



Linux TTY framework(2)_软件架构

作者：wowo 发布于：2016-9-27 22:42 分类：TTY子系统

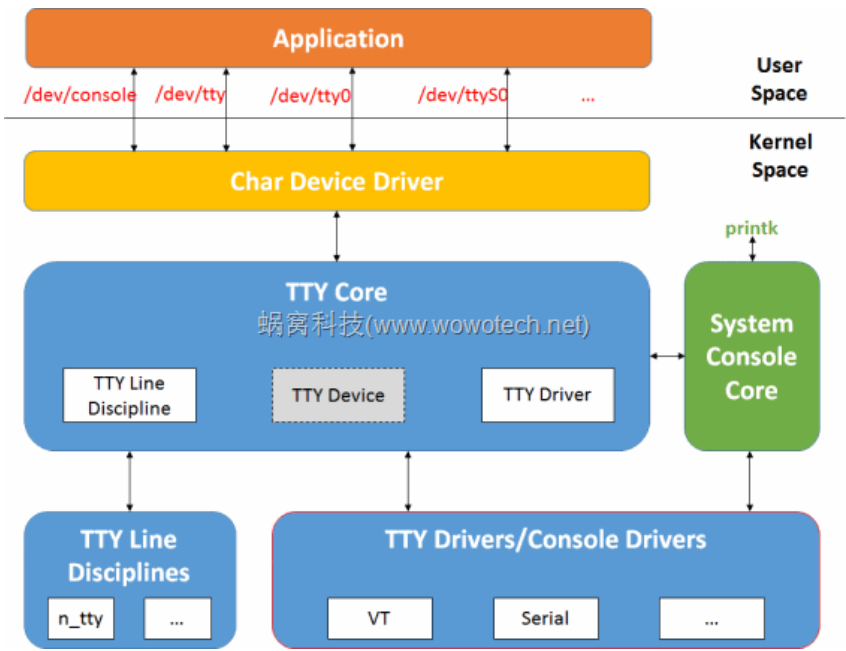
1. 前言

由“Linux TTY framework(1)_基本概念”的介绍可知，在Linux kernel中，TTY就是各类终端（ Terminal ）的简称。为了简化终端的使用，以及终端驱动程序的编写，Linux kernel抽象出了TTY framework：对上，向应用程序提供使用终端的统一接口；对下，提供编写终端驱动程序（ 如serial driver ）的统一框架。

本文是Linux TTY framework分析的第二篇文章，将从整体架构的角度，介绍Linux TTY framework，以便分解出功能相对独立的子模块，以便后续的分析。

2. 软件架构

Linux kernel TTY framework位于“drivers/tty”目录中，其软件框架如下面图片1所示：



图片1 Linux TTY framework框架

和Linux其它的framework类似，TTY framework通过TTY core屏蔽TTY有关的技术细节，对上以字符设备的形式向应用程序提供统一接口，对下以TTY device/tty driver的形式提供驱动程序的编写框架。具体请参考后续章节介绍。

2.1 TTY Core

TTY core是TTY framework的核心逻辑，功能包括：

1) 以字符设备的形式，向用户空间提供访问TTY设备的接口，例如：

设备号(主, 次)	字符设备	备注
(5, 0)	/dev/tty	控制终端（ Controlling Terminal ）
(5, 1)	/dev/console	控制台终端（ Console Terminal ）

站内搜索

搜索

功能

留言板
评论列表

最新评论

- luke
- 因公司网络限制，文章里的图片不能显示，看的好辛苦啊！！
- linuxer
- @hit201j：已经修改，多谢！
- bigpillow
- Hi Linuxer, 我们这边有写一套完整的GPIO &...
- hit201j
- 错别字：“还以为一个就是bootloader传递过来的bl...
- GrayMonkey
- 膜拜大佬，早找到你的文章就好了,Android开发一枚，一直...
- fy
- @wowo：是的。我只是就这里的“禁止抢占”非必要，问问wo...

文章分类

- Linux内核分析(9)
- 统一设备模型(13)
- 电源管理系统(42)
- 中断子系统(14)
- 进程管理(13)
- 内核同步机制(17)
- GPIO子系统(5)
- 时间子系统(14)
- 通信类协议(7)
- 内存管理(20)
- 图形子系统(1)
- 文件系统(3)
- TTY子系统(5)
- u-boot分析(3)
- Linux应用技巧(13)
- 软件开发(6)
- 基础技术(13)
- 蓝牙(16)
- ARMv8A Arch(13)
- 显示(3)
- 基础学科(9)
- 技术漫谈(12)
- 项目专区(0)
- X Project(28)

随机文章

(4, 0)	/dev/vc/0 or /dev/tty0	虚拟终端 (Virtual Terminal)
(4, 1)	/dev/vc/1 or /dev/tty1	同上
...
(x, x)	/dev/ttyS0	串口终端 (名称和设备号由驱动自行决定)
...
(x, x)	/dev/ttyUSB0	USB转串口终端
...

perfbook memory
barrier (14.2章节) 的中文翻译
(上)
ARMv8之memory model
ACCESS_ONCE宏定义的讲解
Linux内核同步机制之 (二) :
Per-CPU变量
页面回收的基本概念

文章存档

- 2017年11月(1)
- 2017年10月(1)
- 2017年9月(5)
- 2017年8月(4)
- 2017年7月(4)
- 2017年6月(3)
- 2017年5月(3)
- 2017年4月(1)
- 2017年3月(8)
- 2017年2月(6)
- 2017年1月(5)
- 2016年12月(6)
- 2016年11月(11)
- 2016年10月(9)
- 2016年9月(6)
- 2016年8月(9)
- 2016年7月(5)
- 2016年6月(8)
- 2016年5月(8)
- 2016年4月(7)
- 2016年3月(5)
- 2016年2月(5)
- 2016年1月(6)
- 2015年12月(6)
- 2015年11月(9)
- 2015年10月(9)
- 2015年9月(4)
- 2015年8月(3)
- 2015年7月(7)
- 2015年6月(3)
- 2015年5月(6)
- 2015年4月(9)
- 2015年3月(9)
- 2015年2月(6)
- 2015年1月(6)
- 2014年12月(17)
- 2014年11月(8)
- 2014年10月(9)
- 2014年9月(7)
- 2014年8月(12)
- 2014年7月(6)
- 2014年6月(6)
- 2014年5月(9)
- 2014年4月(9)
- 2014年3月(7)
- 2014年2月(3)
- 2014年1月(4)



注1：控制终端、控制台终端、虚拟终端等概念，比较抽象，我会在后续的文章中详细介绍。

2) 通过设备模型中的struct device结构抽象TTY设备，并通过struct tty_driver抽象该设备的驱动，并提供相应的register接口。TTY驱动程序的编写，简化为填充并注册相应的struct tty_driver结构。

注2：TTY framework弱化了TTY设备 (图片1中使用虚线框标注) 的概念，通常情况下，可以在注册TTY驱动的时候，自动分配并注册TTY设备。

3) 使用struct tty_struct、struct tty_port等数据结构，从逻辑上抽象TTY设备及其“组件”，以实现硬件无关的逻辑。

4) 抽象出名称为线路规程 (Line Disciplines) 的模块，在向TTY硬件发送数据之前，以及从TTY设备接收数据之后，进行相应的处理 (如特殊字符的转换等) 。

2.2 System Console Core

Linux kernel的system console主要有两个功能：

- 1) 向系统提供控制台终端 (Console Terminal) ，以便让用户登录进行交互操作。
- 2) 提供printf功能，以便kernel代码进行日志输出。

System console core模块使用struct console结构抽象system console功能，具体的driver不需要关心console的内部逻辑，填充该接口并注册给kernel即可。

2.3 TTY Line Disciplines

线路规程 (Line Disciplines) 在TTY framework中是一个非常优雅的设计，我们可以把它看成设备驱动和应用接口之间的一个适配层。从字面意思理解，就是辅助TTY driver，将通过TTY设备键入的字符转换成一行一行的数据^[3]，当然，实际情况远比这复杂，例如在蜗窝x project所使用的kernel版本中，存在如下的Line Disciplines (以n_为前缀，我们后续的文章会更为详细的介绍)：

```
pengo@DESKTOP-CH8SB7C:~/work/xprj/linux$ ls drivers/tty/n_*
drivers/tty/n_gsm.c  drivers/tty/n_r3964.c  drivers/tty/n_tracesink.c  drivers/tty/n_tty.c
drivers/tty/n_hdlc.c  drivers/tty/n_tracerouter.c  drivers/tty/n_tracesink.h
```

2.4 TTY Drivers以及System Console Drivers

最后，对内核以及驱动工程师来说，更关注的还是具体的TTY设备驱动。在kernel为我们搭建的如此beauty的框架下面，编写相应的driver就成为一件比较简单的事情了。当然的kernel中，主要的TTY driver有两类：

- 1) 虚拟终端 (Virtual Terminal , VT) 驱动，位于drivers/tty/vt中，负责实现VT (后续文章会详细介绍) 有关的功能。
- 2) 串口终端驱动，也即我们所熟知的serial subsystem (话说终于到重点了，哈哈)，位于drivers/tty/serial中。

3. 总结

本文对Linux TTY framework的软件框架作了一个简单的介绍，目的是从整体上了解Linux TTY有关的软件实现。基于本文的描述，后续计划从如下角度继续TTY framework的分析：

- 控制终端、控制台终端、虚拟终端等概念的理解及解释；
- TTY core的分析；
- System Console Core的分析；
- Serial subsystem (串口子系统) 的分析；
- 虚拟终端 (VT) 的分析；

常用线路规程 (Line Disciplines) 的介绍和分析 ;
等等。

4. 参考文档

[1] TTY驱动分析

[2] 控制终端 (controlling terminal) , <https://linux.die.net/man/4/tty>

[3] <https://utcc.utoronto.ca/~cks/space/blog/unix/TTYLineDisciplineWhy>

原创文章，转发请注明出处。蜗窝科技，www.wowotech.net。

标签: Linux Kernel 架构 Architecture tty



« X-012-KERNEL-serial early console的移植 | TLB flush操作»

评论：

emeralddream
2016-09-28 09:47

wowo 什么时候讲下canbus啊。这个网上资料很少啊

回复

wowo
2016-09-28 21:35

@emeralddream：多谢关注。can一般工业控制用的比较多，我们这些搞消费类的，确实很少接触。有时间了可以写一下，不过您也知道，我们时间太少，见谅哈~~~

回复

发表评论：

昵称

邮件地址 (选填)

个人主页 (选填)

f 22B H

发表评论