

2023.3.21

- 3) addi x0,x0,0 2) jalr x0,x0,0
- 3) auipc x1,offset[31:12] jalr x1,x1,offset 4) addi rd,rs,0
- 5) srrs rd,cycle[h],x0 6) addiwn rd,rs,0
- 7) 1) slti t3,t2,0 slt t4,t0,1

- 2) addu t0,t1,t2 btu t0,t1,overflow

3) 在x86架构中,当执行算术指令时条件代码将设置在CPU状态字中,有些指令执行与否取决于条件代码。

在ARM架构中,通过CPSR的状态寄存器反映当前指令的溢出状态。

- 8) 1) \pm \times $-$ \times

不会抛出异常,结果会是Infinity。

2) M代表invalid operation, DZ代表divide by zero, OF代表overflow, UF代表underflow, NX代表inexact。

不会陷入系统调用。

3) 在x86架构中会产生0号中断,会产生Core Dump,可由程序自行处理,或运行库压制或运行时包装并抛出。

在ARM架构中,会进入HardFault中断。

- 12) 1) 管理员模式 2) 机器模式 3) 机器模式
- 4) 用户模式 5) 用户模式

B.vecMul:

addi sp, sp, -32

sw ra, 4(sp)

sw s0, 4(sp)

addi s0, sp, 32

sw a0, -12(s0)

sw a1, -16(s0)

sw a2, -20(s0)

li a0, 0

sw a0, -4(s0)

loop:

lw a1, -4(s0)

li a0, 99

blt a0, a1, end

lw a0, -16(s0)

lw a1, -4(s0)

sll a2, a1, 2

add a0, a0, a2

lw a0, 0(a0)

lw a1, -20(s0)

mul a0, a0, a1

lw a1, -12(s0)

add a1, a1, a2

sw a0, 0(a1)

lw a0, -24(sp)

addi a0, a0, 1

sw a0, -24(sp)

j loop

end:

lw a0, -12(sp)

lw a0, 0(a0)

lw ra, 8(sp)

lw s0, 24(sp)

addi sp, sp, 32

ret

16. swap: addi sp, sp, -32

sw ra, 28(sp)

sw s0, 24(sp)

addi s0, sp, 32

sw a0, -12(s0)

sw a1, -16(s0)

lw a0, -12(s0)

lw a0, 0(a0)

sw a0, -20(s0)

lw a0, -16(s0)

lw a0, 0(a0)

lw a1, -12(s0)

sw a0, 0(a1)

lw a0, -20(s0)

lw a1, -16(s0)

sw a0, 0(a1)

lw ra, 28(sp)

lw s0, 24(sp)

addi sp, sp, 32

ret

15. start: sw to, 0(to)

addi t1, x0, 3

sw t1, 4(to)

slli t2, t1, 2

add t2, t2, t1

sw t1, 0(t2)

17. $A_1 = 2^{30}$

14. start: beg a1 a0, if

add a2, a0, a1

1. sub a2, a0, a1