



Instructions			
Instr. Kind	Instr. Name	Operators	Description
ALU Instructions without Immediate	add	Reg1, Reg2, Reg3	Reg1 := Reg2 + Reg3
	sub	Reg1, Reg2, Reg3	Reg1 := Reg2 - Reg3
	sll	Reg1, Reg2, Reg3	Reg1 := Reg2 << Reg3
	srl	Reg1, Reg2, Reg3	Reg1 := Reg2 >> Reg3 logical (fill space with 0)
	sra	Reg1, Reg2, Reg3	Reg1 := Reg2 >> Reg3 arith- metical (fill space with sign bit)
	xor	Reg1, Reg2, Reg3	Reg1 := Reg2 xor Reg3
	or and	Reg1, Reg2, Reg3 Reg1, Reg2, Reg3	Reg1 := Reg2 or Reg3 Reg1 := Reg2 and Reg3
ALU Instructions with Immediate	addi	Reg1, Reg2, Imm	Reg1 := Reg2 + Imm
	subi	Reg1, Reg2, Imm	Reg1 := Reg2 - Imm
	slli	Reg1, Reg2, Imm	Reg1 := Reg2 << Imm
	srli	Reg1, Reg2, Imm	Reg1 := Reg2 >> Imm logical (fill space with 0)
	srai	Reg1, Reg2, Imm	Reg1 := Reg2 >> Imm arith- metical (fill space with sign bit)
	xori	Reg1, Reg2, Imm	Reg1 := Reg2 xor Imm
	ori andi	Reg1, Reg2, Imm Reg1, Reg2, Imm	Reg1 := Reg2 or Imm Reg1 := Reg2 and Imm
Memory Instructions	lw	Reg1, Reg2, Imm	addr:=Reg2(base)+Imm(offset), Reg1:=(Mem_word[addr])
	lb	Reg1, Reg2, Imm	addr:=Reg2(base)+Imm(offset), Reg1:=(Mem_byte[addr])
	sw	Reg1, Reg2, Imm	addr:=Reg2(base)+Imm(offset), Mem_word[addr]:=Reg1
	sb	Reg1, Reg2, Imm	addr:=Reg2(base)+Imm(offset), Mem_byte[addr]:=Reg1
	flush	Reg, Imm	addr:=Reg(base)+Imm(offset), flush_cashline(addr)
Branch Instructions	beq	Reg1, Reg2, Label	pc:=Label if Reg1==Reg2, else pc+=1
	bne	Reg1, Reg2, Label	pc:=Label if Reg1!=Reg2
	bltu	Reg1, Reg2, Label	pc:=Label if u(Reg1)<u(Reg2)
	bleu	Reg1, Reg2, Label	pc:=Label if u(Reg1)<=u(Reg2)
	bgtu	Reg1, Reg2, Label	pc:=Label if u(Reg1)>u(Reg2)
	bgeu	Reg1, Reg2, Label	pc:=Label if u(Reg1)>=u(Reg2)
	blts	Reg1, Reg2, Label	pc:=Label if s(Reg1)<s(Reg2)
	bles	Reg1, Reg2, Label	pc:=Label if s(Reg1)<=s(Reg2)
	bgts	Reg1, Reg2, Label	pc:=Label if s(Reg1)>s(Reg2)
	bges	Reg1, Reg2, Label	pc:=Label if s(Reg1)>=s(Reg2)
Special Instructions	rdtsc	Reg	Reg:=cyclecount(number of exe- cuted execution unit ticks)
	fence	none  2	all instr in the EU unit at the point of issueing the fence are exe- cuted before the fence ist exe- cuted; no new instructions are is- sued before the fence is executed

