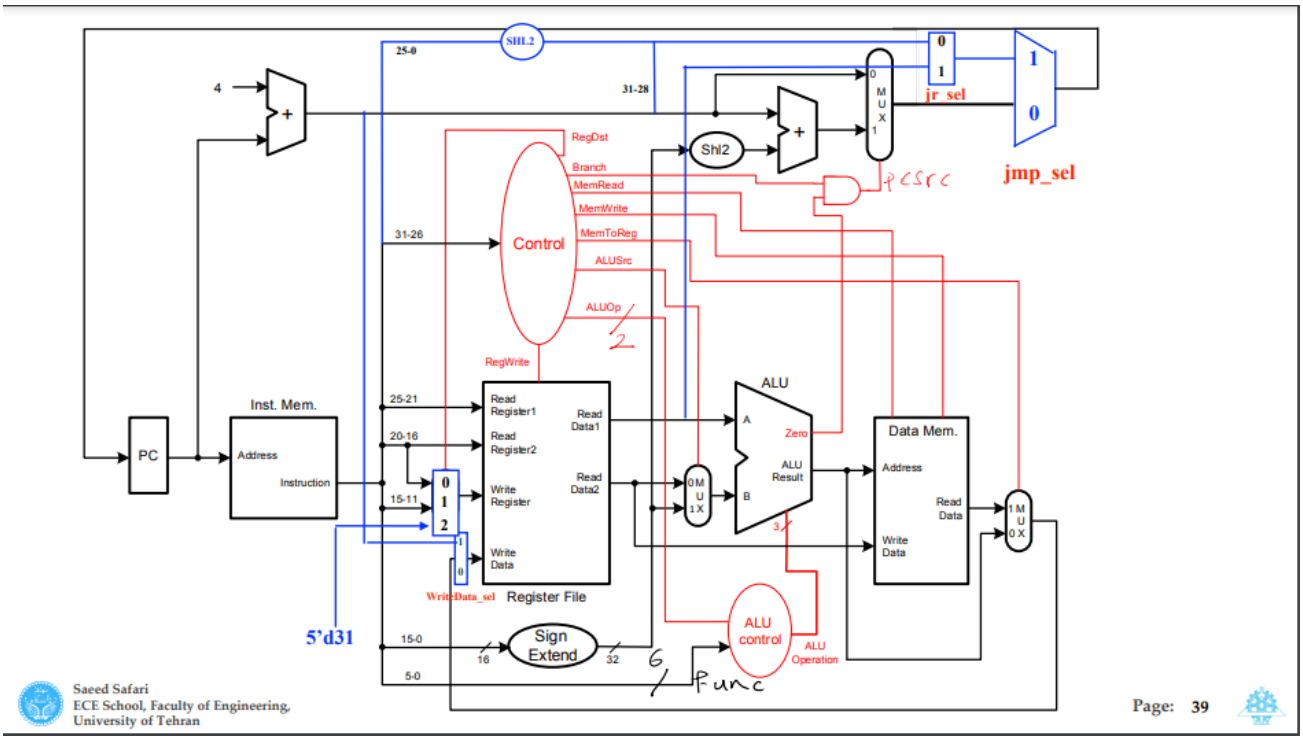
First, we complete the data path to meet the requirements.



Then we need to issue the right control signals.

	RegDst	RegWrite	ALUSrc	MemRead	MemWrite	MemToReg
RT	01	1	0	0	0	0
J	00	0	0	0	0	0
Jal	10	1	0	0	0	0
Jr	00	0	0	0	0	0
lw	00	1	1	1	0	1
sw	00	0	1	0	1	0
beq	00	0	0	0	0	0
addi	00	1	1	0	0	0
slt	01	1	0	0	0	0
slti	00	1	1	0	0	0

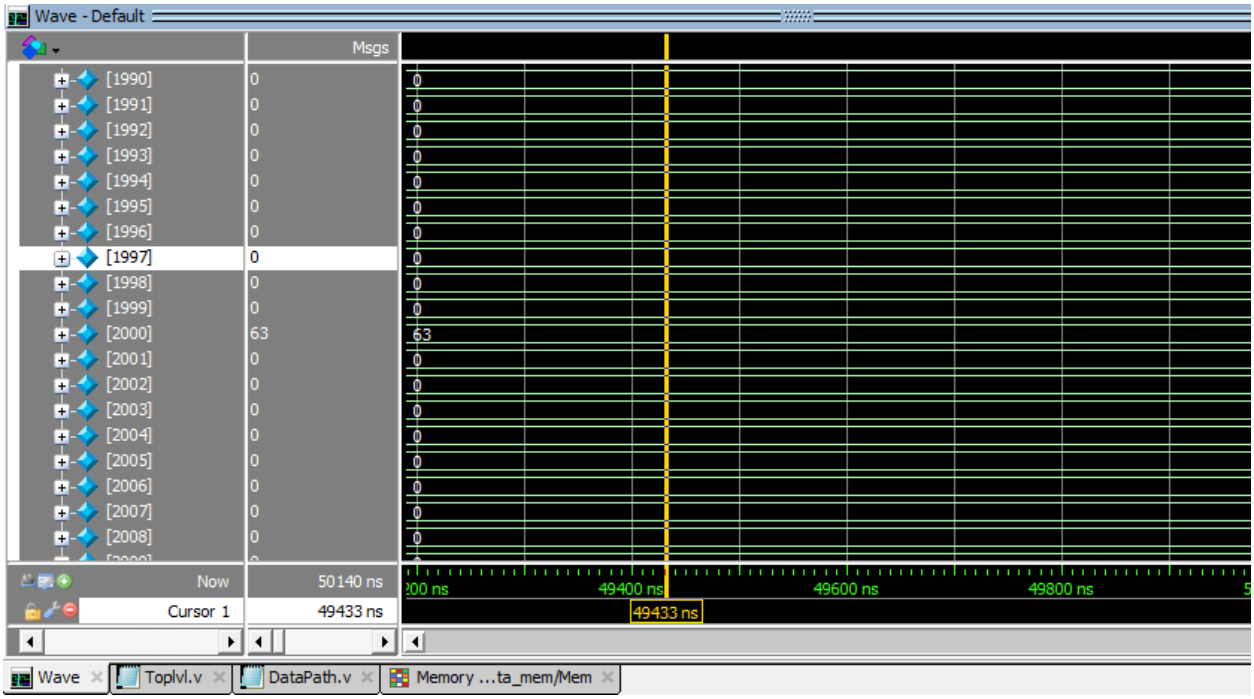
	Branch	ALUop	Jmp_sel	Jr_sel	WriteData_sel
RT	0	10	0	0	00
J	0	00	1	0	00
Jal	0	00	1	0	01
Jr	0	00	1	1	00
lw	0	00	0	0	00
sw	0	00	0	0	00
beq	1	01	0	0	00
addi	0	00	0	0	00
slt	0	10	0	0	10
slti	0	11	0	0	10

Now we test our MIPS processor.

```
        addi    R5, R0, 10
        add     R3, R0, R0
        add     R1, R0, R0
        add     R2, R0, R0
Loop:   beq     R1, R5, END
        lw      R4, 1000(R2)
        add     R3, R3, R4
        addi    R2, R2, 4
        addi    R1, R1, 1
        J       Loop
END:    sw      R3, 2000(R0)
```

Note that we have stored 1, 2, 4, 8, 16, 32 and 64 in the following addresses: 1000, 1004, 1008, 1012, 1016, 1024.

This is the final result.



Note that we have also updated our Memory File; So, this is how our file will look in the end.