CS334 lab3 report

CS334 lab3 report

SID: 12012919 Name: 廖铭骞

请详细描述本节课最小化内核的启动过程

ELF和BIN文件的区别是什么

链接脚本的作用是什么

在init.c (截图)使用cputs函数,使得在最小化内核启动后通过cputs打印出"SUSTechOS"(截图)

在stdio.c中参考cputs()函数实现double_puts()函数(截图),将输出的字符串每个字符打印两次,如double_puts("SUSTech")应输出"SSUUSSTTeecchh"。在init.c中调用该函数(截图),并使得最小化内核启动后输出"IILLOOVVEEOOSS"(截图)。

SID: 12012919

Name: 廖铭骞

请详细描述本节课最小化内核的启动过程

将 entry.S 和 init.c, stdio.c 等几个 c 文件编译成为 .o 目标文件,然后链接器将 .o 文件链接成可执行文件 kernel (elf 文件),最后使用 objcopy 把 elf 文件转化成为 可执行二进制文件 ucore.bin,再使用 OpenSBI 作为启动器,启动内核

ELF和BIN文件的区别是什么

ELF 文件比较复杂,包括文件头和一些冗余的调试信息,指定了程序每个 section 的内存布局,需要一个完整的操作系统才能直接执行。并且 ELF 文件当中的数据段位置并不是实际的物理位置。而 BIN 文件则将 ELF文件当中的对应内存信息映射到一块真实的线性的数据当中,可以被 CPU 直接执行

链接脚本的作用是什么

将输入文件链接成输出文件,描述了怎样把输入文件的 section 映射到输出文件的 section,同时规定这些 section 的内存布局

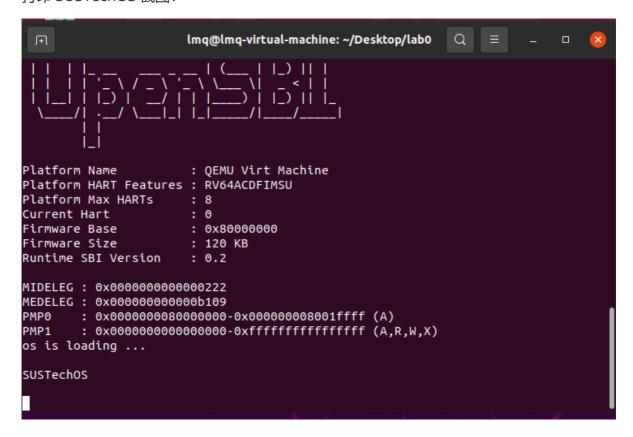
在init.c (截图) 使用cputs函数,使得在最小化内核启动后通过cputs打印出"SUSTechOS"(截图)

init.c 截图:

```
int kern_init(void) {{
    extern char edata[], end[];
    memset(edata, 0, end - edata);

const char *message = "os is loading ...\n";
    cputs(message);
    const char *sustech = "SUSTechOS\n";
    cputs(sustech);
```

打印 SUSTechOS 截图:



在stdio.c中参考cputs()函数实现double_puts()函数(截图),将输出的字符串每个字符打印两次,如double_puts("SUSTech")应输出"SSUUSSTTeecchh"。在init.c中调用该函数(截图),并使得最小化内核启动后输出"IILLOOVVEEOOSS"(截图)。

实现 double_puts() 截图:

```
int double_puts(const char *str){
   int cnt = 0;
   char c;
   while((c = *str++) != '\0'){
      cputch(c, &cnt);
      cputch(c, &cnt);
   }
   cputch('\n', &cnt);
   return cnt;
}
```

调用 double_puts() 截图:

```
int kern init(void) {
18
         extern char edata[], end[];
19
         memset(edata, 0, end - eda+-).
                                     (char [19]) "os is loading ...\n"
20
21
         const char *message = "os is loading ...\n";
22
         cputs(message);
23
         const char *sustech = "ILOVEOS\n";
24
         double puts(sustech);
25
```

输出 "IILLOOVVEEOOSS" 截图:

```
lmq@lmq-virtual-machine: ~/Desktop/lab0
                                                   Q =
Platform Name
                    : OEMU Virt Machine
Platform HART Features : RV64ACDFIMSU
Platform Max HARTs
                    : 8
Current Hart
                    : 0
Firmware Base
                    : 0x80000000
Firmware Size
                    : 120 KB
Runtime SBI Version
                   : 0.2
MIDELEG: 0x0000000000000222
MEDELEG : 0x000000000000b109
      : 0x0000000080000000-0x000000008001ffff (A)
       os is loading ...
IILLOOVVEEOOSS
Ш
```