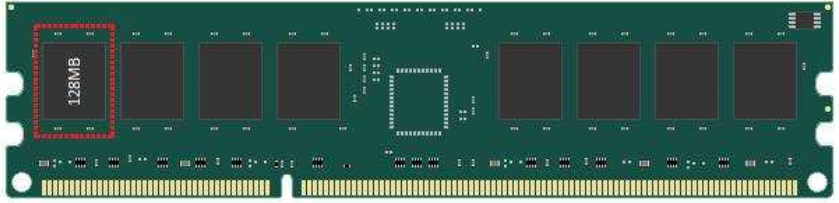
1. **组成**



这是一个2GB的内存条，我们可以看到，内存条的物理组成：金手指、存储芯片、还有各种控制电路和元器件，还有就 PCB板。我们今天要谈的组成不是这些，而是它作为一个数据容器的组成：

**Rank**

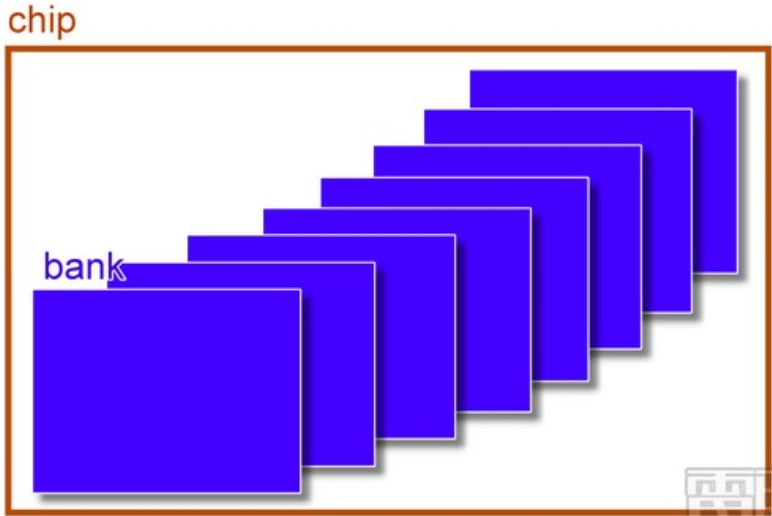
CPU与内存之间的接口位宽是64bit，也就意味着CPU在一个时钟周期内会向内存发送或从内存读取64bit的数据。可是，单个内存颗粒的位宽仅有4bit、8bit或16bit，个别也有32bit的。因此，必须把多个颗粒并联起来，组成一个位宽为64bit的数据集合，才可以和CPU互连。生产商把64bit集合称为一个物理BANK（Physical BANK），简写为P-BANK。为了和逻辑BANK相区分，也经常把P-BANK称为RANK或Physical RANK,把L-BANK则简称为BANK。如果每个内存颗粒的位宽是8bit，应该由8个颗粒并联起来，组成一个RANK（64bit）；同理，如果颗粒的位宽是16bit，应该由4个颗粒组成一个RANK。由此可知：**Rank其实就是一组内存颗粒位宽的集合，也可以叫Chips，在PCB上，往往把一面上的内存颗粒组成一个Rank,另一面是另外一个Rank(假若有的话)，这样也可以将Rank理解内存条的Side(面)**。具体说，当颗粒位宽×颗粒 数=64bits时，这个模组就是有一个RANK。为了保证和CPU的沟通，一个模组至少要有一个RANK。但是，为了保证有一定的内存容量，目前，DDR2内存，经常是采用一个模组两个RANK的架构。“模组构成”中的“R”“RANK”的意思。“2R”是说组成模组的RANK数（Number of ranks of memory installed）是2个。有“1R”和“2R”两种；“模组构成”中的“×8”是颗粒的位宽(bit width)，有×4、×8和×16三种.

**Chip(Memory)**

存储芯片，又叫内存颗粒。真正提供存储的器件。

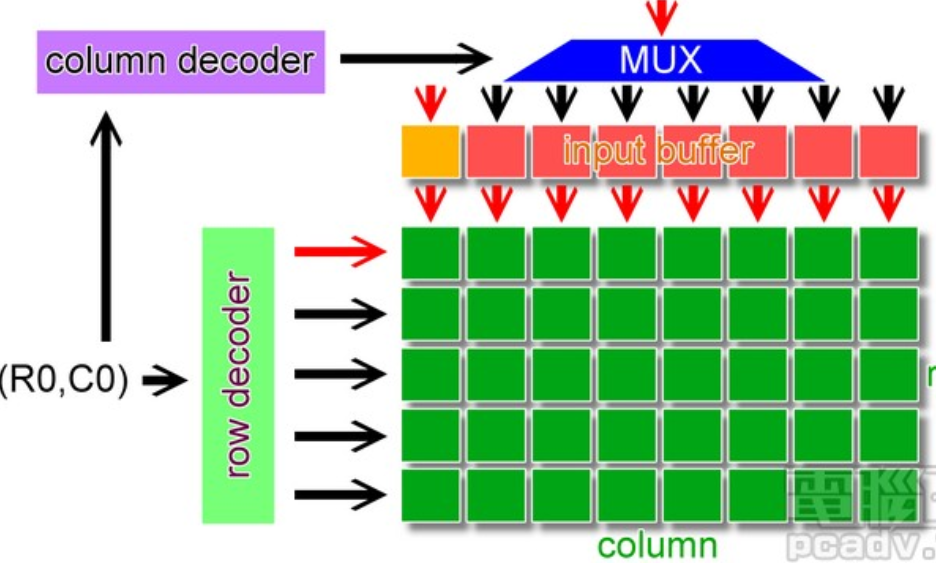
**Bank**

Chip里包含多个Bank。一个Bank就是一个存储矩阵库。由Chip的BA线的位宽决定个数



**Cell**

存储单元，可以保存1bit的数据，Bank由很多的Cell组成，由行列来确定一个Cell。



Rank指的是连接到同一个cs(Chip Select，片选)的所有内存颗粒chips，内存控制器能够对同一个rank的所有chips同时进行读写操作，而在同一个rank的chip也分享同样的控制信号。rank1和rank2共享同一组addr/command信号线，利用cs片选线选择欲读取或是写入的那一组，之后将存储内容经由MUX多路器送出。bank再往下分就是实际存储单位元的电路，一般来说横向选择排数的线路称为row(row enable, row select, word line)，纵向负责传送信号的线路称为column(bitline)，每组bank的下方还会有个row buffer(sense amplifer)，负责将读出的row内容暂存。

1. **编址**

前面我们知道了DRAM颗粒以及内存模块是如何扩展字长和容量的。一个内存可能是8位，也可能是64位，容量可能是1M，也可能是1G。那么内存是如何编地的呢？和地址总线，计算机字长之间又有什么关系呢？内存条容量的大小和它芯片扩展方式有关。比如我们内存模块是采用 16M\*8bit的内存颗粒，那么我们使用4个颗粒进行位扩展，成为16M\*32bit，使用4个颗粒进行字容量扩展变为64M\*32bit。那么我们内存模块使用了16个内存颗粒，实际大小是256MB。

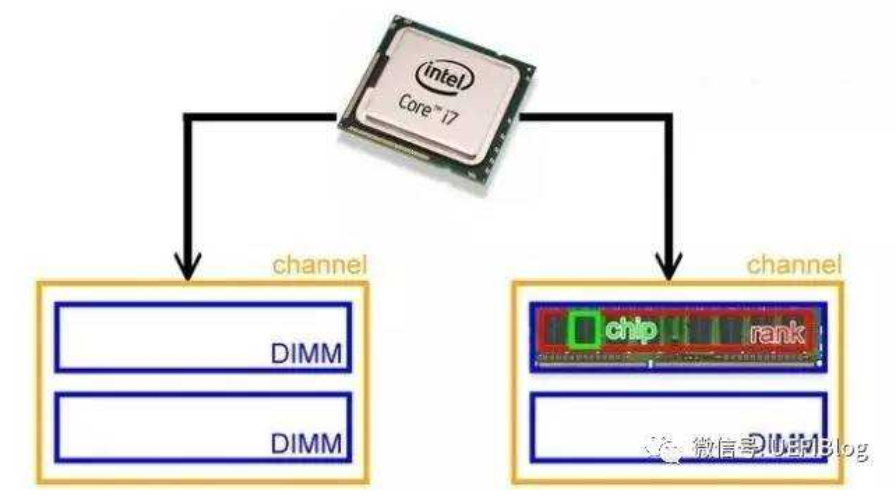
 我们需要对这个256M的内存进行编址以便CPU能够使用它，通常我们多种编址方式：

1. 按字编址：    对于这个256M内存来说，它的寻址范围是64M，而每个内存地址可以存储32bit数据。
2. 按半字编址：对于这个256M内存来说，它的寻址范围是128M，而每个内存地址可以存储16bit数据。
3. 按字节编址：对于这个256M内存来说，它的寻址范围是256M，而每个内存地址可以存储8bit数据。

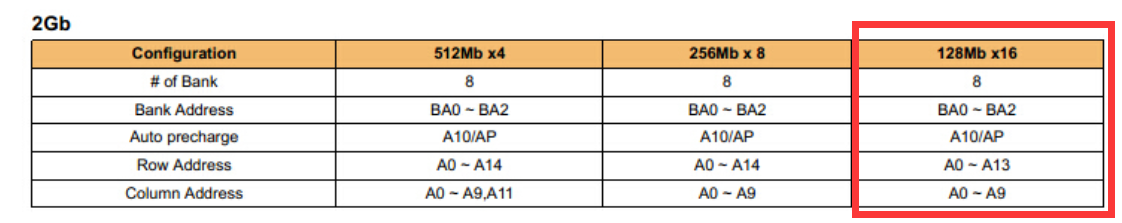
对于我们现在的计算机来说，主要都是采用按字节编址的方式。所以我们可以把内存简单的看成一个线性数组，数组每个元素的大小为8bit，我们称为一个存储单元，对这些存储单元我们从0x00000000开始沿着Rank的访问编址，每8个Cell加1，这样数组里的每个元素都由地址，超过一个Rank的，从上一个Rank的最后一个Cell的地址加1作为基地址。这一点很重要， 这也是为什么对于32位计算机来说，能使用的最多容量的内存为4GB。

1. **寻址**

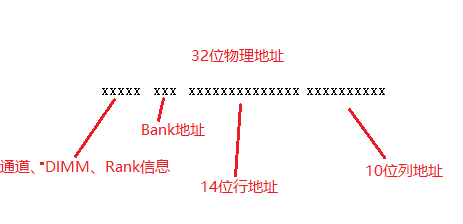
当CPU通过地址总线传过来一个物理地址时，内存条是如何寻找到一个Cell的呢。我们首先要看内存条是如何根CPU连接的。



在看看DDR3 128\*64bit\*16的地址线情况



根据连接情况和组成情况，当物理地址从MMU单元出来时，CPU会把需要访问的地址当做一个数据给内存控制器，如果不是集成在CPU内部，那么通过DMI总线（CPU链接北桥的总线）传送给北桥，当然这个是会加上一些标识以方便北桥识别，这个数据是需要访问的内存地址， 当北桥（内存控制器）收到这个地址后，通过集成在控制器中的"逻辑映射"， 内存控制器就知道这个地址对应了了那个rank（p-bank），那个L-bank，L-bank中的那个row地址，L-bank中的那个Column地址，及确定了那个CS#信号有效（选择rank），那组bank#信号有效（选择L-bank），  然后北桥在遵从SPD设定的内存的时序下按照固定的时序发送CS#信号 bank#信号 row address column address 特殊状态下还会有DQM信号 BL信号等等。也就是说从内存控制器出来，然后是选择DIMM,根据片选到Rank,在到Bank，然后在Chip的地址解码器里输出行地址和列地址，定位到Cell。其实这也就是说物理地址要包含行地址、列地址，bank地址、rank信息等。根据上面的表，物理地址**可能**的构成如下：



1. **读写方式**

读取时首先内存控制器将1组位址现传到内存上，控制器跟着传送控制信号；如果是多rank的情况，CS也会送到对应信号选择的目标rank上。接着由于每个rank是由多个chip组成，每个chip仅负责部分的资料读取，chip接收到位址信号后，将位址放入内部的row/column解码器找出对应的bank位址（每家厂商每款产品内部的bank组合可能不同，因此相应地也会略有不同），接着开启row线，同一排row的内部内容就会流到row buffer内部，row buffer判断信号为0或是1之后就输出存储内容。

写入时除了位址资料以外，还会传送欲写入的内容至芯片内部的input buffer，同样的也是按照row/column解码器找出对应位址之后写入。