老师同学们大家好，那么我们说针对内存系统可以有两个方面的优化，一个是前面说的降低访问延时，另一个则是GPU更加侧重的提高的访问带宽。

为此，人们将**GPU内存系统**划分为多个**可并行的访问通道**，通过一定的映射规则，将地址映射到这些通道上，从而提高他的**并行访存能力**，进而提高GPU的带宽。

首先来看一下伪随机交织方法

它是通过地址的二进制序列上，选取若干位作为基础，成为channel bit，然后再找其他位与channel bit做异或操作，从而得到伪随机的**内存通道地址**。

但是他在实际应用上存在两点问题

首先是对于**不同的内存地址交织方式**可能最终导致访问的地址重叠，又得冲突停顿了

其次是这是一种相对固定的方式，GPU一旦启动，这个**内存地址交织方式**便不能修改

针对上述缺点

人们又提出一种解决方案，基于扩展页表的优化方法

主要思想是在页表里头加入任务类型信息，使得对于每个任务都可以找到最适合它的访存映射方式。

主要步骤如下：

逻辑运算单元ALU接收GPU应用所发送的指令以及任务类型的相关信息；然后根据指令完成相应的操作，并在访问内存时向内存管理单元**发起虚拟地址请求**

内存管理单元接受请求后，通过扩展页表将虚拟地址转换为物理地址，然后将物理地址发送至**地址转换单元**。同时！基于其中的任务类型信息，查找对应任务类型标识，然后发送给**映射方案配置寄存器**，这是GPU的片上寄存器

从而查找对应的最佳映射方式

如何确定最佳映射方式呢，这里是通过计算一定时间段里头每个映射方式下**GPU访问请求在不同通道上的分布信息熵**的时间平均值，从而评估其分布均匀性，选出信息熵最高的也就是分布最均匀的一种映射方式。

然后将该映射方式发送至地址转换单元和之前的物理地址结合形成**内存通道地址**。

这样子就让GPU处理不同任务时，找到最佳映射方式，使得访问通道冲突的可能性大大降低。