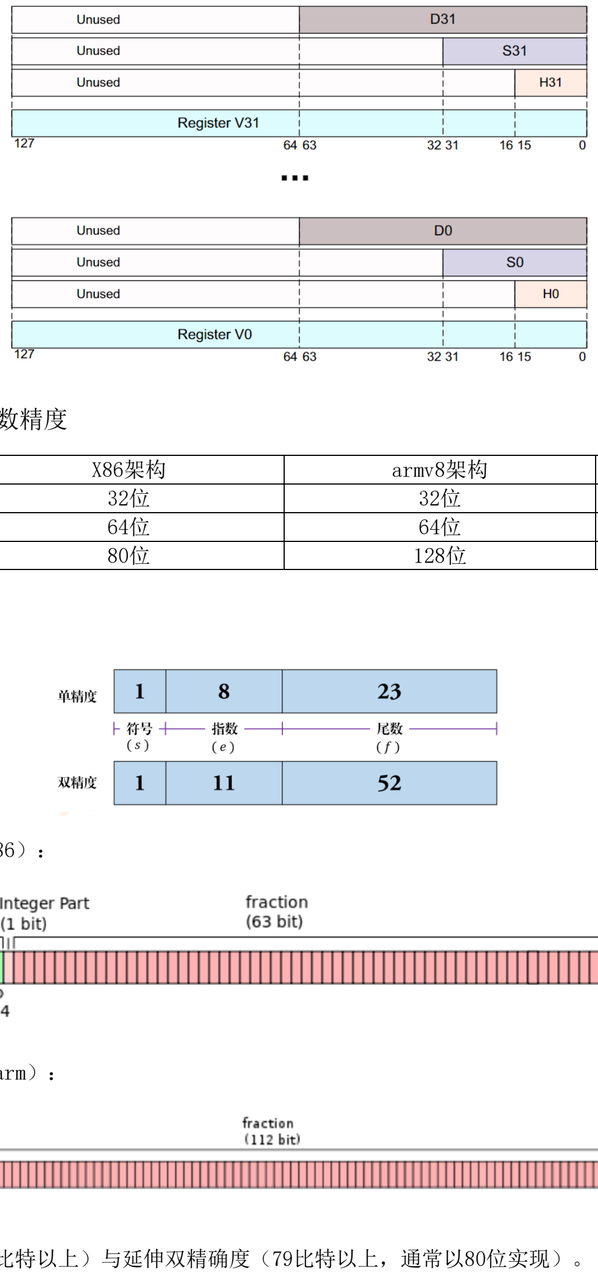
x86架构和armv8架构比对

一、浮点寄存器：

1. x86的FPU（浮点运算单元）中包含8个80位可以直接进行浮点运算的寄存器R0-R7，浮点数以双精度格式存储在寄存器中；浮点寄存器包括8个80位的通用寄存器，两个48位寄存器（指令指针寄存器和数据指针寄存器），三个16位寄存器（控制寄存器、状态寄存器和标志寄存器）。
2. armv8寄存器：AArch64的NEON架构使用32×128位寄存器；单精度浮点值使用低位32位，而双精度值使用128位寄存器的低位64位。在对标量数据进行操作的 NEON 和浮点指令中，浮点和 NEON 寄存器的行为与主要通用整数寄存器类似。因此，仅访问较低位，而未使用的高位在读取时被忽略并在写入时设置为零。

- 32×64位D寄存器D0-D31。D寄存器称为双精度寄存器，包含双精度浮点值
- 32×32位S寄存器S0-S31。S寄存器称为单精度寄存器，包含单精度浮点值
- 32×16位H寄存器H0-H31。H寄存器称为半精度寄存器，包含半精度浮点值
- 如下图所示，



二、x86、armv8、IEEE754下浮点数精度

	X86架构	armv8架构	IEEE754标准
单精度浮点数	32位	32位	32位
双精度浮点数	64位	64位	64位
长双精度浮点数	80位	128位	80位

1. 单精度和双精度格式的位模式：

2. 长双精度（80位）格式的位模式（x86）：

3. 长双精度（128位）格式的位模式（arm）：

4. IEEE754标准中，延伸单精确度（43比特以上）与延伸双精确度（79比特以上，通常以80位实现）。

5.

	+	-
0	0x00000000	0x80000000
无穷	0x7F800000	0XFF800000
NaN	指数全部为1，尾数非0	

三、x86和armv8架构下浮点运算的汇编指令

x86指令	说明	arm指令	说明
finit	初始化控制和状态寄存器	ldr	从内存中加载单精度或双精度浮点数到寄存器中
flds value	加载内存中的单精浮点到寄存器堆栈	ldrh	从内存中加载半精度浮点数到寄存器中
fldl value	加载内存中的双精浮点到fpu寄存器堆栈	ldrsh	从内存中加载有符号的半精度浮点数到寄存器中
fldt value	加载内存中的扩展精度点到fpu寄存器堆栈	ldrb	从内存中加载字节大小的浮点数到寄存器中
fadd	浮点加法	fadd	FP加法指令
fdiv	浮点除法	fsub	FP减法指令
fdivr	反向浮点除法	fmul	FP乘法指令
fmul	浮点乘法	fdiv	FP除法指令
fsub	浮点减法	fsqrt	FP开根号指令
fsubr	反向浮点减法	frcp	FP取倒数指令
fcom	浮点数比较指令	fcmp	FP比较指令