CSE 3038 Computer Organization

Programming Project 2 Report

Koray Emre Şenel – 150117037

Alperen Köker – 150117035

Ahmet Turgut – 150117046

Mehmet Etka Uzun – 150118504

	Instr	Type	Code	Syntax	Meaning
1.	bltz	I-type	opcode=1	bltz $rs, Target	if R[rs] < 0, branches to PC-relative address (formed as beq & bne do)
6.	nori	I-type	opcode=13	nori $rt, $rs, imm16	Put the logical NOR of register $rs and the zero-extended immediate into register $rt.
10.	balrz	R-type	funct=22	balrz $rs, $rd	if Status [Z] = 1, branches to address found in register $rs link address is stored in $rd (which defaults to 31)
13.	jal	J-type	opcode=3	jal Target	jump to pseudo-direct address (formed as j does), link address is stored in register 31
14.	jm	I-type	opcode=16	jm imm16($rs)	jumps to address found in memory (indirect jump)
19.	sllv	R-type	func=4	sllv $rd, $rt, $rs	shift register $rt to left by the value in register $rs, and store the result in register $rd.

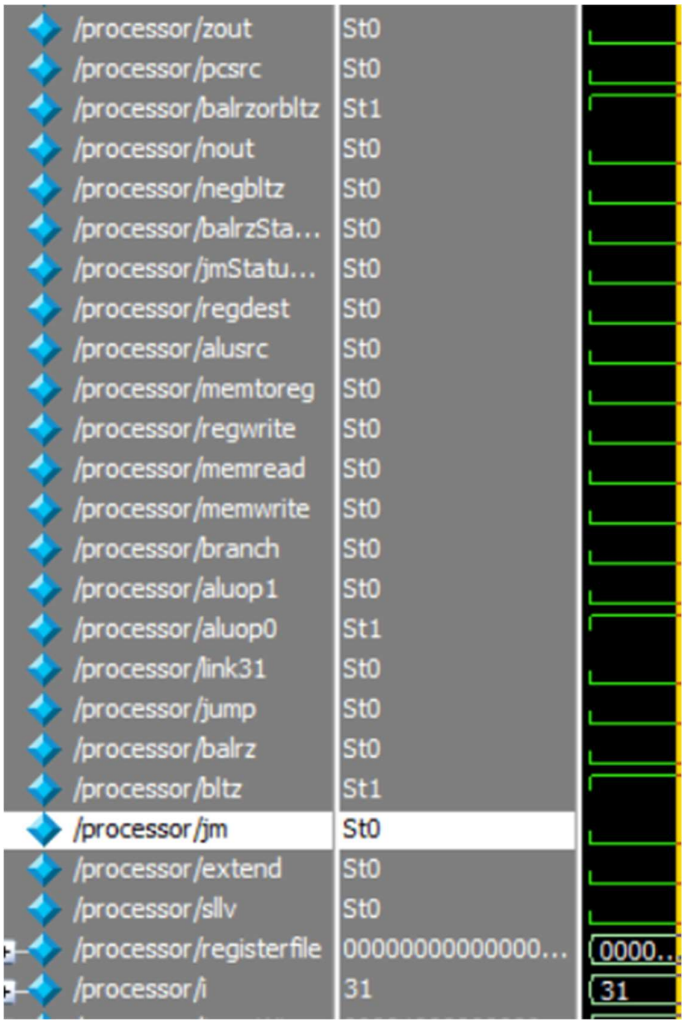
1- BLTZ

BLTZ

OPCODE-000001	RS	RT	IMMEDIATE
6 bits	5 bits	5 bits	16 bits

Instruction is fetched from instruction memory. Signals are set properly in the control. Then registers are read. In ALU the read register rs value will be added with zero. If the output is negative branches to PC-relative address, else next instruction continued as PC+4.

Operation	RegDst	Jump	Branch	Memread	MemToReg	ALUOp	MemWrite	ALUSrc	RegWrite	Bltz	extend	Jm	balrz	Link31
Balrz	X	0	0	x	1	10	0	x	1	0	x	0	1	1



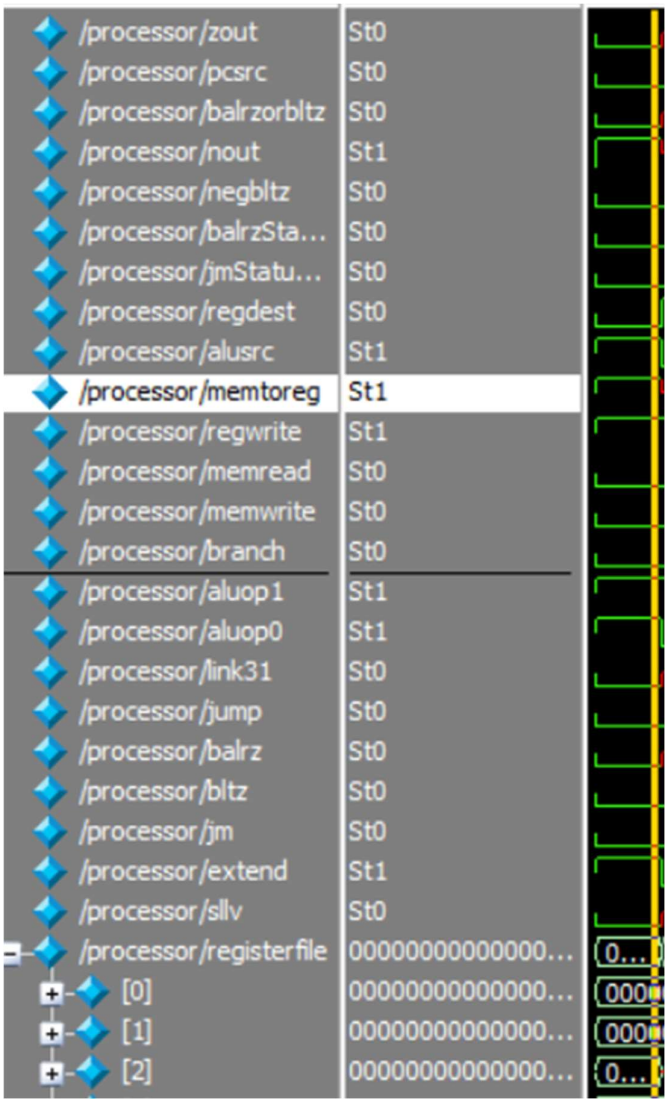
2- NORI

NORI

OPCODE-001101	RS	RT	IMMEDIATE
6 bits	5 bits	5 bits	16 bits

The instruction fetched from instruction memory. Signals are set properly in the control. Then registers are read. Read register rs and zero extended immediate value send to ALU to make the nor operation. The output of the ALU put into register rt. Next instruction continued as PC+4.

Operation	RegDst	Jump	Branch	Memread	MemToReg	AluOp0	AluOp1	MemWrite	AluSrc	RegWrite	Bltz	extend	balrz	Link31	Jm
nori	0	0	0	x	1	1	1	x	1	1	0	1	0	0	0



3- BALRZ

BALRZ

OPCODE-000000	RS	RT	RD	00000.	FUNCTION-011001
6 bits	5 bits	5 bits	5 bits	5 bits	6 bits

Instruction is fetched from instruction memory. Signals are set properly in the control and ALU control. Registers are read. If the status register of zero flag is 1 it branches to address read from the register rs . Link address (PC+4) is stored in register rd (register 31). Else it continues to next instruction PC+4.

Operation	RegDst	Jump	Branch	Memread	MemToReg	ALUOp	MemWrite	ALUSrc	RegWrite	Bltz	extend	Jm	balrz	Link31
Balrz	X	0	0	x	1	10	0	x	1	0	x	0	1	1

4- JAL

JAL

OPCODE-000011	ADDRESS
6 bits	26 bits

Instruction is fetched from instruction memory. Signals are set properly in the control. Then jumps to target address and link address (PC+4) is stored in register 31.

Operation	RegDst	Jump	Branch	Memread	MemToReg	ALUOp0	ALUOp1	MemWrite	ALUSrc	RegWrite	Bltz	extend	balrz	Link31	Jm
jal	X	1	x	x	x	x	x	x	x	1	0	x	0	1	0

5- JM

JM

OPCODE-001000	RS	RT	IMMEDIATE
6 bits	5 bits	5 bits	16 bits

Instruction is fetched from instruction memory. Signals are set properly in the control. Then the register rs and sign extended immediate value added in ALU. Then output is send to PC as new PC value.

Operation	RegDst	Jump	Branch	Memread	MemToReg	ALUOp0	ALUOp1	MemWrite	ALUSrc	RegWrite	Bltz	extend	balrz	Link31	Jm
Jm	X	x	x	1	0	0	0	0	1	x	0	0	0	x	1

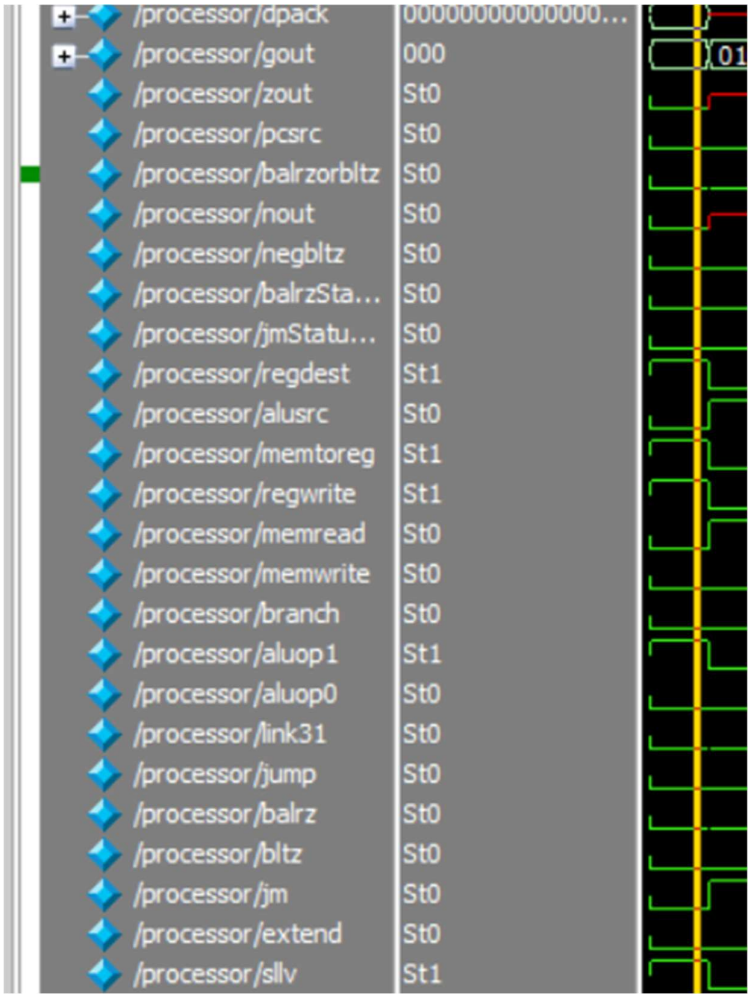
6- SLLV

SLLV

OPCODE-000000	RS	RT	RD	00000.	FUNCTİON-000100
6 bits	5 bits	5 bits	5 bits	5 bits	

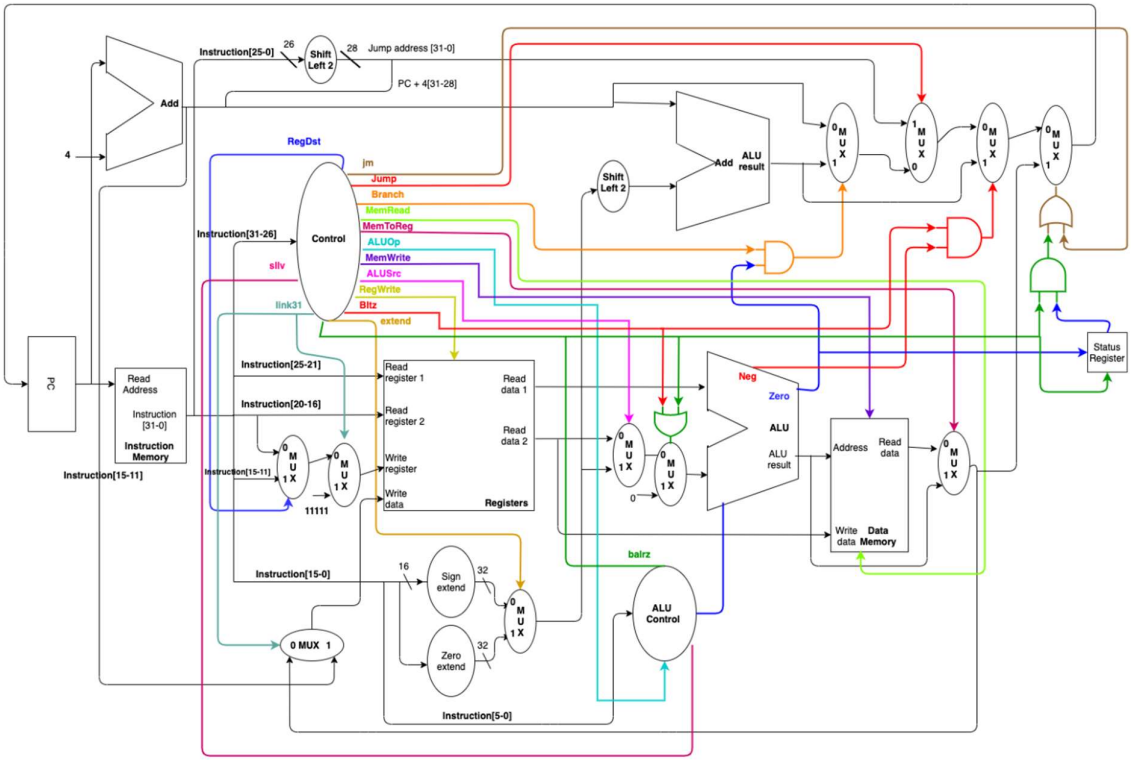
Instruction is fetched in instruction memory. Signals are set properly in control and ALU control. Registers are read and send to ALU . In ALU rt is shifted left by the register rs value. Output is put into register rd. Next instruction continued as PC+4.

Operation	RegDst	Jump	Branch	Memread	MemToReg	ALUOp0	ALUOp1	MemWrite	ALUSrc	RegWrite	Bltz	extend	balrz	Link31	Jm
sllv	1	x	x	x	1	0	1	x	0	1	0	x	0	0	0


/processor/dpack
/processor/gout
/processor/zout
/processor/pcsrc
/processor/balrzorbltz
/processor/nout
/processor/negbltz
/processor/balrzSta...
/processor/jmStatu...
/processor/regdest
/processor/alusrc
/processor/memtoreg
/processor/regwrite
/processor/memread
/processor/memwrite
/processor/branch
/processor/aluop1
/processor/aluop0
/processor/link31
/processor/jump
/processor/balrz
/processor/bltz
/processor/jm
/processor/extend
/processor/sllv
000
St0
St1

Single Cycle Datapath

We modified the datapath to be able to make these new 6 instructions. In this figure you can see the new datapath.



New components ->

7 new 2 to 1 Multiplexers.

2 new or gates.

3 new and gates.

Zero extension module.

Status register for zero of past instruction.

Shift left 2 module.