



北京航空航天大学

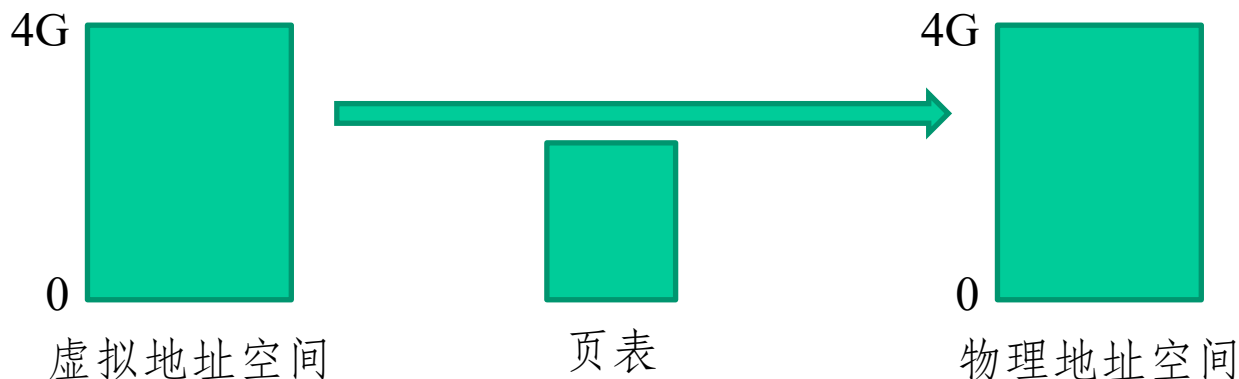
Beijing University of Aeronautics and Astronautics

# 内存自映射

# 页目录自映射

## ■ 基本事实：

- 页表的作用是将虚拟地址空间映射到物理地址空间



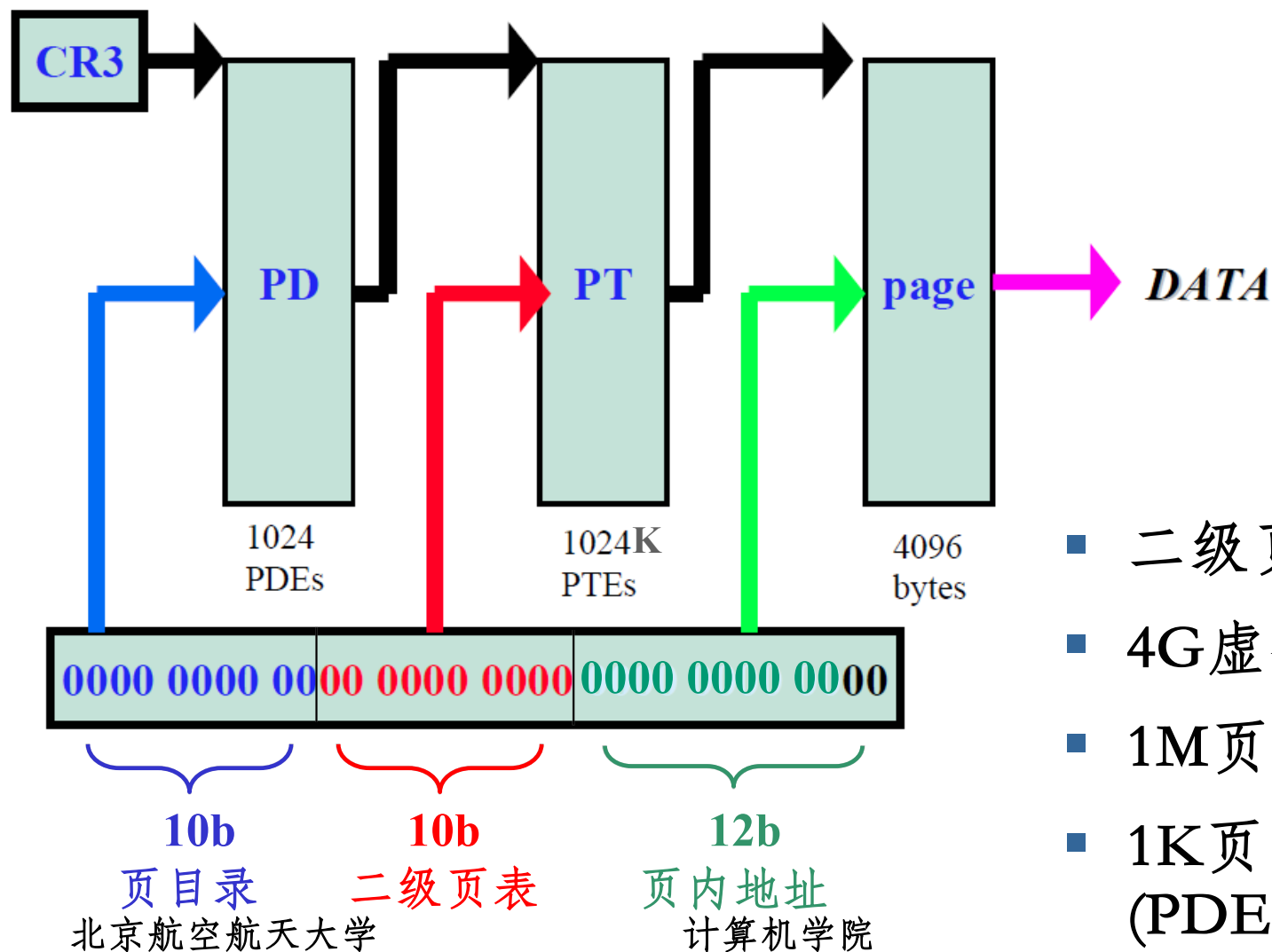
- 对于32位地址长度，可寻址空间为4GB
- 采用12位页内偏移，表明内存页大小为4KB
- 每个页表项负责记录1页（4KB）的地址映射关系
- 整个4GB地址空间被划分为 $4GB/4KB=1M$ 页，所以需要1M个页表项来记录逻辑-物理映射关系

Virtual Page #(20bit)

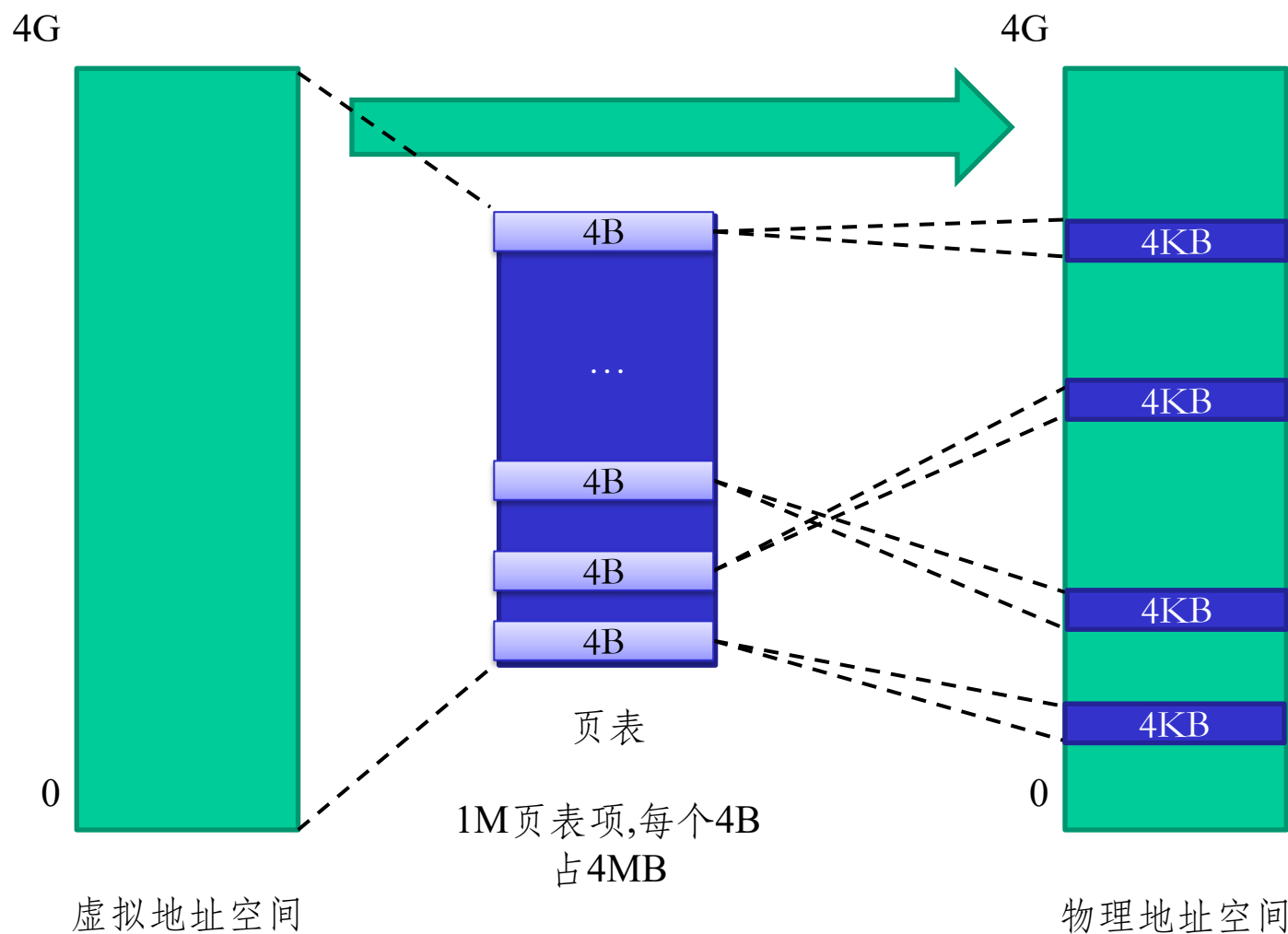
Offset(12bit)

多级页表中，每级页表大小应该正好在一页。一个页表项4Byte， $4KB/4B=1K=1024$ ，所以二级页表10位，剩余一级页表10位，页表合计20位

## Virtual Address Translation



# 页目录自映射

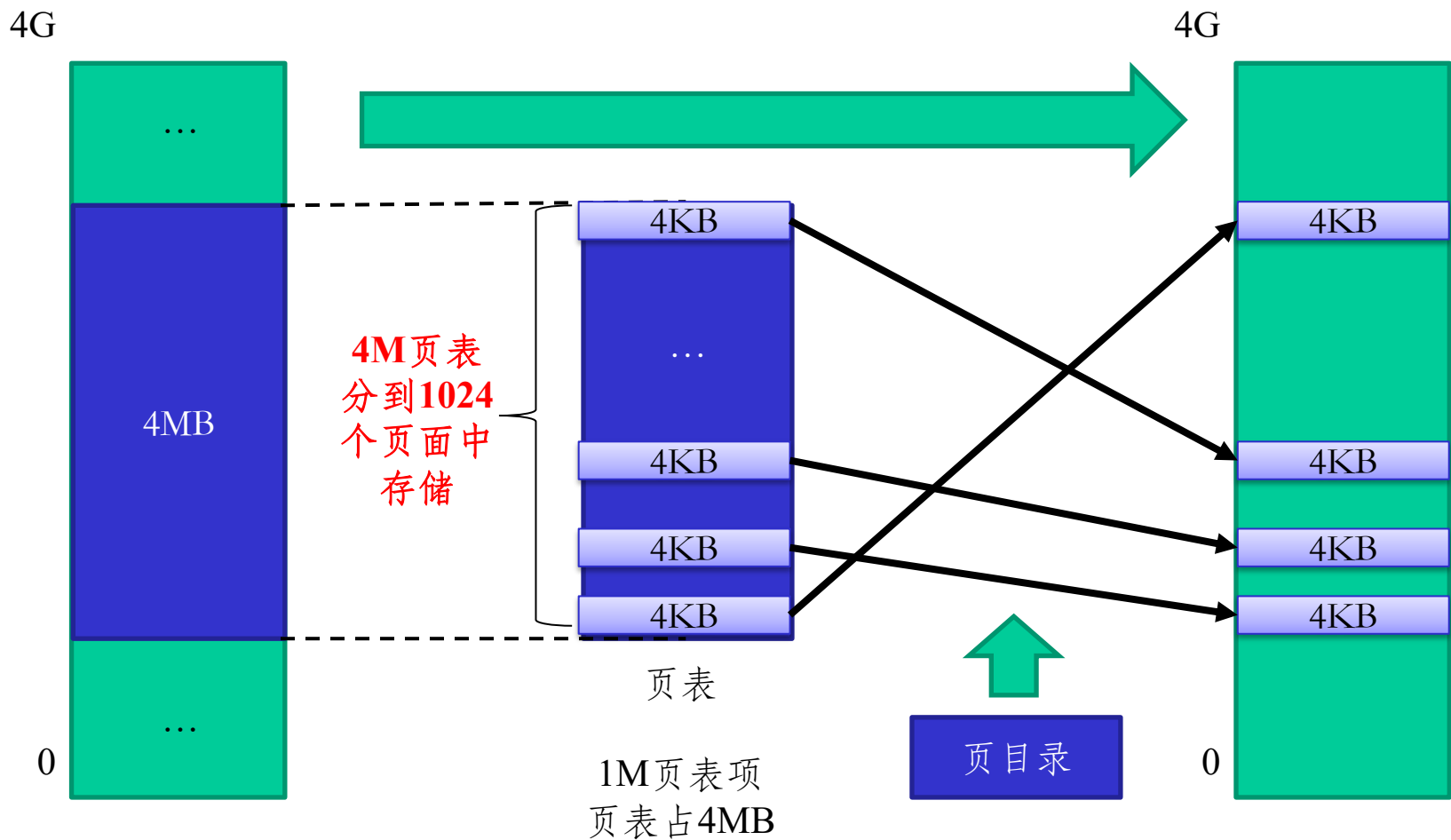


# 页目录自映射

## ■ 页表存储在内存中

- 每个页表项需要4字节，所以1M个页表项需要4MB字节存储，所以整个页表占用的内存大小就是4MB
- 4MB页表也要分页存储，共需要 $4\text{MB}/4\text{KB}=1024$ 个页面存储（我们把存放页表的页称为：页表页）

# 页目录自映射



# 页目录自映射

## ■ 页目录也是页表

- 1024个存放页表的页面逻辑上连续，物理上可以分散，其对应逻辑-物理映射关系记录在页目录中
- 页目录有1024项（页目录项），每一项指向一个页表，页目录占1页（4KB）空间。
- 每一页目录项对应4MB内存，1024个页目录项正好对应4GB内存（整个地址空间）

# 页目录自映射

## ■ 关键点

- 页表也被映射到4G逻辑地址空间内。存储页表的4MB地址空间中是整个4GB虚拟地址空间中的一部分，所以页表也需要完成自己所对应逻辑地址空间到物理地址空间的映射。OS设计者可以规定其所在位置（4MB对齐），例如0x7fc00000
- 一方面根据页目录的定义：记录这4MB（连续）地址空间到物理地址空间映射关系的，是一个4KB的页目录
- 另一方面根据页表的定义：记录这4MB（连续）地址空间到物理地址空间映射关系的，需要1K页表项，即一个大小4KB的页表，它是1K个页表其中的一个。
- 所以，页目录和上述页表内容相同，页目录无需额外分配单独的存储空间

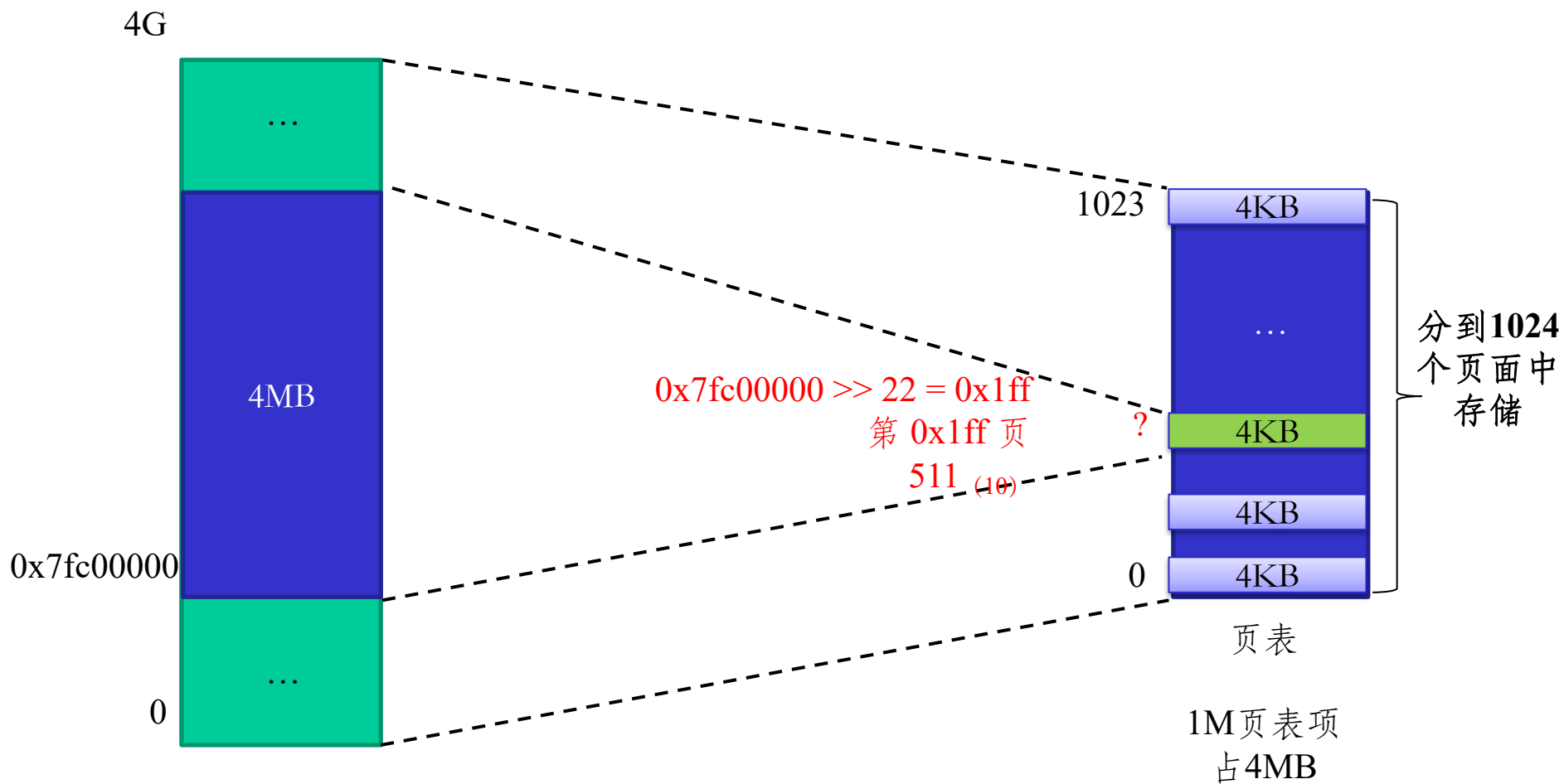


# 页目录自映射

## ■ 页目录在哪？

- 第一种计算方式：页目录对应1K个页表的那一个？
- 给定页表虚拟地址起始位置，例如0x7fc00000
- 可知，从这个地址开始的4MB是存储页表的空间
- 这4MB地址空间是整个4GB地址空间中第  $(0x7fc00000 >> 22)$  个4MB地址空间，因此其逻辑-物理映射关系应该记录在第  $(0x7fc00000 >> 22 = 0x1ff)$  个页表中
- 每个页表大小4KB，所以页目录对于页表虚拟地址起始位置的地址偏移为  $0x1ff << 12 = 0x1ff000$

# 页目录自映射



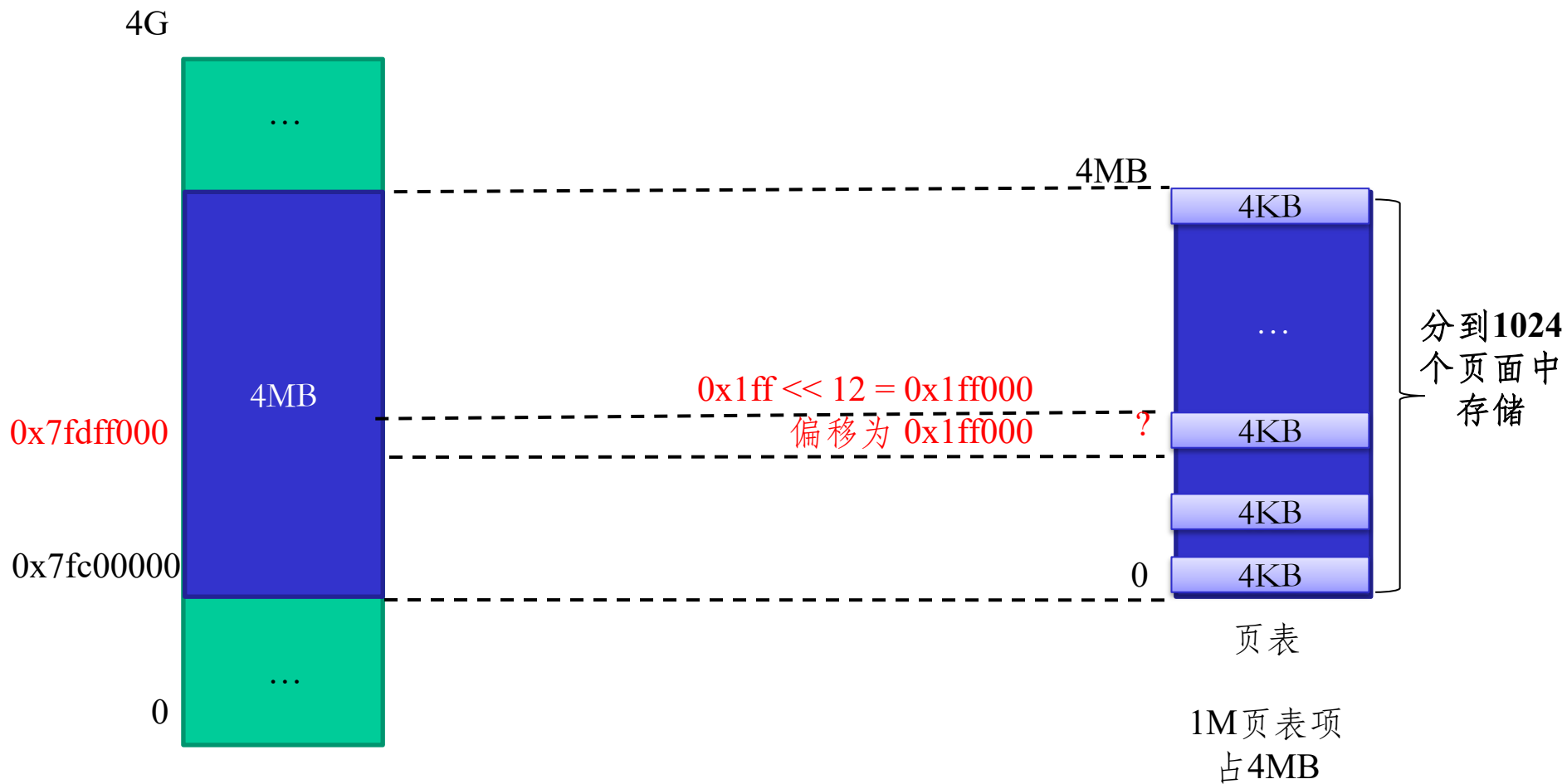
虚拟地址空间

北京航空航天大学

计算机学院

王雷

# 页目录自映射

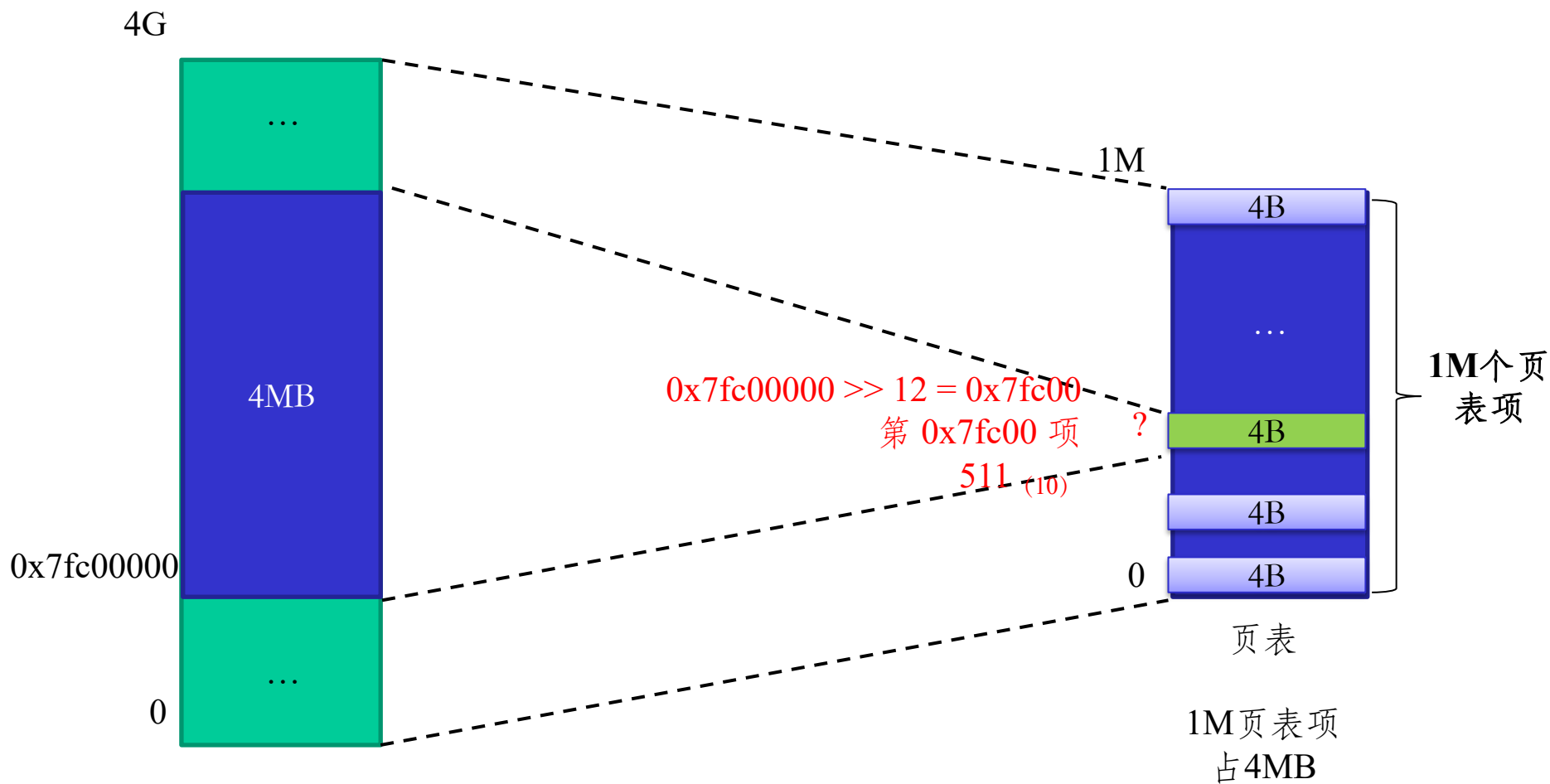


# 页目录自映射

## ■ 页目录在哪？

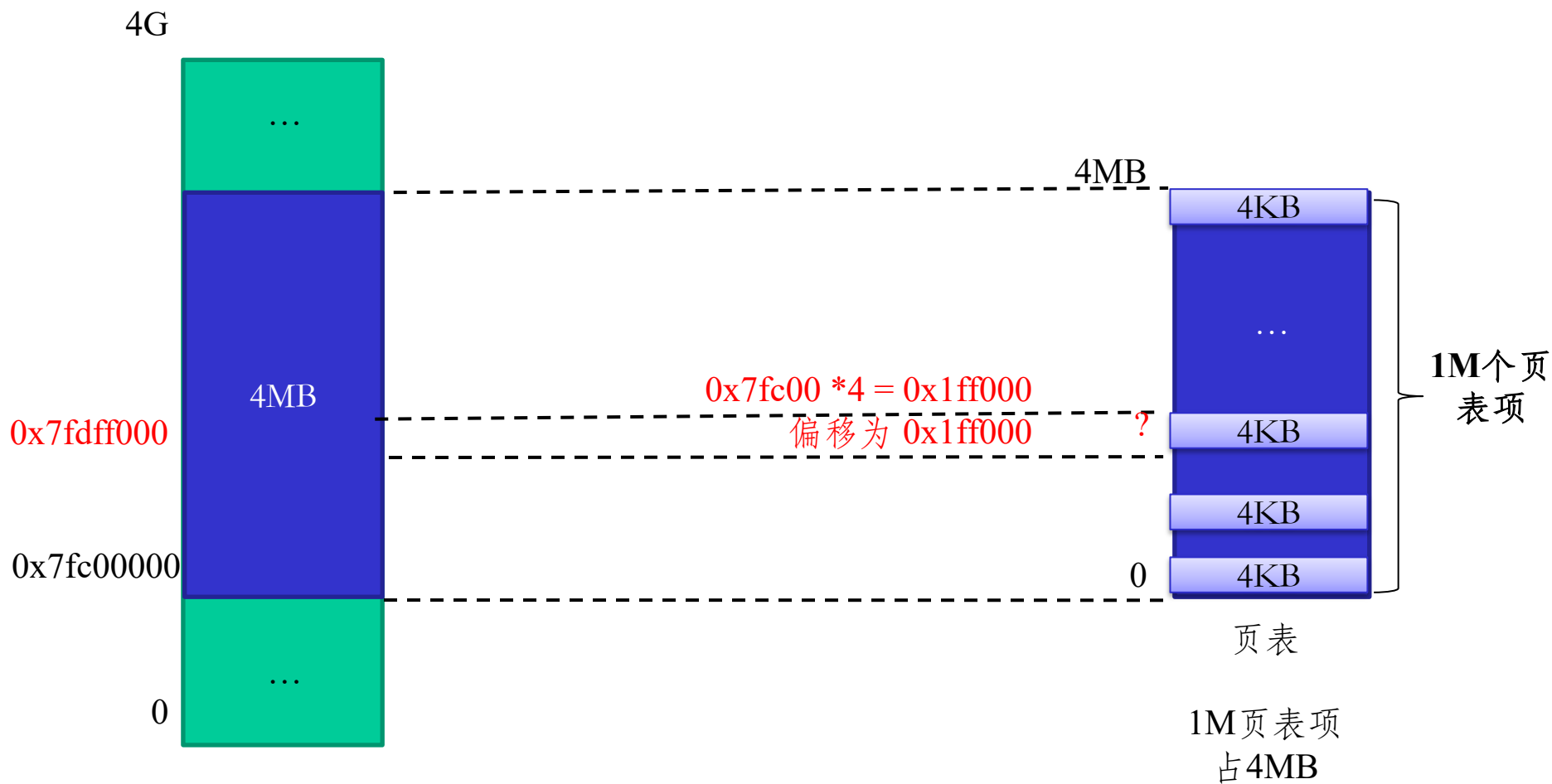
- 第二种计算方式：页目录对应1M个页表项的那一个？
- 给定页表虚拟地址起始位置，例如0x7fc00000
- 将整个4GB地址空间划分为1M个4KB页
- 上述地址对应于第  $(0x7fc00000 >> 12)$  个4KB页，因此其逻辑-物理映射关系应该记录在第  $(0x7fc00000 >> 12)$  个页表项中
- 每个页表项4个字节，所以页目录对于页表虚拟地址起始位置的地址偏移为  $(0x7fc00000 >> 12) * 4 = 0x1ff000$

# 页目录自映射

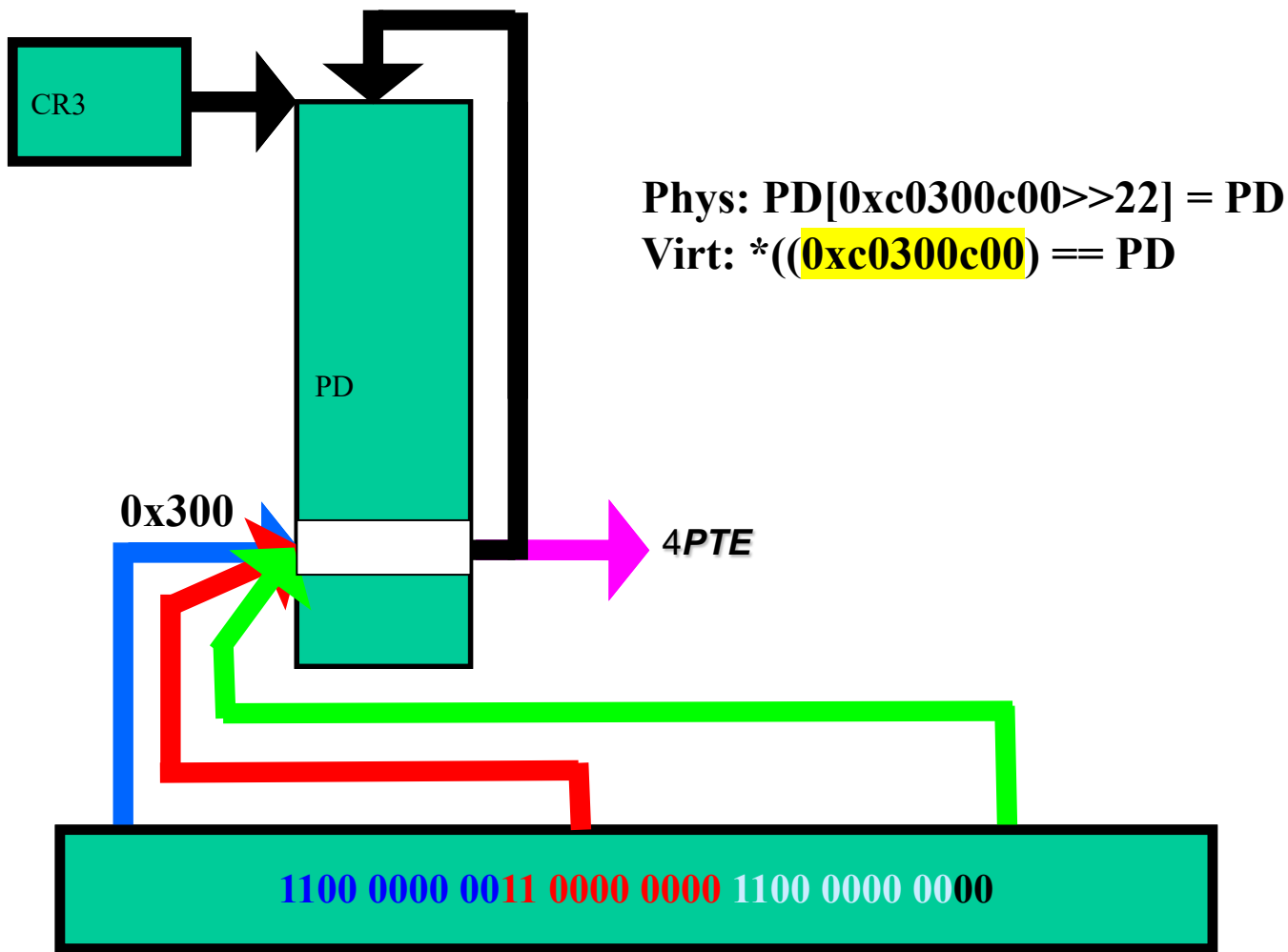


虚拟地址空间

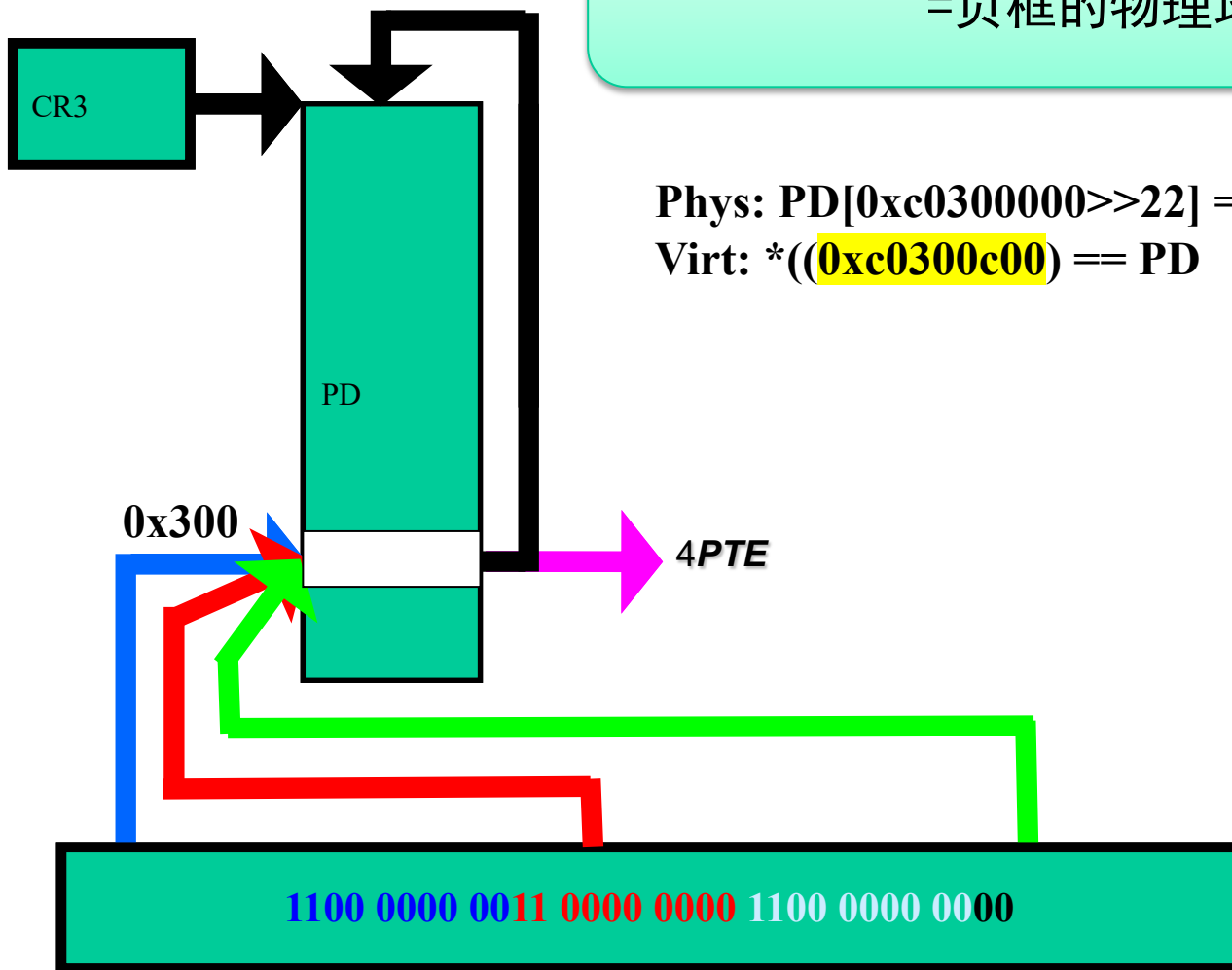
# 页目录自映射



# 自映射机制



# 自映射机制



页目录物理基址=二级页表物理地址  
=页框的物理地址

Phys:  $PD[0xc0300000 \gg 22] = PD$

Virt:  $*((0xc0300c00) == PD$



# 页目录自映射

## ■ 简化计算

- 对于32位地址字长，2级页表，4KB页面大小
- 某个虚拟地址 $va$ 所对应的页目录虚拟地址？
- $((PMMPTE)((((ULONG)(va)) >> 12) << 2) + PTE\_BASE))$
- 某个虚拟地址 $va$ 所对应的页表项地址？
- $((PMMPTE)((((ULONG)(va)) >> 22) << 2) + PDE\_BASE))$

## ■ 练习：

- 页表起始地址 $0x80000000$ ，页目录起始地址=?
  - $va = PTE, 0x80000000 >> 10 + 0x80000000 = 0x80200000$
- 如果给定页目录起始地址，求页表起始地址？
  - E.g. 页目录起始地址 $0xC0300000$ ，页表起始？

## 2025第8次课堂小测试

