CSE331
COMPUTER ORGANIZATION
FALL 2022
FINAL PROJECT

Burcu Sultan ORHAN

1901042667

Truth Table for Main Control:

	opcode	func	regDst	ALUSrc	MemtoReg	RegWrite	MemRead	MemWrite	Branch	NotBranch	Jump	ALUOp
add	00 0000	10 0000	1	0	0	1	0	0	0	0	0	1 1 1
subtract	00 0000	10 0010	1	0	0	1	0	0	0	0	0	1 1 1
and	00 0000	10 0100	1	0	0	1	0	0	0	0	0	1 1 1
or	00 0000	10 0101	1	0	0	1	0	0	0	0	0	1 1 1
shift left logical	00 0000	00 0000	1	0	0	1	0	0	0	0	0	1 1 1
shift right logical	00 0000	00 0010	1	0	0	1	0	0	0	0	0	1 1 1
set on less than	00 0000	10 1010	1	0	0	1	0	0	0	0	0	1 1 1
jump register	00 0000	00 1000	1	0	0	1	0	0	0	0	0	1 1 1
load immediate			0	1		1	0	0	0	0	0	0 0 0
add immediate	00 1000	xx xxxx	0	1	0	1	0	0	0	0	0	0 0 0
and immediate	00 1100	xx xxxx	0	1	0	1	0	0	0	0	0	1 0 0
or immediate	00 1101	xx xxxx	0	1	0	1	0	0	0	0	0	1 0 1
load word	10 0011	xx xxxx	0	1	1	1	1	0	0	0	0	0 0 0
store word	10 1011	xx xxxx	0	1	x	0	0	1	0	0	0	0 0 0
branch on equal	00 0100	xx xxxx	0	0	x	0	0	0	1	0	0	0 1 0
branch on not equal	00 0101	xx xxxx	0	0	x	0	0	0	0	1	0	0 1 0
set on less than immediate	00 1010	xx xxxx	0	1	0	1	0	0	0	0	0	1 1 0
jump	00 0010	xx xxxx	0	1	0	0	0	0	0	0	1	0 0 0
jump and link	00 0011	xx xxxx	0	1	0	1	0	0	0	0	1	0 0 0

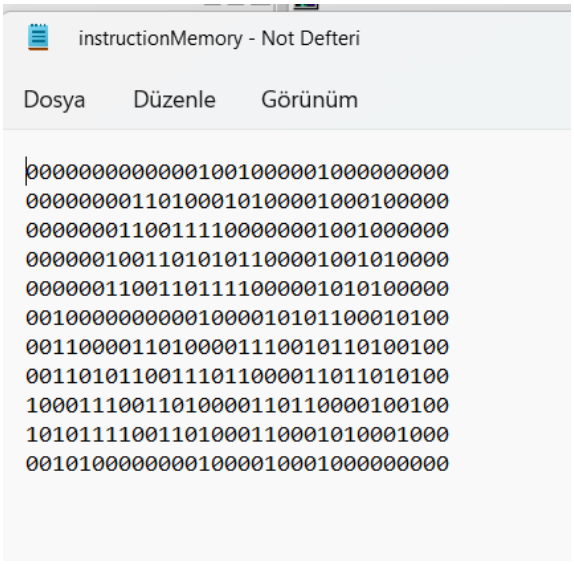
Truth table for ALU Control:

	ALUOp	Func		ALUControl	Action
add	1 1 1	10 0000		0 0 0	add
subtract	1 1 1	10 0010		0 0 1	sub
and	1 1 1	10 0100		1 0 0	and
or	1 1 1	10 0101		1 0 1	or
shift left logical	1 1 1	00 0000		0 1 0	mult
shift right logical	1 1 1	00 0010		0 1 0	mult
set on less than	1 1 1	10 1010		1 1 0	slt
jump register	1 1 1	00 1000		0 0 0	add
load immediate	0 0 0	xxxxxx		0 0 0	add
add immediate	0 0 0	xxxxxx		0 0 0	add
and immediate	1 0 0	xxxxxx		1 0 0	and
or immediate	1 0 1	xxxxxx		1 0 1	or
load word	0 0 0	xxxxxx		0 0 0	add
store word	0 0 0	xxxxxx		0 0 0	add
branch on equal	0 1 0	xxxxxx		0 0 1	sub
branch on not equal	0 1 0	xxxxxx		0 0 1	sub
set on less than immediate	1 1 0	xxxxxx		1 1 0	slt
jump	0 0 0	xxxxxx		0 0 0	add
jump and link	0 0 0	xxxxxx		0 0 0	add

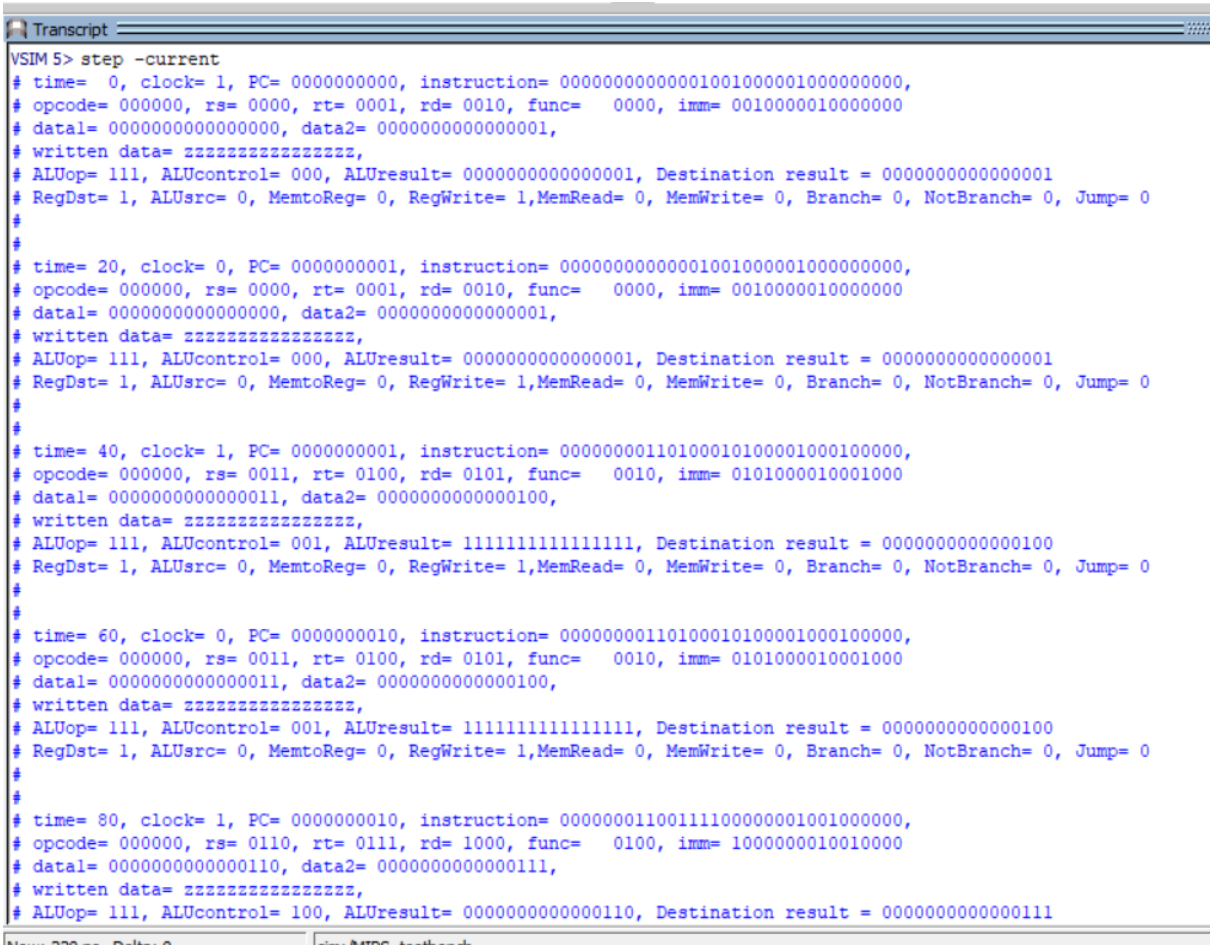
Test Benches:

- MIPS Test bench

  For these instructions:



  Result:



- Instruction Memory Test bench

```
add wave -position insertpoint sim:/instructionMemory_testbench/*
VSIM 21> step -current
# time=  0, PC address= 0000000000, instruction= 00000000000001001000001000000000, clk= 1
# time= 20, PC address= 0000000000, instruction= 00000000000001001000001000000000, clk= 0
# time= 40, PC address= 0000000001, instruction= 00000000110100010100001000100000, clk= 1
# time= 60, PC address= 0000000001, instruction= 00000000110100010100001000100000, clk= 0
# time= 80, PC address= 0000000010, instruction= 00000001100111100000001001000000, clk= 1
VSIM 22> step -current

VSIM 22>
```

Now: 100 ps Delta: 1 | sim:/instructionMemory_testbench

- Main Control Test bench

```
# time=  0,
# opcode= 000000,
# RegDst= 1,
# ALUsrc= 0,
# MemtoReg= 0,
# RegWrite= 1,
# MemRead= 0,
# MemWrite= 0,
# Branch= 0,
# NotBranch= 0,
# Jump= 0,
# ALUop= 111
#
# time= 20,
# opcode= 001000,
# RegDst= 0,
# ALUsrc= 1,
# MemtoReg= 0,
# RegWrite= 1,
# MemRead= 0,
# MemWrite= 0,
# Branch= 0,
# NotBranch= 0,
# Jump= 0,
# ALUop= 110
#
# time= 40,
# opcode= 001100,
# RegDst= 0,
# ALUsrc= 1,
# MemtoReg= 0,
# RegWrite= 1,
# MemRead= 0,
# MemWrite= 0,
# Branch= 0,
# NotBranch= 0,
# Jump= 0,
# ALUop= 100
```

Now: 220 ps Delta: 1 | sim:/MainContro

- Select Destination Test bench

```
# Loading work.SelectDestination
VSIM 28> step -current
# time =  0, rt=0101, rd=1011, regDest=0 , Destination=0101
# time = 20, rt=0101, rd=1010, regDest=1 , Destination=1010

VSIM 29>
```

- Register Operations Test bench
  For these register values

registers - Not Defteri

Dosya Düzenle Görünüm

```
0000000000000000
0000000000000001
0000000000000010
0000000000000011
0000000000000100
0000000000000101
0000000000000110
0000000000000111
0000000000001000
0000000000001001
0000000000001010
0000000000001011
0000000000001100
0000000000001101
0000000000001110
0000000000001111
```

Results

```
add wave -position insertpoint sim:/registerOperations_testbench/*
VSIM 32> step -current
# time =  0, clock = 1 rs = 0001, rt = 0010, output1 = 0000000000000001, output2 = 0000000000000010, rd = 011 , data = 0000000111110000, RegWrite = 0
# time = 20, clock = 0 rs = 0011, rt = 0100, output1 = 0000000000000011, output2 = 0000000000000100, rd = 011 , data = 0000001111000011, RegWrite = 1

VSIM 33>
```

- ALU Control Test bench

```
# Loading work.ALUControl
add wave -position insertpoint sim:/ALUControl_testbench/*
VSIM 36> step -current
# time=  0,AluOp= 111,func= 100000,AluControl= 000
#
# time= 20,AluOp= 111,func= 100010,AluControl= 001
#
# time= 40,AluOp= 111,func= 100100,AluControl= 100
#
# time= 60,AluOp= 111,func= 100101,AluControl= 101
#
# time= 80,AluOp= 111,func= 000000,AluControl= 010
#
# time= 100,AluOp= 111,func= 000010,AluControl= 010
#
# time= 120,AluOp= 111,func= 101010,AluControl= 110
#
# time= 140,AluOp= 111,func= 001000,AluControl= 000
#
# time= 160,AluOp= 000,func= 110110,AluControl= 000
#
# time= 180,AluOp= 100,func= 110110,AluControl= 100
#
# time= 200,AluOp= 101,func= 110110,AluControl= 101
#
# time= 220,AluOp= 010,func= 110110,AluControl= 001
#
# time= 240,AluOp= 110,func= 110110,AluControl= 110
#
VSIM 37> step -current

VSIM 37>
```

- Mux2x1 Test bench

```
VSIM 40> step -current
# time =  0 ,a=0101010101010101,b=1010111010101011, Selection=0 ,Result=0101010101010101
# time = 20 ,a=0101010101010101,b=1111001010111010, Selection=1 ,Result=1111001010111010

VSIM 41>
```

- ALU Test bench

```
#         Region: /ALU32bit_testbench/gl/g3/fl
add wave -position insertpoint sim:/ALU32bit_testbench/*
VSIM 13> step -current
# time =  0, a =0000000000000011, b=0000000000000001, op=000, sum=0000000000000100
# time = 20, a =1111100000000011, b=1111111111111111, op=001, sum=1111100000000100
# time = 40, a =0000000000000011, b=0000000000000001, op=011, sum=0000000000000010
# time = 60, a =0000000000000000, b=0000000000000001, op=100, sum=0000000000000000
# time = 80, a =1001010100000000, b=1111110010100001, op=101, sum=1111110110100001
# time = 100, a =1001010100000000, b=1111110010100001, op=110, sum=0000000000000001
# time = 120, a =1001010100000000, b=1111110010100001, op=111, sum=0000001001011110

VSIM 14>
```

- Memory Operations Test bench

```
VSIM 43> step -current
# time =  0, address = 0000000000000001, data = 0000000000011111, myOutput = 0000000000011111, MemRead = 1, MemWrite = 0, clk = 1
# time = 20, address = 0000000000000001, data = 0000000000011111, myOutput = 0000000000011111, MemRead = 1, MemWrite = 1, clk = 0
# time = 40, address = 0000000000000001, data = 0000000000011111, myOutput = 0000000000011111, MemRead = 1, MemWrite = 1, clk = 1
# time = 60, address = 0000000000000010, data = 0001001000111110, myOutput = 0001001000111110, MemRead = 1, MemWrite = 1, clk = 0
# time = 80, address = 0000000000000010, data = 0001001000111110, myOutput = 0001001000111110, MemRead = 1, MemWrite = 1, clk = 1
# time = 100, address = 0000000000000111, data = 0100001000011100, myOutput = 0100001000011100, MemRead = 1, MemWrite = 1, clk = 0
# time = 120, address = 0000000000000111, data = 0100001000011100, myOutput = 0100001000011100, MemRead = 1, MemWrite = 1, clk = 1

VSIM 44>
```