

Date. Page. 3. 7. 8. 12. 13. 14. 15. 16. 17

第五周

3. 1) addi x0, x0, 0

2) jalr x0, x1, 0

3) auipc x6, offset[31:12]

jalr x1, x6, offset[11:0]

4) addi rd, rs, 0

5) rdcycle to

addi rd, to, 0

6) slli rd, rs, 32

srai rd, rd, 32

7. 1) sub t3, to, t1

mv t4, t2

2) add to, t1, t2

slt t3, to, t1

bne t3, x0, overflow.

3) 在X86架构中，一般采用ZF标志位，是在执行加减指令后由处理器设置的一种标志位，如果ZF被设置为1，则标志溢出。

在ARM架构中，一般采用C和V两个标志位来检测加法是否溢出。C标志指示无符号加法是否发生进位，V标志位用于指示有符号加法是否发生溢出。

8. 1) $Op = DIVU$ 时 $rd = 0xFFFFFFFFFFFFFFFFF$

$Op = REMU$ 时 $rd = 0$

$Op = DIV$ 时 $rd = 0xFFFFFFFFFFFFFFFFF$

$Op = REM$ 时 $rd = 0$

2) NV: 发生了某种不支持的操作

DZ: 发生了某种动态非数值情况。

OF: 计算结果溢出了

UF: 计算结果下溢

NX: 执行了无效操作

当 fflags 被置位时，不会直接导致处理器陷入系统调用

而是可以通过指令跳转或处理器异常。在 RISC-V 中，浮点异常处理需要通过相应的浮点异常指令来进行显式处理。

3) 在 X86 架构中，整数除法使用 idiv 指令，被除数有符号

将相应浮点异常标志位置 1。ARM 也置 1，但由

UDIV 和 SDIV 指令执行。

12. 1) 管理员模式

2) 机器模式

3) 管理员模式

4) 管理员模式

5) 用户模式

13. addi a₂, x₀, 1

addi a₃, x₀, 100

loop: bge a₂, a₃, exit

slli a₄, a₂, 2

add a₅, a₄, t₀

add a₆, a₄, t₁

lw a₅, 0(a₅)

lw a₆, 0(a₆)

mul a₅, a₆, t₂

sw a₅, 0(a₅)

addi a₂, a₂, 1

j loop

exit: lw t₀, v1t₀)

ret

14. bgt a₀, a₁, big

j small

big: add a₂, a₀, a₁

small: sub a₂, a₀, a₁

15. sw to, 0(t0).

addi t1, x0, 3.

sw t1, 41 to1.

sw t1, 121 to1.

16. mv t2, t0

mv t0, t1.

mv t1, t2.

17. 将寄存器 a. 的值赋为 2^{30}