

CS2318

Accessing Memory \leftarrow pointer
(address)

09/08/16

WORD ACCESS

LW - Load Word

SW - store word

LW \$Rd, offset(\$Rd) ↑ pointer

LW \$Rd ↑ pointer
(X)

LW \$5, 20(\$7)

(\$7) - Effective operand

Memory pointed to by \$7

M[\$7]

LW \$5, 20(\$7)

\$5 \leftarrow M[\$7 + 20]

2

$\text{sw } \$R + \text{ offset}(\$R_5) \quad \begin{matrix} \text{static} \\ \text{dynamic} \end{matrix}$

$M[\$R_5 + \text{offset}] \leftarrow \$R +$

$LW \$6, 200(\$11) \quad \$6 \leftarrow M[11 + 200]$

$\text{sw } \$6, 200, (\$11) \quad M[11 + 200] \leftarrow \6

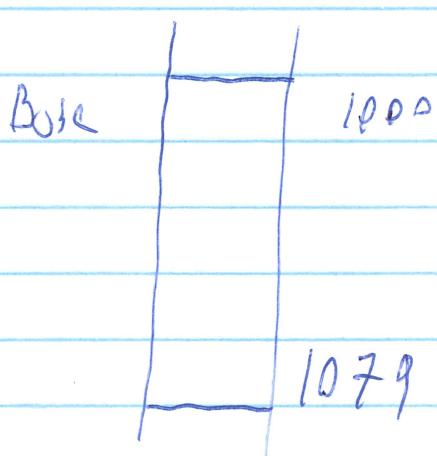
Offset is a pos / neg byte offset

int A[20];

$g = h + A[8];$

$\$S1 + \$S2 \quad A$

Base in $\$S3$
1000



lw \$t0, 32(\$S3)

$\$t0 \leftarrow M[32 + \$S3] = M[1032]$

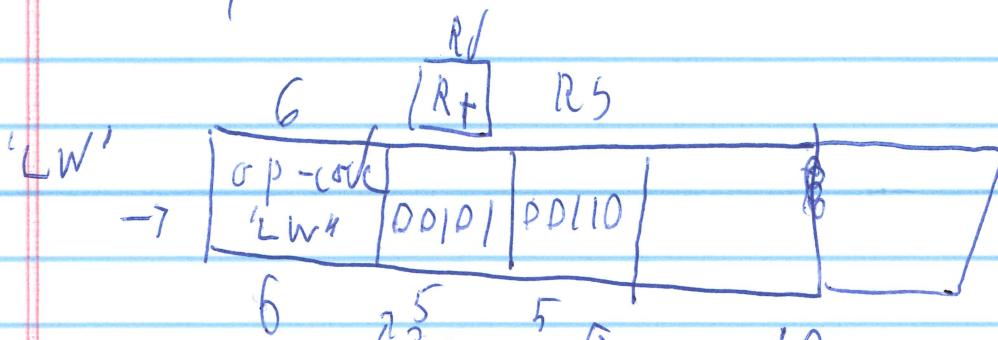
Add $\$S1, \$L, \$t0$

3

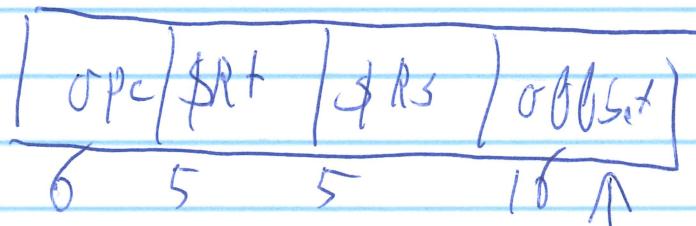
int A[20]

$$A[6] \leftarrow h + A[10]$$

↑ |
64 40



LW \$5, ~~6~~ 200(\$6)



offset is 516
 constant
 data
 Immediate

4

I type instructions I-Format

why I? contains Immediate within
the instruction

I - Format

1) LW / SW

2) Arith immediate

3) Conditional change of flow

Arithmetic Immediate

add \$5, \$6, \$7 $\$5 \leftarrow \$6 + \$7$

addi \$5, \$6, 75 $\$5 \leftarrow \$6 + 75$

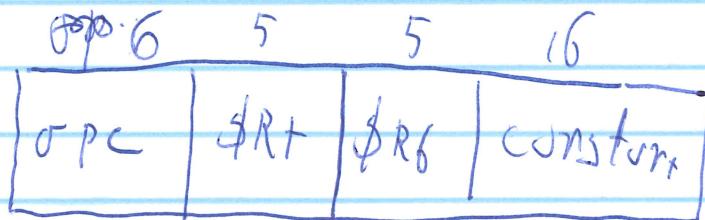
li \$5, 1 Load immediate

$\$5 \leftarrow 1$ `loadi $5, $0, 1`

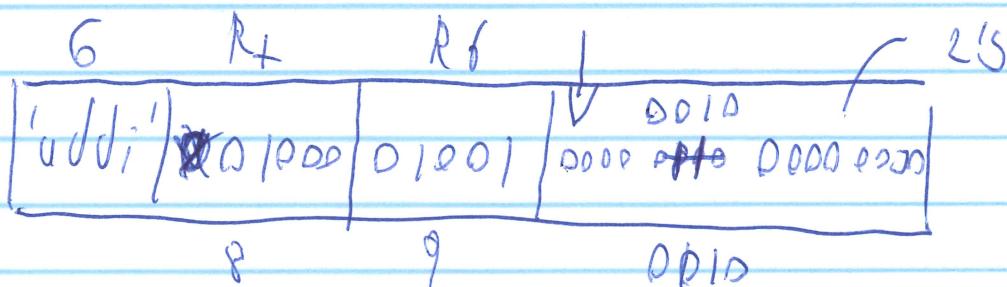
5 op*i* add*i* [sub*i* \$5,\$17,20
 or*i* add*i* \$5,\$17 -L0
 xor*i* sub*i* \$5,\$6,~1
 ... ou*lli* \$5,\$6,1

op*i* \$R_t, \$R_s, constant

\$R_t ← \$R_s op constant



add*i* \$8,\$9,0X200



S16 add*i* signed

U16 or*i* xor*i* Unsigned

6

~~Condition branch~~

→ Pseudo inst for Real inst

→ #define PI 314159

add \$5, \$6, \$7 => some
add \$9, \$10, -20

Conditional Branch

gray BEQ

BEQ \$R_t, \$R_s, target

BNE

BNE \$R_t, \$R_s, TARGET

7 1000 BEQ \$5, \$10, out
 1004 [↑ Next inst
 1020 out; might change pc

Target is a function of Label

if ($C \$5 == \10) go to out

otherwise execute the next inst (1004)

$i \neq j$ $f = g + h;$

else $f = g - h;$

BNE \$50, \$51, - ElseD
~~\$10 \$2, \$53, \$54~~

→

- ElseD SUB \$53, \$54

8

out: buy \$0, \$0, out infinite loop

buy \$0, \$0, out

out:

unconditional

for out ≡ buy \$0, \$0, out

if (i == j) f = g + h

else f = g - h

lne \$50, \$1, ; Else

Add \$50, \$51, \$52

f - Exit 0

~ Else Sub \$50, \$51, \$52

- Exit 0

9) if ($a > b$) $a \leq b$ $a \geq b$

can be done using SLT

SLT \$R_1, \$R_2, \$R_3

$R_f \leftarrow 1$ if \$R_2 < \$R_3

On.W. $R_f \in 0$