Alan Achtenberg

CSCE 350

HW 1

2.10

A[0]=&A[1]

F= &A[0]+&A[1]

2.11

OP RS RT RD/immediate decimal

Addi 8 22 8 4

Add 0 22 0 9

Sw 43 8 9 0

Lw 35 8 8 0

Add 0 9 8 16

2.12.1

0x1500000

2.12.2

the result is the desired, no overflow

2.12.3

0xffb00000

2.12.4

There has been overflow, because s1 is bigger than s0

2.12.5

0x1d00000

2.12.6

The result is as desired, no overflow

2.13.1

128=0x80, values of s1 > 0xffffff7f will result in overflow

2.13.2

128=0x80, values of s1 > 0x80 will result in overflow

2.13.3

128=0x80, values of s1 < 0x80 will result in overflow

2.14

Op code is 000000 =add which is register type

Add $16, $16, $16= Add $s0, $s0, $s0

2.15

Sw $t1, 32($t2) is immediate type , 43 10 9 32 dec

Hex representation= 0xad490020

2.19.1

Sll discards all t0

Value is 0x12345678

2.19.2

Sll discards 4 bits t2=0xaaaaaaa0

-1=0xFFFFFFFF in twos complement

So andi leave t2=0xaaaaaaa0

2.19.3

t2= 0x5545

2.21

Use exclusive or with bit string of 1’s ex. 1010 xor 1111=0101

Addi $t3, $t2, 0

Slr $t3, 16

Xori $t1, $t2, 0xFFFF //bottom half

Xori $t3,$t3, 0xFFFF //upper half

Sll $t3, 16

Or $t1,$t1, $t3

2.23

T0 is greater than 0 so t2 is set to 1

T2 is not equal to 0 so addi 3

t2= 0x3

2.26.1

T2 is set to 1

Loop is entered 10 times, s2=20

2.26.2

While(1){

If(0<t1){

t2=1;}

If (t2==0){

Break;}

t1-=1;

s2+=2

}

2.26.3

For case 1 its 5 instructions plus 1 for ending loop

For case 0 its 1 for ending loop

For k its 5k+1

N\*5+1

2.29

i=0;

S0=0;

Do while(t2!=s0){

s1=\*s0;

s2+=s1;

++s0;

++I;

If (i<100)

t2=1;

}

2.47.1 70% a 2 10% load/store 6 20% branch 3

2\*.7+6\*.1+3\*.2=2.6 CPI

2.47.2

25% improvement=2.6-2.6\*.25= 1.95 CPI

1.95-(.6)-(.6)=.7\*x

X=1.07 cycles on average

2.47.3

50% improvement=2.6-1.3=1.3 CPI

1.3-1.2=.1=.7x

X=.14 cycles on average, only possible if multiple arithmetic operations are available per cycle