

11. 1) 立即数寻址模式

2) 偏移量寻址模式

3) 立即数寻址模式

4) 寄存器寻址模式

5) 寄存器间接寻址模式

3. (1) nop 为 ADDI x0, x0, 0

(2) ret. 为 jalr x0, 0(x1)

(3). call offset 为 auipc xb, offset[31:12]

jalr x1, xb, offset

(4) mv. rd, rs 为 addi rd, rs, 0

(5) rdcycle 为 csrrs rd, cycle[th], x0

(6) sext.w rd, rs 为 addiw rd, rs, 0

7. (1). slt. t3, r\$0, r\$1

slt t4, r\$0, r\$2

(2). bgt t0, t1, overflow.

(3). ARM 通过 CPSR 寄存器反映当前指令的溢出状态

MIPS 通过指令触发中断的方式产生溢出信号。

8. (1) $2^{XLEN} - 1 - x$

不会抛出异常，使整个架构更为轻巧。

(2). NV. Invalid operation.

DZ Divided by Zero

OF Overflow

UF Underflow

不会使得处理器陷入系统调用

13) 除数为零产生“除以零”异常

12 (1) M (2) N (3) S (4) S (5) U

13. part: add t₃ a₃, a

part:

add t₃, x₀, x₀

addi t₄, x₀, 100

Loop: bge t₃, t₄, exit

mula t₅, t₂, 0(t₁)

add t₀, x₀, t₅ 重叠区域同上

lw t₀, 4(t₀)

lw t₁, 4(t₁)

addi t₃, t₃, 1

j loop

14. bge a₀, a₁, more, less.

add a₂, a₀, a₁

leg: sub x a₀, a₁

int x lw a₂, 0(a₀)

15. add $t_0, x_0, 0(t_0)$. add x_3, x_0, t_0
~~addi $t_1, x_0, 3$~~ sw $x_3, 0(t_0)$
~~(lw $1(t_0), t_1$~~ sw $t_1, 4(t_0)$.
~~(lw $3(t_0), t_1$~~ add t_0, t_1, t_0
sw $t_1, 0(t_0)$.

16. ~~lw $x_3, 0(t_0)$~~
~~lw $x_3, 0(t_1)$~~
~~sw $t_2, 0(t_0)$~~
→ lw $x_3, 0(t_0)$
lw $x_4, 0(t_1)$
lw t_0 → sw $x_3, 0(t_1)$
lw t_1 → sw $x_4, 0(t_0)$

17. 将 a_i 中的数向左移 1 位并重复 30 次即令 a_i 乘以 2^{30}