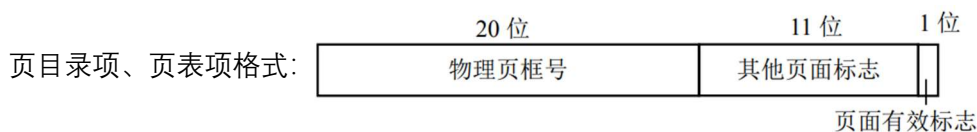
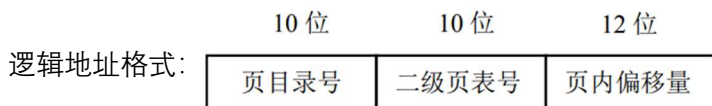
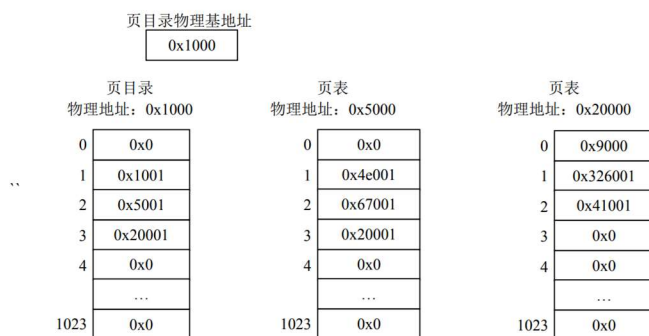


1. 一个 32 位的页式虚拟存储系统采用两级页表管理，其逻辑地址中，第 22 到 31 位是第一级页表（页目录）索引，第 12 位到 21 位是第二级页表索引，页内偏移占第 0 到 11 位。每个页表（目录）项包含 20 位物理页框号和 12 位标志位，其中最后 1 位为页面有效标志。



- 1) 请问进程整个的地址空间有多少字节？一页有多少字节？
共有 $2^{32}=4G$ 字节
一页有 $2^{12}=4K$ 字节
- 2) 一个进程如果从 0x80000000 开始映射 4MB 大小页表空间，求第一级页表（页目录）的起始逻辑地址，并指出从哪个逻辑地址可以读出第一级页表（页目录）所在的物理页框号。说明理由。（注意 B 代表字节，一个 32 位地址占 4 字节）
 $PTbase = 0x8000_0000$
 $PDbase = PTbase | ((PTbase \gg 12) \ll 2) = 0x8020_0000$
 $PDself_mapping = PTbase | ((PTbase \gg 12) \ll 2) | (PTbase \gg 20) = 0x80200800$
- 3) 如果当前进程的页目录物理地址、页目录和相应页表内容如图下所示，请描述访问以下逻辑地址时系统进行地址转换的过程，如可行给出最终访存获取到的数据。逻辑地址：0x0、0x00803004、0x00402001



1. 逻辑地址 0x0 => 页目录号：0 二级页表号：0 页内偏移 0
访问页目录 0x1000 第 0 项，得到 0x0，有效标志位 0，触发缺页中断
2. 逻辑地址 0x00803004 => 页目录号：2 二级页表号：3 页内偏移：4
访问页目录 0x1000 第 2 项，得到 0x5001，有效标志位 1
访问 0x5000 处页表第 3 项，得到 0x2_0001，有效标志位 1
物理页面的地址为 0x20000，页内偏移为 4，得到数据 0x0
3. 逻辑地址 0x00402001 => 页目录号：1 二级页表号：2 页内偏移 1
访问页目录 0x1000 第 1 项，得到 0x1001，物理页框号 1 有效标志 1
二级页表基地址为 0x1000，访问第 2 项，得到 0x5001，物理页框号 5，有效位 1
访问物理地址 0x5000 的第 1 项，得到数据 0x4e001
- 4) 要想访问物理地址 0x326028，需要使用哪个逻辑地址？
 $0x326028 \gg 12 = 0x326$ ，因此其物理页框号为 0x326000，页内偏移为 0x028
 在 0x20000 的页表存有此页框，因此其二级页表基地址为 0x20000，二级页表号为 1
 存有 0x20000 的页目录项为第三项，因此页目录号为 3
 对应虚拟地址为 0xc0_1028

2. 一个 32 位的虚拟存储系统采用两级页表管理，其逻辑地址形式如下：

第一级页表（10 位）	第二级页表（10 位）	页内偏移（12 位）
-------------	-------------	------------

物理地址为 32 位，形式为：

物理页框号（20 位）	页内偏移（12 位）
-------------	------------

页表项（PTE）格式为：

高 20 位	低 12 位（为标志位）
物理页框号	第 0 位为有效位，如果为 0，表示 Invalid；如果为 1 表示 Valid。 第 1 位为读写位，如果为 0，表示 Read Only；如果为 1 表示 Read/Write。

假设当前进程第一级页表的物理地址为 0x00200000，利用后面物理内存的信息，请在下表中写出以下指令的执行结果。

对于 Load 指令，如果成功执行，写出读入的数据（读取一个字节），否则写 Error；

对于 Store 指令，如果成功执行，写 OK，否则写 Error。并给出寻址过程。

指令	结果
Load [0x00001022]	一级页表号：0x0 => 访问 0x200000 => 得到数据 0x100007（物理页框号：0x100，有效位为 1） 二级页表号：0x1 => 访问 0x100004 => 得到数据 0x00004067（物理页框号：0x4，有效位 1） 页内偏移：0x022 => 访问 0x4022 => 得到数据 0x20
Store [0x00C07222]	一级页表号：0x3 => 访问 0x20000C => 得到数据 0x103007（物理页框号：0x103，有效位 1） 二级页表号：0x7 => 访问 0x10301C => 得到数据 0xEEFF0001（物理页框号：0xEEFF0，有效位 1，可写位 0，只读）=> Error
Store[0x00C005BF]	一级页表号：0x3 => 访问 0x20000C => 得到数据 0x103007（物理页框号：0x103，有效位 1） 二级页表号：0x0 => 访问 0x103000 => 得到数据 0x11220067（物理页框号：0x11220，有效位 1 可写位 1，） 页内偏移：0x5BF => 写入 0x112205BF，OK
Load [0x00003013]	一级页表号：0x0 => 访问 0x200000 => 得到数据 0x100007（物理页框号：0x100，有效位为 1） 二级页表号：0x30 => 访问 0x1000C0 => 得到数据 0x4007（物理页框号：0x4，有效位：1） 页内偏移 0x013: => 访问 0x4013 => 得到数据 0x19
Load [0xFF80078F]	一级页表号：0x3FE => 访问 0x200FF8 => 得到数据 0x1FE007（物理页框号：0x1FE，有效位 1） 二级页表号：0x0 => 访问 0x1FE000 => 得到数据 0x04150000（有效位 0）=> Error
Load [0xFFFFF005]	一级页表号：0x3FF => 访问 0x200FFC => 得到数据 0x1FF007（物理页框号 0x1FF，有效位 1） 二级页表号：0x3FF => 访问 0x1FFFFC => 得到数据 0x103067（物理页框号 0x103，有效位 1） 页内偏移：0x005 => 访问 0x103005 => 得到数据 0x66