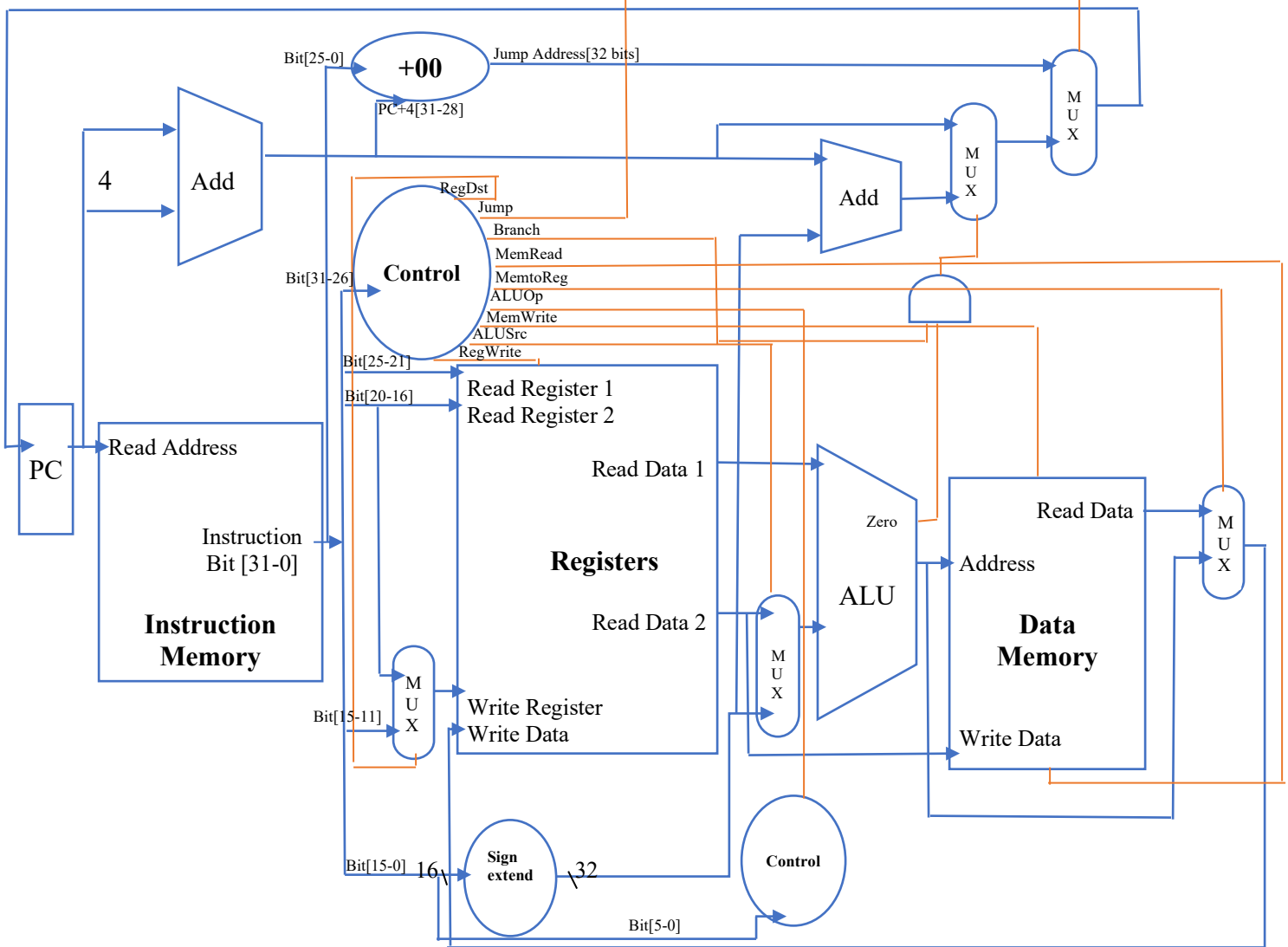


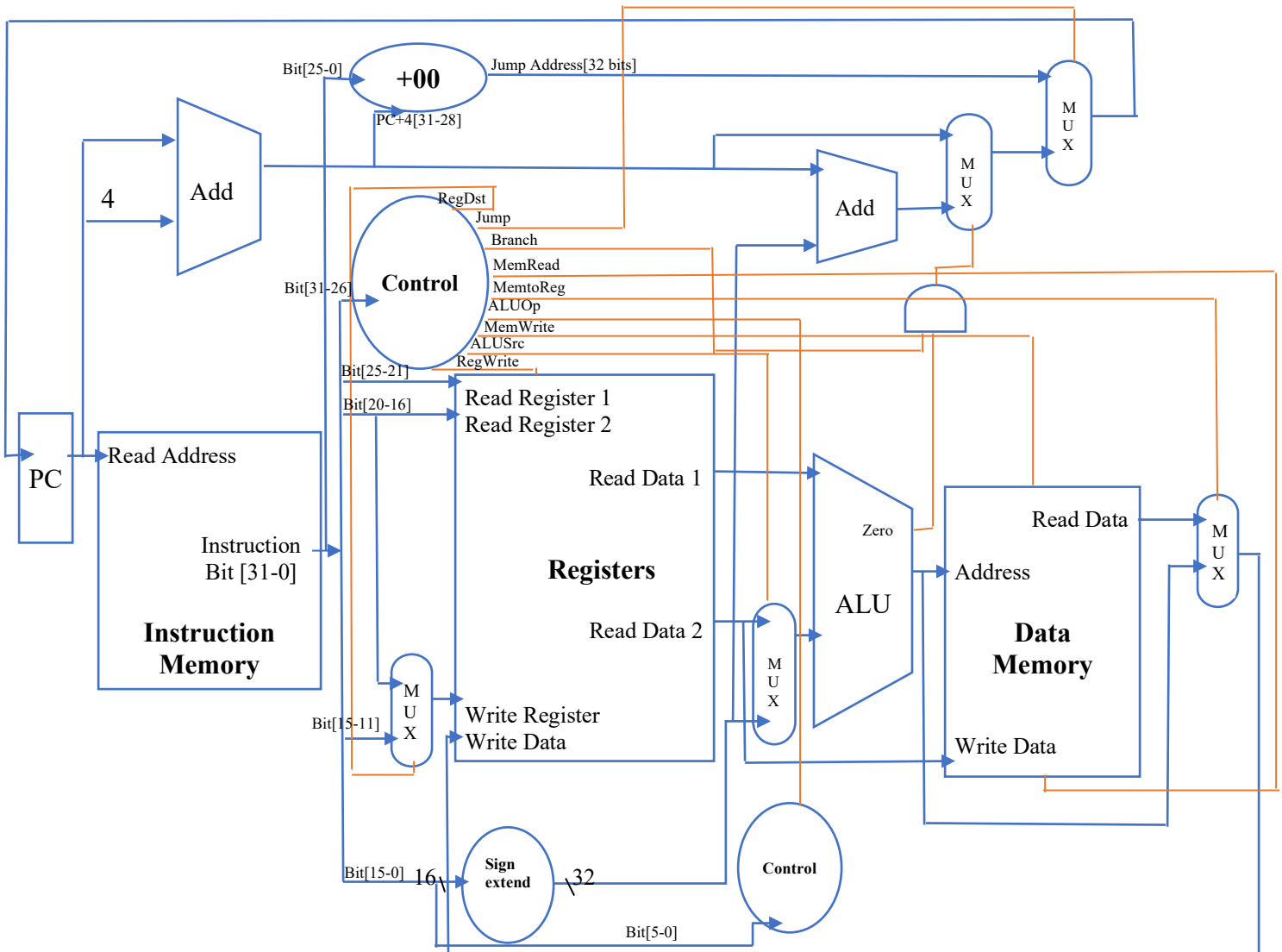
### Designing Single Cycle Data-paths and Control-paths

1. Graph the three instructing formats for MIPS. Make sure to specify each field and the number of bits in each field.
2. Fill out the instruction fields for the following add instruction: add \$t1, \$t3, \$t4
3. Fill out the instruction fields for the following branch instruction: beq \$t2, \$t3, offset
4. Fill out the instruction fields for the following load word instruction: lw \$t2, offset(\$t3)
5. Fill out the instruction fields for the following store word instruction: sw \$t2, offset(\$t3)
6. Fill out the instruction fields for the following jump instruction: j Loop (address)

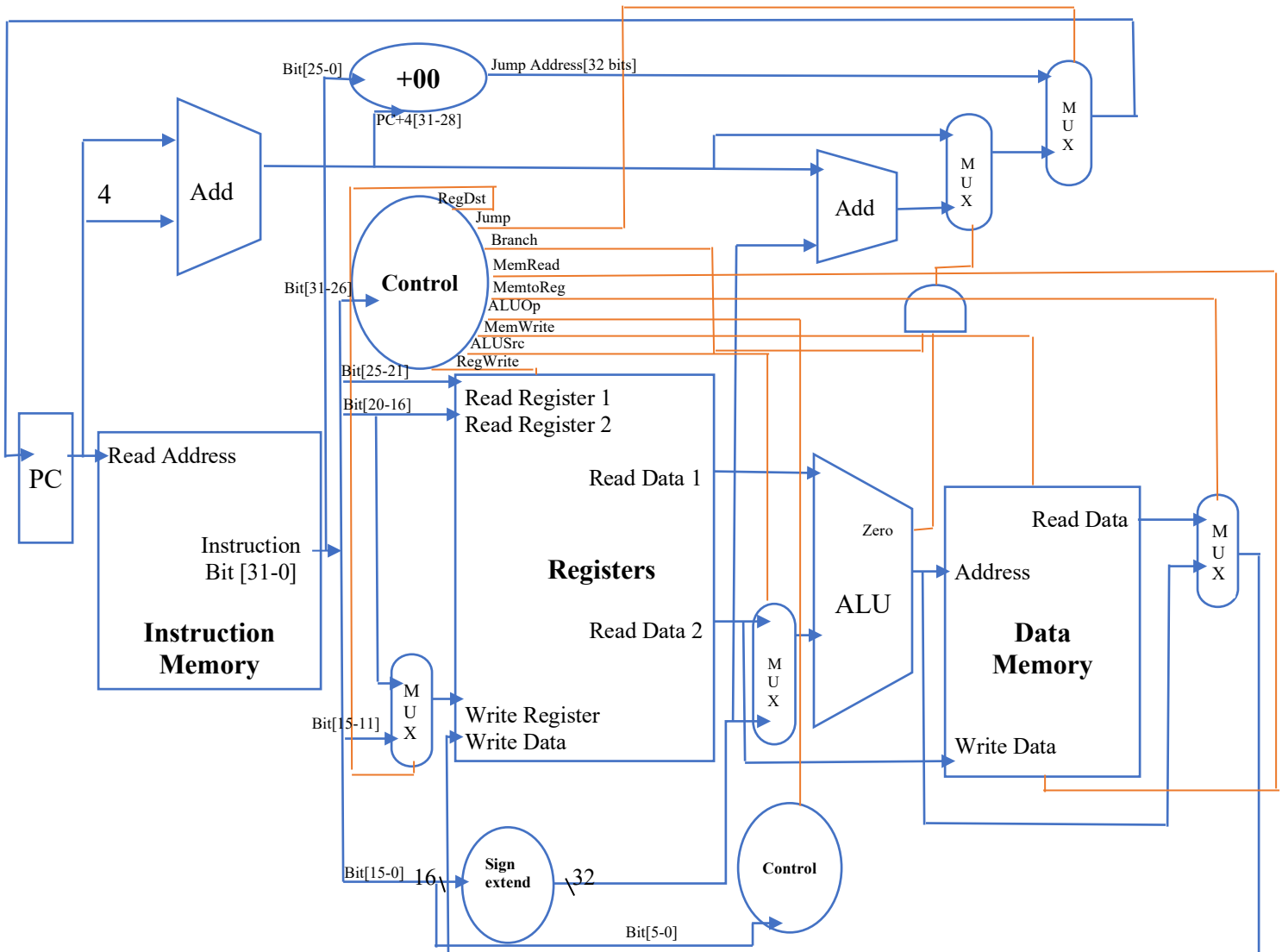
7. **Highlight** the data path and control path for the following add instruction: add \$t1, \$t3, \$t4



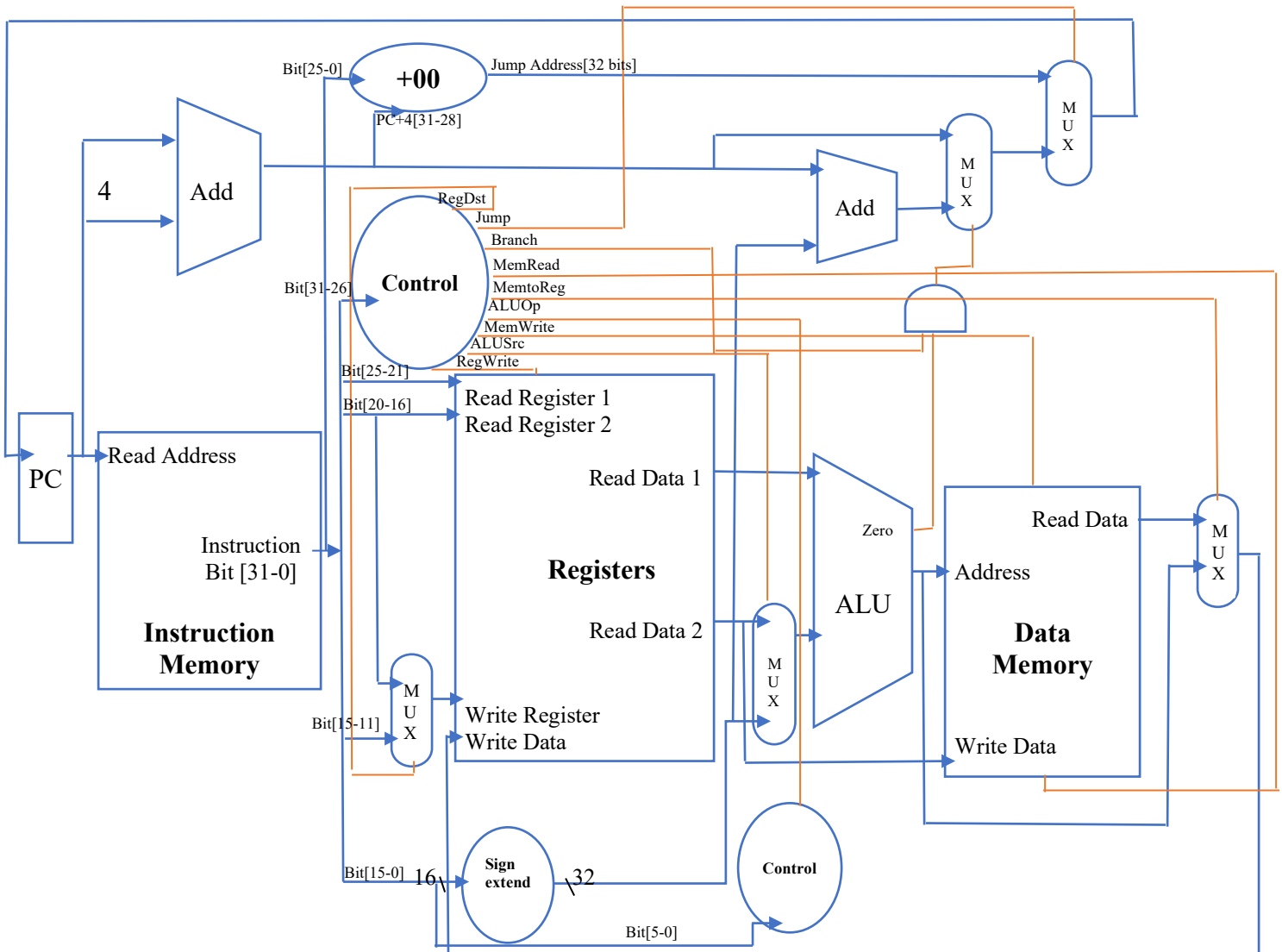
8. **Highlight** the data path and control path for the following branch instruction: beq \$t2, \$t3, offset



9. **Highlight** the data path and control path for the following load word instruction: lw \$t2, offset(\$t3)



10. **Highlight** the data path and control path for the following jump instruction: j Loop (address)



**Solutions**

- Graph the three instructing formats for MIPS. Make sure to specify each field and the number of bits in each field.

R – type format

op (6 bits)	rs (5 bits)	rt (5 bits)	rd (5 bits)	shamt (5 bits)	func (6 bits)
-------------	-------------	-------------	-------------	----------------	---------------

I – type format

op (6 bits)	rs (5 bits)	rt (5 bits)	Immediate (16 bits)
-------------	-------------	-------------	---------------------

J – type format

op (6 bits)	Address (26 bits)
-------------	-------------------

- Fill out the instruction fields for the following add instruction: add \$t1, \$t3, \$t4

op = 0	rs = 11 (\$t3)	rt = 12 (\$t4)	rd = 9 (\$t1)	shmat = 0	func = 32 (add)
--------	----------------	----------------	---------------	-----------	-----------------

- Fill out the instruction fields for the following branch instruction: beq \$t2, \$t3, offset

op = 4 (beq)	rs = 10 (\$t2)	rt = 11 (\$t3)	Immediate = offset
--------------	----------------	----------------	--------------------

- Fill out the instruction fields for the following load word instruction: lw \$t2, offset(\$t3)

op = 35 (lw)	rs = 11 (\$t3)	rt = 10 (\$t2)	Immediate = offset
--------------	----------------	----------------	--------------------

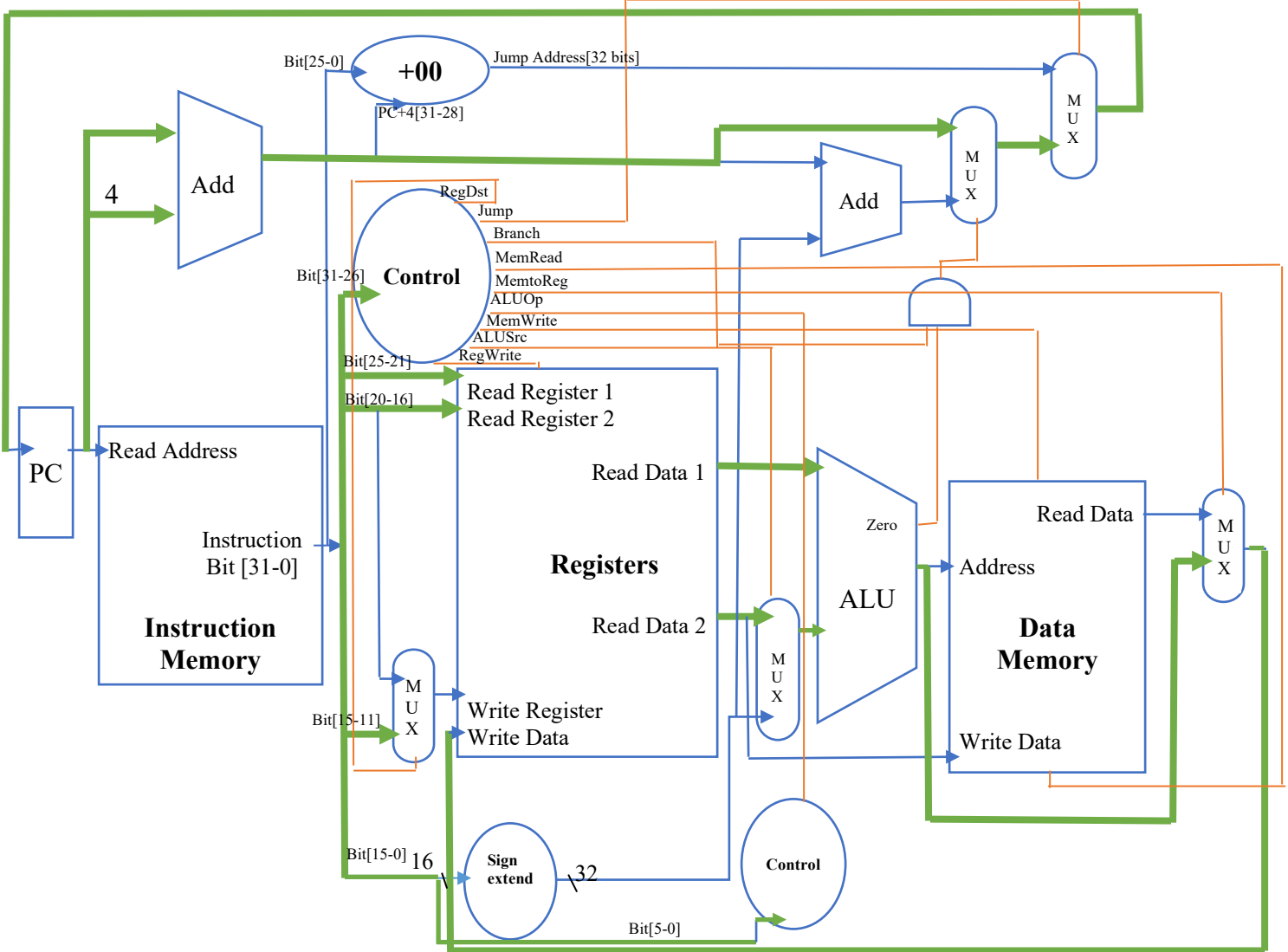
- Fill out the instruction fields for the following store word instruction: sw \$t2, offset(\$t3)

op = 43 (sw)	rs = 11 (\$t3)	rt = 10 (\$t2)	Immediate = offset
--------------	----------------	----------------	--------------------

- Fill out the instruction fields for the following jump instruction: j Loop (address)

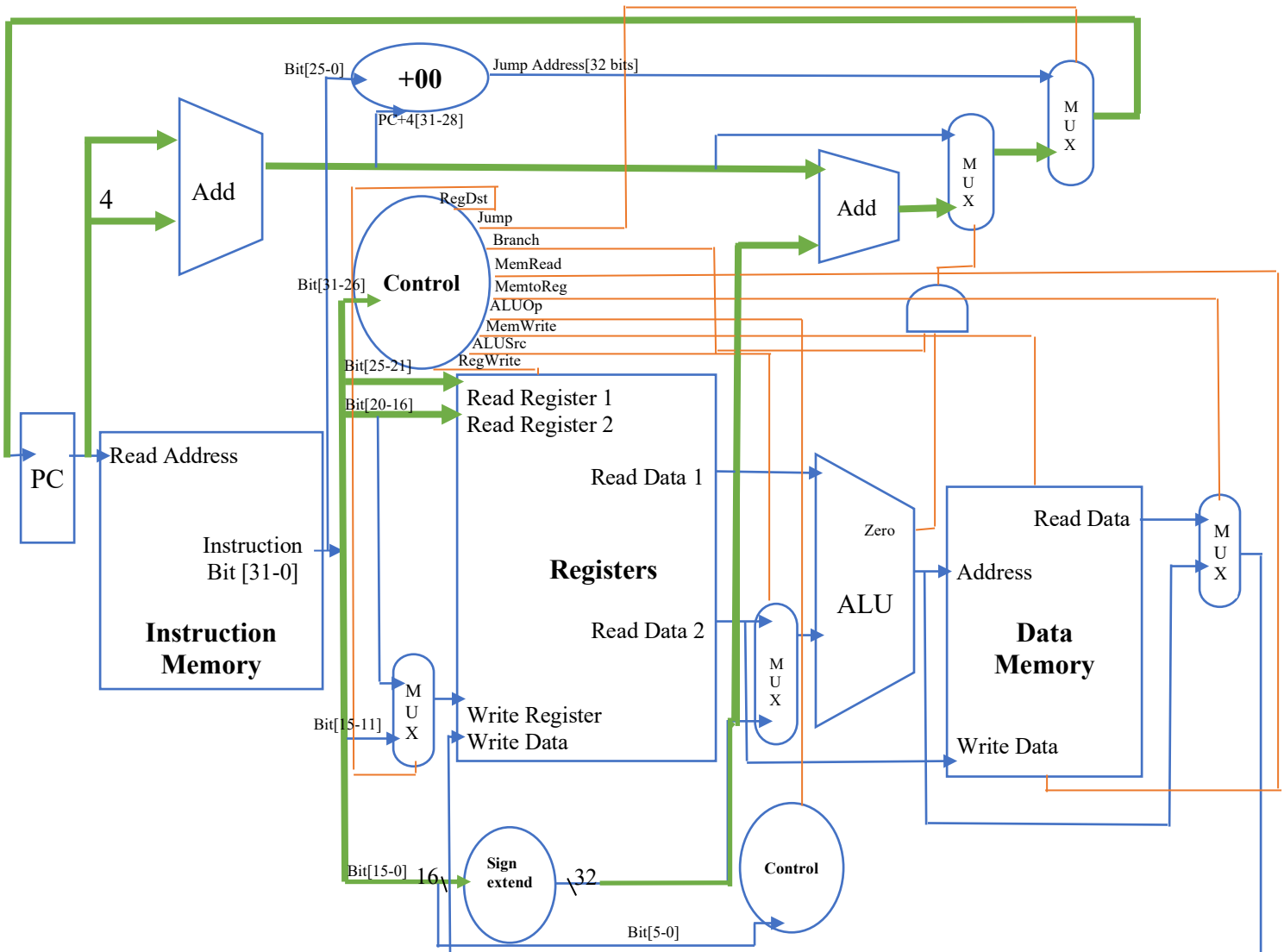
op = 2 (j)	Address = address
------------	-------------------

7. **Highlight** the data path and control path for the following add instruction: add \$t1, \$t3, \$t4



Control Path = RegDest, RegWrite

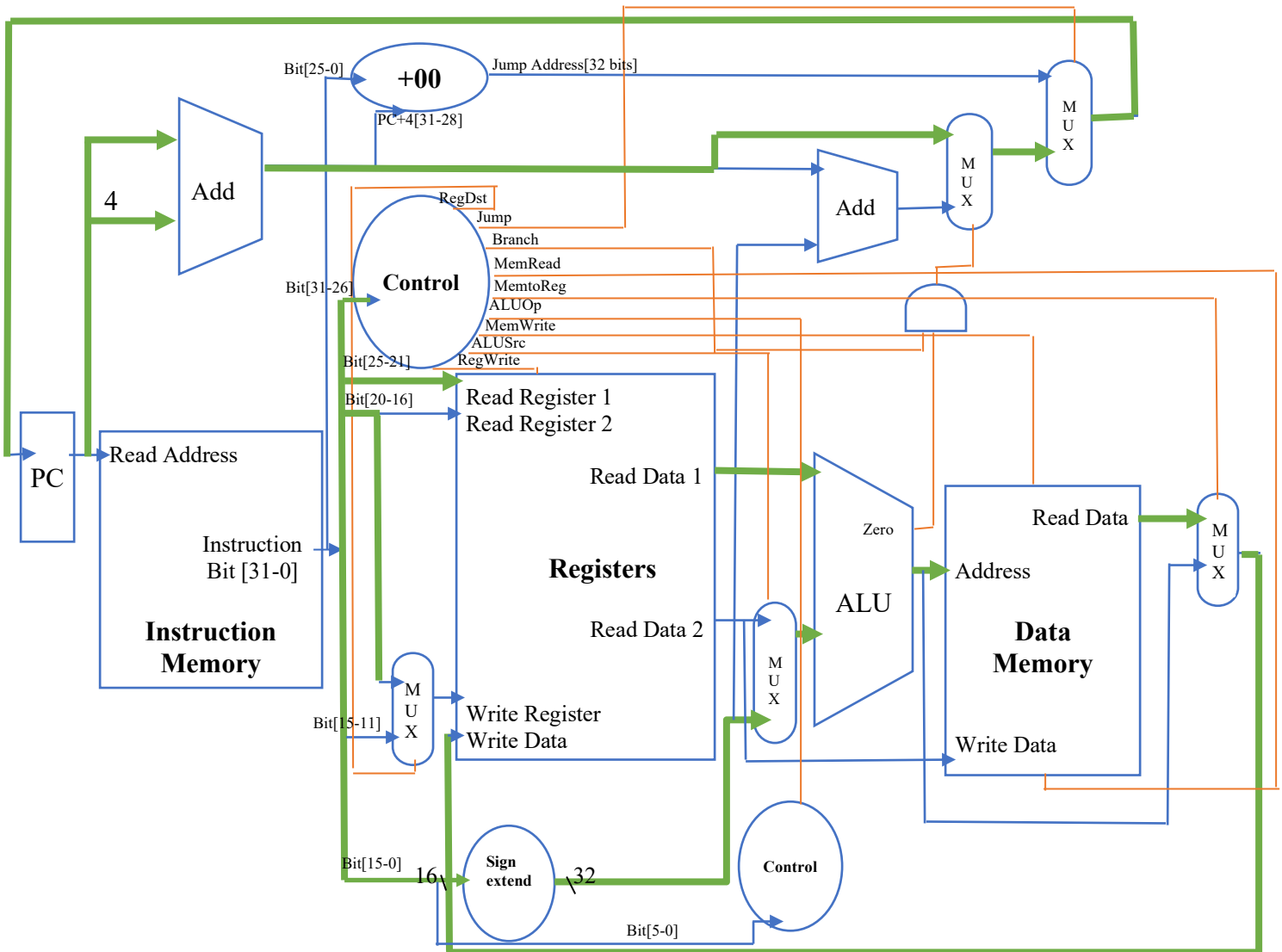
8. Highlight the data path and control path for the following branch instruction: beq \$t2, \$t3, offset



Control Path = Branch, ALUOp

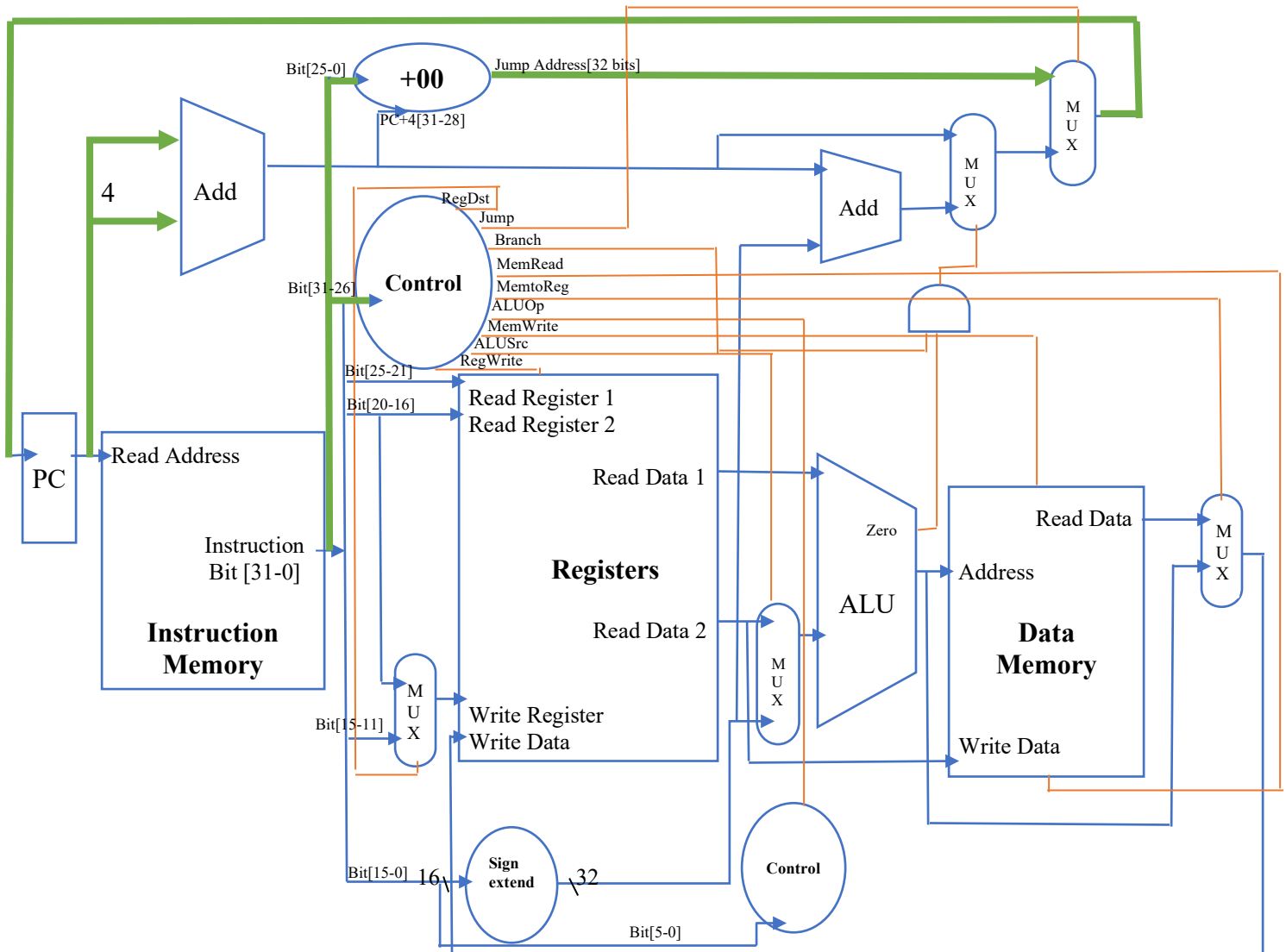


9. **Highlight** the data path and control path for the following load word instruction: lw \$t2, offset(\$t3)



Control Path = MemRead, MemtoReg, ALUOp, ALUSrc, RegWrite

10. Highlight the data path and control path for the following jump instruction: j Loop (address)



Control Path = Jump, ALUOp