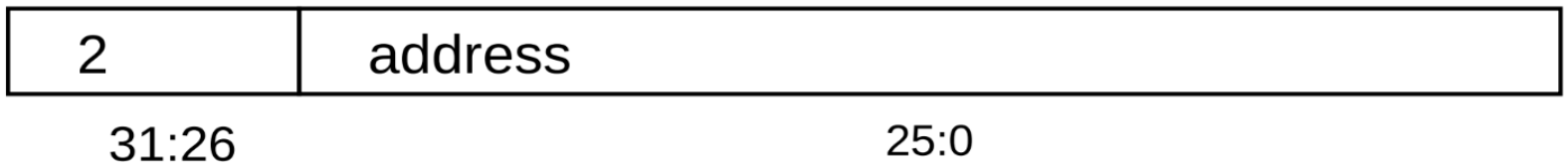
CENG 311
Computer Architecture

LAB 9

Instruction Addition to MIPS Single Cycle Datapath

Jump-Jal-Jr

Implementing Jumps



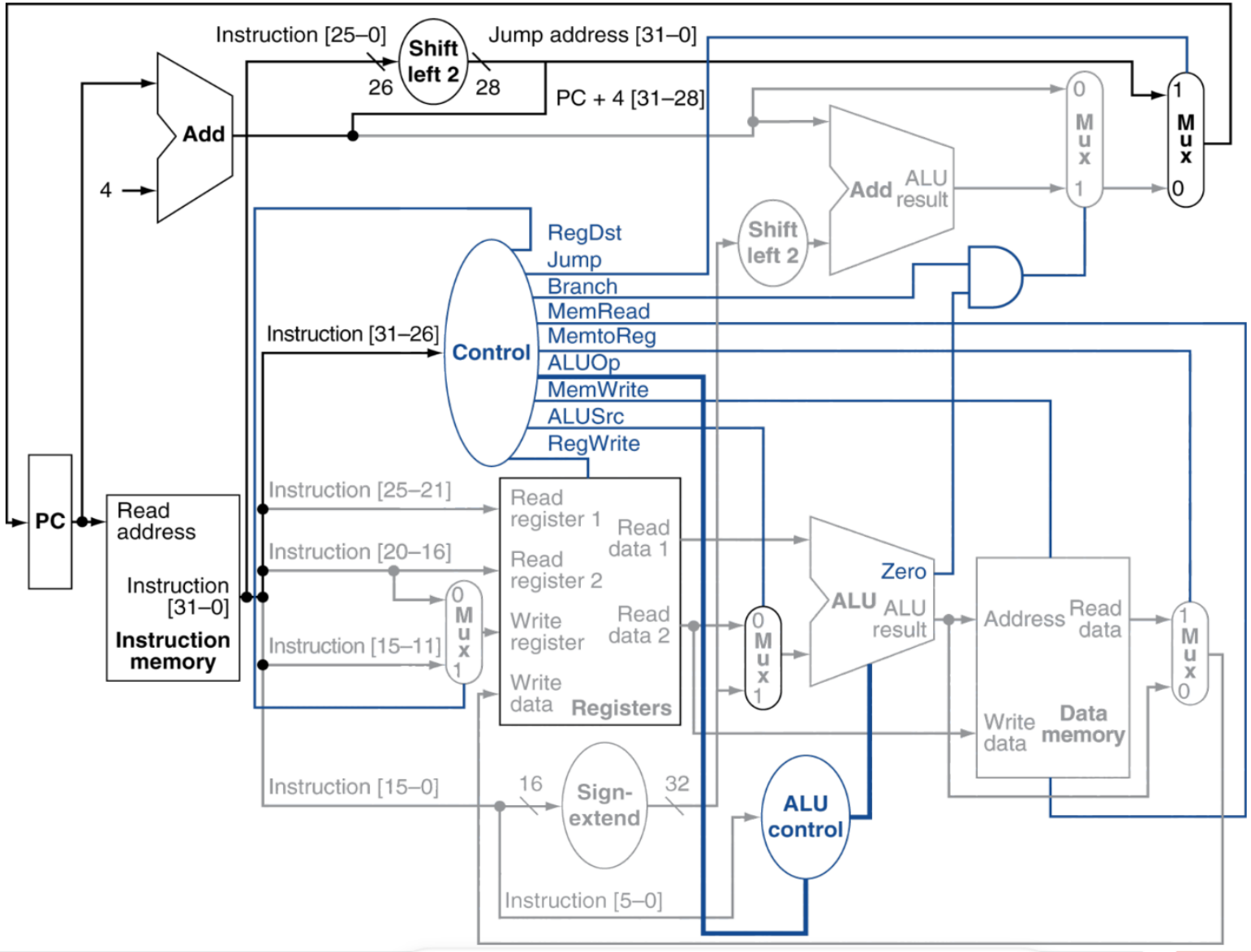
Jump uses word address

Update PC with concatenation of

- Top 4 bits of old PC
- 26-bit jump address
- The bits 00 (shift left 2)

Need an extra control signal decoded from opcode

Datapath with Jumps

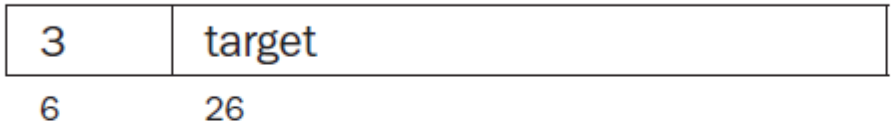


Implementing jal (jump and link)

- Saves the return address **(PC+4)** to $ra register before jumping to the target address.

Jump and Ilnk

`jal target`

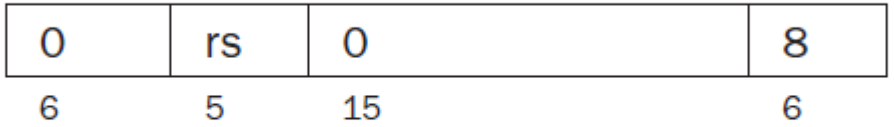


Unconditionally jump to the instruction at target. Save the address of the next instruction in register `$ra`.

Implementing jr (jump register)

Jump register

`jr rs`



Unconditionally jump to the instruction whose address is in register `rs`.

`jr $ra` (register 31)

op	rs	rt	rd	shamt	funct
0	31	0	0	0	8
000000	11111	00000	00000	00000	001000