به نام خدا

دانشگاه اصفهان

University of Isfahan

پروژه‌ی درس معماری کامپیوتر

استاد زهره بیکی – بهار ۱۴۰۰

پریسا شهابی‌نژاد – ۹۸۳۶۱۳۰۳۲

نگار نادیان – ۹۸۳۶۱۳۰۵۵

پردازنده ۱۶بیتی:

ISA:

R-type:

opcode	rs	rt	rd
4bit	4bit	4bit	4bit

I-type:

opcode	rs	rt	imm
4bit	4bit	4bit	4bit

J-type:

opcode	imm
4bit	12bit

Registers:

Reg num	name	usage	Value assigned(4bit)
0	$z0	always zero	0000
1	$t0	temporary	0001
2	$t1	temporary	0010
3	$t2	temporary	0011
4	$t3	temporary	0100
5	$s0	save	0101
6	$s1	save	0110
7	$s2	save	0111
8	$s3	save	1000
9	$s4	save	1001
10	$s5	save	1010
11	$sp	stack pointer	1011
12	$v0	return funct value	1100
13	$ra	return address	1101
14	$a0	arg to funct	1110
15	$a1	arg to funct	1111

Instructions:

Instruction type		Instruction	opcode
R-type	arithmetic	Add	0000
	arithmetic	sub	0001
	logical	and	0010
	logical	or	0011
	logical	nor	0100
	logical	xor	1110
I-type	data transfer	lw	0101
	data transfer	sw	0110
	conditional branch	beq	0111
	arithmetic	addi	1001
	logical	andi	1010
	logical	ori	1011
	logical	sll	1100
	logical	srl	1101
J-type	jump	j	1111

Syntax:

add: add $s0,$s1,$t0

sub: sub $s0,$s1,$t0

sll: sll $s0,$t0,imm

and: and $s0,$s1,$t0

beq: beq $s0,$t0,L

lw: lw $s0,$s1,offset

addi: addi $s0,$s1,$t0

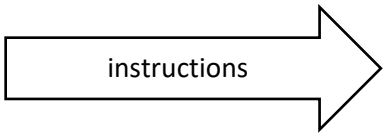
j: j,L

Control signals:

operation	opcode	REG write	REG dst	ALU src	Mem read	ALU op	Branch	jump	load	store	B'	Carry in	L/R	Shift
add	0000	1	0	0	0	100	0	0	0	0	1	0	0	0
sub	0001	1	0	0	0	100	0	0	0	0	0	1	0	0
and	0010	1	0	0	0	000	0	0	0	0	1	0	0	0
or	0011	1	0	0	0	001	0	0	0	0	1	0	0	0
nor	0100	1	0	0	0	010	0	0	0	0	1	0	0	0
lw	0101	1	1	1	1	000	0	0	1	0	0	0	0	0
sw	0110	0	1	1	1	000	0	0	0	1	0	0	0	0
beq	0111	0	0	0	0	000	1	0	0	0	0	1	0	0
addi	1001	1	1	1	0	100	0	0	0	0	1	0	0	0
andi	1010	1	1	1	0	000	0	0	0	0	1	0	0	0
ori	1011	1	1	1	0	001	0	0	0	0	1	0	0	0
sll	1100	1	1	0	0	000	0	0	0	0	0	0	0	1
srl	1101	1	1	0	0	000	0	0	0	0	0	0	1	1
xor	1110	1	0	0	0	011	0	0	0	0	1	0	0	0
jump	1111	0	0	0	0	000	0	1	0	0	0	0	0	0

Example: fibonacci code for calculate 5th sequence:

```
addi $t0,$z0,1
addi $t1,$z0,1
addi $t2,$z0,5
addi $s0,$z0,1
beq $s0,$t2,5
addi $s0,$s0,1
add $t3,$z0,$t1
add $t1,$t1,$t0
add $t0,$z0,$t3
j 4
sw $t1,$s0,2
```




```
9011
9021
9035
9051
7355
9551
204
122
401
f004
6522
6522
```