The Processor

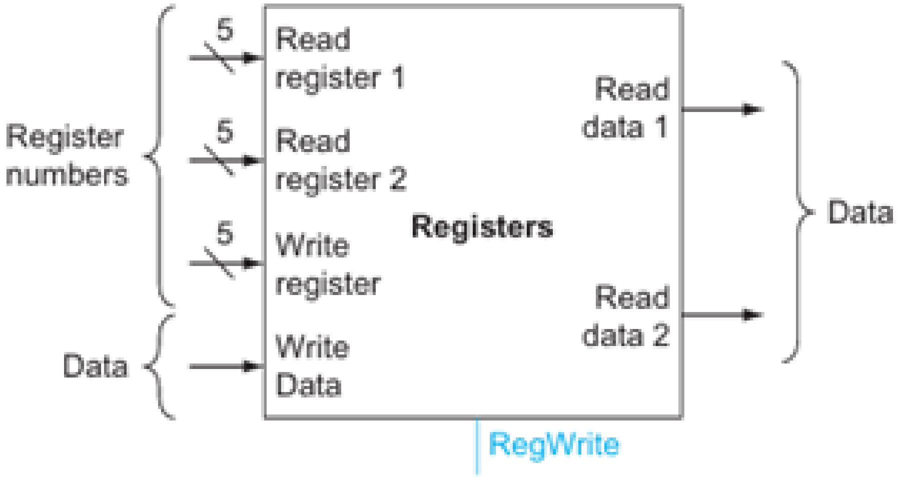
Dr. Rajib Ranjan Maiti

CSIS, BITS-Pilani, Hyderabad

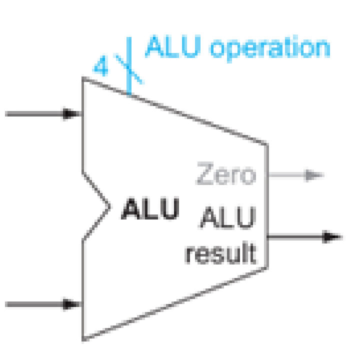
Building a Datapath: Execution of Load and Store

- lw $1, 100($0)
- sw $t3, 12($t0)

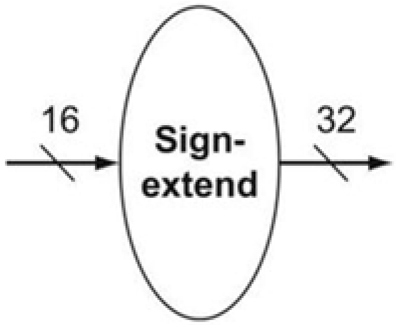
- Four MIPS elements needed



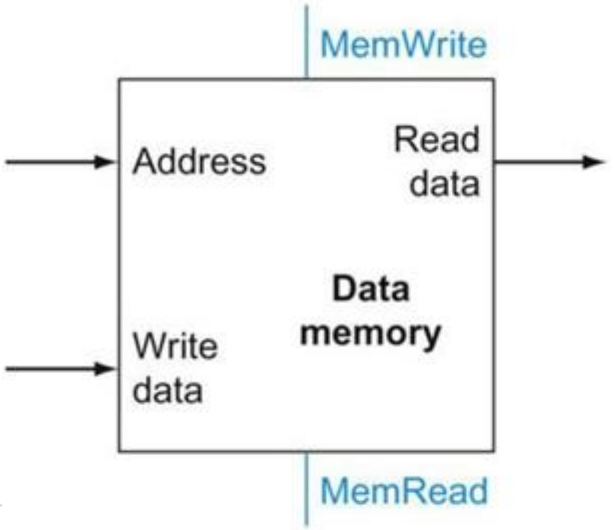
the register file



the ALU

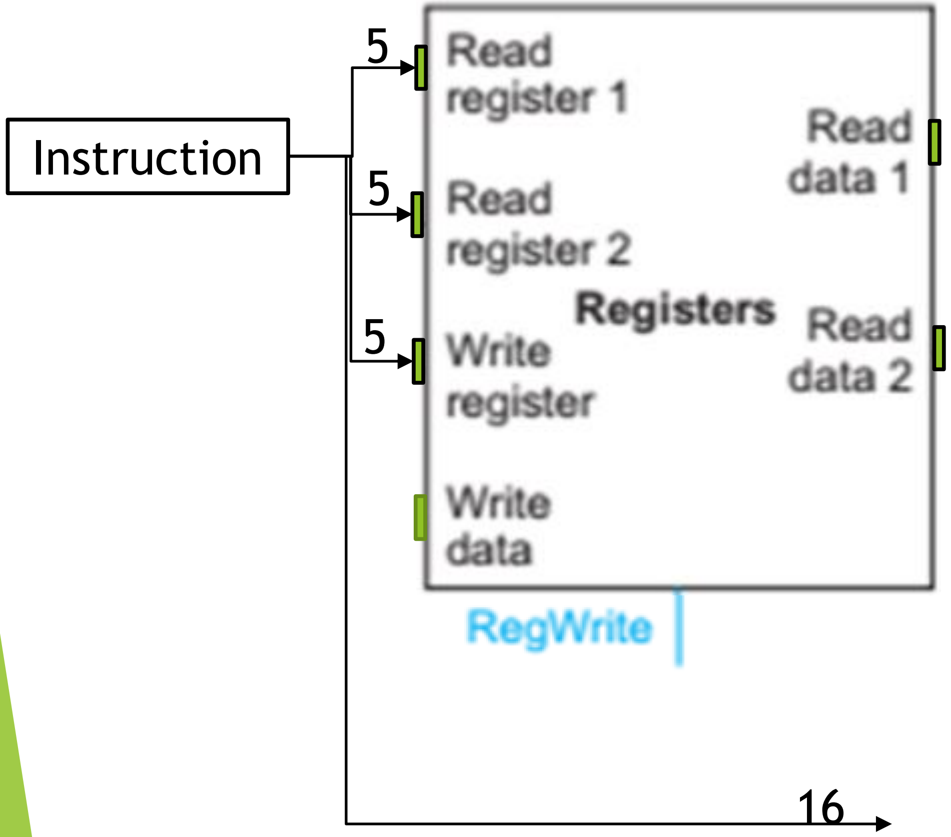


a unit to sign-extend the 16-bit off set field in the instruction to a 32-bit signed value

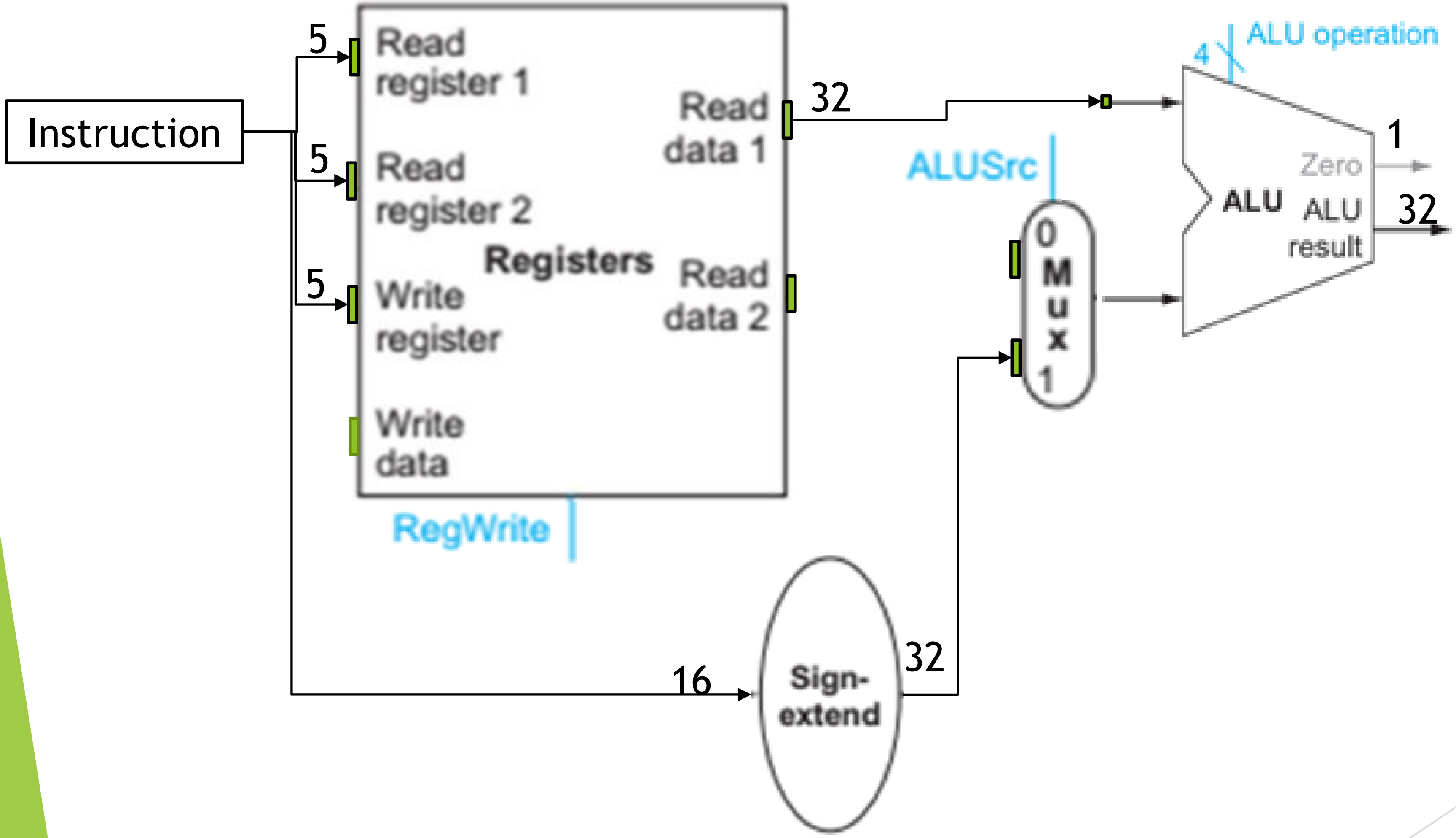


a data memory unit

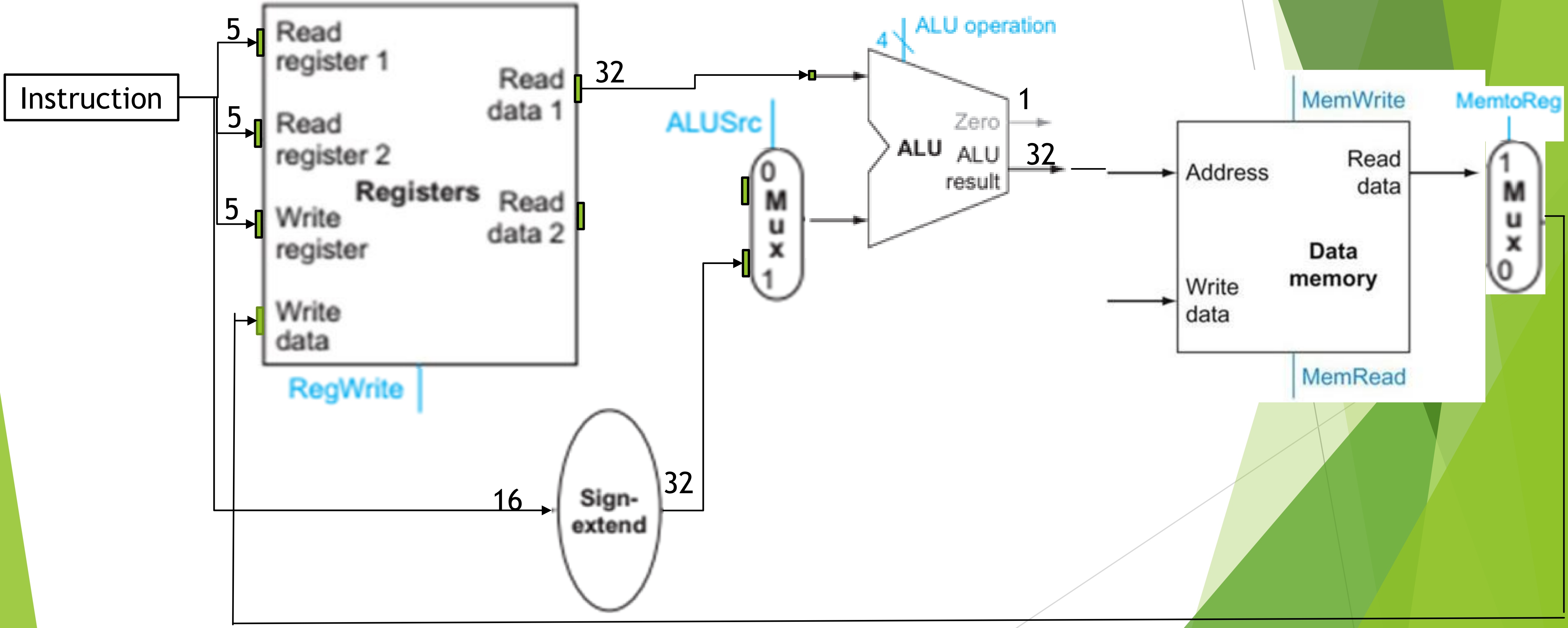
Datapath: memory instructions



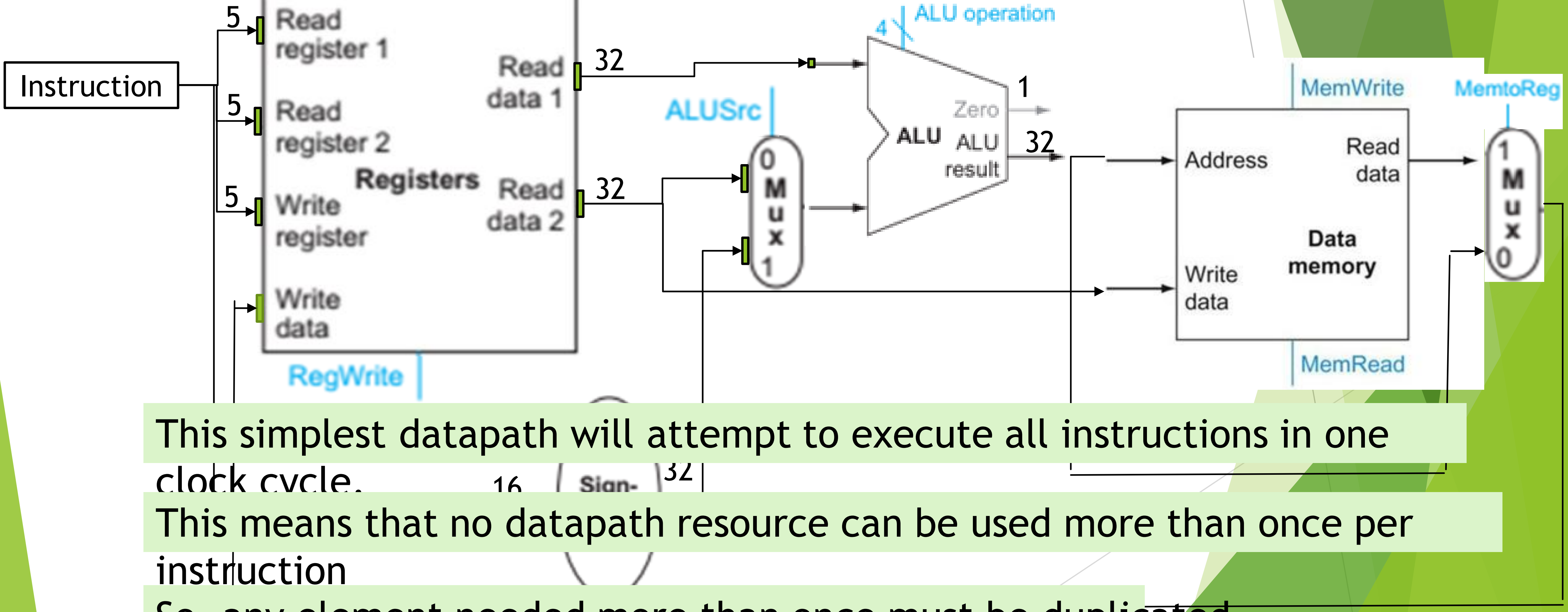
Datapath: memory instructions



Datapath: memory instructions



Datapath: memory instructions



This simplest datapath will attempt to execute all instructions in one clock cycle.

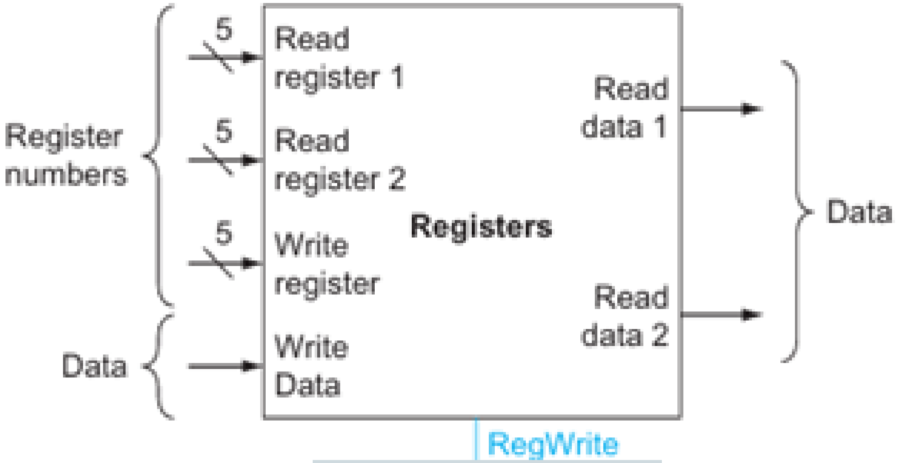
This means that no datapath resource can be used more than once per instruction

So, any element needed more than once must be duplicated

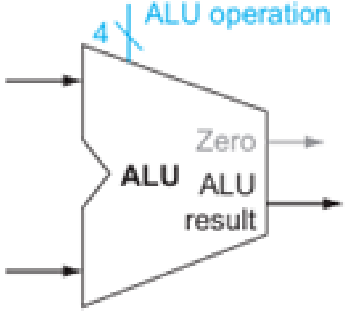
Building a Datapath: Branch instruction

Building a Datapath: Branch instruction

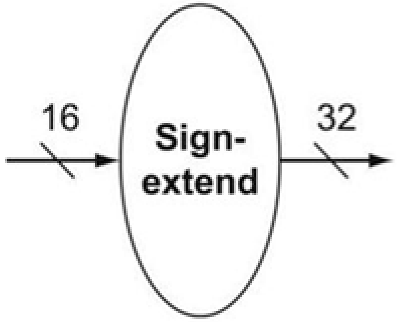
- beq $t1,$t2,offset
- Five MIPS elements needed



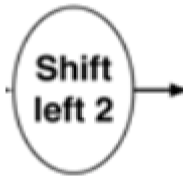
the register file

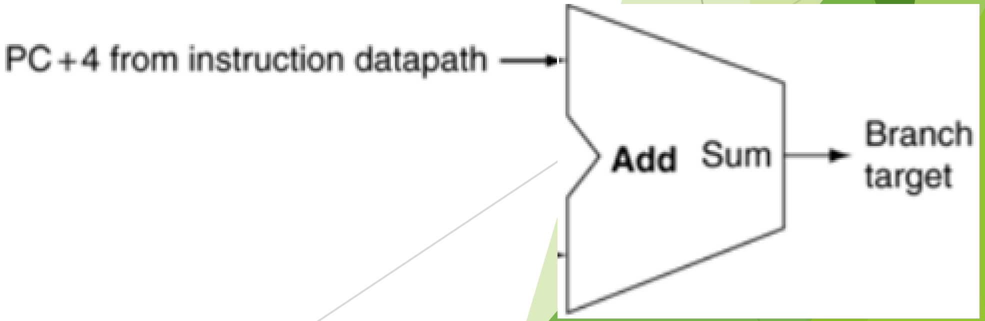


the ALU

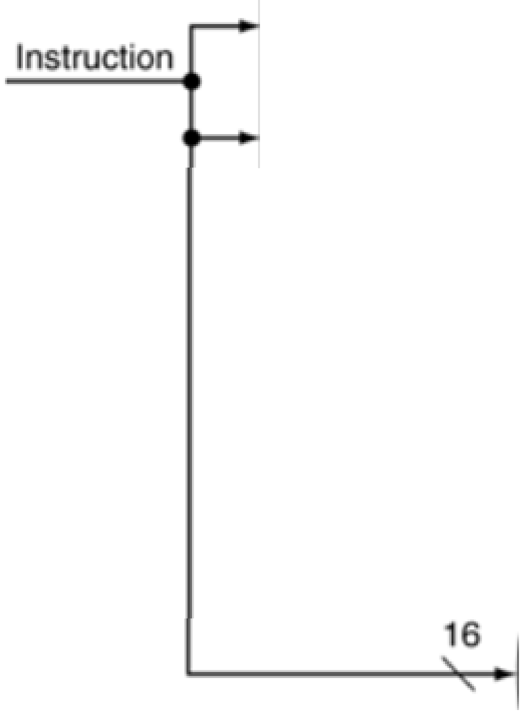


a unit to sign-extend the 16-bit off set field in the instruction to a 32-bit signed value





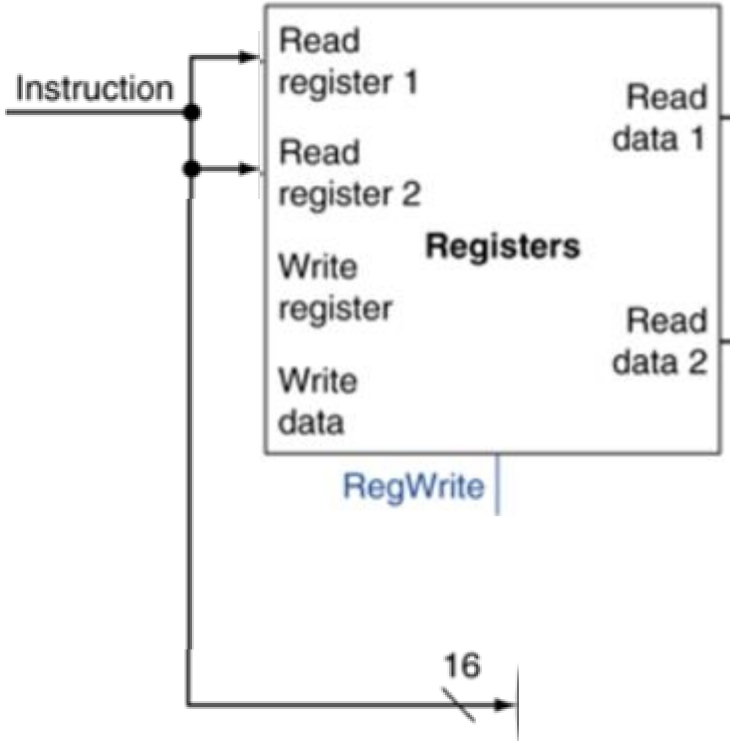
Building a Datapath: Branch instruction



To Compute Label:

Temp1 = Branch Target – (PC + 4)
Temp2 = RightShift2 (temp1)
Label = SignReduce16(Temp2)

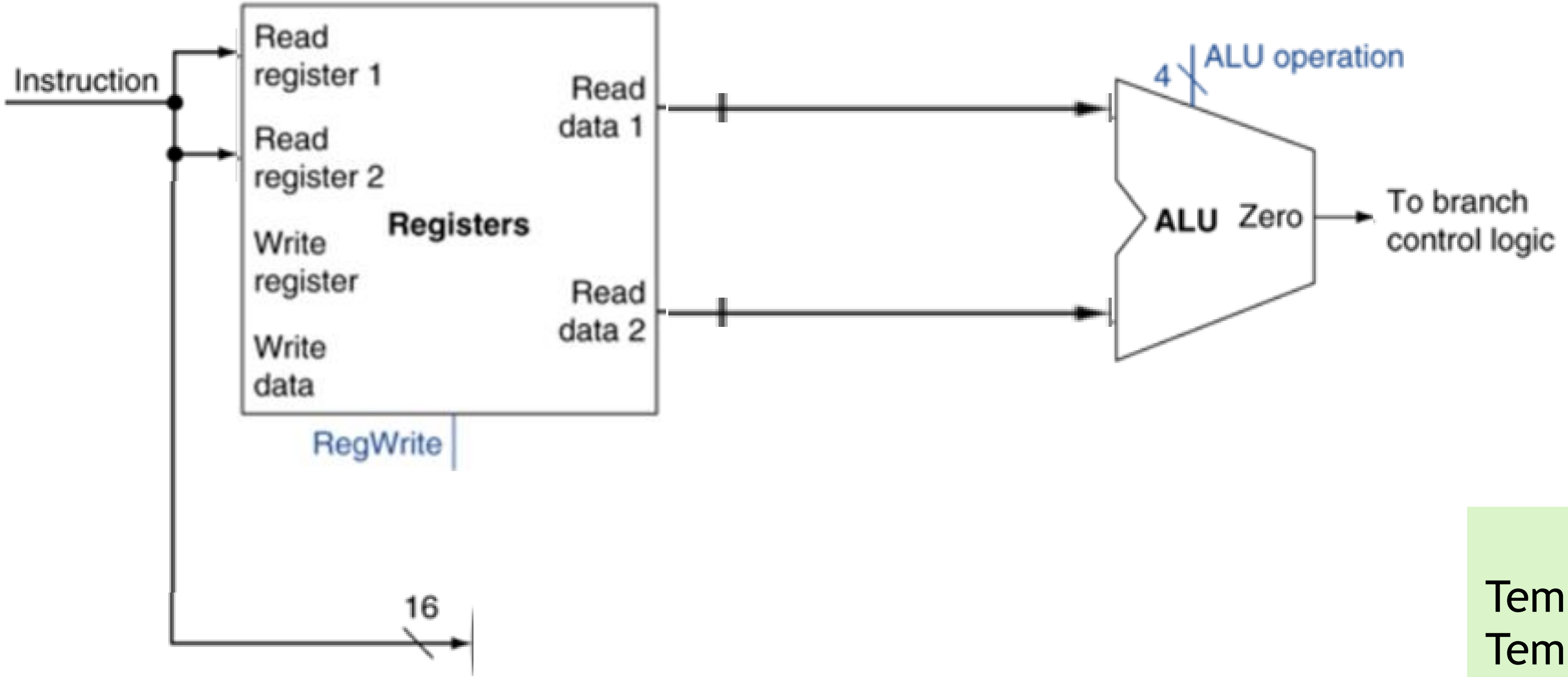
Building a Datapath: Branch instruction



To Compute Label:

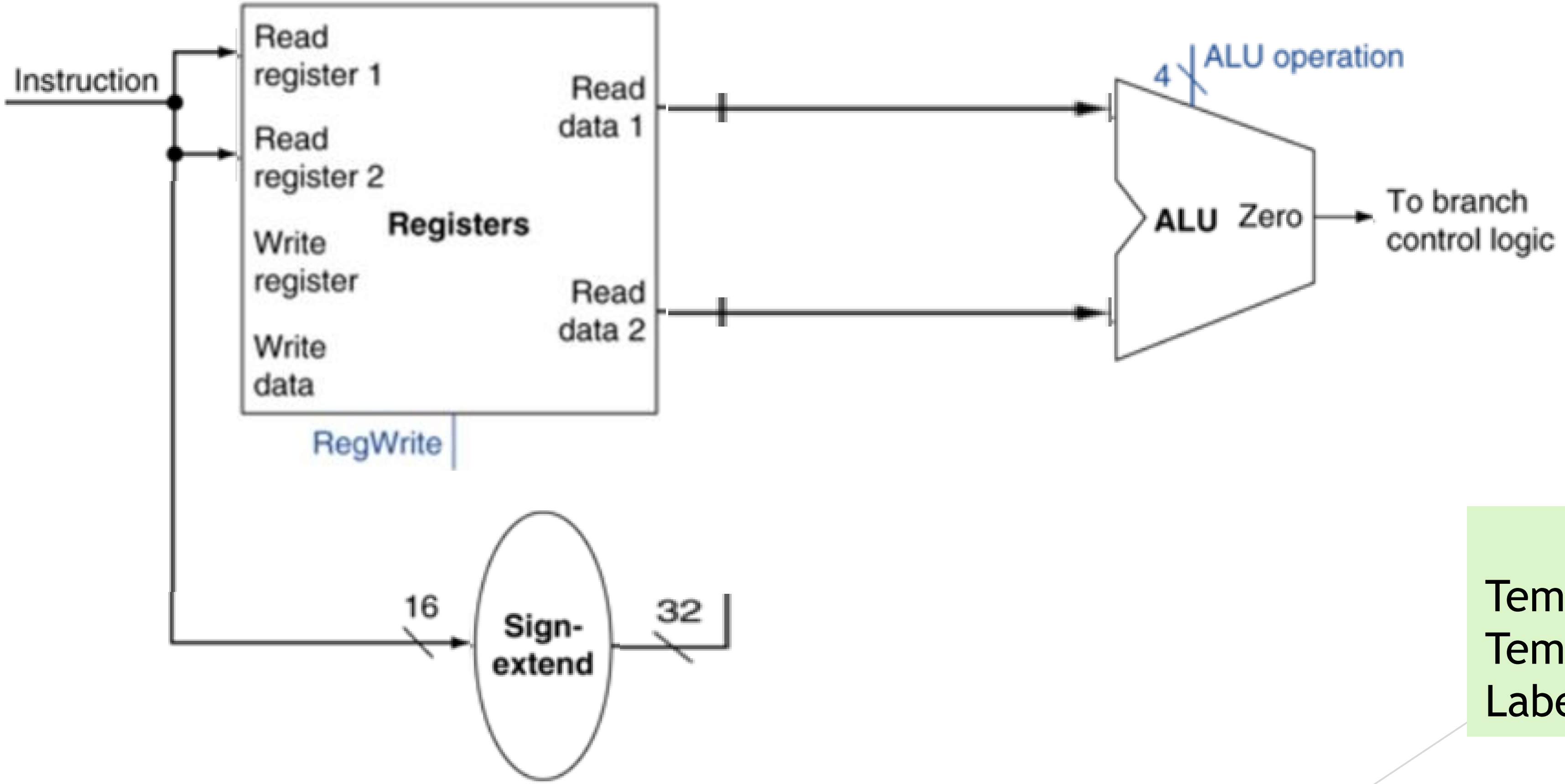
Temp1 = Branch Target – (PC + 4)
Temp2 = RightShift2 (temp1)
Label = SignReduce16(Temp2)

Building a Datapath: Branch instruction



To Compute Label:
Temp1 = Branch Target – (PC + 4)
Temp2 = RightShift2 (temp1)
Label = SignReduce16(Temp2)

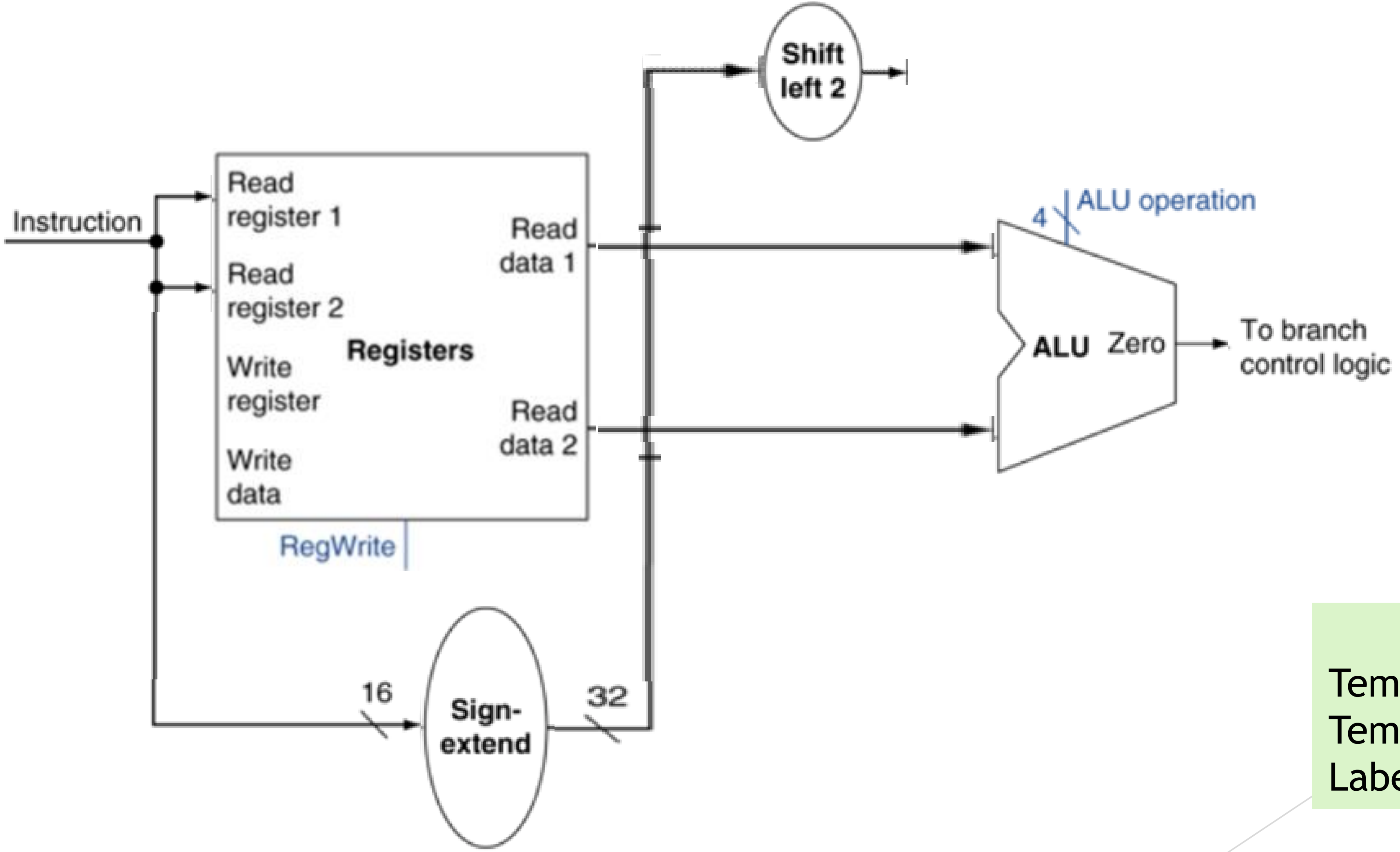
Building a Datapath: Branch instruction



To Compute Label:

Temp1 = Branch Target – (PC + 4)
Temp2 = RightShift2 (temp1)
Label = SignReduce16(Temp2)

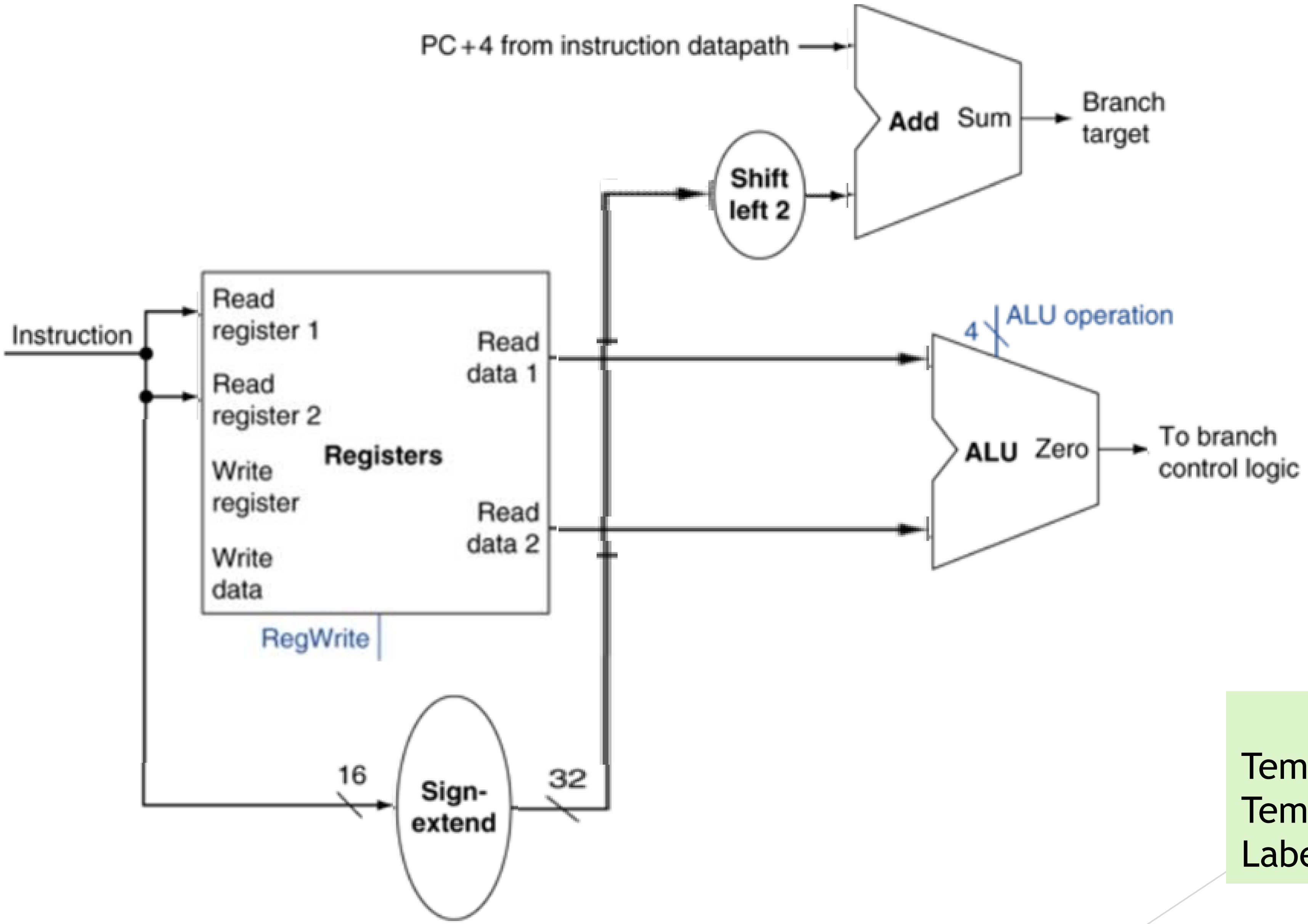
Building a Datapath: Branch instruction



To Compute Label:

Temp1 = Branch Target – (PC + 4)
Temp2 = RightShift2 (temp1)
Label = SignReduce16(Temp2)

Building a Datapath: Branch instruction



To Compute Label:

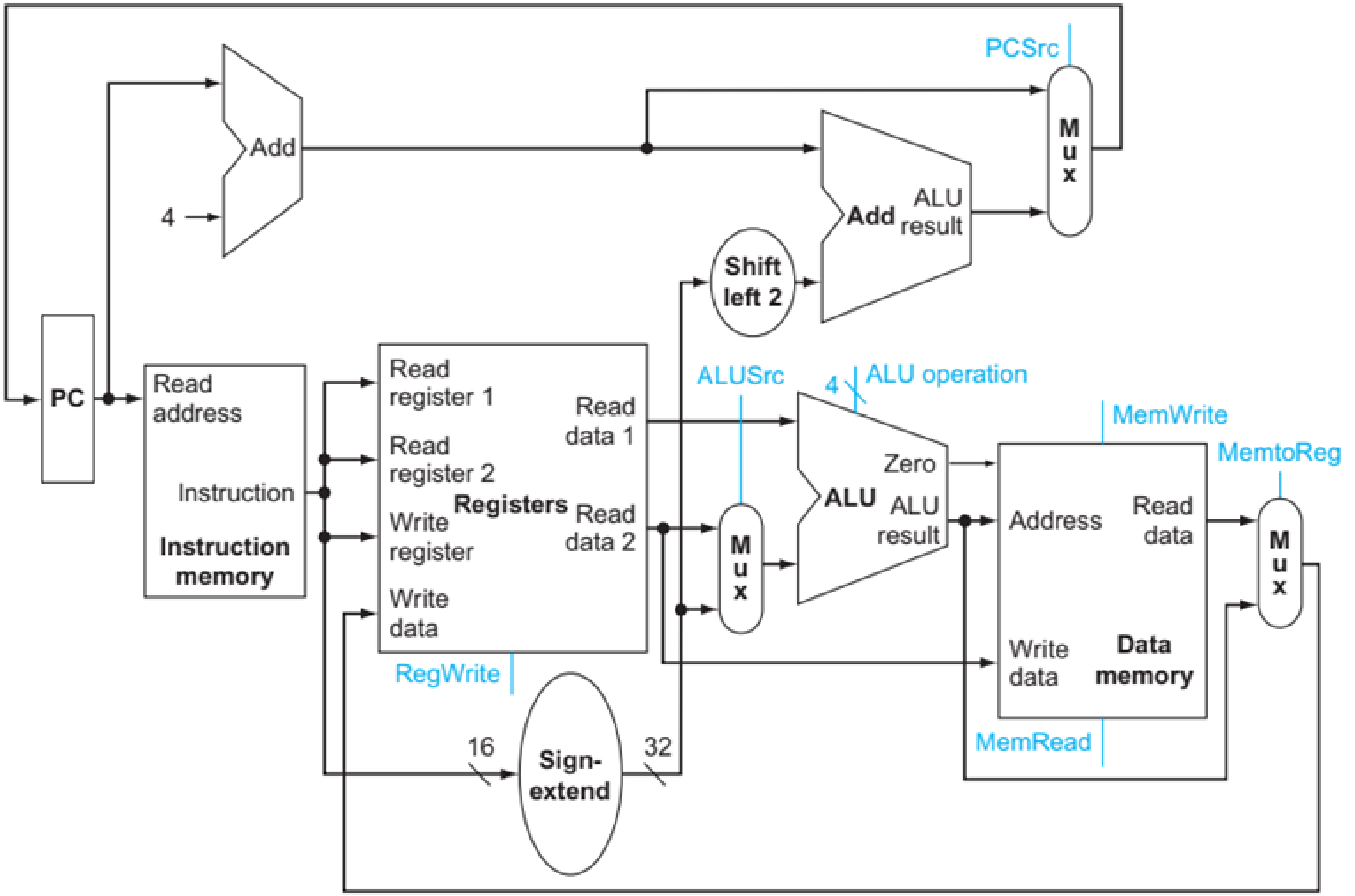
Temp1 = Branch Target – (PC + 4)
Temp2 = RightShift2 (temp1)
Label = SignReduce16(Temp2)

Building a Datapath: Jump instruction

- The jump instruction
  - operates by replacing the lower 28 bits of the PC with the lower 26 bits of the instruction shifted left by 2 bits
  - Simply concatenating 00 to the jump off set accomplishes this shift

Detailed MIPS Architecture Datapaths integrated from instruction fetch, R-type, load and store instructions and branch types.

Datapath for jump instruction not included



Thank You