

kmalloc() 和 vmalloc() 介绍

kmalloc()

用于申请较小的、连续的物理内存

1. 以字节为单位进行分配，在<linux/slab.h>中
2. void *kmalloc(size_t size, int flags) 分配的内存物理地址上连续，虚拟地址上自然连续
3. gfp_mask 标志：什么时候使用哪种标志？如下：

情形	相应标志
进程上下文，可以睡眠	GFP_KERNEL
进程上下文，不可以睡眠	GFP_ATOMIC
中断处理程序	GFP_ATOMIC
软中断	GFP_ATOMIC
Tasklet	GFP_ATOMIC
用于 DMA 的内存，可以睡眠	GFP_DMA GFP_KERNEL
用于 DMA 的内存，不可以睡眠	GFP_DMA GFP_ATOMIC

4. void kfree(const void *ptr)

释放由 kmalloc() 分配出来的内存块

vmalloc()

用于申请较大的内存空间，虚拟内存是连续的

1. 以字节为单位进行分配，在<linux/vmalloc.h>中
2. void *vmalloc(unsigned long size) 分配的内存虚拟地址上连续，物理地址不连续
3. 一般情况下，只有硬件设备才需要物理地址连续的内存，因为硬件设备往往存在于 MMU 之外，根本不了解虚拟地址；但为了性能上的考虑，内核中一般使用 kmalloc()，而只有在需要获得大块内存时才使用 vmalloc()，例如当模块被动态加载到内核当中时，就把模块装载到由 vmalloc() 分配的内存上。
4. void vfree(void *addr)，这个函数可以睡眠，因此不能从中断上下文调用。

malloc(), vmalloc() 和 kmalloc() 区别

[*]kmalloc 和 vmalloc 是分配的是内核的内存，malloc 分配的是用户的内存

[*]kmalloc 保证分配的内存存在物理上是连续的，vmalloc 保证的是在虚拟地址空间上的连续，malloc 不保证任何东西（这点是自己猜测的，不一定正确）

[*]kmalloc 能分配的大小有限，vmalloc 和 malloc 能分配的大小相对较大

[*]内存只有在要被 DMA 访问的时候才需要物理上连续

[*]vma11oc 比 kma11oc 要慢