CS2318

9/13/16

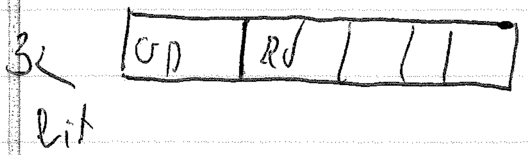
"Real" instructions

↓

Machine code

```
Add $7, $8, $9
```

↓



Pseudo instruction

↓

Real instruction

↓

Machine code

```
mov $5, $6
```

↓

```
or $5, $6, $0
```

↓

Machine code

2

~~if i =~~   s0   s1   ↙ set k

```
     ↑    ↑
While (i == k) {
  i = i+1  }

while (B-Exp) {
  Exp
  }
```

You check for !(B-exp)

```
_while0: bne $s0, $s1, exit
    [Exp]
    addi $s0, $s0, 1        inc
    b _while0
Exit:
```

3 while $(\overset{s_0}{i} < \overset{s_1}{K})$

$i = i+1$

$!(i < k) \Rightarrow (i \geq k)$

```
_whilen  bge $s0, $s1, exit
         inc $s0
         b _wilen
exit
```

BGE
BLT
BLE
BGT
BZ

BEQ BNE SLT SLTi

```
blt $7, $8, exit
```

if ($7 < $8) go to exit

```
4    slt $Rd, $Rs, $Rt

     slt $1, $2, $7

     $1 ← 0      ~ cond

     if ($2 < $7)  $1 ← 1
```

$1 0 condition does not hold

$1 1 " holds

```
             BLT
     blt  ~~$t00~~  $2, $7, exit
                                     $1 ← 1
     slt  $1, $2, $7                 only if
                                     $2 < $7

     bne $1, $0,
```

5 blt $2, $7, exit10

→ Beq

→ SLT $\boxed{\$1}$ $2, $7   2 inst

→ bne $1, $0, exit10

bgt   SLT Set on Less Than

blti → when you are

a Human compiler

playing the role of assembler

abs $2, $3

⇒ default assume $3 ≥ 0

does not hold ?

then correct

6

constant

~~SLEi $10, $9, out~~

~~Assume default~~

SLEi $1, $5, 20

Assume default

Fix if wrong

SLEi $t0, $t1, 20

ABS $5, $6

↓ Use $1 ($at) at your discresion

↑ dst can be used as a second tempory

7 MIPS Instruction

Britton   List of pseudo inst.

List of Real inst.

(List of directives)

add   addi

sub   subi

MIPS   3 Formats   5 family

R   –

I < Arith, Immediate
Conditional Branch
Load/store

J   Jump   J   Jal

8) Branch ~~(Jump)~~

conditional   un conditional

1) BEQ BNE   1) B

1) BLT

1) BZ   2) J (JAL)

3) JR

BEQ $5,$19, out

6	5	5	16
'BEQ'	R_t	R_s	constant
	19	5	

B is Relative to pc

$2^{15}-1$ Forwar

PC → B 100000 B out

→

2^{15} backward

~~1000000 BEQ $1,$2, out~~ $2^{15}-1$ Back

2^{15} Forw

out $2^{15}-1$ For

B 1000000   BEQ $1,$2,$3

2^{15} Back

↑

words

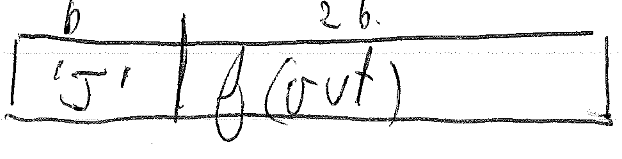
Great for Loops if then

Relocatable (no need to to use Loader)

B   Limited

J

J out



2^{26} words

10) Mem is divided into 16 segments each of which is 2^{26} words

J can jump to any address in the segment

JR $5

$PC \leftarrow \$5$
↑

J out $\quad$ $PC \Leftarrow f(out)$

B out $\quad$ $PC \Leftarrow f(out)$

JR $R $\quad$ $PC \Leftarrow \$R$