**同济大学计算机系**

**操作系统实验报告**

****

**学 号 2152809**

**姓 名 曾崇然**

**专 业 计算机科学与技术**

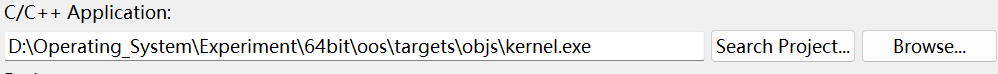
**授课老师 方钰老师**

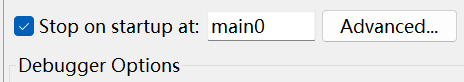
**一. 完成实验准备**

**（1）**我在本次实验中采用的是之前实验二中的可运行程序showStack.exe



**（2）**设置调试对象和调试的起点

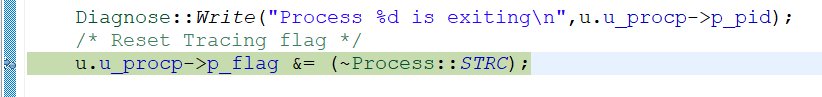




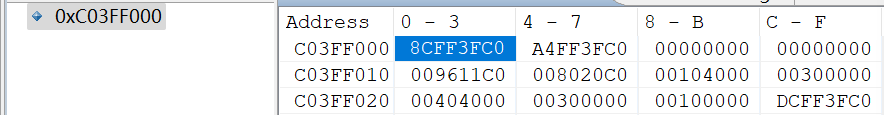
**二. 找到完整的进程图像（题目(1)）**

**（1）调试运行**

设置断点并运行至断点处

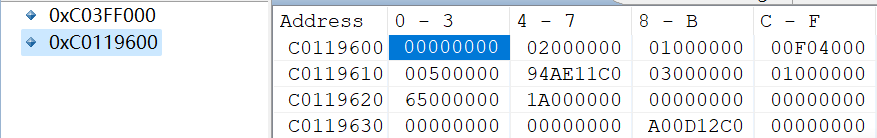


**（2）获取进程的user结构**



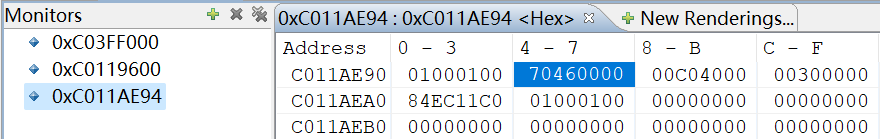
|  |  |  |
| --- | --- | --- |
| 变量名称 | 含义 | 值 |
| Process\* u\_procp | Proc结构的逻辑地址 | 0xC0119600 |
| MemoryDescriptor u\_MemoryDescriptor（定义如下，此处均为逻辑地址） | | |
| PageTable\* m\_UserPageTableArray | 相对映射表首地址 | 0xC0208000 |
| unsigned long m\_TextStartAddress | 代码段起始地址 | 0x00401000=4M+4K |
| unsigned long m\_TextSize | 代码段长度 | 0x00003000=12K |
| unsigned long m\_DataStartAddress | 数据段起始地址 | 0x00404000=4M+16K |
| unsigned long m\_DataSize | 数据段长度 | 0x00003000=12K |
| unsigned long m\_StackSize | 栈段长度 | 0x00001000=4K |

**（3）获取进程的Proc结构**



|  |  |  |
| --- | --- | --- |
| 变量名称 | 含义 | 值 |
| short p\_uid | 用户 ID | 0 |
| int p\_pid | 进程标识数 | 2 |
| int p\_ppid | 父进程标识数 | 1 |
| unsigned long p\_addr | user 结构即 ppda 区的物理地址 | 0x0040F000 |
| unsigned int p\_size | 除共享正文段的长度，以字节单位 | 0x00005000=20K |
| Text\* p\_textp | 指向代码段 Text 结构的逻辑地址 | 0xC011AE94 |
| ProcessState p\_stat | 进程调度状态 | 3=SRUN |
| int p\_flag | 进程标志位 | 1=SLOAD |
| int p\_pri | 进程优先数 | 65 |
| int p\_cpu | cpu值，用于计算 p\_pri | 19 |
| int p\_nice | 进程优先数微调参数 | 0 |
| int p\_time | 进程在盘上(内存内)驻留时间 | 0 |
| unsigned long p\_wchan | 进程睡眠原因 | 0 |

**（4）获取进程代码段的Text结构**



|  |  |  |
| --- | --- | --- |
| 变量名称 | 含义 | 值 |
| int x\_daddr | 代码段在盘交换区上的地址 | 0x00004670 |
| unsigned long x\_caddr | 代码段起始地址（物理地址） | 0x0040C000 |
| unsigned int x\_size | 代码段长度，以字节为单位 | 0x00003000 = 12K |
| Inode\* x\_iptr | 内存inode地址 | 0xC011ECD0 |
| Unsigned short x\_count | 共享正文段的进程数 | 1 |
| unsigned long m\_DataSize | 共享该正文段且图像在内存的进程数 | 1 |
| Unsigned short x\_ccount | 栈段长度 | 0x00001000=4K |

**（5）进程图像完整信息**

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 逻辑地址 | 物理地址 | 大小 |
| 代码段 | 0x00401000 | 0x0040C000 | 12K |
| 可交换部分 | 0xC03FF000 | 0x0040F000 | 20K |
| PPDA区 | 0xC03FF000 | 0x0040F000 | 4K |
| 数据段 | 0x00404000 | 0x00410000 | 12K |
| 堆栈段 |  | 0x00413000 | 1K |

**（6）获取代码段和可交换部分起始地址的方法**

逻辑地址：数据段和代码段：从proc块中该进程的u\_MemoryDescriptor获取

User结构：固定的为0xC03FF000

物理地址：代码段：proc结构->text结构,，根据x\_ccount的值来判断物理地址是在

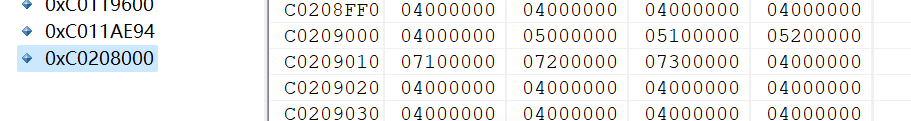
内存的x\_caddr还是盘交换区的x\_daddr

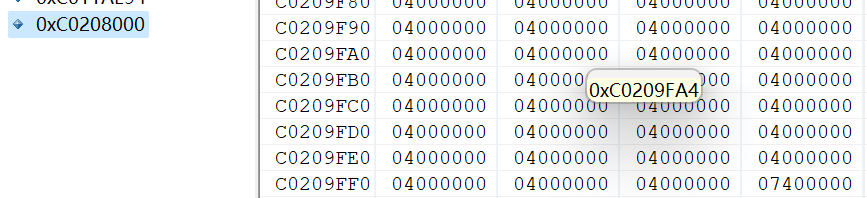
PPDA区：proc结构的p\_addr即可获得

**三. 找到进程完整的页表**

**（1）获取相对虚实地址映射表（题目(2)）**

（只截取了部分）



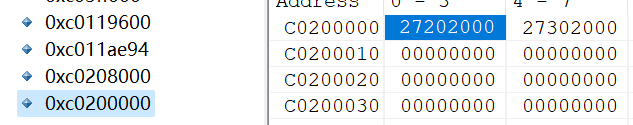


|  |  |  |  |
| --- | --- | --- | --- |
| 页号 | 地址 | 值 | |
| 高20位页框号 | 低12位标志位(u/s r/w p) |
| 0# | \ | \ | \ |
| …… | \ | \ | \ |
| 1024# | \ | \ | \ |
| 1025# | 0xC0208000~0xC0208003 | 0 | 005 |
| 1026# | 0xC0209008~0xC020900B | 1 | 005 |
| 1027# | 0xC020900C~0xC020900F | 2 | 005 |
| 1028# | 0xC0209010~0xC0209013 | 1 | 007 |
| 1029# | 0xC0209014~0xC0209017 | 2 | 007 |
| 1030# | 0xC0209018~0xC020901B | 3 | 007 |
| 1031# | 0xC020901C~0xC020901F | 0 | 004 |
| …… | …… | …… | …… |
| 2047# | 0xC0209FFC~0xC0209FFF | 4 | 007 |

**（2）进程的物理页表（题目(3)）**

物理页表包括0x200号，0x201号，0x202号，0x203号页表，其逻辑地址为0xc0200000，0xc0201000，0xc0202000，0xc0203000

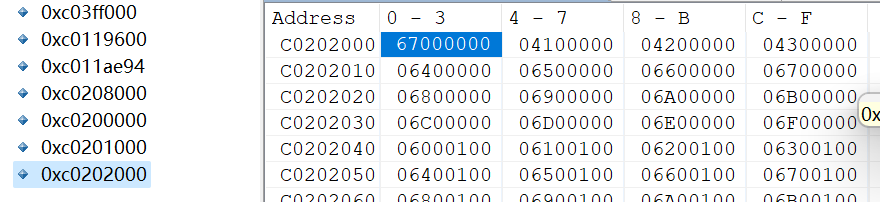
①页目录





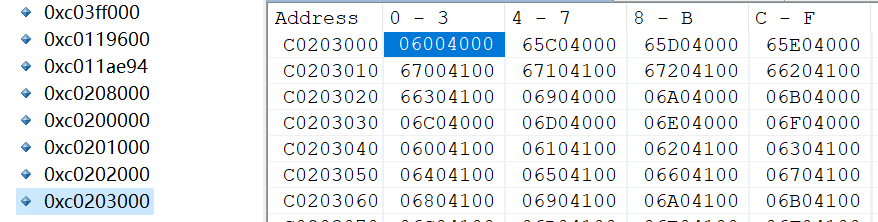
|  |  |  |  |
| --- | --- | --- | --- |
| 页号 | 地址 | 值 | |
| 高20位页框号 | 低12位标志位(u/s r/w p) |
| 0# | 0xC0200000~0xC0200003 | 0x202 | 027（课件上这里是不是写错了） |
| 1# | 0xC0200004~0xC0200007 | 0x203 | 027（课件上这里是不是写错了） |
| …… | …… | …… | …… |
| 768# | 0xC0200C00~0xC0200C03 | 0x201 | 023 |
| …… | …… | …… | …… |

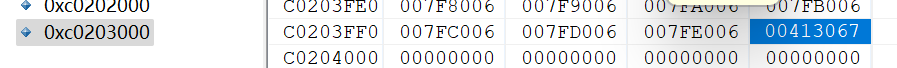
②用户页表1



|  |  |  |  |
| --- | --- | --- | --- |
| 页号 | 地址 | 值 | |
| 高20位页框号 | 低12位标志位(u/s r/w p) |
| 0# | 0xC0202000~0xC0202003 | \ | \ |
| 1# | 0xC0202004~0xC0202007 | \ | \ |
| …… | …… | …… | …… |
| 1023# | 0xC0202FFC~0xC0202FFF | \ | \ |

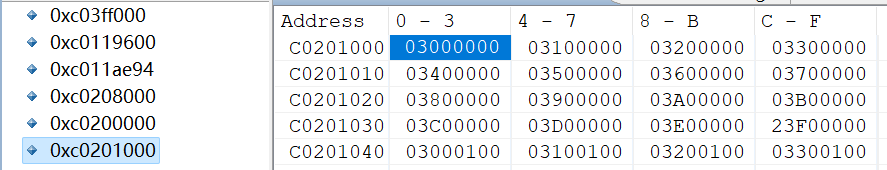
③用户页表2

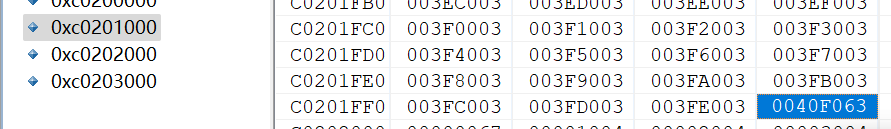




|  |  |  |  |
| --- | --- | --- | --- |
| 页号 | 地址 | 值 | |
| 高20位页框号 | 低12位标志位(u/s r/w p) |
| 0# | 0xC0203000~0xC0203003 | \ | \ |
| 1# | 0xC0203004~0xC0203007 | 0x40C | 065 |
| 2# | 0xC0203008~0xC020300B | 0x40D | 065 |
| 3# | 0xC020300C~0xC020300F | 0x40E | 065 |
| 4# | 0xC0203010~0xC0203013 | 0x410 | 067 |
| 5# | 0xC0203014~0xC0203017 | 0x411 | 067 |
| 6# | 0xC0203018~0xC020301B | 0x412 | 067 |
| …… | …… | …… | …… |
| 1023# | 0xC0203FFC~0xC0203FFF | 0x413 | 067 |

④核心页表





|  |  |  |  |
| --- | --- | --- | --- |
| 页号 | 地址 | 值 | |
| 高20位页框号 | 低12位标志位(u/s r/w p) |
| 0# | 0xC0201000~0xC0201003 | 0x001 | 003 |
| 1# | 0xC0201004~0xC0201007 | 0x002 | 033 |
| …… | …… | …… | …… |
| 1023# | 0xC0201FFC~0xC0201FFF | 0x40F | 063 |

**四. 完整的进程图像**

