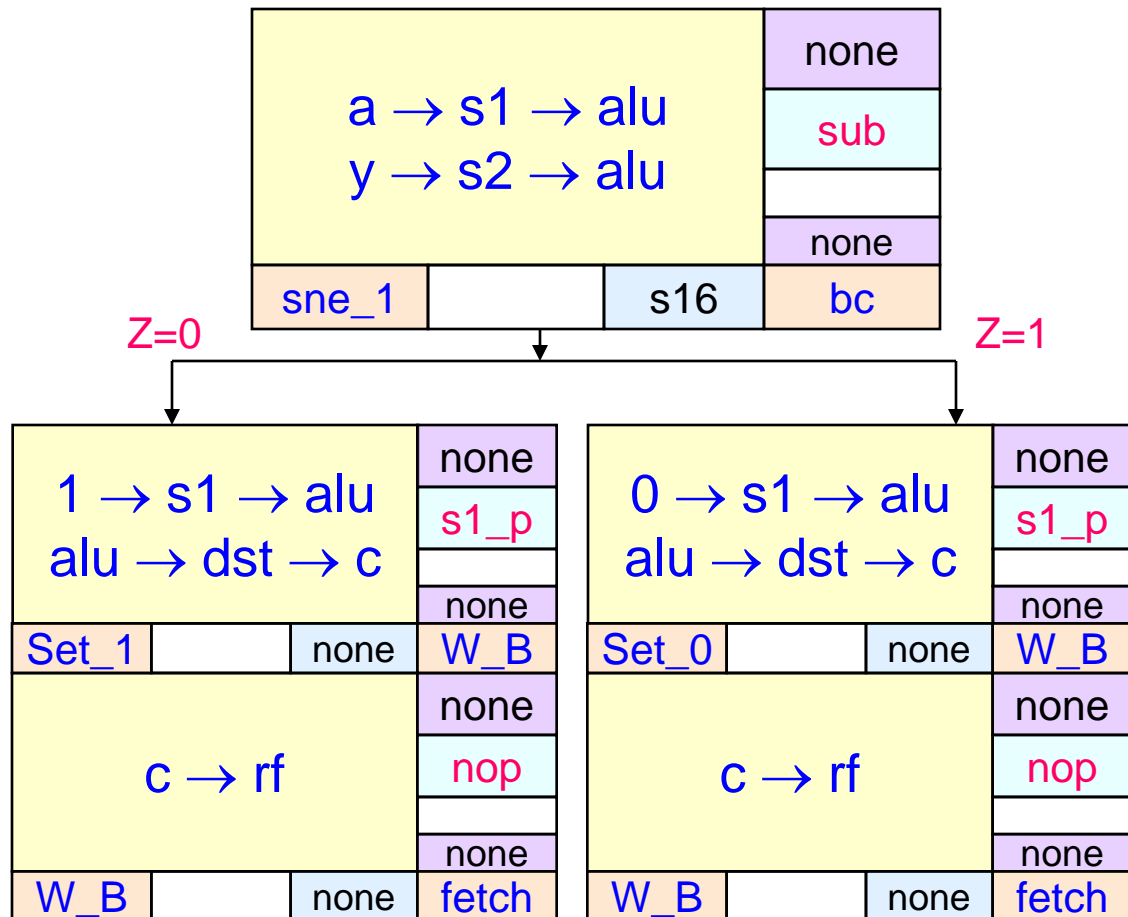
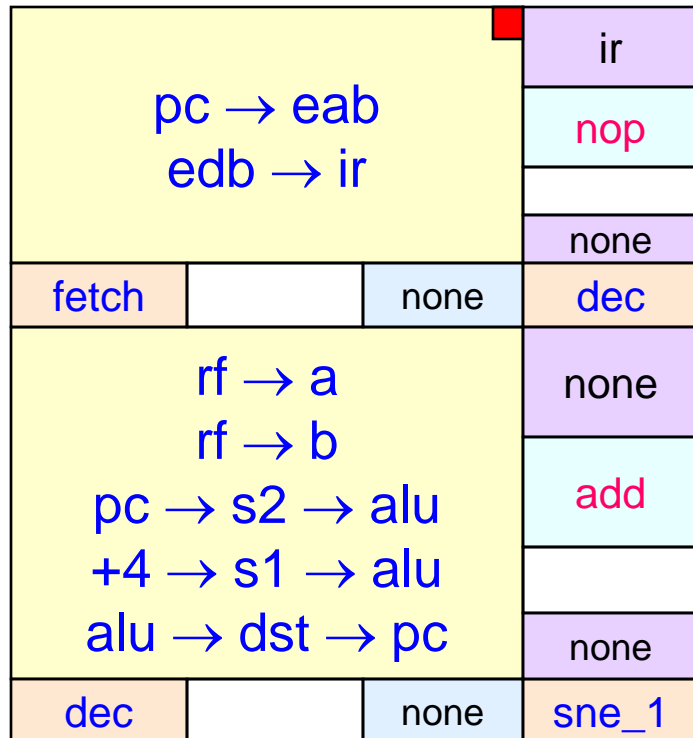
Merged Level 2 Hardware Flowchart

SNE Rd, Rs1, Rs2



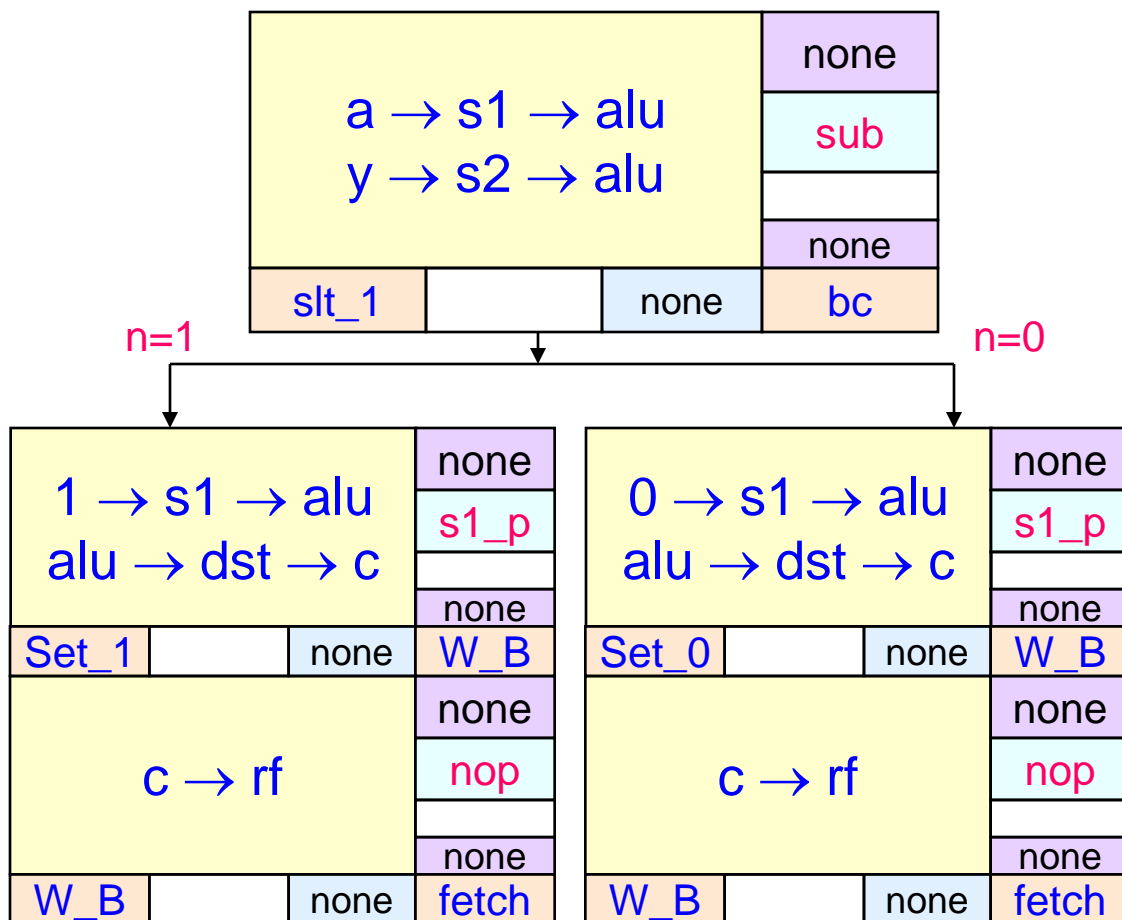
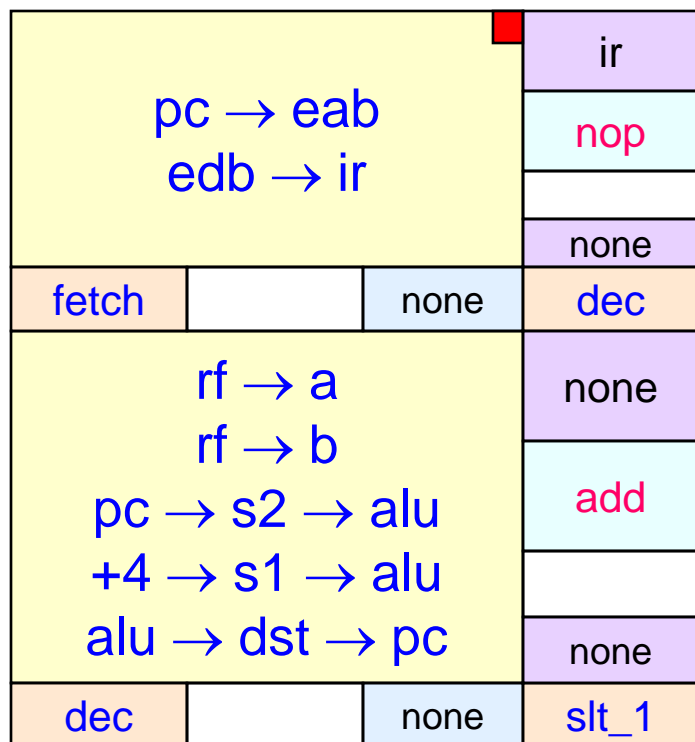
Merged Level 2 Hardware Flowchart

SNE.I Rd, Rs1, Imm



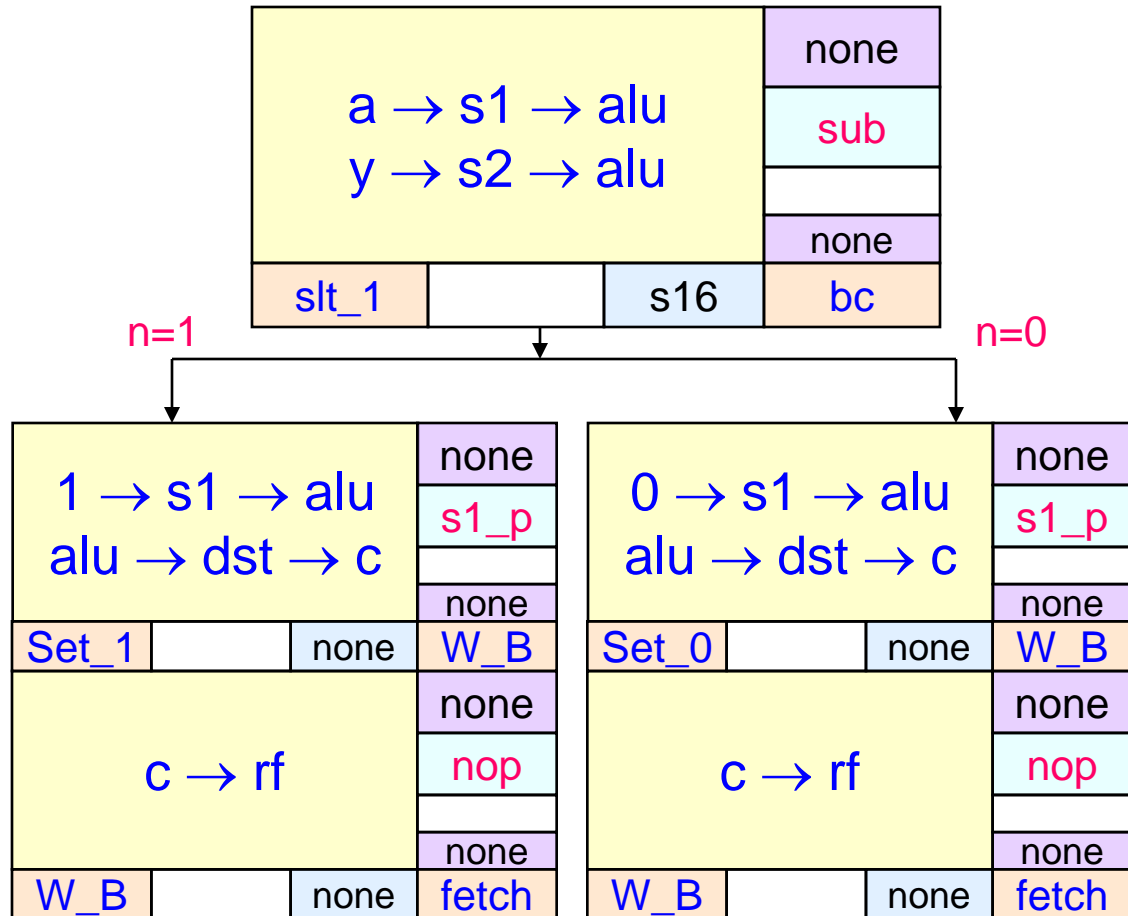
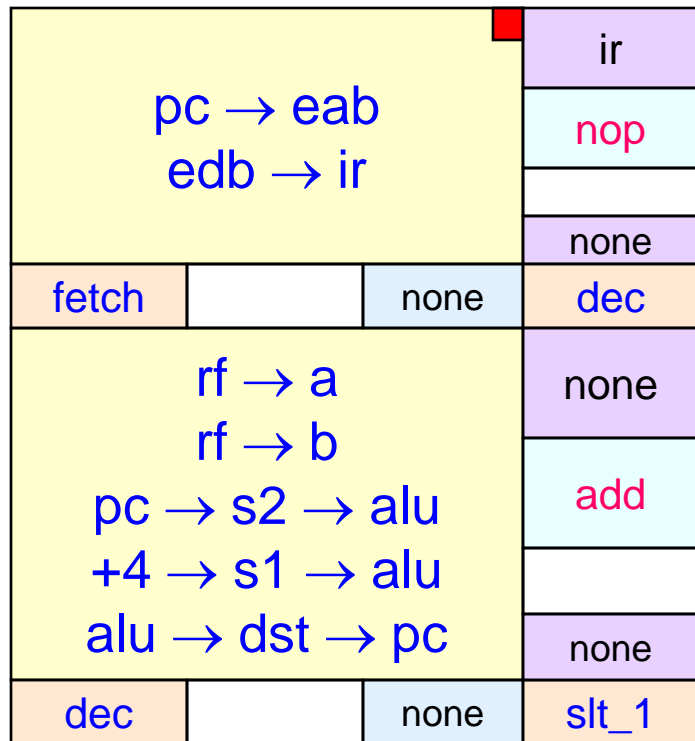
Merged Level 2 Hardware Flowchart

SLT Rd, Rs1, Rs2



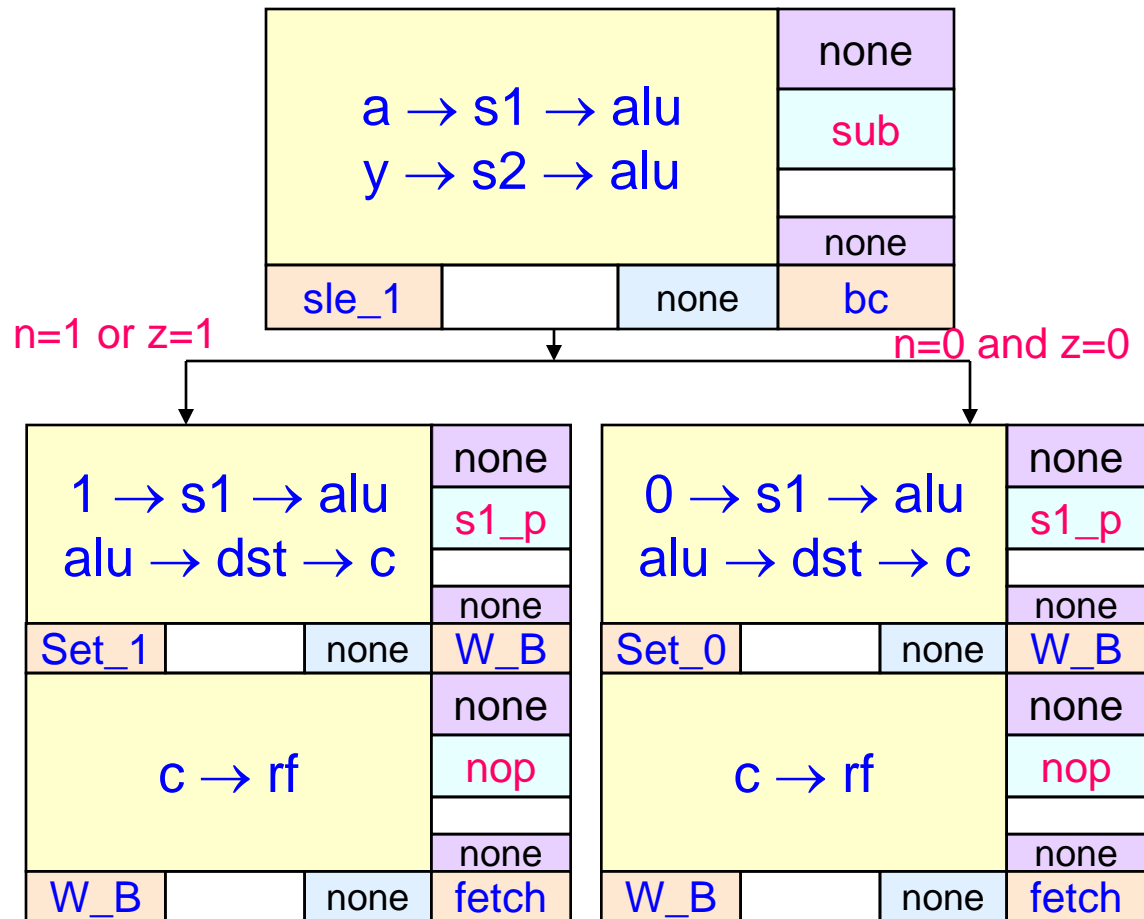
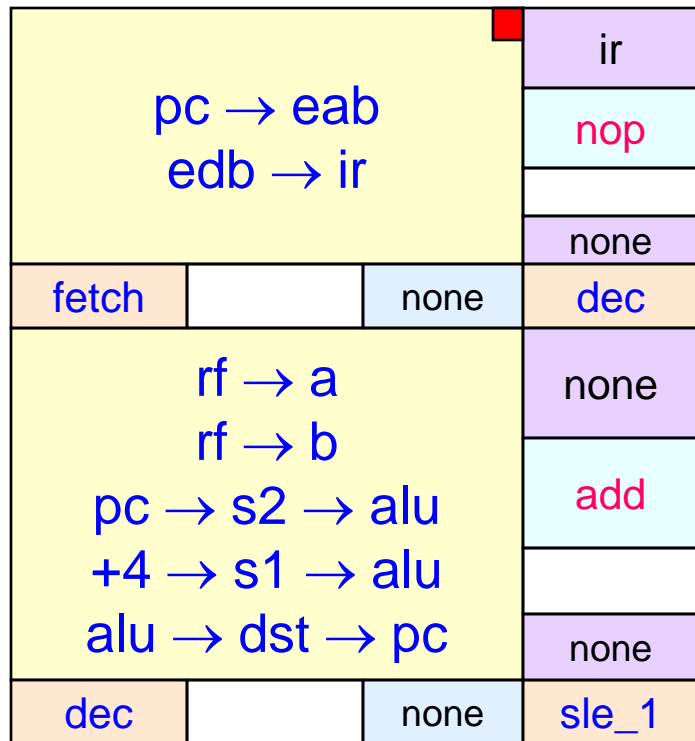
Merged Level 2 Hardware Flowchart

SLT.I Rd, Rs1, Imm



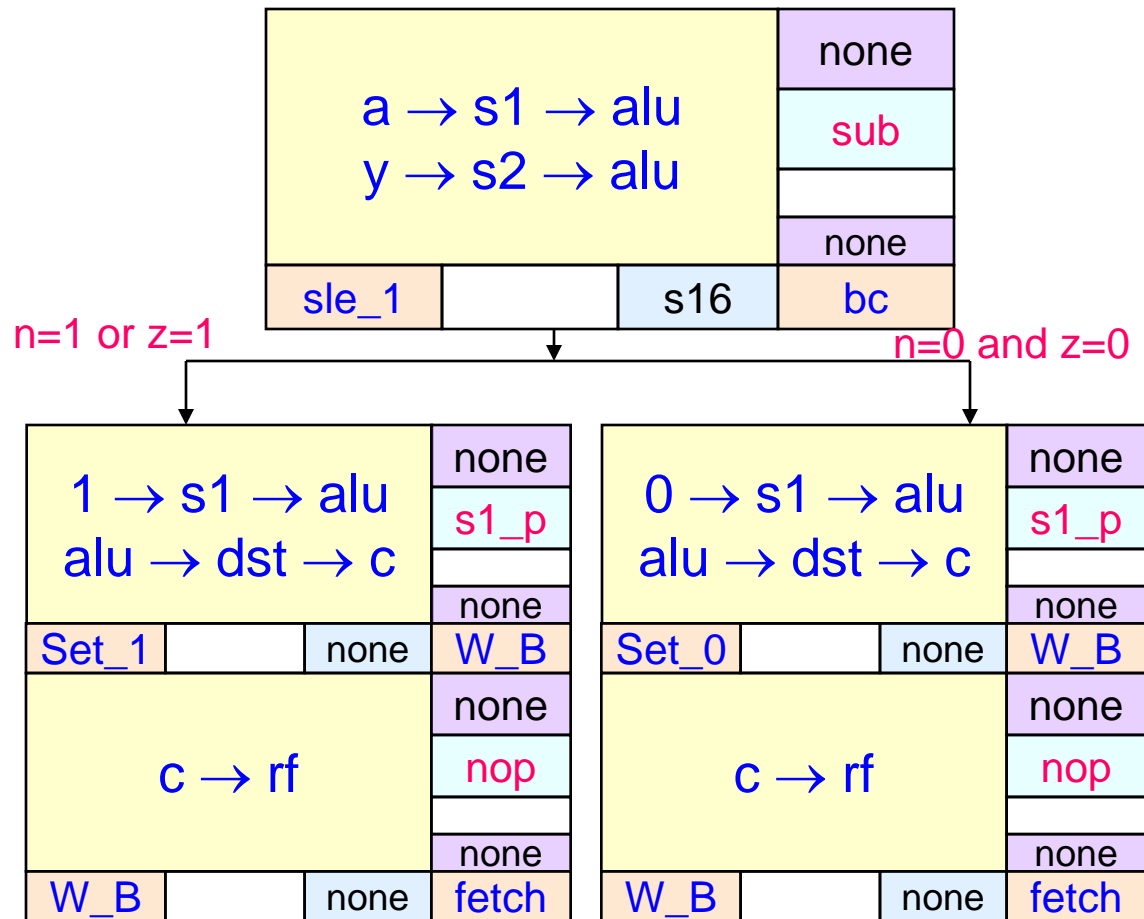
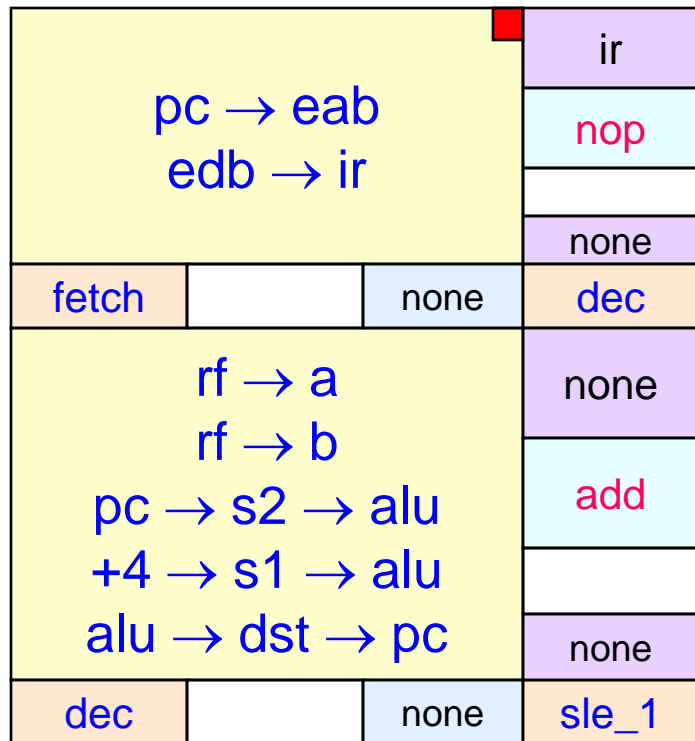
Merged Level 2 Hardware Flowchart

SLE Rd, Rs1, Rs2



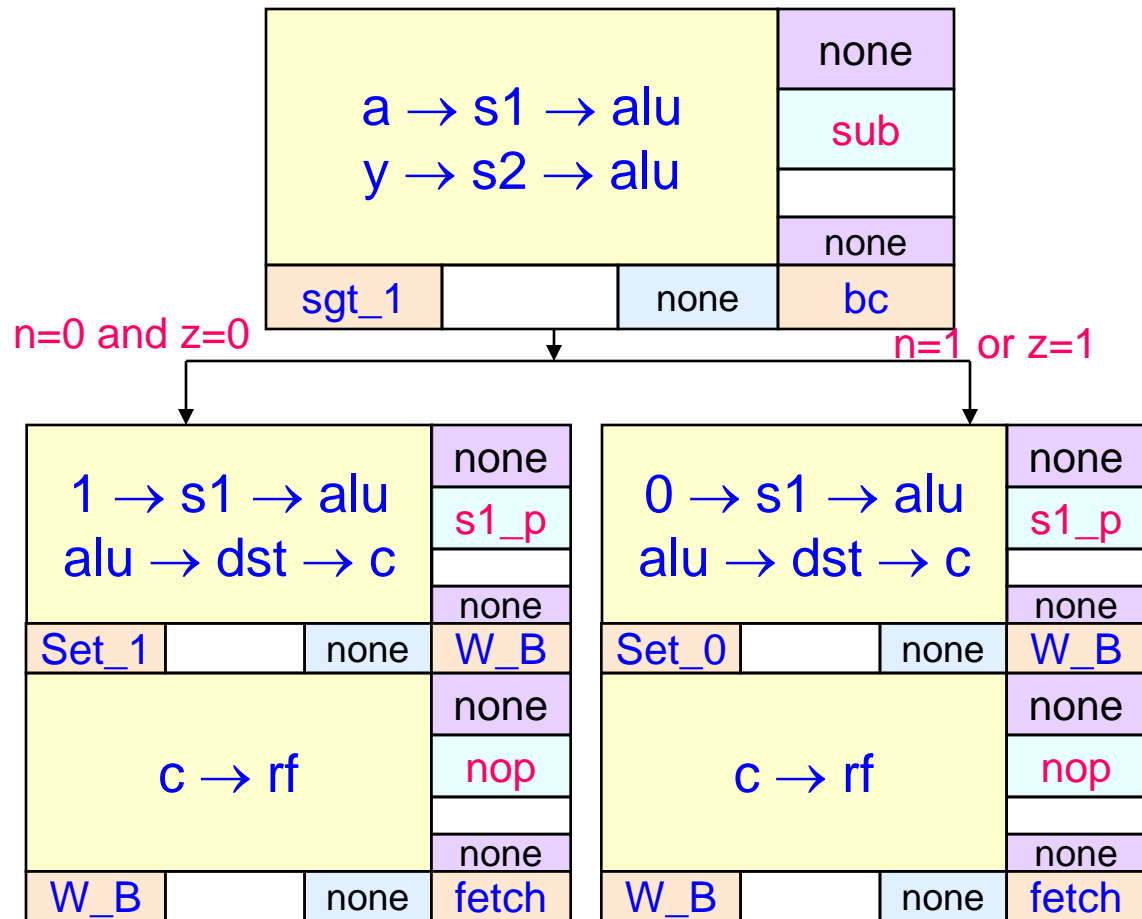
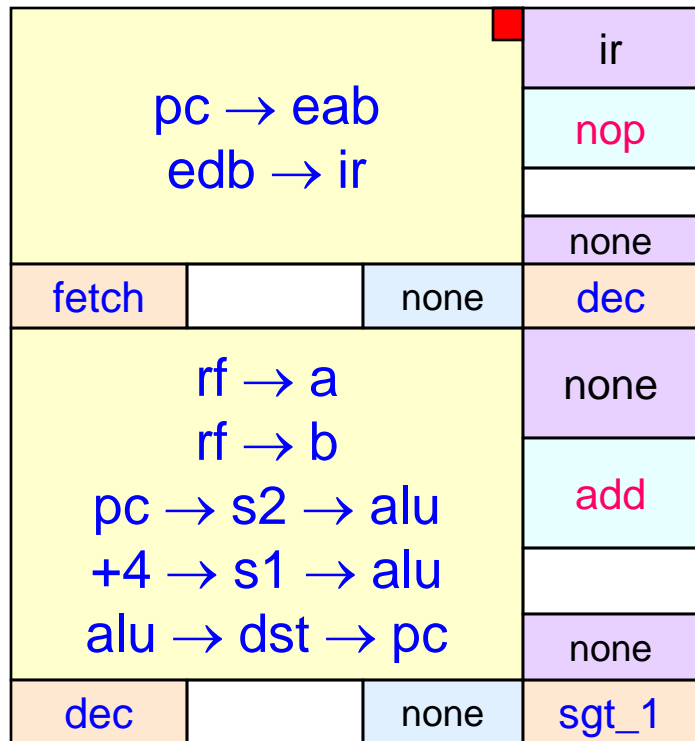
Merged Level 2 Hardware Flowchart

SLE.I Rd, Rs1, Imm



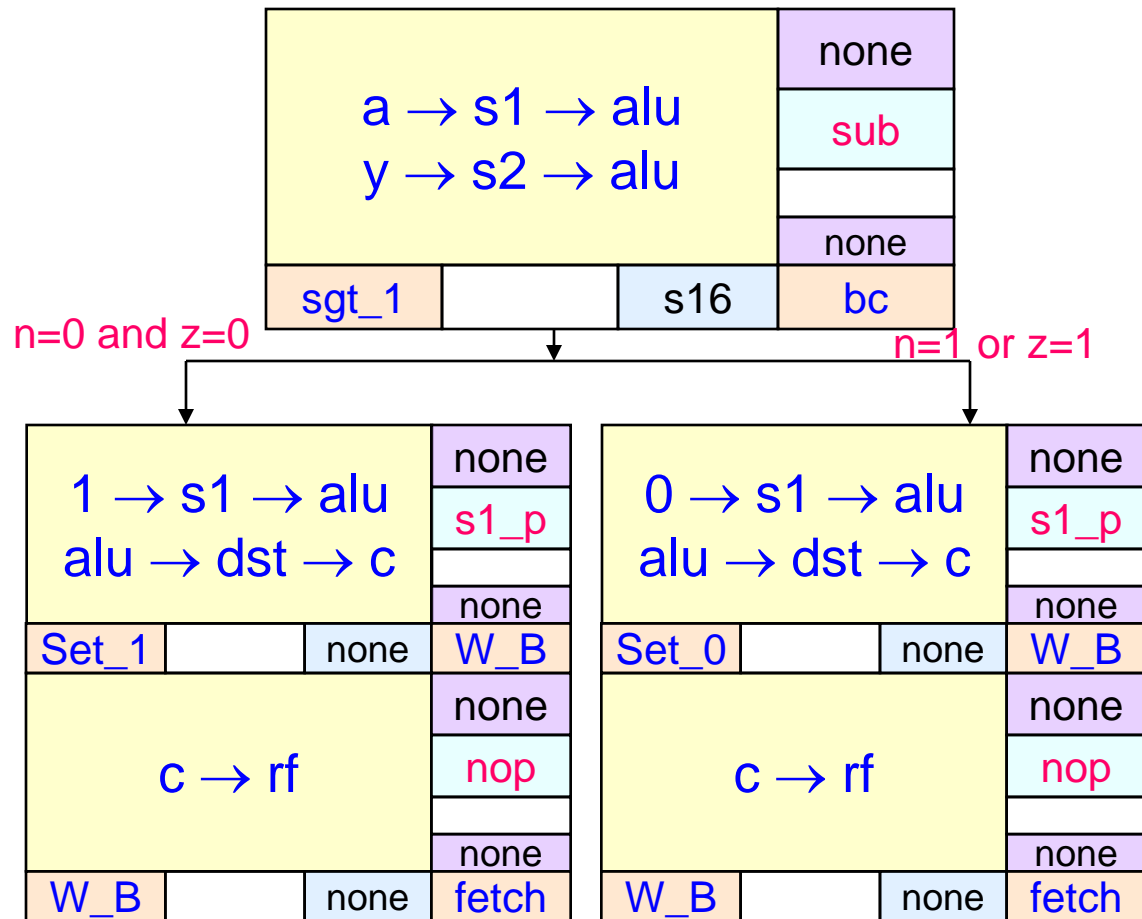
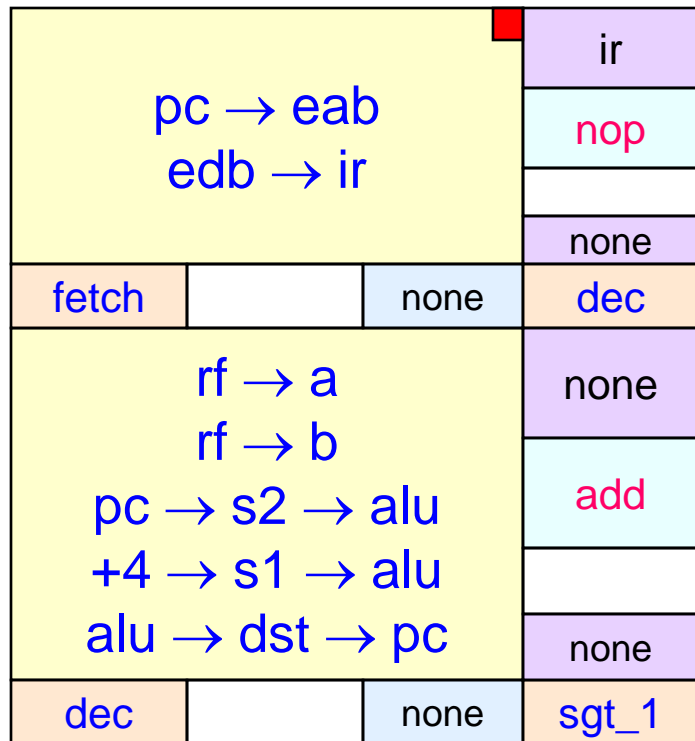
Merged Level 2 Hardware Flowchart

SGT Rd, Rs1, Rs2



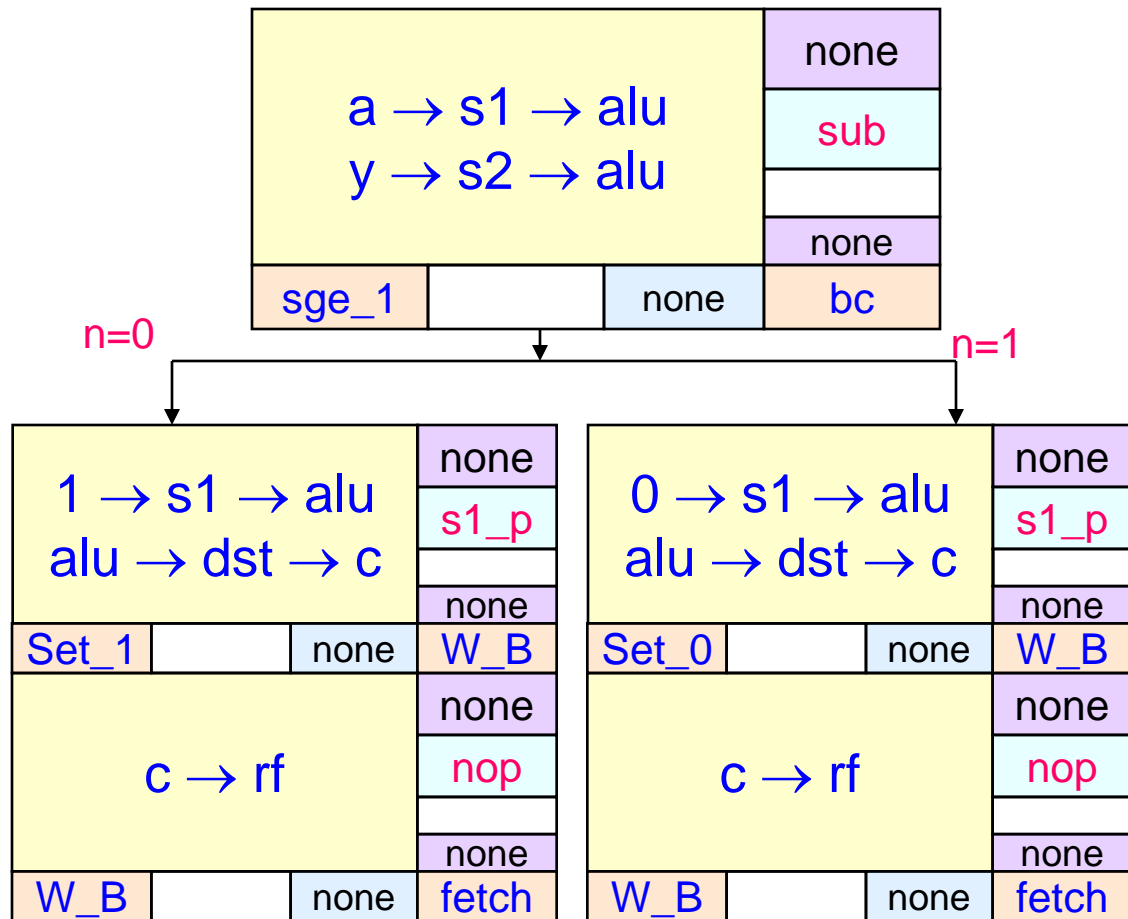
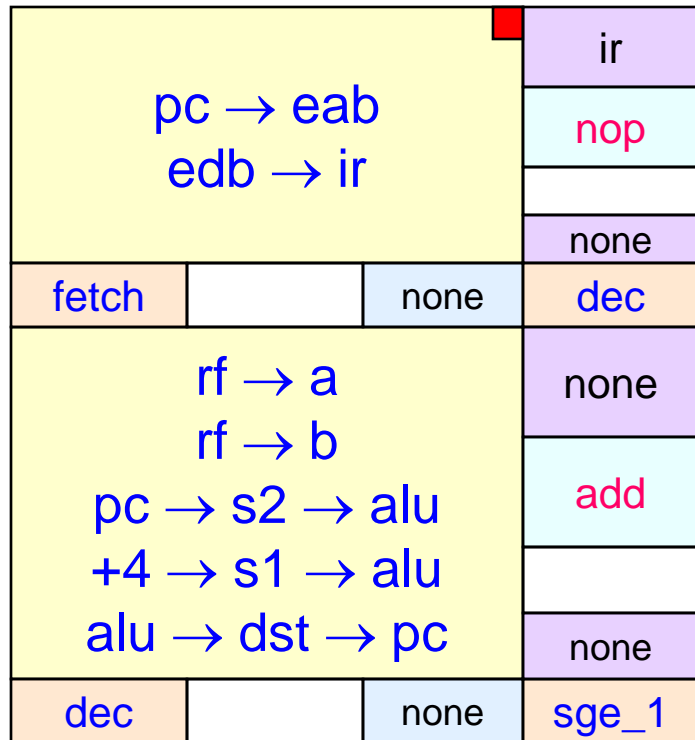
Merged Level 2 Hardware Flowchart

SGT.I Rd, Rs1, Imm



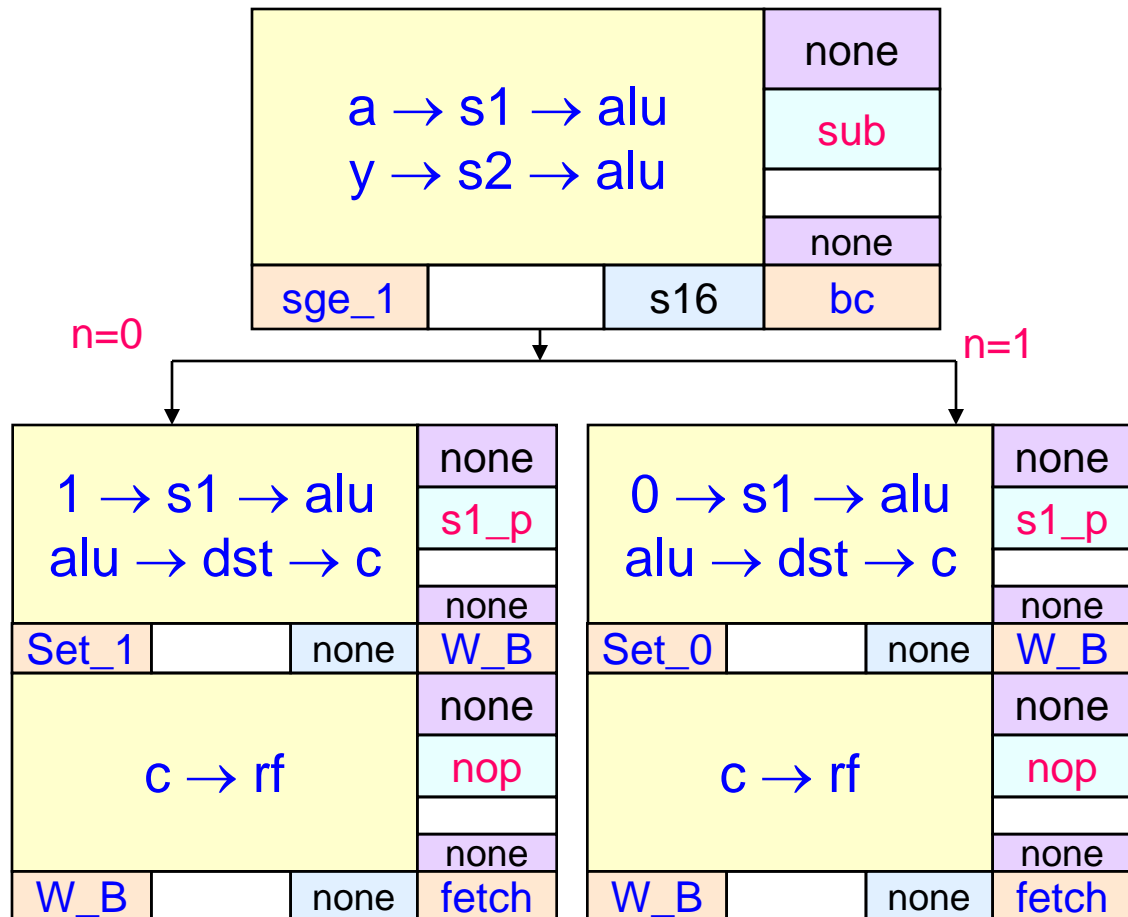
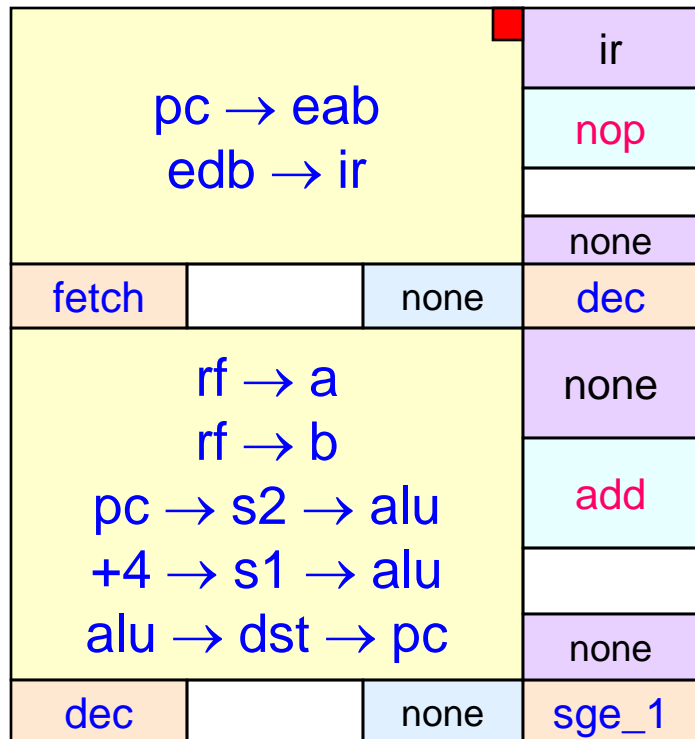
Merged Level 2 Hardware Flowchart

SGE Rd, Rs1, Rs2



Merged Level 2 Hardware Flowchart

SGE.I Rd, Rs1, Imm



Merged Level 2 Hardware Flowchart

BEQZ Rs, Label

$pc \rightarrow eab$ $edb \rightarrow ir$			ir
			nop
			none
			none
fetch		none	dec
$rf \rightarrow a$ $rf \rightarrow b$ $pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$			none
			add
			none
			none
dec		none	breqz

$a \rightarrow s1 \rightarrow alu$ $0 \rightarrow s2 \rightarrow alu$			none
			sub
			none
breqz		none	bc

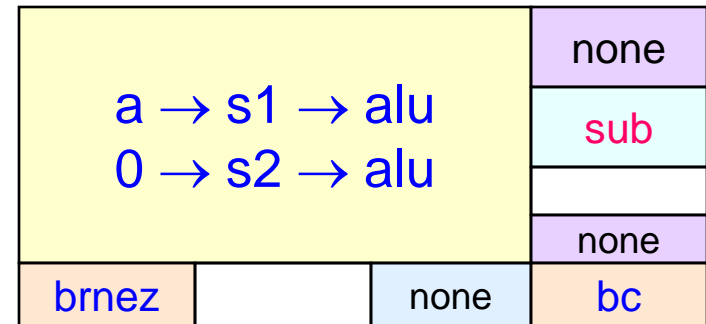
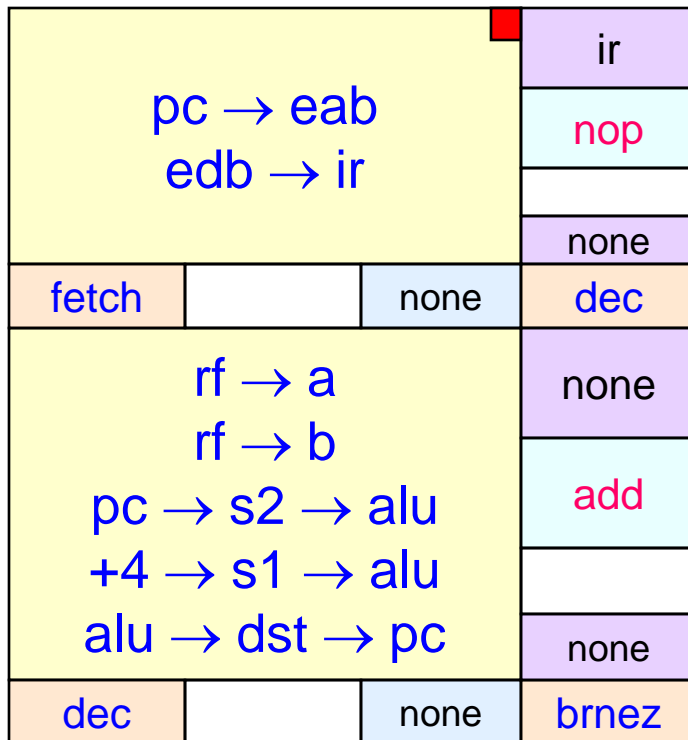
z=1

$ir \rightarrow s1 \rightarrow alu$ $pc \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow mar$			none
			add
			exc
			none
branch		s16	ld_pc
$mar \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$			none
			S2_p
			none
			none
ld_pc		none	fetch

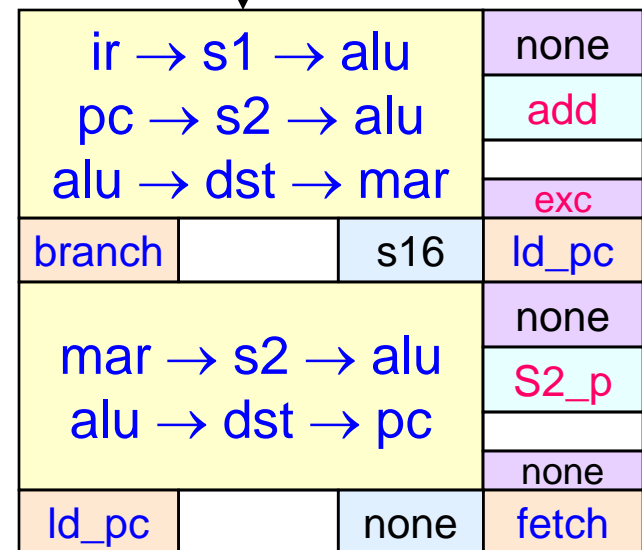


Merged Level 2 Hardware Flowchart

BNEZ Rs, Label



z=0



Merged Level 2 Hardware Flowchart

J Label

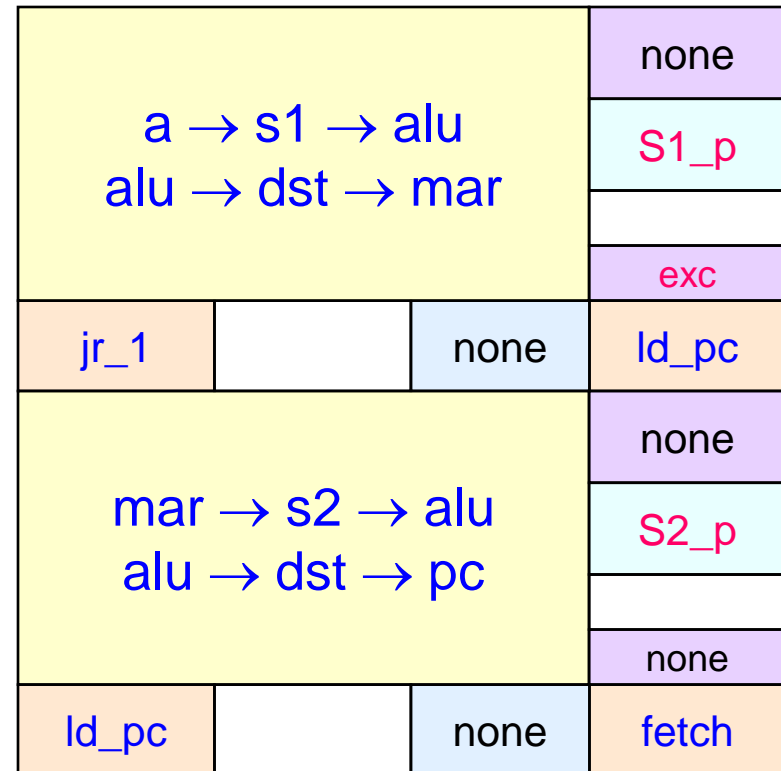
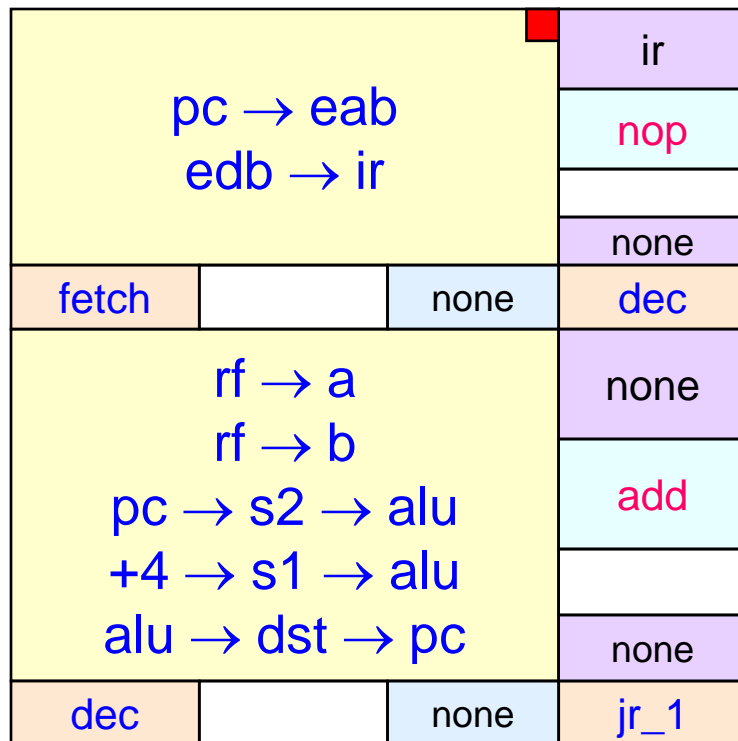
$pc \rightarrow eab$ $edb \rightarrow ir$			ir
			nop
			none
			none
fetch		none	dec
$rf \rightarrow a$ $rf \rightarrow b$ $pc \rightarrow s2 \rightarrow alu$ $+4 \rightarrow s1 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$			none
			add
			none
			none
dec		none	j_1

$ir \rightarrow s1 \rightarrow alu$ $pc \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow mar$			none
			add
			exc
			exc
j_1		s26	ld_pc
$mar \rightarrow s2 \rightarrow alu$ $alu \rightarrow dst \rightarrow pc$			none
			S2_p
			none
			none
ld_pc		none	fetch



Merged Level 2 Hardware Flowchart

JR Rs



Merged Level 2 Hardware Flowchart

pc → eab edb → ir			ir
			nop
			none
fetch		none	dec
rf → a rf → b pc → s2 → alu +4 → s1 → alu alu → dst → pc			none
			add
			none
			none
dec		none	jal_1
pc → s2 → alu alu → dst → c			none
			S2_p
			none
			none
jal_1		none	jal_2

JAL Label

ir → s1 → alu pc → s2 → alu alu → dst → mar c → rf			none
			add
			exc
			exc
jal_2		s26	ld_pc
mar → s2 → alu alu → dst → pc			none
			S2_p
			none
			none
ld_pc		none	fetch

Merged Level 2 Hardware Flowchart

LW Rd, Rs2(Rs1)

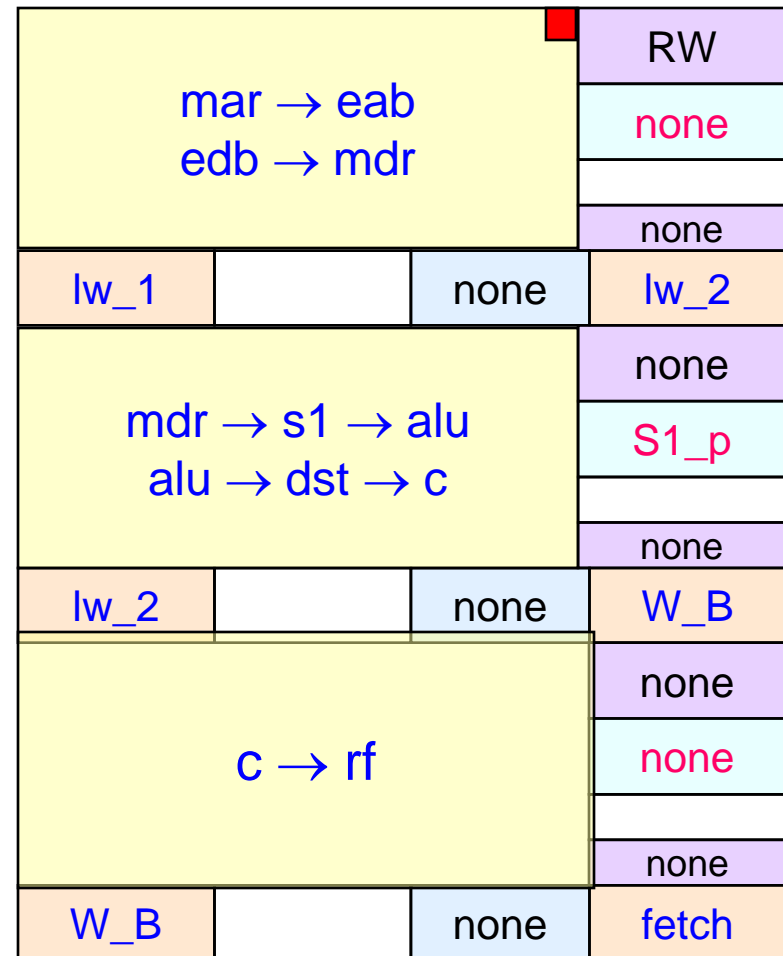
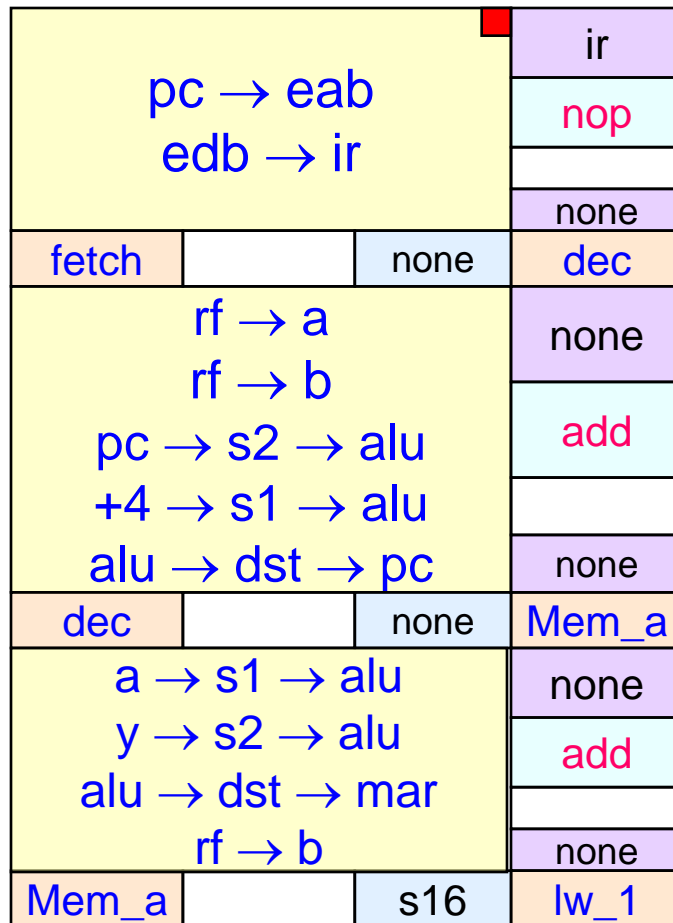
pc → eab edb → ir			ir
			nop
			none
			none
fetch		none	dec
rf → a rf → b pc → s2 → alu +4 → s1 → alu alu → dst → pc			none
			add
			none
			none
			none
dec		none	Mem_a
a → s1 → alu y → s2 → alu alu → dst → mar rf → b			none
			add
			none
			none
Mem_a		none	1w_1

mar → eab edb → mdr			RW
			none
			none
			none
lw_1		none	lw_2
mdr → s1 → alu alu → dst → c			none
			S1_p
			none
			none
lw_2		none	W_B
c → rf			none
			none
			none
			none
W_B		none	fetch



Merged Level 2 Hardware Flowchart

LW.I Rd, Imm(Rs1)



Merged Level 2 Hardware Flowchart

pc → eab edb → ir			ir
			nop
			none
fetch		none	dec
rf → a rf → b pc → s2 → alu +4 → s1 → alu alu → dst → pc			none
			add
			none
			none
			none
dec		none	Mem_a
a → s1 → alu y → s2 → alu alu → dst → mar rf → b			none
			add
			none
			none
Mem_a		none	sw_1

SW Rd, Rs2(Rs1)

b → s2 → alu alu → dst → mdr			none
			S2_p
			none
sw_1		none	sw_2
mar → eab mdr → edb			WW
			none
			none
			none
sw_2		none	fetch



Merged Level 2 Hardware Flowchart

pc → eab edb → ir			ir
			nop
			none
fetch		none	dec
rf → a rf → b pc → s2 → alu +4 → s1 → alu alu → dst → pc			none
			add
			none
			none
dec		none	Mem_a
a → s1 → alu y → s2 → alu alu → dst → mar rf → b			none
			add
			none
			none
Mem_a		s16	sw_1

SW.I Rd, Imm(Rs1)

b → s2 → alu alu → dst → mdr			none
			S2_p
			none
sw_1		none	sw_2
mar → eab mdr → edb			WW
			none
			none
			none
sw_2		none	fetch



Merged Level 2 Hardware Flowchart

LH Rd, Rs2(Rs1)

pc → eab edb → ir			ir
			nop
			none
fetch		none	dec
rf → a rf → b pc → s2 → alu +4 → s1 → alu alu → dst → pc			none
			add
			none
			none
			none
dec		none	Mem_a
a → s1 → alu y → s2 → alu alu → dst → mar rf → b			none
			add
			none
			none
Mem_a		none	lh_1

mar → eab edb → mdr			RH
			none
			none
lh_1		none	lh_2
mdr → s1 → alu 16 → s2 → alu alu → dst → c			none
			sra
			none
			none
lh_2		none	W_B
c → rf			none
			none
			none
			none
W_B		none	fetch



Merged Level 2 Hardware Flowchart

LH.I Rd, Imm(Rs1)

pc → eab edb → ir			ir
			nop
			none
			none
fetch		none	dec
rf → a rf → b pc → s2 → alu +4 → s1 → alu alu → dst → pc			none
			add
			none
			none
			none
dec		none	Mem_a
a → s1 → alu y → s2 → alu alu → dst → mar rf → b			none
			add
			none
			none
Mem_a		s16	lh_1

mar → eab edb → mdr			RW
			none
			none
			none
lh_1		none	lh_2
mdr → s1 → alu 16 → s2 → alu alu → dst → c			none
			sra
			none
			none
lh_2		none	W_B
c → rf			none
			none
			none
			none
W_B		none	fetch



Merged Level 2 Hardware Flowchart

pc → eab edb → ir			ir
			nop
			none
fetch		none	dec
rf → a rf → b pc → s2 → alu +4 → s1 → alu alu → dst → pc			none
			add
			none
			none
dec		none	Mem_a
a → s1 → alu y → s2 → alu alu → dst → mar rf → b			none
			add
			none
			none
Mem_a		none	sh_1

SH Rd, Rs2(Rs1)

b → s2 → alu alu → dst → mdr			none
			S2_p
			none
sh_1		none	sh_2
mar → eab mdr → edb			WH
			none
			none
			none
sh_2		none	fetch



Merged Level 2 Hardware Flowchart

pc → eab edb → ir			ir
			nop
			none
fetch		none	dec
rf → a rf → b pc → s2 → alu +4 → s1 → alu alu → dst → pc			none
			add
			none
			none
dec		none	Mem_a
a → s1 → alu y → s2 → alu alu → dst → mar rf → b			none
			add
			none
			none
Mem_a		s16	sh_1

SH.I Rd, Imm(Rs1)

b → s2 → alu alu → dst → mdr			none
			S2_p
			none
sh_1		none	sh_2
mar → eab mdr → edb			WH
			none
			none
			none
sh_2		none	fetch



Merged Level 2 Hardware Flowchart

LB Rd, Rs2(Rs1)

pc → eab edb → ir			ir
			nop
			none
			none
fetch		none	dec
rf → a rf → b pc → s2 → alu +4 → s1 → alu alu → dst → pc			none
			add
			none
			none
dec		none	Mem_a
a → s1 → alu y → s2 → alu alu → dst → mar rf → b			none
			add
			none
			none
Mem_a		none	lb_1

mar → eab edb → mdr			RB
			none
			none
			none
lb_1		none	lb_2
mdr → s1 → alu 24 → s2 → alu alu → dst → c			none
			sra
			none
			none
lb_2		none	W_B
c → rf			none
			none
			none
			none
W_B		none	fetch



Merged Level 2 Hardware Flowchart

LB.I Rd, Imm(Rs1)

pc → eab edb → ir			ir
			nop
			none
fetch		none	dec
rf → a rf → b pc → s2 → alu +4 → s1 → alu alu → dst → pc			none
			add
			none
			none
dec		none	Mem_a
a → s1 → alu y → s2 → alu alu → dst → mar rf → b			none
			add
			none
			none
Mem_a		s16	lb_1

mar → eab edb → mdr			RB
			none
			none
lb_1		none	lb_2
mdr → s1 → alu 24 → s2 → alu alu → dst → c			none
			sra
			none
			none
lb_2		none	W_B
c → rf			none
			none
			none
			none
W_B		none	fetch



Merged Level 2 Hardware Flowchart

pc → eab edb → ir			ir
			nop
			none
fetch		none	dec
rf → a rf → b pc → s2 → alu +4 → s1 → alu alu → dst → pc			none
			add
			none
			none
dec		none	Mem_a
a → s1 → alu y → s2 → alu alu → dst → mar rf → b			none
			add
			none
			none
Mem_a		none	sb_1

SB Rd, Rs2(Rs1)

b → s2 → alu alu → dst → mdr			none
			S2_p
			none
sb_1		none	sb_2
mar → eab mdr → edb			WB
			none
			none
			none
sb_2		none	fetch



Merged Level 2 Hardware Flowchart

pc → eab edb → ir			ir
			nop
			none
fetch		none	dec
rf → a rf → b pc → s2 → alu +4 → s1 → alu alu → dst → pc			none
			add
			none
			none
dec		none	Mem_a
a → s1 → alu y → s2 → alu alu → dst → mar rf → b			none
			add
			none
			none
Mem_a		s16	sb_1

SB.I Rd, Imm(Rs1)

b → s2 → alu alu → dst → mdr			none
			S2_p
			none
sb_1		none	sb_2
mar → eab mdr → edb			WB
			none
			none
			none
sb_2		none	fetch

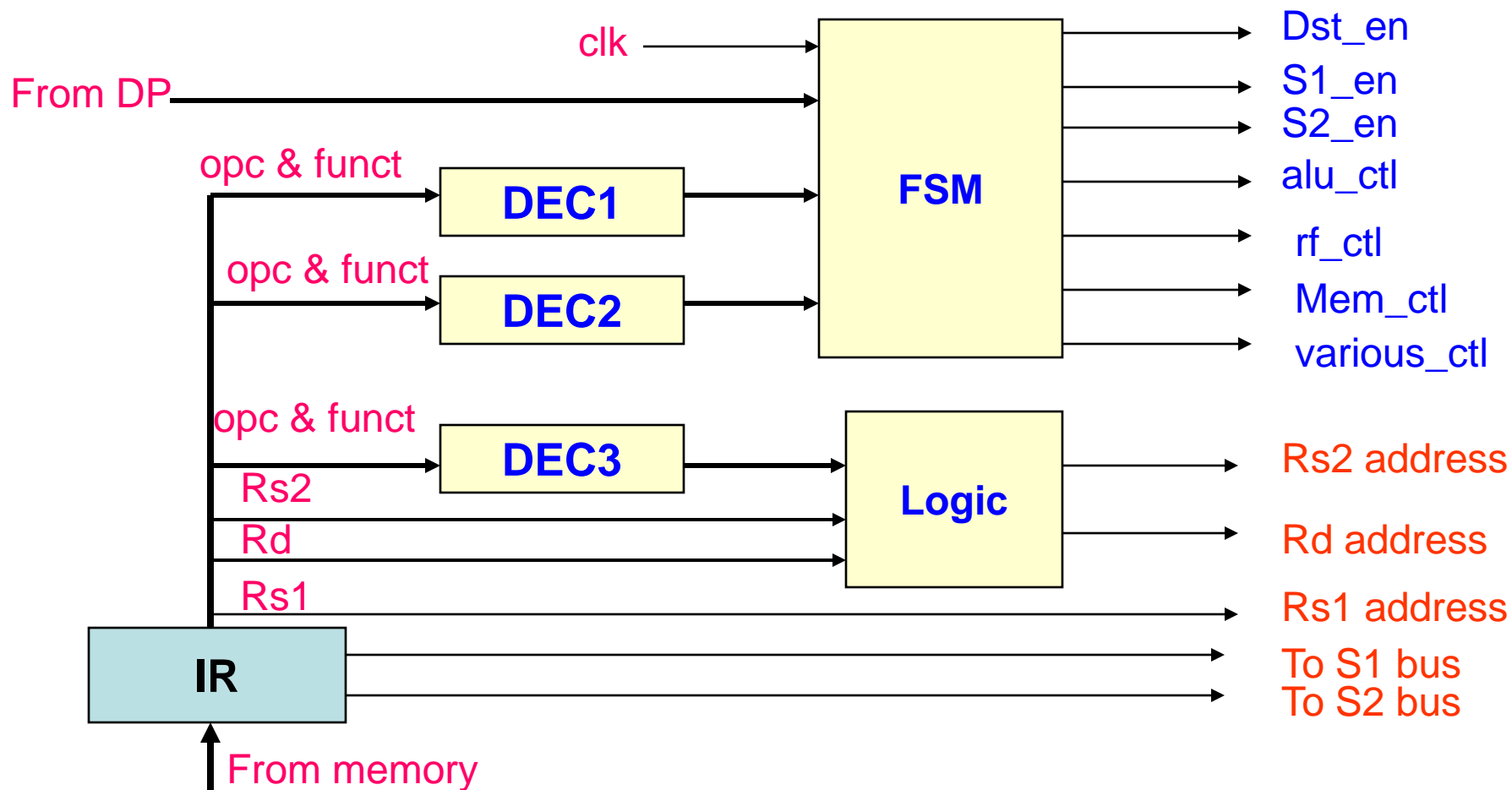


Implementation - States

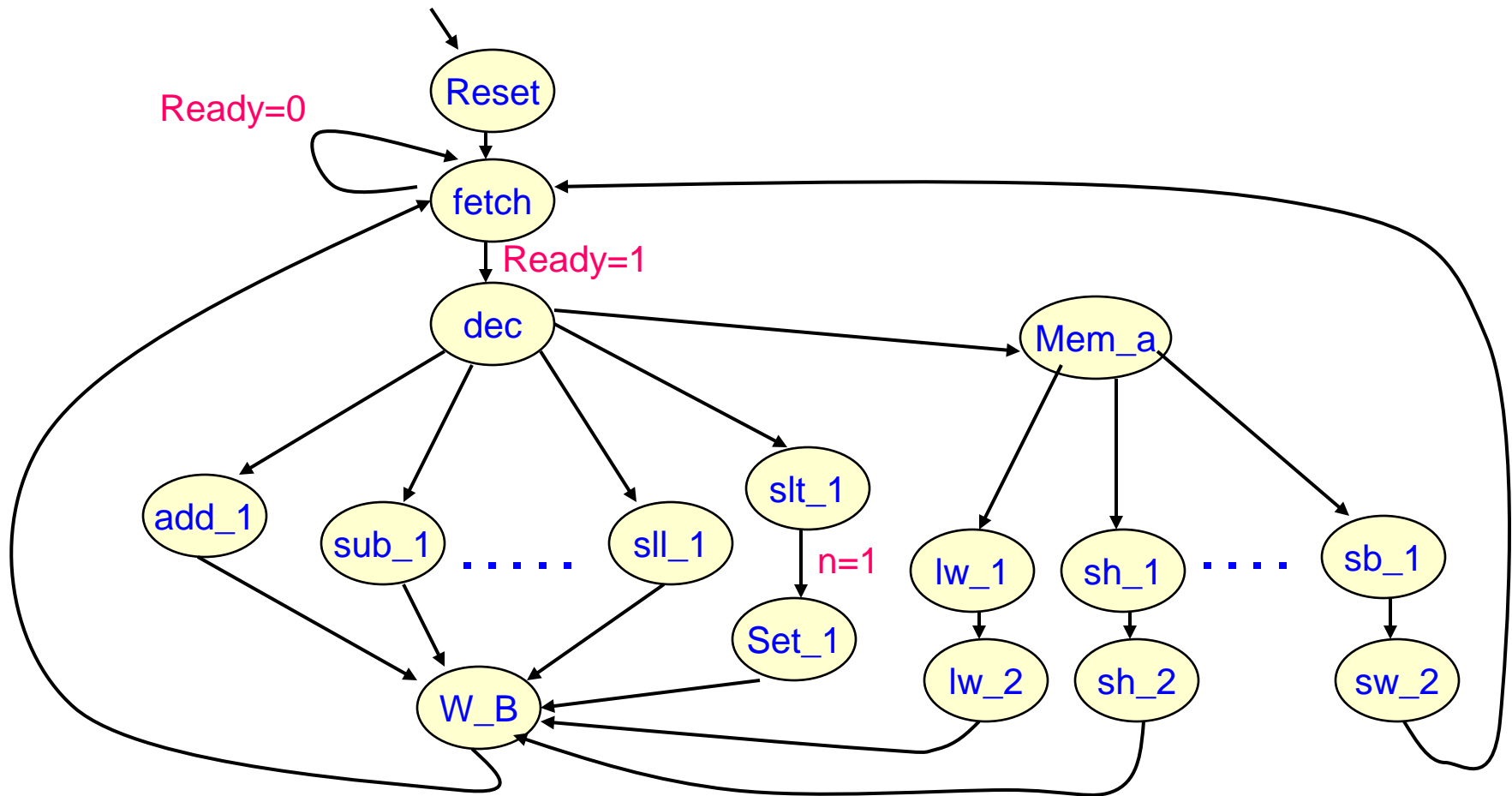
States:

- fetch, dec, mem_a
- add_1, sub_1, and_1, or_1, xor_1, sll_1, srl_1, sra_1
- seq_1, neq_1, slt_1, sle_1, sgt_1, sge_1
- Lw_1, lw_2, lh_1, lh_2, lb_1, lb_2, sw_1, sw_2, sh_1, sh_2, sb_1, sb_2
- breqz, brnez, branch, set_1, set_0
- reset

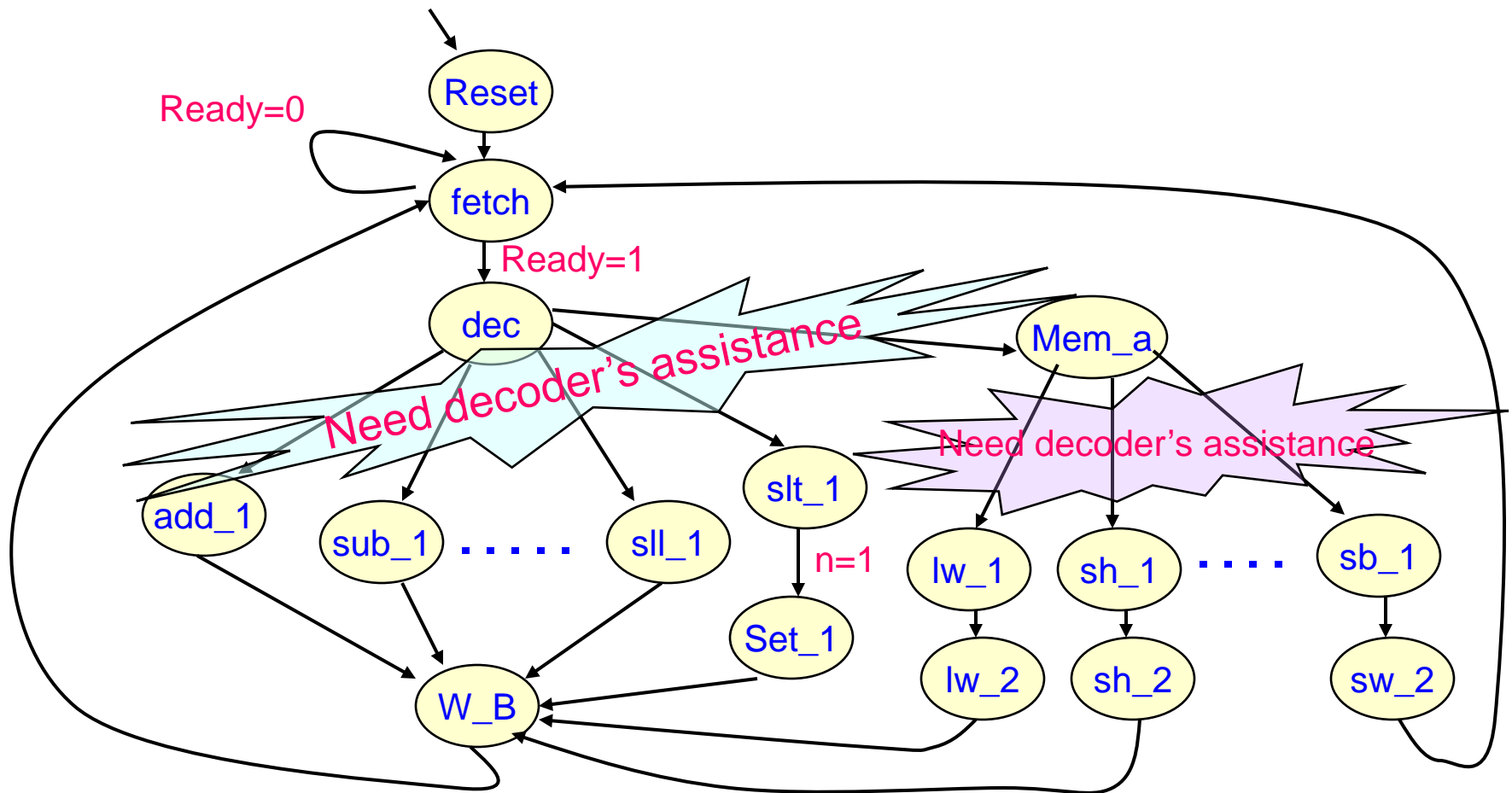
Implementation - Controller



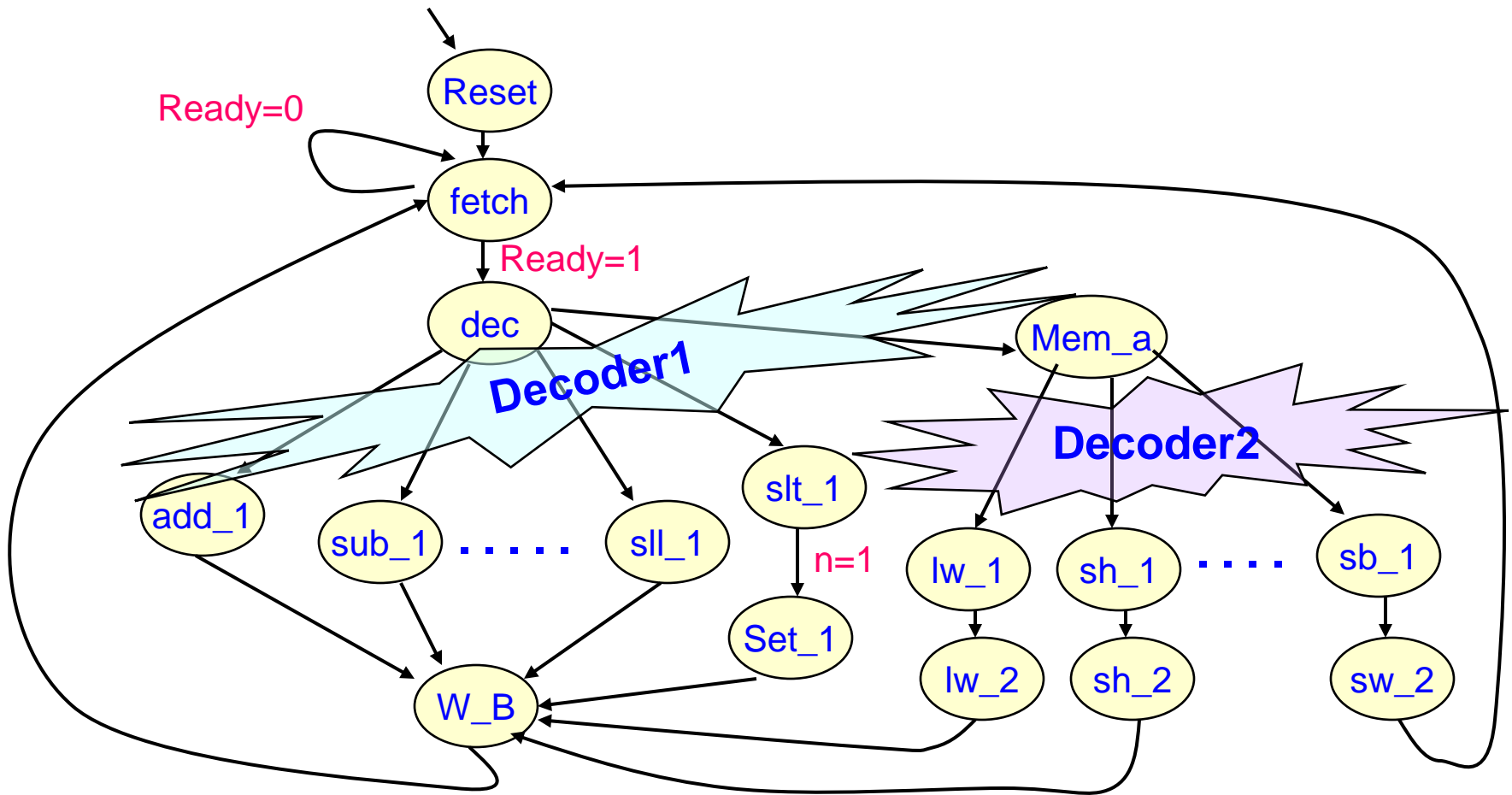
State Transition



State Transition



State Transition



Implementation

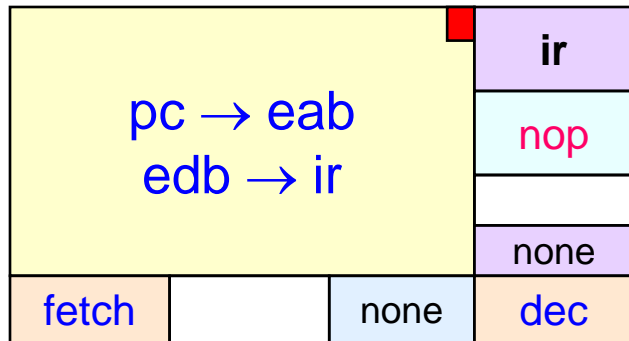
Reset

```
when reset_1 =>  
  s1_enab <= s1_const;  
  s2_enab <= s2_none;  
  alu_op_sel <= alu_S1_p;  
  dest_enab <= dest_pc;  
  const_sel <= const_00;  
  rf_op_sel <= rfop_none;  
  immmed_sel <= imm_dcare;  
  exc_enab <= exc_none;  
  mem_ctrl <= mem_fetch;  
  next_state <= fetch;
```



Implementation

Fetch



```
when fetch =>
  s1_enab <= s1_none;
  s2_enab <= s2_none;
  alu_op_sel <= alu_dcare;
  dest_enab <= dest_none;
  const_sel <= const_dcare;
  rf_op_sel <= rfop_none;
  immmed_sel <= imm_dcare;
  exc_enab <= exc_none;
  mem_ctrl <= mem_fetch;
  if ready = '1' then next_state <= dec;
  else next_state <= fetch;
end if;
```

Implementation

Dec

rf → a rf → b pc → s2 → alu +4 → s1 → alu alu → dst → pc			none
			add
			none
dec		none	Mem_a

when **dec** =>
s1_enab <= s1_const;
s2_enab <= s2_pc;
alu_op_sel <= alu_add;
dest_enab <= dest_pc;
const_sel <= const_04;
rf_op_sel <= rfop_ab_rf;
immed_sel <= imm_dcare;
exc_enab <= exc_none;
mem_ctrl <= mem_none;
next_state <= dec_1_in;



Implementation

Mem_a

dec		none	Mem_a
$a \rightarrow s1 \rightarrow \text{alu}$ $y \rightarrow s2 \rightarrow \text{alu}$ $\text{alu} \rightarrow \text{dst} \rightarrow \text{mar}$ $\text{rf} \rightarrow b$			none
			add
			none
Mem_a		s16	sb_1

```

when Mem_a =>
  s1_enab <= s1_a;
  s2_enab <= s2_y;
  alu_op_sel <= alu_add;
  dest_enab <= dest_mar;
  const_sel <= const_dcare;
  rf_op_sel <= rfop_ab_rfx;
  immmed_sel <= imm_dcare;
  exc_enab <= exc_none;
  mem_ctrl <= mem_none;
  next_state <= dec_2_in;
  
```



Implementation

Add_1

a → s1 → alu y → s2 → alu alu → dst → c			none
			add
			none
add_1		none	W_B

```
when add_1 =>  
  s1_enab <= s1_a;  
  s2_enab <= s2_y;  
  alu_op_sel <= alu_add;  
  dest_enab <= dest_c;  
  const_sel <= const_dcare;  
  rf_op_sel <= rfop_none;  
  immmed_sel <= imm_s16;  
  exc_enab <= exc_none;  
  mem_ctrl <= mem_none;  
  next_state <= W_B;
```

Implementation

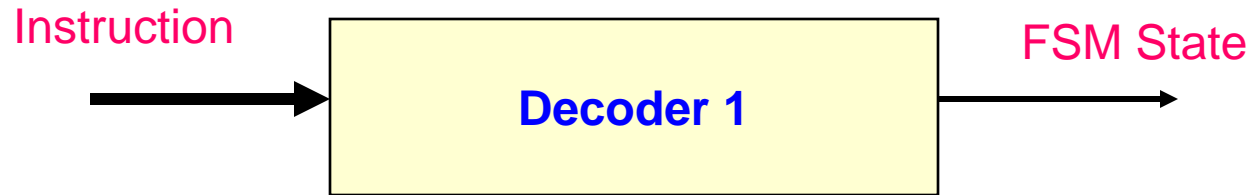
W_B

c → rf			none
			nop
			none
			none
W_B		none	fetch

when **W_B** =>
s1_enab <= s1_none;
s2_enab <= s2_none;
alu_op_sel <= alu_none;
dest_enab <= dest_c;
const_sel <= const_dcare;
rf_op_sel <= rfop_none;
immed_sel <= imm_none;
exc_enab <= exc_none;
mem_ctrl <= mem_none;
next_state <= fetch;



Implementation: Decoder1



```
decoder1 : process(instruct)
begin
  case instruct(0 to 5) is
    when op_rr_alu =>
      case instr_in(26 to 31) is
        when funct_nop =>
          dec1_out <= fetch;
        when funct_add =>
          dec1_out <= add_1;
        .....
        .....
```

Implementation: Decoder1

```
when funct_lw | funct_lh | funct_lb |  
    funct_sw | funct_sh | funct_sb =>  
    dec1_out <= Mem_a;  
when others =>  
    dec1_out <= fetch; // - it should be illegal state  
end case;  
when op_lw_i | op_lh_i | op_lb_i | op_sw_i | op_sh_i |  
    op_sb_i => dec1_out <= Mem_a;  
when op_lhi =>  
    dec1_out <= lhi;  
    when op_add_i =>  
        dec1_out <= add_1;
```

.....

.....

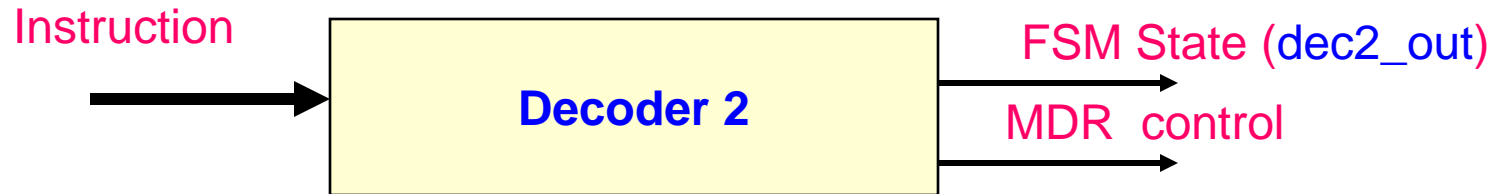


Implementation: Decoder1

```
when op_jal =>  
    dec1_out <= jal_1;  
when op_jr =>  
    dec1_out <= jr_1;  
.....  
.....  
when others =>  
    dec1_out <= fetch; // it should be illegal state  
end case;  
end process decoder1;
```



Implementation: Decoder2



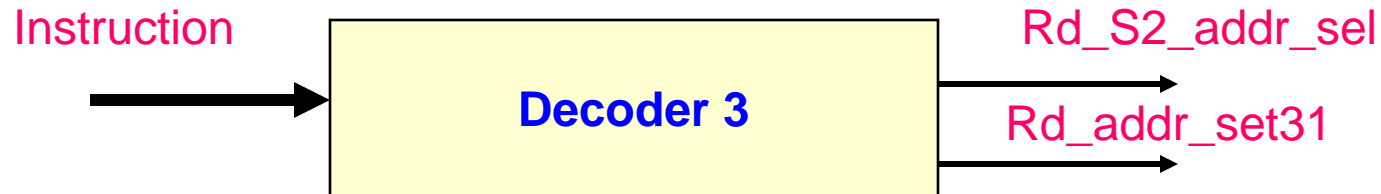
```
decoder2 : process(instruct)
begin
case instruct(0 to 5) is
  when op_rr_alu =>
    case instr_in(26 to 31) is
      when funct_lw =>
        dec2_out <= lw_1;
        mdr_ctrl_out <= mdr_ctl_none;
      when rr_func_lh =>
        dec2_out <= lh_1;          .....
    .....end case;
```

Implementation: Decoder2

```
when op_lw_i =>
    dec2_out <= lw_1;
    mdr_ctrl_out <= mdr_ctl_none;
when op_lh_i =>
    dec2_out <= lh_1;
    mdr_ctrl_out <= mdr_ctl_lh;
.....
.....
when others =>
    dec2_out <= sb_1;
    mdr_ctrl_out <= mdr_ctl_sb;
end case;
end process decoder2;
```



Implementation: Decoder3



```
decoder3 : process(instr_in)
begin
  case instr_in(0 to 5) is
    when op_rr_alu =>
      case instr_in(26 to 31) is
        when
          funct_add | funct_sub | tfunct_and | funct_or |
          funct_xor | funct_seq | funct_sne | funct_slt | funct_sle |
          funct_sgt | funct_sge | funct_lw | funct_lh | funct_lb
          | funct_sw | funct_sh |    funct_sb =>
          rd_s2_adr_sel <= '1' ;
          rd_adr_set31 <= '0' ; .....
```

Implementation: Decoder3

```
when others =>
    rd_s2_adr_sel <= '0' ;
    rd_adr_set31 <= '0' ;
end case;
when op_jal | op_jalr =>
    rd_s2_adr_sel <= '0' ;
    rd_adr_set31 <= '1' ;
.....
.....
when others =>
    rd_s2_adr_sel <= '0' ;
    rd_adr_set31 <= '0' ;
end case;
end process decoder3;
```



Implementation: Controller (FSM)

```
architecture fsm_1 of fsm is
    signal cur_state: fsm_state;
    signal nxt_state: fsm_state;
```

```
State_change: process(clk, reset)
begin
    if (clk'event and clk=1)
        if reset = '1' then
            cur_state <= reset_1;
        else
            cur_state <= nxt_state;
        end if;
    end if;
end process state_change;
```



Implementation: Controller (FSM)

```
Next_state_logic: process(cur_state, reset, ready, dec_1_in,
dec_2_in, alu_neg, alu_z)
begin
  case cur_state is
    when reset_1 =>
      nxt_state <= fetch;
    when fetch =>
      if ready = '1' then
        nxt_state <= dec;
      else
        nxt_state <= fetch;
      when dec =>
        nxt_state <= dec_1_in;
        . . . . .
```



Implementation: Controller (FSM)

```
when Mem_a =>
    nxt_state <= dec_2_in;
    . . . . .
when slt_1 =>
    if alu_neg = '1' then
        nxt_state <= Set_1;
    else
        nxt_state <= Set_0;
    end if;
    . . . . .
    . . . . .
end case;
end process Next_state_logic;
```



Implementation: Controller (FSM)

```
Output_logic: process(cur_state)
begin
  case cur_state is
    when reset_1 =>
      s1_enab <= s1_const;
      s2_enab <= s2_none;
      alu_op_sel <= alu_S1_p;
      dest_enab <= dest_pc;
      const_sel <= const_00;
      rf_op_sel <= rfop_none;
      immed_sel <= imm_dcare;
      exc_enab <= exc_none;
      mem_ctrl <= mem_fetch;
```



Implementation: Controller (FSM)

```
when fetch =>  
  s1_enab <= s1_none;  
  s2_enab <= s2_none;  
  alu_op_sel <= alu_dcare;  
  dest_enab <= dest_none;  
  const_sel <= const_dcare;  
  rf_op_sel <= rfop_none;  
  immед_sel <= imm_dcare;  
  exc_enab <= exc_none;  
  mem_ctrl <= mem_fetch;  
  
  .....  
  
  .....  
  
  .....
```



Implementation: Controller (FSM)

```
when W_B =>
  s1_enab <= s1_none;
  s2_enab <= s2_none;
  alu_op_sel <= alu_none;
  dest_enab <= dest_c;
  const_sel <= const_dcare;
  rf_op_sel <= rfop_none;
  immmed_sel <= imm_none;
  exc_enab <= exc_none;
  mem_ctrl <= mem_none;
  . . . . .
  . . . . .
end case;
end process output_logic;
```



Thank You

