件开发 > C++ > 正文

关于 quot;setconsole.c" Alesssandro Rubini 的邮件回复

个评论   来源：The . L   收藏   我要投稿

关于 "setconsole.c" Alesssandro Rubini 的邮件回复

抱着试一试的心态给偶像写了一封求助e-mail，回复了！！！我的小心脏啊~

呵呵~Rubini是谁不解释了...

我的HELP原文主要的关于setconsole.c的问题，这个是LDD3的一个小程序

```
setconsole.c

/*
 * setconsole.c -- choose a console to receive kernel messages
 *
 * Copyright (C) 1998,2000,2001 Alessandro Rubini
 *
 *    This program is free software; you can redistribute it and/or modify
 *    it under the terms of the GNU General Public License as published by
 *    the Free Software Foundation; either version 2 of the License, or
 *    (at your option) any later version.
 *
 *    This program is distributed in the hope that it will be useful,
 *    but WITHOUT ANY WARRANTY; without even the implied warranty of
 *    MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
 *    GNU General Public License for more details.
 *
 *    You should have received a copy of the GNU General Public License
 *    along with this program; if not, write to the Free Software
 *    Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307, USA.
```

```
 */


#include

#include

#include

#include

#include

#include


int main(int argc, char **argv)
{
    char bytes[2] = {11,0}; /* 11 is the TIOCLINUX cmd number */


    if (argc==2) bytes[1] = atoi(argv[1]); /* the chosen console */


    else {
        fprintf(stderr, "%s: need a single arg\n",argv[0]); exit(1);
    }
    if (ioctl(STDIN_FILENO, TIOCLINUX, bytes)<0) {    /* use stdin */
        fprintf(stderr,"%s: ioctl(stdin, TIOCLINUX): %s\n",
                argv[0], strerror(errno));
        exit(1);
    }
    exit(0);
}
```

但是在虚拟控制台之间测试一直不成功，很纠结~ 我用*某*搜索引擎检索了很多blog，都是水...都是
直接copy原书的内容，仅发现一个有对程序进行测试的，但是测试没有成功，google不能用，不知道情
况如果，用TTT检索出来的一样，没有实质性的对setconsole.c的讨论和测试！

    下面是Rubini 对我的回复：

发件人 Alessandro Rubini ⭐

主题 **Re: HELP for setconsole.c(LDD3)**

收件人 我 <jasonleaster@gmail.com> ☆

Hello Jason.

[...]
So I am writing this e-mail for your **help**.

You are welcome.

> I am gona to use this code to redirect terminal
I/O in current console to another console.
Something like that if I printf("hello world!\n"); in current console
and it would print out in another console.

BUT IT DOESN'T WORK...

I think you misunderstood the role of the program (or, my text is not
clear enough in explaining it).

LDD is about kernel drivers and kernel code, so your printf("h, w") is
not affected by code in the book. What it affects is printk("...\n"),
i.e. messages generated by kernel code -- either from the stock
kernel+drivers or by the modules you built and loaded.

In this context the console is the text screen. If you are working
on a PC or another modern machine, most likely you run the graphic
environment. The "terminal" or "tty" in your graphic environment is not
a "console".

To reach the text console, please try typing ctrl-alt-F1. Usually
distributions leave 6 text consoles, and the graphic environment is
on console 7. So, with ctrl-alt-F3 you reach console 3. To go
back to the graphic environment use alt-F7 or ctrl-alt-F7.

Please note, also, that your printk (e.g. if you load hello.ko from
the LDD examples) is only printed to the console if it's message
priority is bigger than the current threshold. Thus the "setlevel"
program, but you can also "echo 8 > /proc/sys/kernel/printk" to enable
debug messages.

I hope this helps you in making sense of the section about the console
and kernel messages. If the core misunderstanding is clarified,
everything should be easier.

/alessandro

```
<喝�http://www.2cto.com/kf/ware/vc/" target="_blank"

class="keylink">vcD4KPHA+PC9wPgo8cD48YnI+CjwvcD4KPHA+ytfPyNa4s/bBy87StcTO88f4PC9wPgo8cD4xLiF
------------------------------------------------------------------ // setconsole.cpp // //

This utility allows a user possessing root privilege // to redirect 'printk' output to a

designated console. // // compile using: // root# gcc -o setconsole setconsole.cpp //

root# chmod a+s setconsole // // execute using: // user$ setconsole 4 // // Code used

here is from an example by Alesandro Rubini, // "Linux Device Drivers (2nd Edition),"

pages 99-100. // // programmer: ALLAN CRUSE // written on: 24 NOV 2002 // revised on: 24

JUN 2007 -- to use "/dev/console" device //------------------------------------------

------------------ #include // for open() <--- added #include // for fprintf() #include //

for errno #include // for exit() #include // for STDIN_FILENO #include // for strerror()

#include // for ioctl() #include // for TIOCLINUX int main( int argc, char **argv ) {

char bytes[ 2 ] = { 11, 0 }; // 11 is the TIOCLINUX command-number if ( argc == 2 )

bytes[1] = atoi( argv[1] ); // console id-number else { fprintf( stderr, "%s: need a

single argument\n", argv[0] ); exit(1); } int fd = open( "/dev/console", O_RDWR ); // <--
```
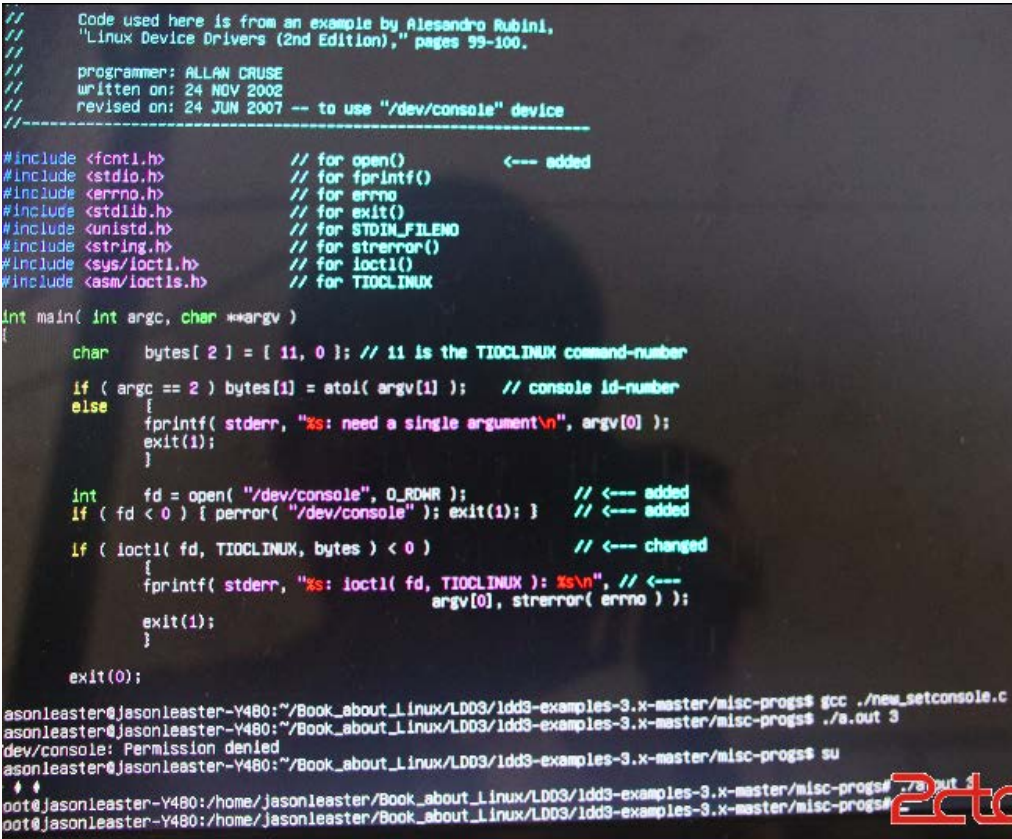
```
- added if ( fd < 0 ) { perror( "/dev/console" ); exit(1); } // <--- added if ( ioctl( fd,
TIOCLINUX, bytes ) < 0 ) // <--- changed { fprintf( stderr, "%s: ioctl( fd, TIOCLINUX ):
%s\n", // <--- argv[0], strerror( errno ) ); exit(1); } exit(0); }
```
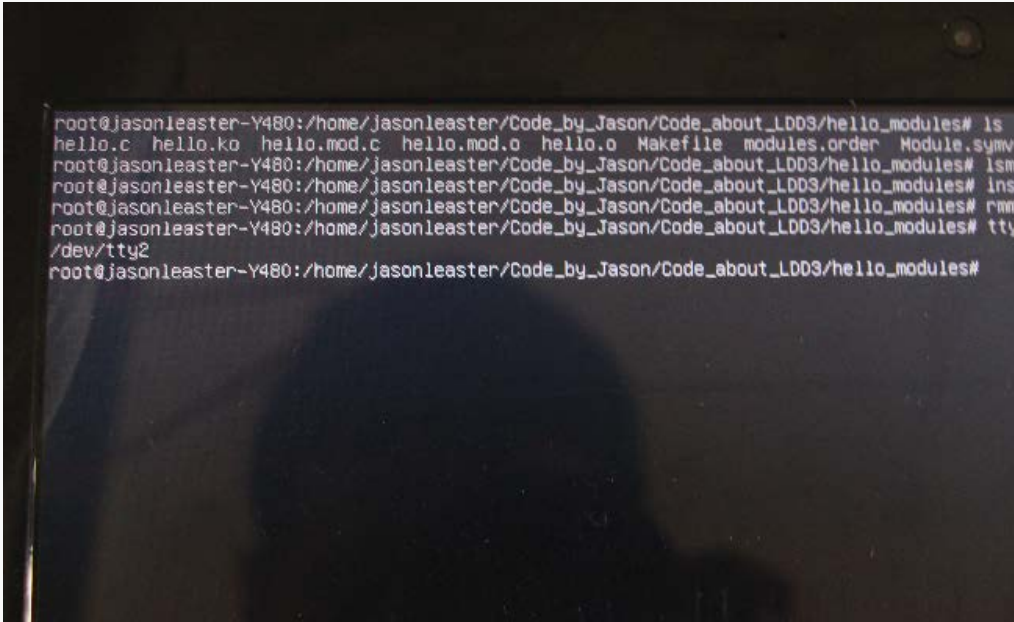
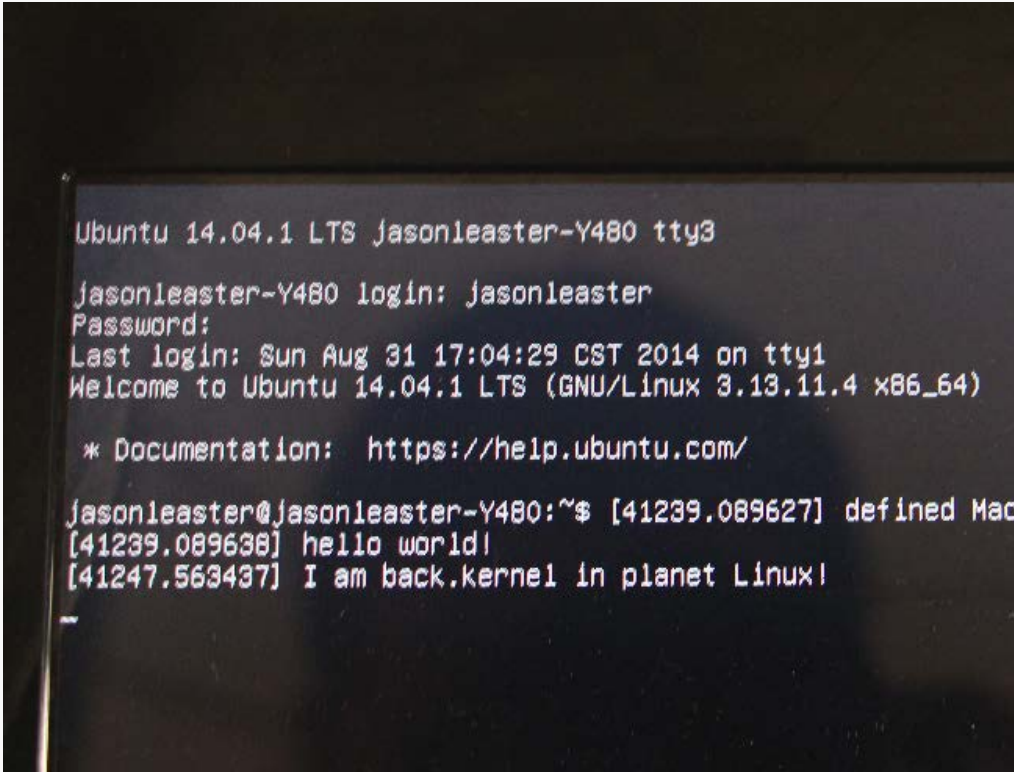是可以在console之间重定向IO的！！

这幅图看到的是tty4，我把控制台的IO重定向到了console 3,也就是/dev/tty3

切换到tty2，我们插入hello.ko 模块，调用printk，打印hello world

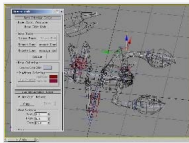我故意执行了tty这个shell程序，提示读者当前我们在tty2,进行insmod操作

明明是有hello world的为什么不打印呢？去哪儿了？console 3！

我们ctrl + alt + F3 切换到console 3看看

终于搞定了！利用ioctl实现了对于不同console之间的IO重定向！

最后谢谢偶像~ Alesssandro Rubini & Allan Cruse

c++游戏编程　　app开发报价单　　ios工程师月薪　　达内的　　编程学习入门

点击复制链接 与好友分享!　　回本站首页

上一篇： Leetcode dfs Path Sum

下一篇： EasyUI实现异步加载tree(整合Struts2)
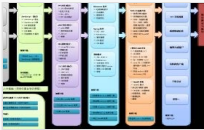
相关文章

- 用C++ Builder实现电子邮件群发
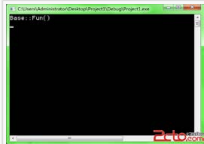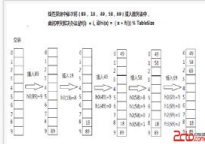- 邮件、短信、相机、图库的使用

图文推荐

app开发报价单　　达内的　　c++游戏编程　　c语言学习网站
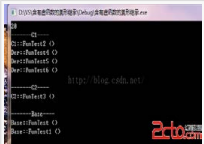
含有虚函数菱形的虚拟 C++多态篇1—静态联编 数据结构与算法——散 初识C++之虚函数

我有话说(0条评

来说两句吧...

发布

微博登录 QQ登录 手机登录

还没有评论，快来抢沙发吧！

推广

推广