Happy Friday!!

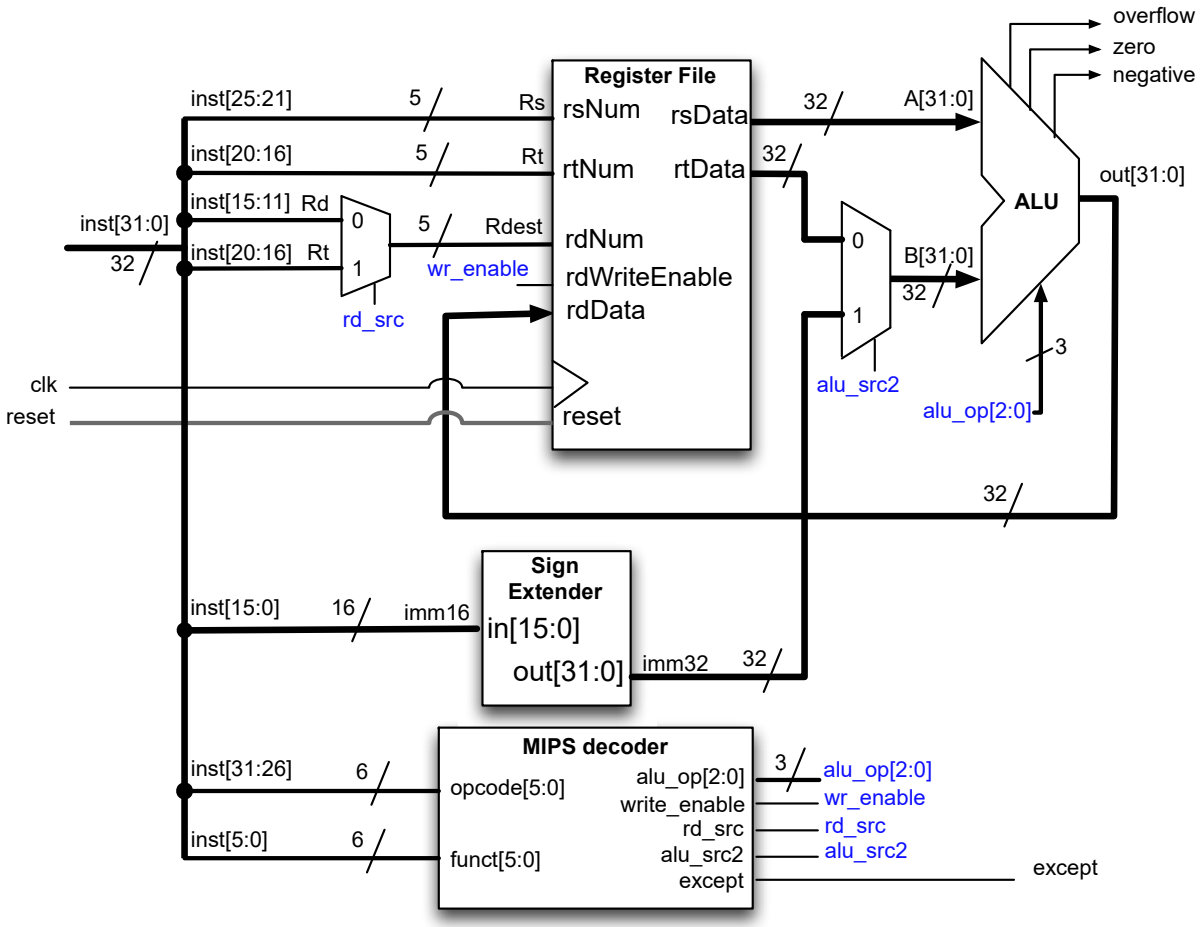
Instruction Decoding

Example Control and Datapath problem on Piazza

Sign up for Second chance exam 1

Bring MIPS “Green Sheet” to lecture and discussion

By the End of Today's Lecture

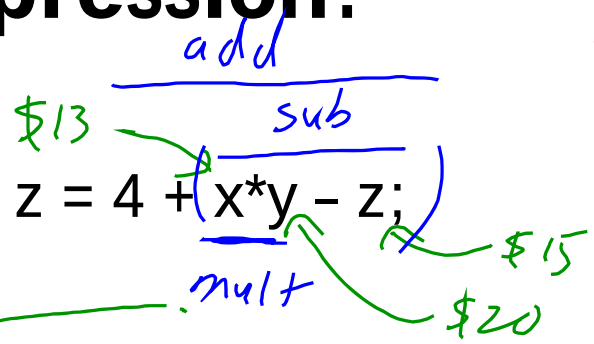


Today's lecture

- Instruction Encoding
  - R-type & I-type encodings
- Instruction Decoding
  - Operands
  - Sign-extending the immediate
  - Decoding the ALU operation

How can we write MIPS code to compute the following expression?

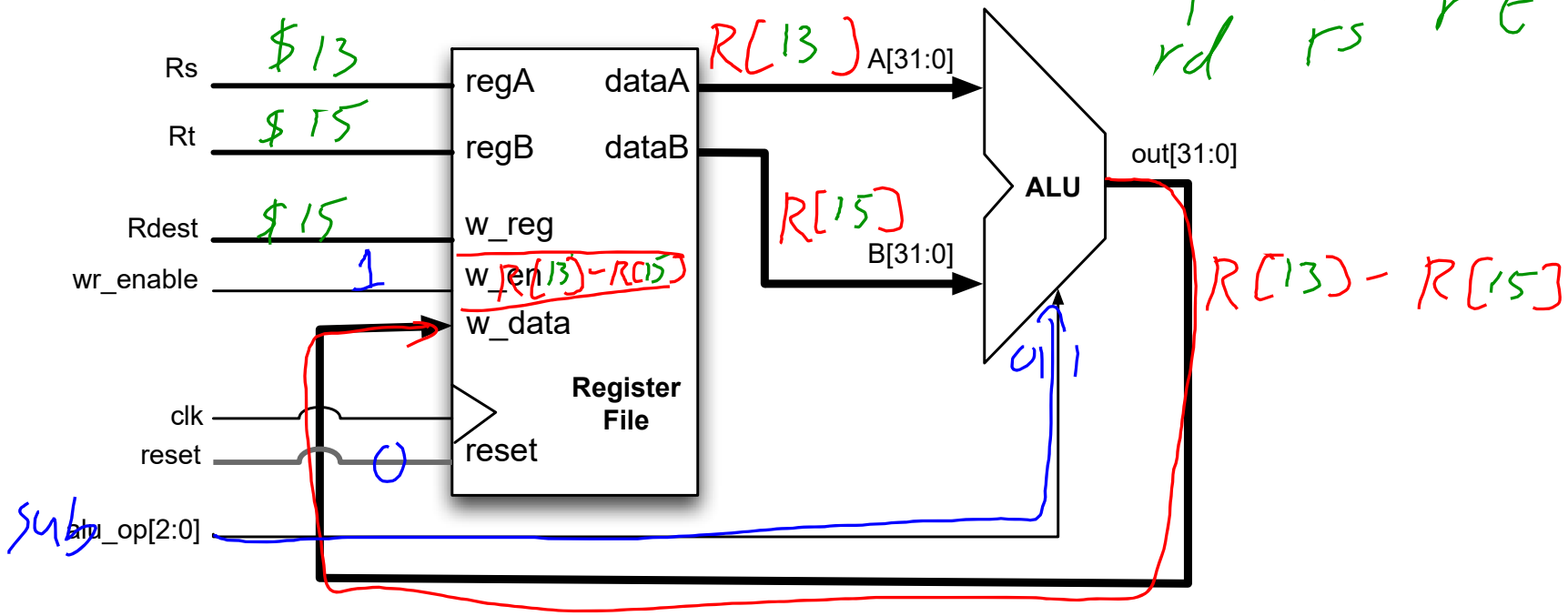
operation dest, src1, src2



- Assume the following register allocation:
  - $13 = x, $20 = y, $15 = z

```
mult $13, $13, $20    # x*y
sub  $15, $13, $15    # (x*y) - z
add  $15, $15, 4
```

How do instructions control the datapath?

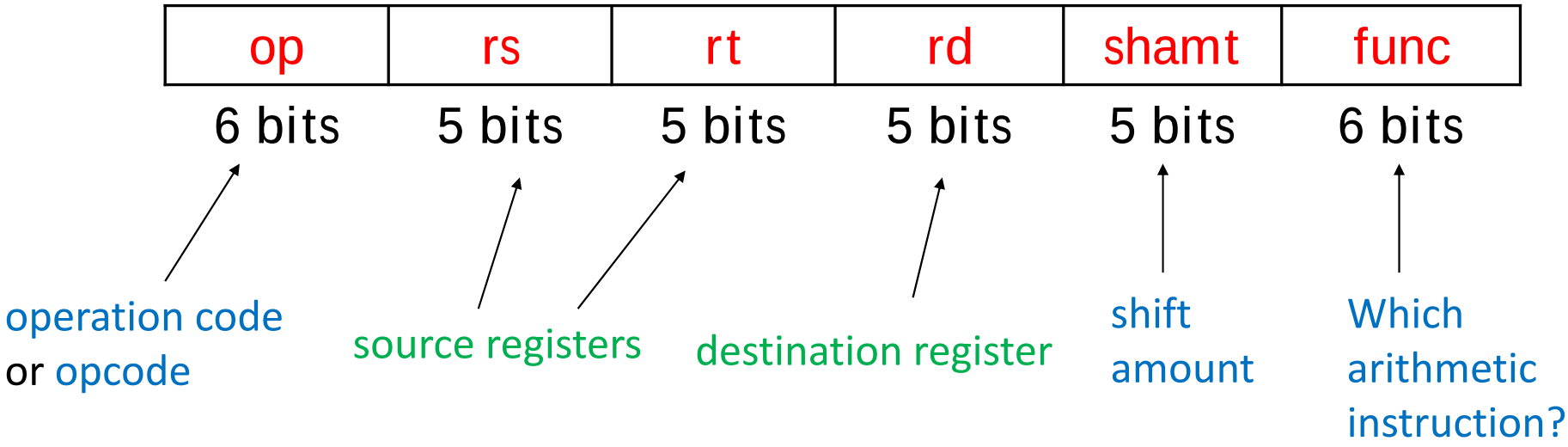


- First step is to learn how instructions are encoded

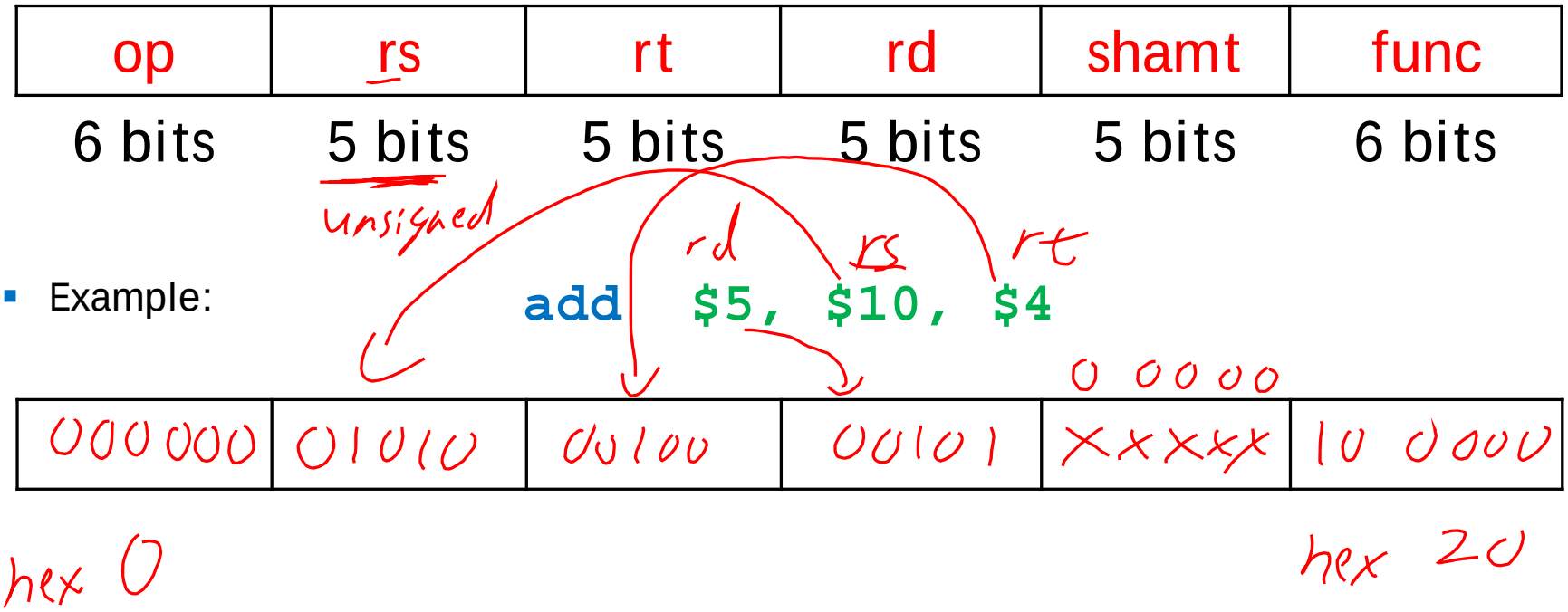
Machine language is a binary format that can be stored in memory

- MIPS machine language is easy to decode
  - Each MIPS instruction is 32 bits wide
  - There are three instruction formats

Register-to-register arithmetic instructions use the R-type format

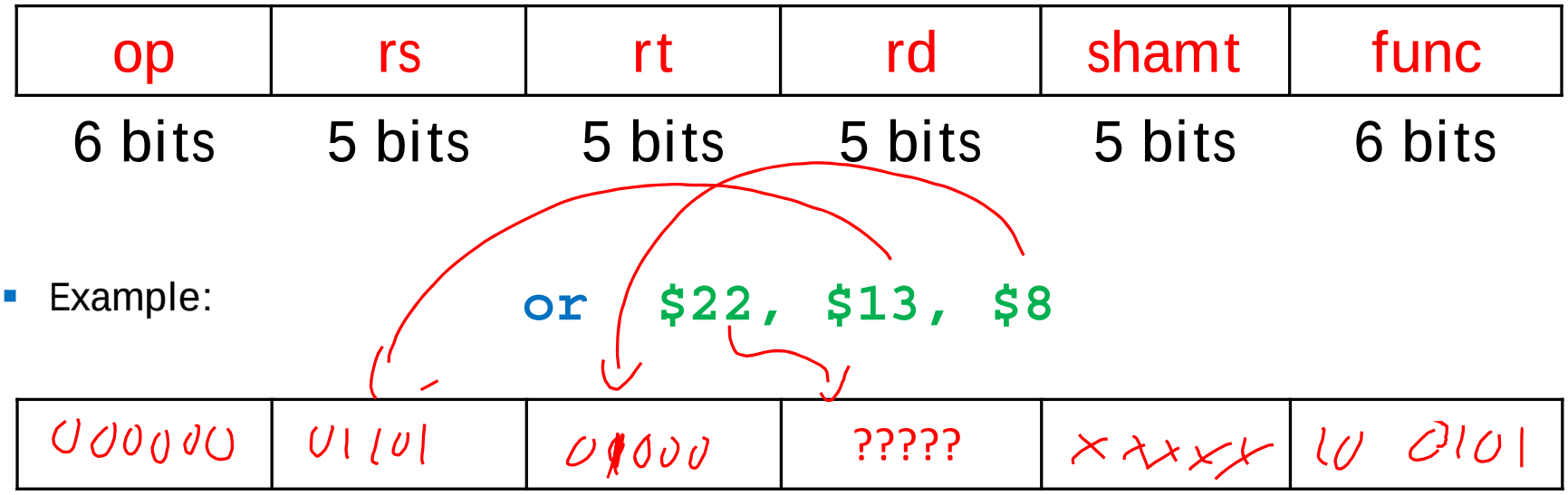


Register-to-register arithmetic instructions use the R-type format



- Example:

Register-to-register arithmetic instructions use the R-type format



- Example:

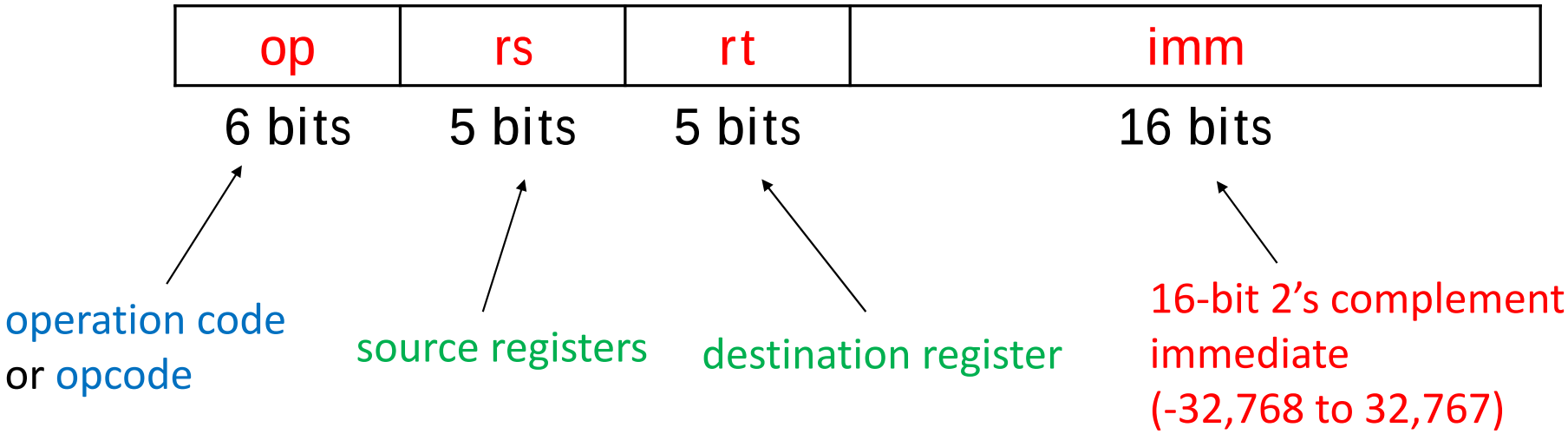
```
or $22, $13, $8
```


x0

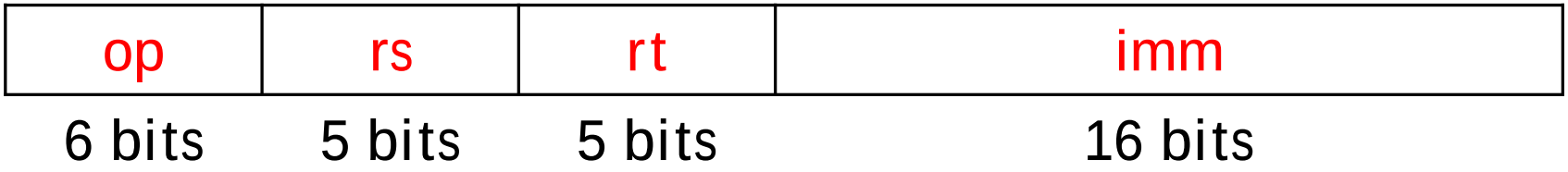
a) xxxxx
b) 01000
c) 01101
d) 10110

x 25

Instructions with immediates all use the I-type format.



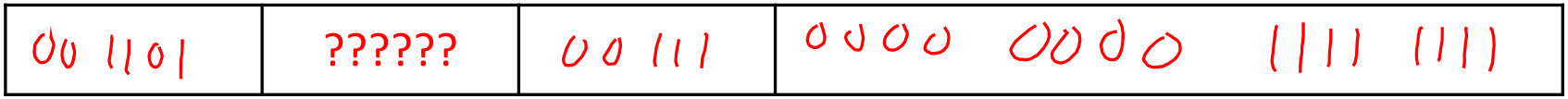
Instructions with immediates all use the I-type format.



- Example

```
ori  $7, $2, 0xff
```

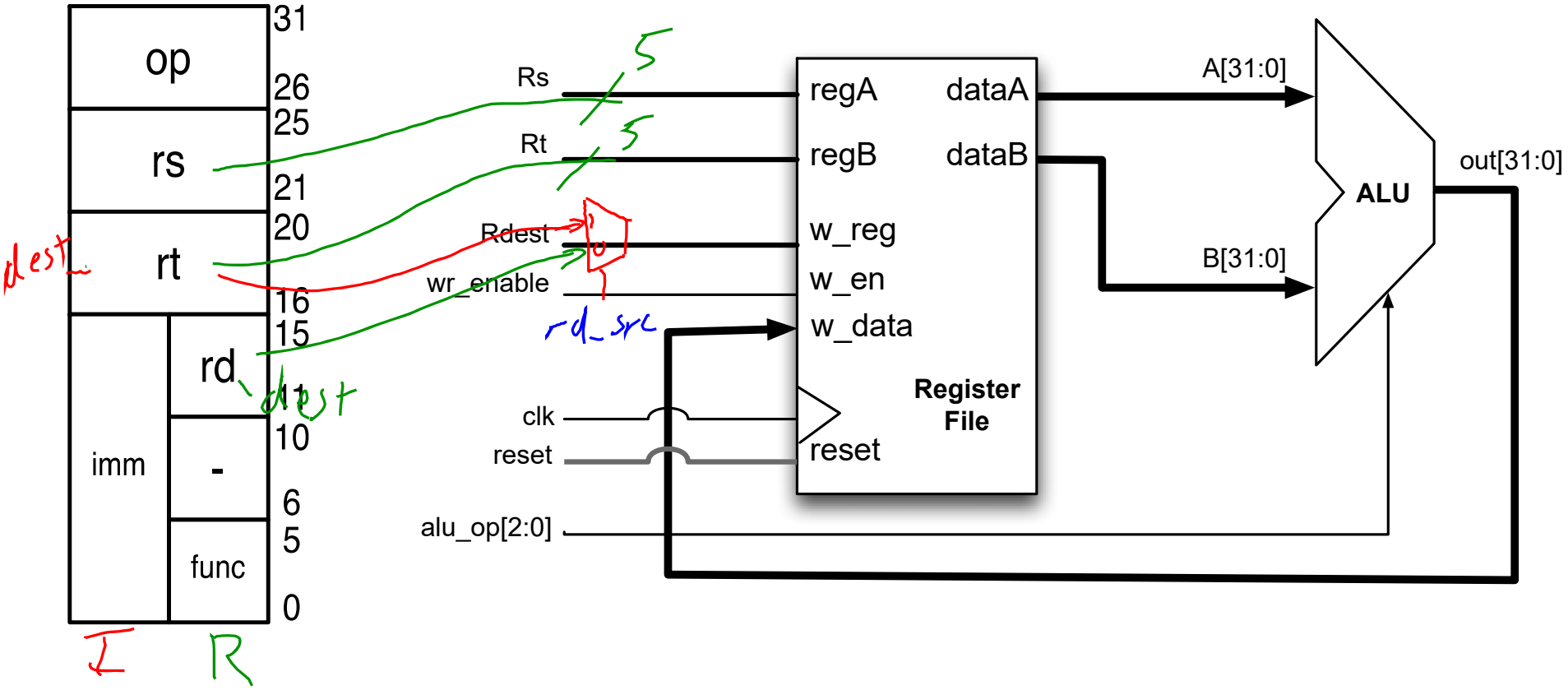
rt rs imm



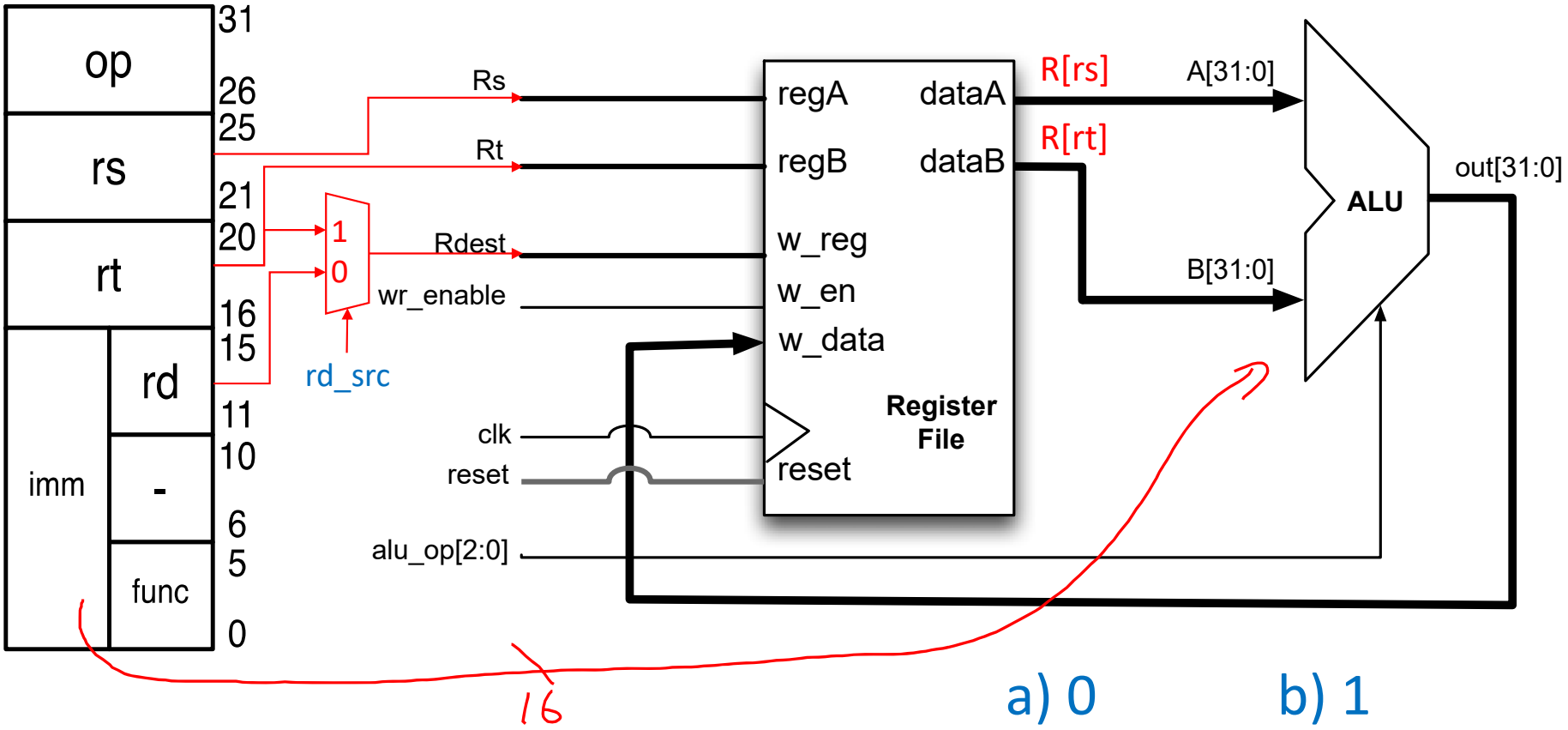
x d

a) 01101
b) 00111
c) 00010
d) 11111

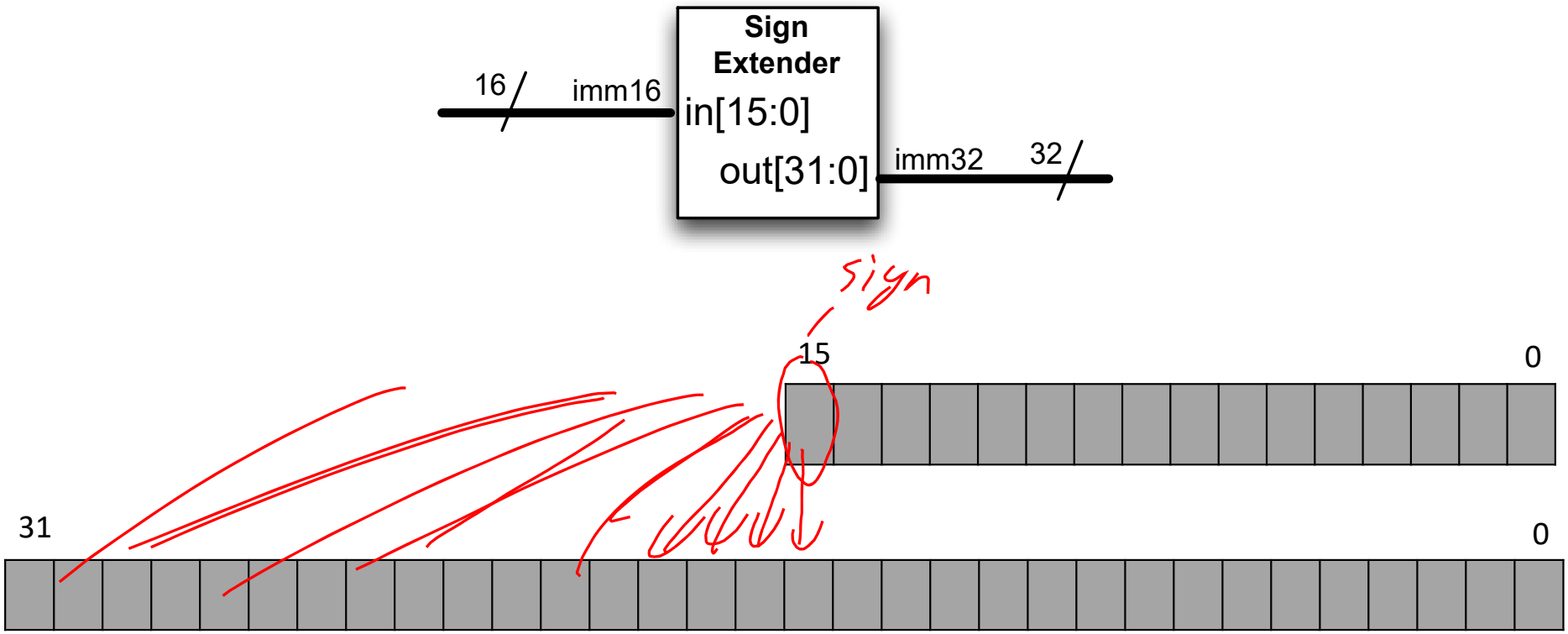
Some control signals are encoded in the instruction



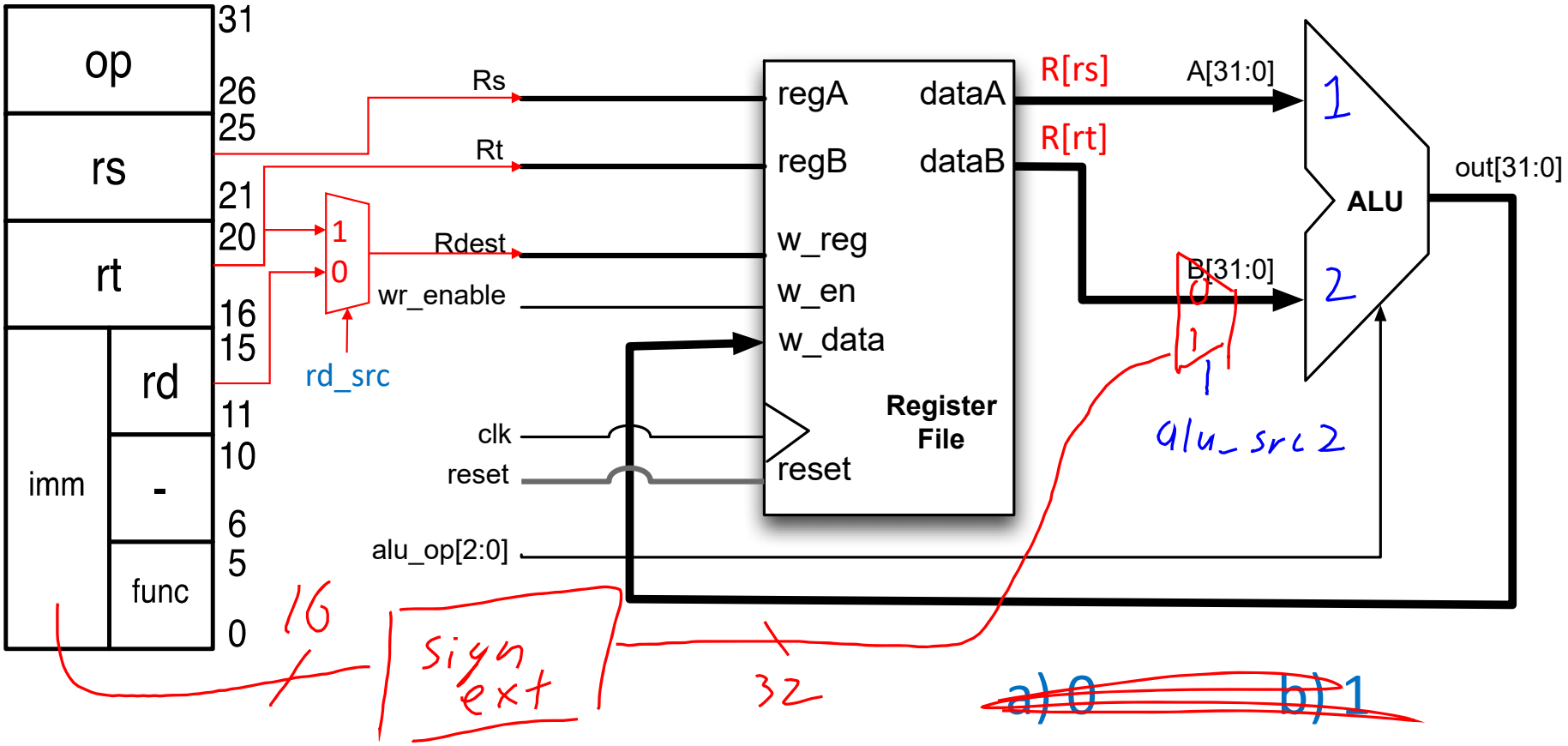
If we have an I-type instruction, what should the control signal rd_src be?



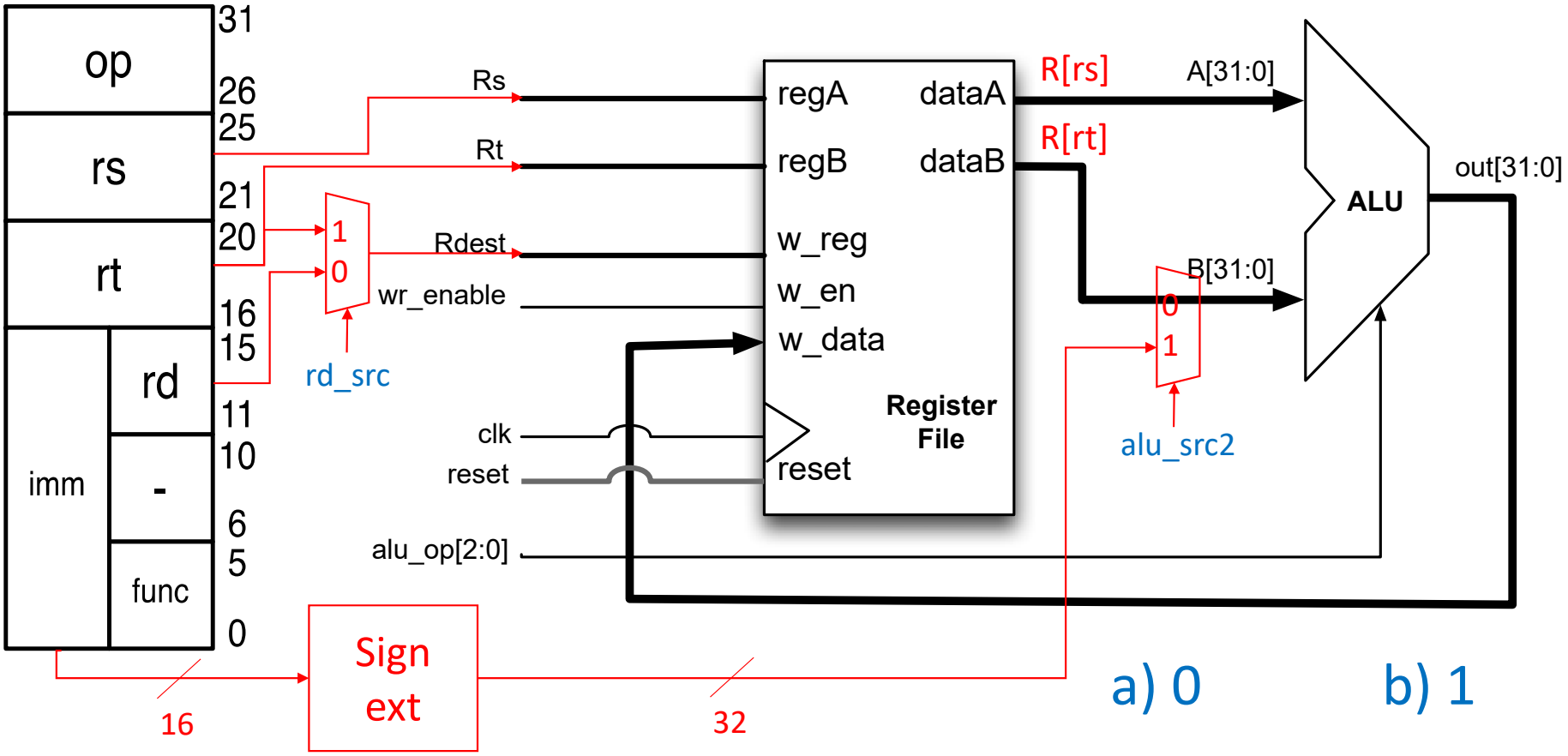
Sign Extension replicates the MSb of imm16



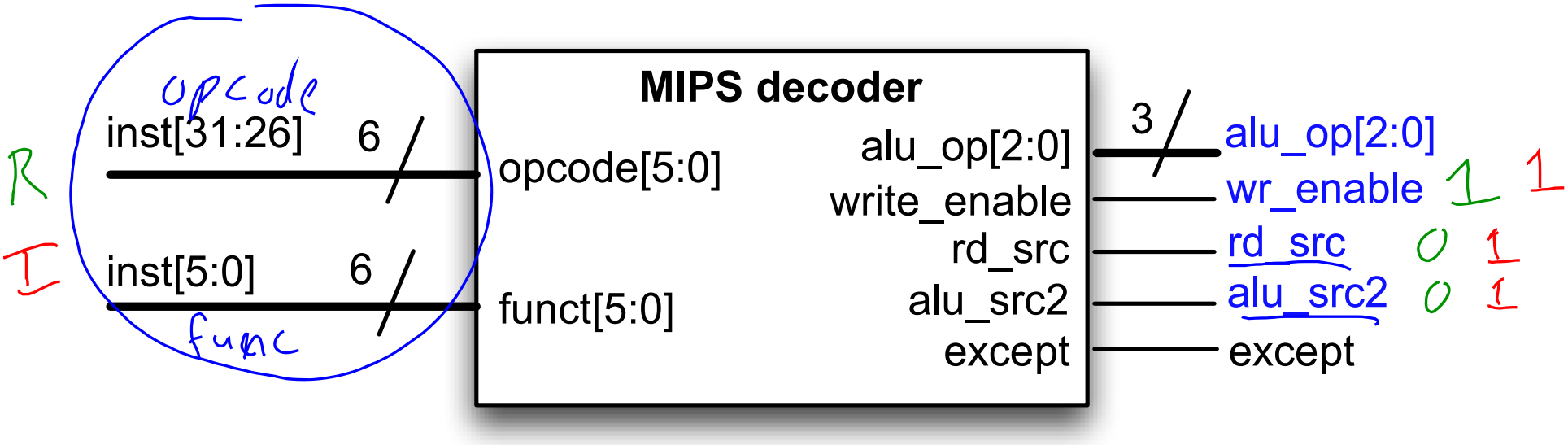
Select behavior of the ALU B input based on instruction type



If we have an R-type instruction, what should the control signal alu_src2 be?



The instruction decoder translates bits from the instruction into control signals



Use a table to decode instructions into control signals

	Instruction	opcode	func	alu_op	rd_src	alu_src2	wr_enable
		Code		Control signals			
R	add	000000	10 0000	010	0	0	1
	sub	000000	10 0010	011	0	0	1
	and			:	0	0	1
	or				0	0	1
	nor			:	0	0	1
	xor				0	0	1
I	addi	001000	??????	010	1	1	1
	andi						
	ori						
	xori						

Arithmetic Machine Datapath