

9.1) jal 指令包含 20 位的有符号立即数编码, 其中最高位是符号位。该指令会将当前 PC (程序计数器) 的值加上符号扩展后的偏移量, 然后将结果存在 ra 中, 并将 PC 的值设置为跳转目标的地址。

因此, 可以访问到地址空间的任何位置。

2) bne 包含 12 位的有符号立即数编码, 其中最高位是符号位, 这类指令可以跳转到相对于当前 PC 偏移量确定的地址。

可以访问到相对于当前 PC 偏移量确定的地址空间。

3) 不可以。

10. 条件(以下同时满足):

1. 指令需是支持 RVC 压缩的指令类型之一 (即 load/store 型指令、ALU 型指令、Branch 指令、Jump 指令)

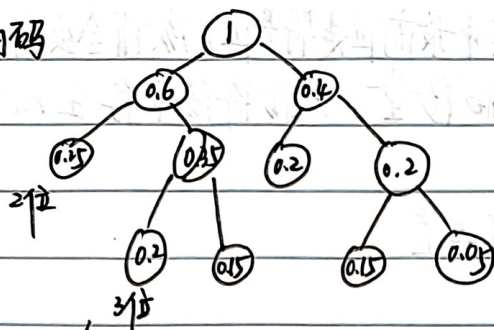
2. 指令的操作码、目标寄存器等信息能通过 16 位编码表示

3. 指令的参数值通过压缩后可以用指令中的字段表示。

是。



18. 霍夫曼编码



平均长度: $\sum_{i=1}^6 p_i \cdot l_i = 3 \times 0.55 + 0.45 \times 2 = 1.65 + 0.9 = 2.55$

操作码的最短平均长度: $H = -\sum_{i=1}^6 p_i \cdot \log_2 p_i = 2.466$

信息冗余量: $R = 1 - \frac{H}{\sum_{i=1}^6 p_i \cdot l_i} = 1 - \frac{2.466}{2.55} = 3.29\%$

19. 1) 函数调用栈是函数的活动记录, 每一次函数的调用, 都会在调用栈上维护一个独立的栈中帧。而内存是有限的, 所以层数过多时, 栈会溢出。

2) 例如将函数的不断的递归, 修改为将循环等递归调用的部份放在另一个函数, 使得占用的栈可以及时得到释放, 而没有过多的尾递归优化, 在函数返回的时候, 调用自身本身, 并且 return 语句不包含表达式, 从而递归本身无论使用多少次, 都只使用一个栈中帧, 不会出现栈溢出的情况

20.

ra(F1)

ra(F2)

a1(F2)

a0(F2)

s1(F2)

t1(F2)

t0(F2)

ra(F1)

其他可能的局部变量或参数或中间值

ra(F3)

so(F3)

##

扫码使用

夸克扫描王

