Name	Fields					Comments
Field size	3 bits	3 bits	3 bits	3 bits	4 bits	All MIPS-L instructions 16 bits
R-format	op	rs	rt	rd	funct	Arithmetic instruction format
I-format	op	rs	rt	Address/immediate		Transfer, branch, immediate format
J-format	op	target address				Jump instruction format

Name	Format	Example					Comments
		3 bits	3 bits	3 bits	3 bits	4 bits	
add	R	0	2	3	1	0	add $1,$2,$3
sub	R	0	2	3	1	1	sub $1,$2,$3
and	R	0	2	3	1	2	and $1,$2,$3
or	R	0	2	3	1	3	or $1,$2,$3
slt	R	0	2	3	1	4	slt $1,$2,$3
jr	R	0	7	0	0	8	jr $7
lw	I	4	2	1	7		lw $1, 7 ($2)
sw	I	5	2	1	7		sw $1, 7 ($2)
beq	I	6	1	2	7		beq $1,$2, 7
addi	I	7	2	1	7		addi $1,$2, 7
j	J	2	500				j 1000
jal	J	3	500				jal 1000
slti	I	1	2	1	7		slti $1,$2, 7

1. **Add : R[rd] = R[rs] + R[rt]**
2. **Subtract : R[rd] = R[rs] - R[rt]**
3. **And: R[rd] = R[rs] & R[rt]**
4. **Or : R[rd] = R[rs] | R[rt]**
5. **SLT: R[rd] = 1 if R[rs] < R[rt] else 0**
6. **Jr: PC=R[rs]**
7. **Lw: R[rt] = M[R[rs]+SignExtImm]**
8. **Sw : M[R[rs]+SignExtImm] = R[rt]**
9. **Beq : if(R[rs]==R[rt]) PC=PC+1+BranchAddr**
10. **Addi: R[rt] = R[rs] + SignExtImm**
11. **J : PC=JumpAddr**
12. **Jal : R[7]=PC+2;PC=JumpAddr**
13. **SLTI: R[rt] = 1 if R[rs] < imm else 0**

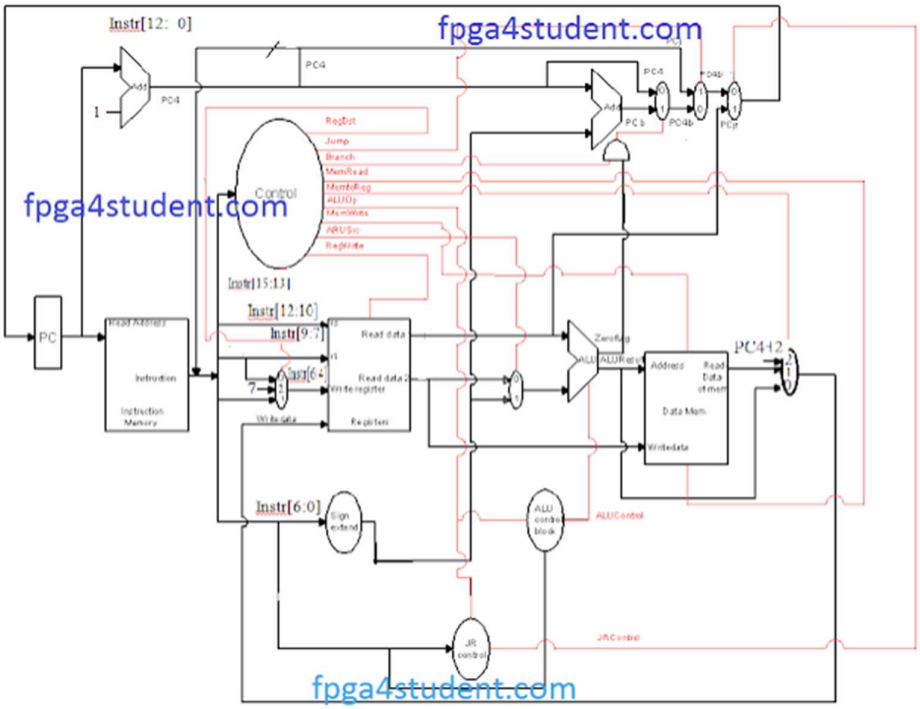
SignExtImm = { 9{immediate[6]}, imm

JumpAddr = { (PC+1)[15:13], address}

BranchAddr = { 7{immediate[6]}, immediate, 1'b0 }

Control signals									
Instruction	Reg Dst	ALU Src	Memto Reg	Reg Write	MemRead	Mem Write	Branch	ALUOp	Jump
R-type	1	0	0	1	0	0	0	00	0
LW	0	1	1	1	1	0	0	11	0
SW	0	1	0	0	0	1	0	11	0
addi	0	1	0	1	0	0	0	11	0
beq	0	0	0	0	0	0	1	01	0
j	0	0	0	0	0	0	0	00	1
jal	2	0	2	1	0	0	0	00	1
slti	0	1	0	1	0	0	0	10	0

ALU Control				
ALU op	Function	ALUcnt	ALU Operation	Instruction
11	xxxx	000	ADD	Addi,lw,sw
01	xxxx	001	SUB	BEQ
00	00	000	ADD	R-type: ADD
00	01	001	SUB	R-type: sub
00	02	010	AND	R-type: AND
00	03	011	OR	R-type: OR
00	04	100	slt	R-type: slt
10	xxxxxx	100	slt	i-type: slti


Instr[12: 0]
fpga4student.com
PC4
Add
1
RegDst
Jump
Branch
MemRead
MemtoReg
ALUOp
MemWrite
ALUSrc
RegWrite
Control
fpga4student.com
Instr[15:13]
Instr[12:10]
Instr[9:7]
Instr[6:4]
7
Read data
Read data 2
Write register
Registers
Write data
PC
Instruction
Instruction Memory
Zero flag
ALU
ALU Result
Address
Read data
Data Mem
Write data
PC442
Instr[6:0]
Sign extend
ALU control block
ALUControl
JR control
JRControl
fpga4student.com