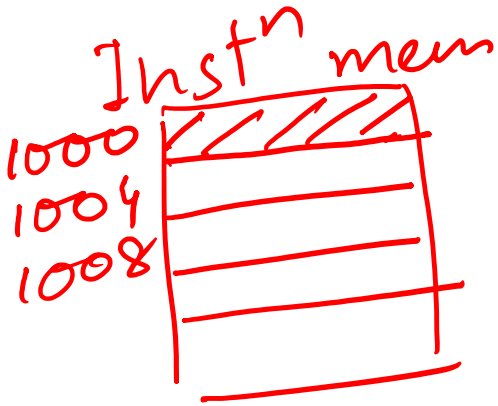
CSE 112: Computer Organization (Section A)

Instructor: Sujay Deb

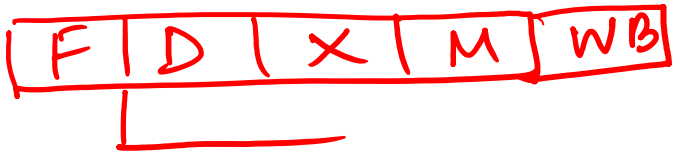
Lecture 15

INDRAPRASTHA INSTITUTE *of*
INFORMATION TECHNOLOGY
DELHI

Encoding Instructions



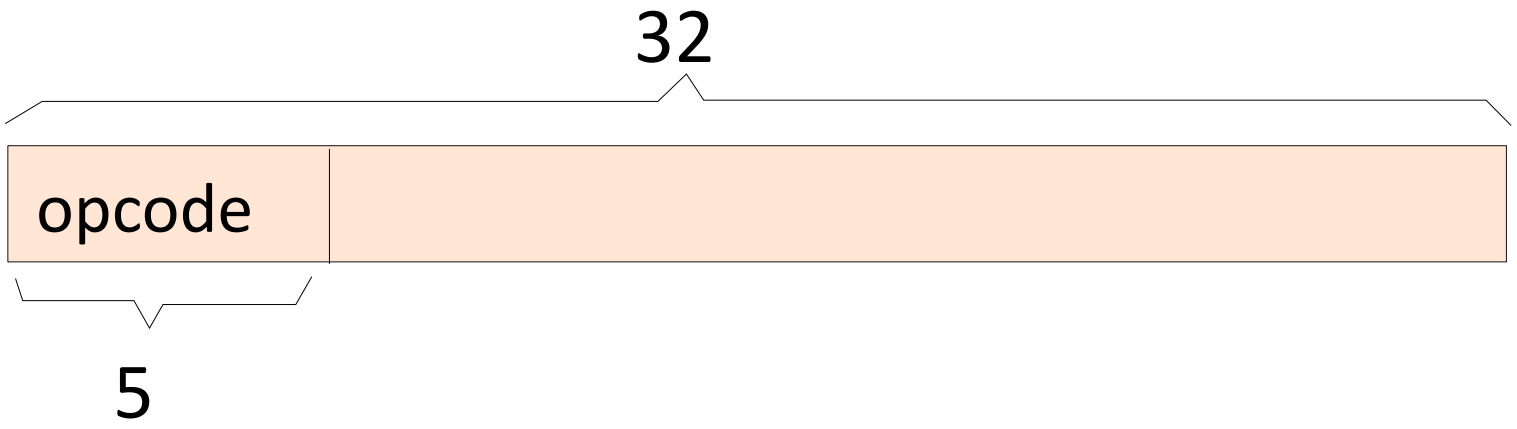
- Encode the SimpleRisc ISA using 32 bits.
- We have 21 instructions. Let us allot each instruction an unique code (opcode)



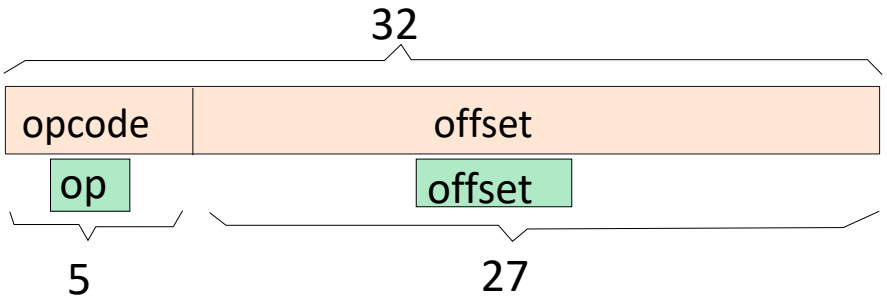
Instruction	Code	Instruction	Code	Instruction	Code
add	00000	not	01000	beq	10000
sub	00001	mov	01001	bgt	10001
mul	00010	lsl	01010	b	10010
div	00011	lsr	01011	call	10011
mod	00100	asr	01100	ret	10100
cmp	00101	nop	01101		
and	00110	ld	01110		
or	00111	st	01111		

0-Address Instructions

- nop and ret instructions



1-Address Instructions

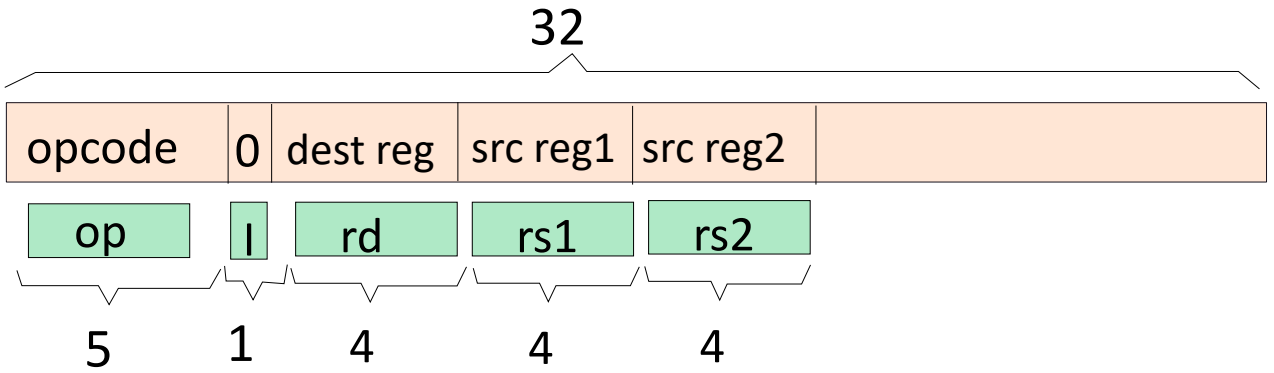


- Instructions – call, b, beq, bgt
- Use the branch format
- Fields :
  - 5 bit opcode
  - 27 bit offset (PC relative addressing)
  - Since the offset points to a 4 byte word address
    - The actual address computed is : PC + offset * 4

3-Address Instructions

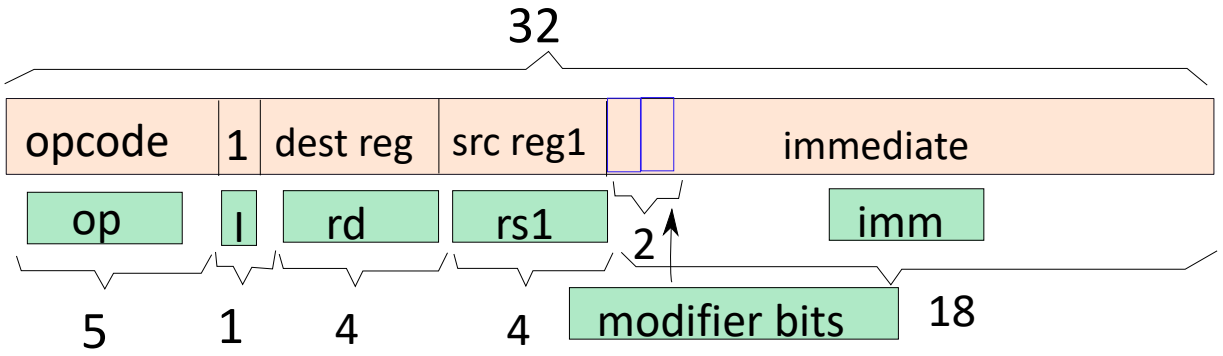
- Instructions – add, sub, mul, div, mod, and, or, lsl, lsr, asr
- Generic 3 address instruction
  - <opcode> rd, rs1, <rs2/imm>
- Let us use the I bit to specify if the second operand is an immediate or a register.
  - I = 0 → second operand is a register
  - I = 1 → second operand is an immediate
- Since we have 16 registers, we need 4 bits to specify a register

Register Format



- opcode → type of the instruction
- I bit → 0 (second operand is a register)
- dest reg → rd
- source register 1 → rs1
- source register 2 → rs2

Immediate Format

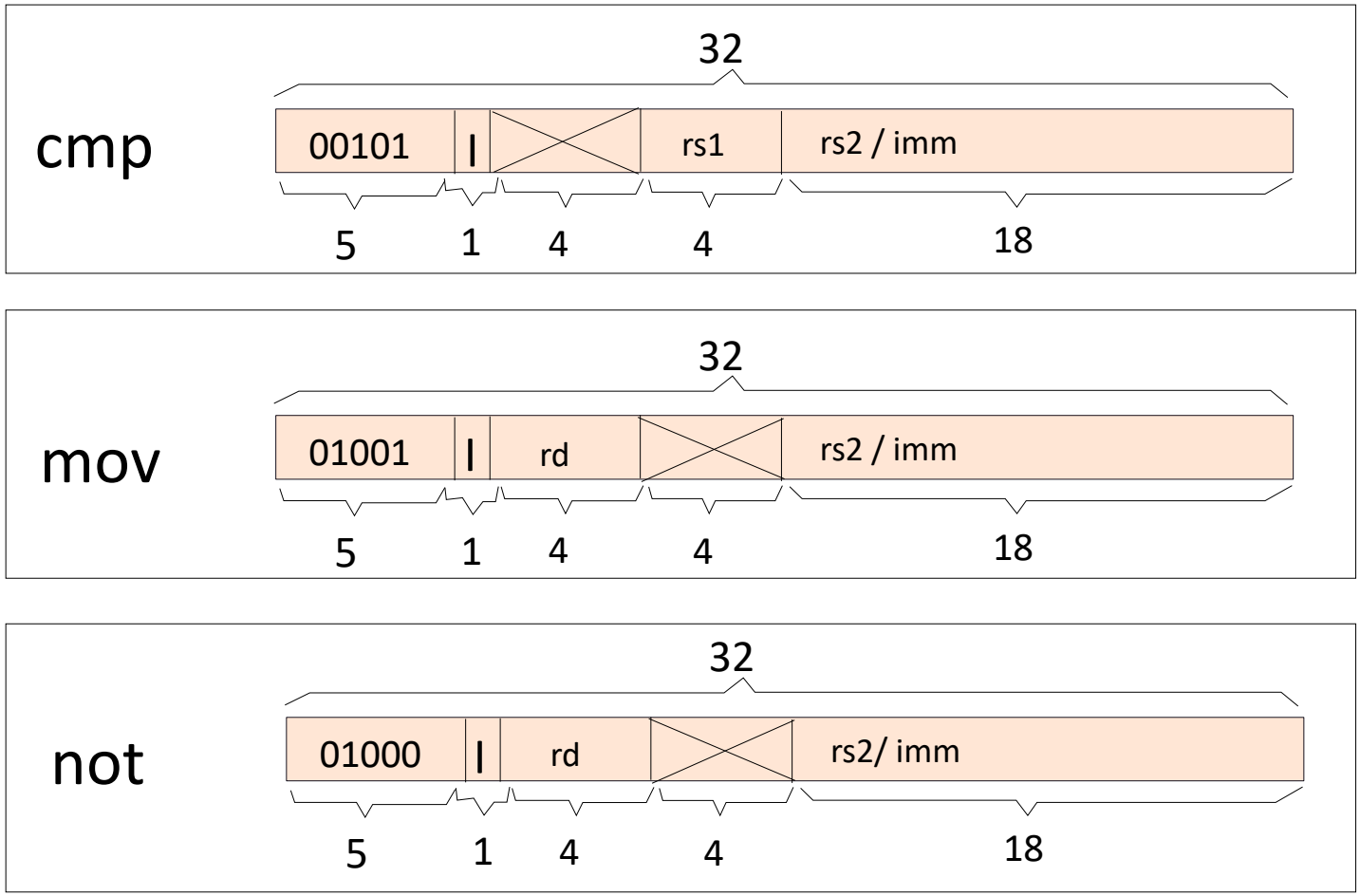


- opcode → type of the instruction
- I bit → 1 (second operand is an immediate)
- dest reg → rd
- source register 1 → rs1
- Immediate → imm
- modifier bits → 00 (default), 01 (u), 10 (h)

2 Address Instructions

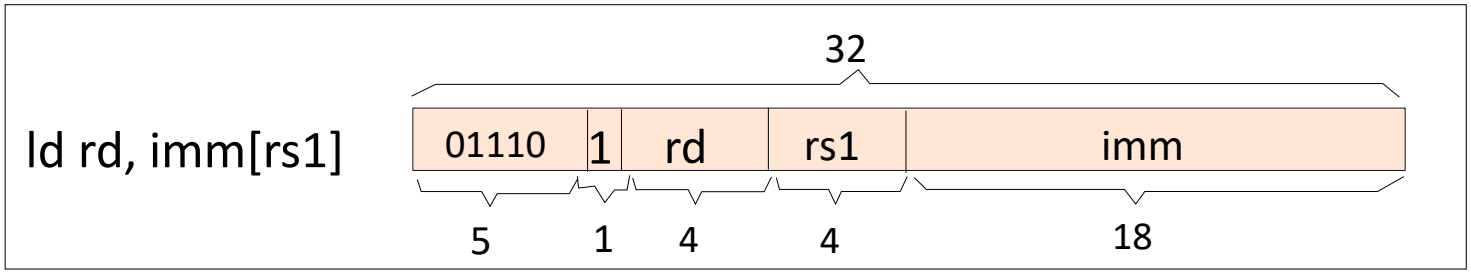
- cmp, not, and mov
- Use the 3 address : immediate or register formats
- Do not use one of the fields

cmp, not, and mov



Load and Store Instructions

- ld rd, imm[rs1]
- rs1 → base register
- Use the immediate format.

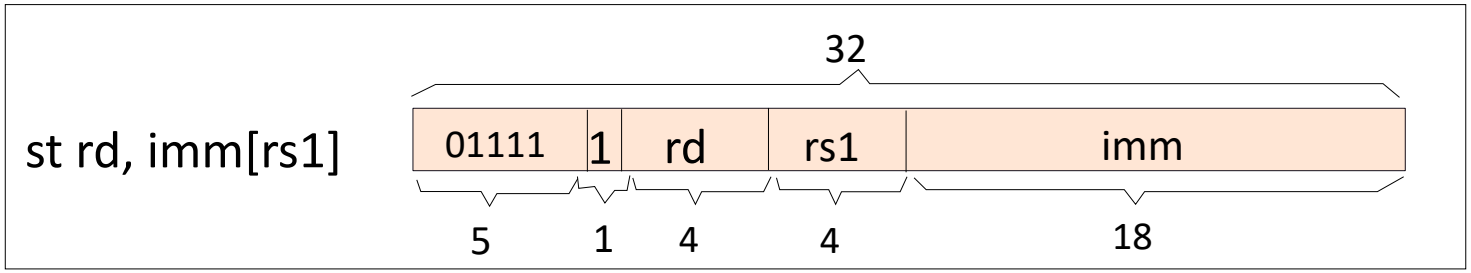


Store Instruction

- Strange case of the store inst.
- st reg1, imm[reg2]
- has two register sources, no register destination, 1 immediate
- Cannot fit in the immediate format, because the second operand can be either be a register OR an immediate (not both)
- Should we define a new format for store instructions ?

Store Instruction

- Let us make an exception and use the immediate format.
- We use the rd field to save one of the source registers
- st rd, imm[rs1]



Summary of Instruction Formats

Format	Definition				
branch	*op* (28-32)	*offset* (1-27)			
register	*op* (28-32)	I (27)	*rd* (23-26)	*rs* 1(19-22)	*rs* 2(15-18)
immediate	*op* (28-32)	I (27)	*rd* (23-26)	*rs* 1(19-22)	*imm* (1-18)
op → opcode, *offset* → branch offset, *I* → immediate bit, *rd* → destination register					
*rs*1 → source register 1, *rs*2 → source register 2, *imm* → immediate operand					

- branch format → nop, ret, call, b, beq, bgt
- register format → ALU instructions
- immediate format → ALU, ld/st instructions