

们也会说明对整个数据向量进行操作的指令出现的情况。后文中的网络旁注 OPT: SIMD 更全面地说明了如何利用 SSE 和 AVX 的 SIMD 功能读者可能希望参考 AMD 和 Intel 对每条指令的说明文档[4, 51]。和整数操作一样, 注意我们表述中使用的 ATT 格式不同于这些文档中使用的 Intel 格式。特别地, 这两种版本中列出指令操作数的顺序是不同的。

如图 3-45 所示, AVX 浮点体系结构允许数据存储在 16 个 YMM 寄存器中, 它们的名字为 `%ymm0~%ymm15`。每个 YMM 寄存器都是 256 位(32 字节)。当对标量数据操作时, 这些寄存器只保存浮点数, 而且只使用低 32 位(对于 `float`)或 64 位(对于 `double`)。汇编代码用寄存器的 SSE XMM 寄存器名字 `%xmm0~%xmm15` 来引用它们, 每个 XMM 寄存器都是对应的 YMM 寄存器的低 128 位(16 字节)。

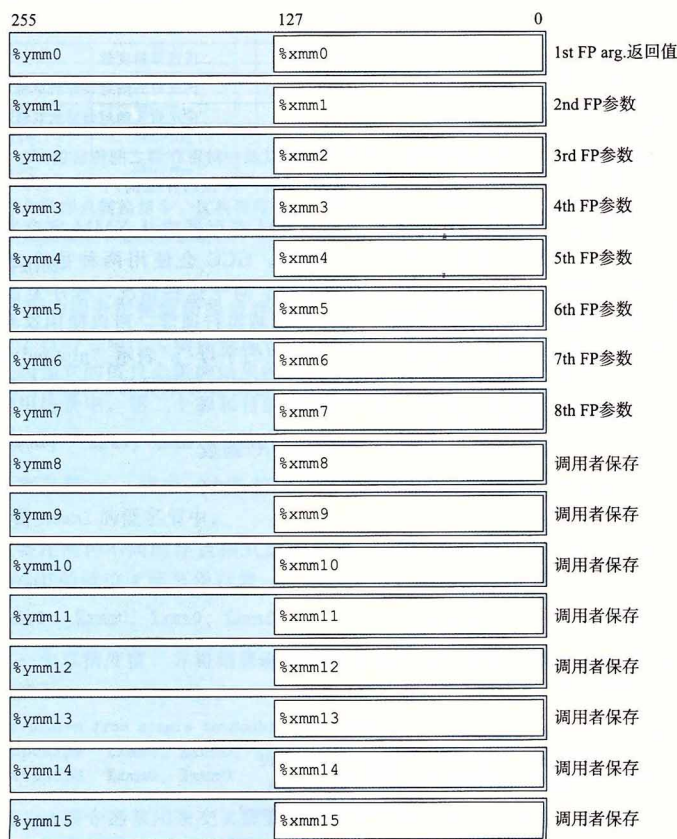


图 3-45 媒体寄存器。这些寄存器用于存放浮点数据。每个 YMM 寄存器保存 32 个字节。低 16 字节可以作为 XMM 寄存器来访问

### 3.11.1 浮点传送和转换操作

图 3-46 给出了一组在内存和 XMM 寄存器之间以及从一个 XMM 寄存器到另一个不