

RISC-V指令特点

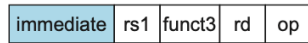
xyfjASON

- 由于 RISC-V 中寄存器的大小是 64 位，成组的 64 位字会频繁出现，故一般称 64 位（8 个字节）为一个双字，32 位（4 个字节）为一个字，16 位（2 个字节）为一个半字。
32 位一个字是一般而言的，如果上下文有专门定义应以上下文为准。
- RISC-V 有 32 个寄存器，命名为 x0, x1, ..., x31. 约定为：

寄存器	助记符	释义
x0	zero	固定值为0
x1	ra	返回地址(Return Address)
x2	sp	栈指针(Stack Pointer)
x3	gp	全局指针(Global Pointer)
x4	tp	线程指针(Thread Pointer)
x5-7	t0-2	临时寄存器
x8	s0/fp	save寄存器/帧指针(Frame Pointer)
x9	s1	save寄存器
x10-11	a0-1	函数参数 / 函数返回值
x12-17	a2-7	函数参数
x18-27	s2-11	save寄存器
x28-31	t3-6	临时寄存器

- RISC-V中，只有存/取指令（load/save）才能访问内存，其余指令只能在寄存器之间进行操作。
- 如无特殊说明，几乎所有的体系结构都是按字节寻址的，因此，连续双字的地址相差 8。
- RISC-V 有 4 种寻址方式：
 1. 立即数寻址：操作数是指令本身的常量
 2. 寄存器寻址：操作数在寄存器中
 3. 基址或偏移寻址：操作数在内存中，其地址是寄存器和指令中的常量之和
 4. PC 相对寻址：分支地址是 PC 和指令中的常量之和

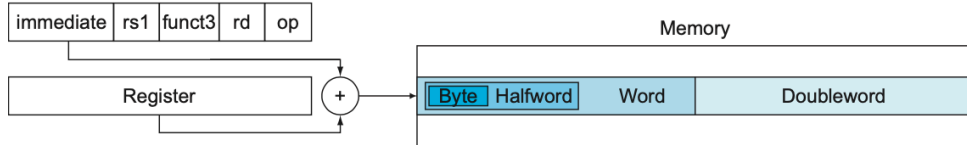
1. Immediate addressing



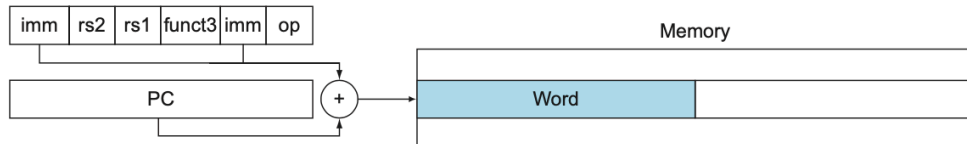
2. Register addressing



3. Base addressing



4. PC-relative addressing



- J 型指令格式与 U 型很像（仅仅立即数顺序不同），因而也叫做 UJ 型；B 型指令格式与 S 型很像（仅仅立即数顺序不同），因而也叫做 SB 型。
- 只有一个 J 型指令，即 `jal`，其作用是把 `PC+4` 存入寄存器 `R[rd]`，然后 `PC=PC+{imm, 1'b0}` 跳转到新的地址。J 型指令的 `imm` 有 20 位，并且省略了最后一位的默认 0，因此可以跳转的相对范围是 $[-2^{20}, 2^{20} - 2]$ 。
- 只有两个 U 型指令，分别是 `lui` 和 `auipc`。

`lui` 作用是加载 `R[rd]={32b'imm<31>, imm, 12'b0}`，即符号扩展前 32 位，低 12 位置零。

`auipc` 作用是 `R[rd]=PC+{imm, 12'b0}`。

- R `rd, rs1, rs2`
- I `rd, rs1, imm`
- I `rd, offset(rs1)` (`load`、`jalr`)
- save `rs2, offset(rs1)`
- B `rs1, rs2, offset`
- U `rd, imm`
- J `rd, imm`