

Cálculo em

Vírgula

Flutuante no

MIPS

- O MIPS inclui um coprocessador aritmético (Coprocessador 1) capaz de efetuar operações aritméticas em vírgula flutuante

tem o seu próprio espaço de armazenamento composto por um conjunto de 32 registros de 32 bits cada, e o seu próprio conjunto de instruções (ISA).

⇒ FPU : Floating Point Unit

- Cada par de registros consecutivos $[\$f_n, \$f_{n+1}]$ (com n par) pode funcionar como um registro de 64 bits para armazenar valores em precisão dupla.

- Registro por ($\$10, \$12, \$14, \dots$)

- Suffixes:

↳ Presumed samples: 0.8

↳ 4 displa : - d

- Conversões: não efetuadas pela FPU

↳ r: float (single)

6 d: double

↳ w : integers

- Instruções de Transferência

mtc1 CPUsrc, FPdst \Rightarrow mtc1 \$t0, \$f4
 ↓ ↓ move (+t0)
 registro do registro da Coprocessor 1
 CPU FPU

↓
registro do
CPU

↓
registro da
FPU

move (+)
Copro cernor 1

mfc1 CPUdst, FPdst \Rightarrow mfc1 \$a0, \$f6
move from coprocessor 1

move from processor 1

movl, %FPdst, %FPsrc \Rightarrow movl, %\$4, %\$8
 \hookrightarrow single

↳ single

mov.d FPdst, FPsrc \Rightarrow mov.d \$f2, \$f0
 \hookrightarrow double

↳ double

Copiam o conteúdo integral do registro fonte para o registro destino

↓

Transferência de informação entre registros do CPU e da FPU, e entre registros da FPU

l.s FPdst, offset(CPUreg) \Rightarrow l.s \$f0, 4(\$a0)
 registro do FPU endereço de memória load float from memory

s.s FPsrc, offset(CPUreg) \Rightarrow s.s \$f0, 0(\$a0)
 store float into memory

l.d FPdst, offset(CPUreg) \Rightarrow l.d \$f4, 0(\$a1)
 load double from memory

s.d FPsrc, offset(CPUreg) \Rightarrow s.d \$f4, 16(\$a0)
 store double into memory

Transfereência de informação entre registros do FPU e a memória

Exemplo:

float a, b;

...

if (a > b) a = a + b;

else a = a - b

#a: \$f0 b: \$f2

...

if:

c.le.s \$f0, \$f2 if (a > b)

bc1t else

add.s \$f0, \$f0, \$f2 a = a + b

j endif

else:

else

sub.s \$f0, \$f0, \$f2 a = a - b

endif:

...

• Registos para passar parâmetros para sub-rotinas (do tipo float ou double): \$f12, \$f14

• Registos para devolução de resultados das sub-rotinas: \$f0

• Registos que podem ser livremente usados e alterados pelas sub-rotinas ("caller-saved"): \$f0 a \$f18

• Registos que não podem ser alterados pelas sub-rotinas ("callee-saved"): \$f20 a \$f30