

成固定USB转串口设备节点

转数 通雪长安 于 2020-12-17 20:22:34 发布 2120 收藏 19

分类专栏: Linux常用技巧 文章标签: linux shell python usb

Linux常用技巧 专栏收录该内容 0 订阅 5 篇文章 订阅专栏

目录

[点击这里查看所有博文](#)

本系列博客所述资料 **均来自网络**，并不是本人原创（只有博客是自己写的）。出于热心，本人将自己的所学笔记整理并推出相对应的使用教程，方便其他人学习。若出现侵权现象，请告知本人，本人会立即停止更新，并删除相应的文章和代码。

1. Linux下USB设备节点名不固定问题经常会遇到

以USB转串口设备为例，通常设备节点名为ttyUSBx (x为0~n) ,Linux内核会根据设备插入的先后顺序进行编号的分配，比如第一个插入的设备编号为ttyUSB0，然后依此加1，变为ttyUSB1，ttyUSB2.....

如果仅仅以设备节点ttyUSBx来区别具体是哪个设备，因为末位的编号是随时会变的，所以就会造成混乱。无法保证A设备就是ttyUSB0，B设备就是ttyUSB1。

Linux在设备文件/dev目录下并没有提供固定显示ttyUSB的方法，但是，其实，每个USB端口都有唯一的端口号，相当于每个门店的门牌号，只要我們依据端口号来进行设备的区分，那么问题就迎刃而解了。简单点来说就是找到端口号，然后根据端口号找到挂载在这个端口号上面的USB设备是ttyUSB0还是 ttyUSB1....(这个是变化的，前面讲到了)。

2. 关于端口号的查看方法

连接好两个USB转串口设备之后执行命令：`ls -l /sys/class/tty/ttyUSB*`

```
1 [smart@localhost multiTesting]$ ls -l /sys/class/tty/ttyUSB*
2 lrwxrwxrwx 1 root root 0 Dec 17 18:39 /sys/class/tty/ttyUSB0 -> ../../devices/platform/soc/3f980000.usb/usb1/1-
3 lrwxrwxrwx 1 root root 0 Dec 17 18:39 /sys/class/tty/ttyUSB1 -> ../../devices/platform/soc/3f980000.usb/usb1/1-
4 lrwxrwxrwx 1 root root 0 Dec 17 18:39 /sys/class/tty/ttyUSB2 -> ../../devices/platform/soc/3f980000.usb/usb1/1-
5 lrwxrwxrwx 1 root root 0 Dec 17 18:39 /sys/class/tty/ttyUSB3 -> ../../devices/platform/soc/3f980000.usb/usb1/1-
6 [smart@localhost multiTesting]$ ls -l /sys/class/tty/ttyUSB*
7 lrwxrwxrwx 1 root root 0 Dec 17 18:41 /sys/class/tty/ttyUSB0 -> ../../devices/platform/soc/3f980000.usb/usb1/1-
8 lrwxrwxrwx 1 root root 0 Dec 17 18:41 /sys/class/tty/ttyUSB1 -> ../../devices/platform/soc/3f980000.usb/usb1/1-
9 lrwxrwxrwx 1 root root 0 Dec 17 18:41 /sys/class/tty/ttyUSB2 -> ../../devices/platform/soc/3f980000.usb/usb1/1-
10 lrwxrwxrwx 1 root root 0 Dec 17 18:41 /sys/class/tty/ttyUSB3 -> ../../devices/platform/soc/3f980000.usb/usb1/1-
```

我这里分别进行了两次测试

第1次先插入CH340(ttyUSB0)，后插入4G模组(虚拟出来的ttyUSB1、ttyUSB2、ttyUSB3)。

第2次先插入4G模组(虚拟出来的ttyUSB0、ttyUSB1、ttyUSB2)，后插入CH340(ttyUSB3)。

由上面的日志可知：不管先后插入的顺序如何，CH340对应的USB端口都是1-1.3.1.0，知道端口后我们该如何确定设备对应的ttyUSBx设备节点呢。

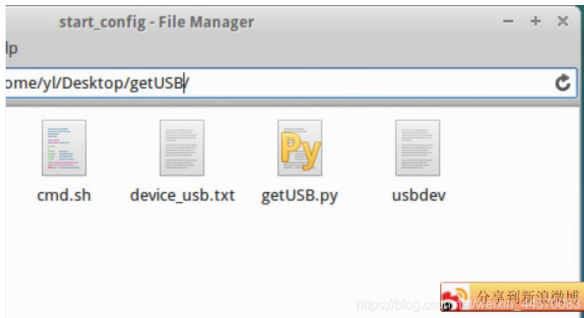
3. 实际问题，实际解决

实际工程中，碰到的一个问题是：硬件上连接有两个USB转串口设备，硬件只要一上电，两个USB设备几乎是同时上电的，这将导致ttyUSB0或者ttyUSB1无法每次固定的对应到上电的那个相同端口，上层软件需要通过串口设备节点/dev/ttyUSBx来打开一个串口，并从串口获取数据，但是这个ttyUSBx设备并不是一直都指向固定的一个usb端口号，这直接导致我们无法往下操作了。

解决办法：

这里使用bash语言加Python正则表达式的相关知识解决这个问题：解决问题的思路是：

- 第一次上电的时候，我们需要确定哪个端口上的数据是我们所需要的：
`ls -l /sys/class/tty/ttyUSB*`
假设是3-1.1这个端口是我们的data端口。
- 以后每次上电，我们要找到3-1.1这个端口后面挂载的ttyUSB设备是ttyUSB0还是ttyUSB1，并建立一个软链接将当时获取到的ttyUSBx生成一个软链接，名字固定为ttydata.那么以后每次打开/dev/ttydata就能找到正确3-1.1这个端口，并获取数据了。
- 建立一个文件夹getUSB，该文件夹下面包含：



其中cmd.sh是利用bash脚本获取/sys/class/tty/ttyUSB*的一些信息保存在device_usb.txt中。
cmd.sh如下：

```
1 #!/bin/bash
2 declare -i a=0
3 declare -i b=0
4 while [[ ! -e "/sys/class/tty/ttyUSB0" ]]
5 do
6     sudo sleep 0.01s
7     a=a+1
8     if [ $a -eq 300 ];then #等待一段时间没有检测到ttyUSB0设备到会自动跳出while
9         break
10    fi
11 done
12
13 while [[ ! -e "/sys/class/tty/ttyUSB1" ]]
14 do
15     sudo sleep 0.01s
16     b=b+1
17     if [[ $b -eq 300 || $a -ne 0 ]];then #if USB0 been detected ,also get out of while
18         break
19     fi
20 done
21
22 if [[ ! -e /sys/class/tty/ttyUSB0&& ! -e /sys/class/tty/ttyUSB1 ]]; then #如果不存在ttyUSB设备
23     echo "Not have ttyUSB0 or not have ttyUSB1"
24 else #如果完美检测到了两个ttyUSB设备，则将信息Log到device_usb.txt当中
25     tty1=$(ls -l /sys/class/tty/ttyUSB0)
26     tty2=$(ls -l /sys/class/tty/ttyUSB1)
```

March 16-19, 2023
March 16-19, 2023

Visit Site
TOHRU NAKAMURA
ALVIN LEUNG
RENE FRANK

目录
1. Linux下USB设备节点名不固定问...
2. 关于端口号的查看方法
3. 实际问题，实际解决
4. udev

分类专栏
杂项 12篇
机器视觉/人工智能 1篇
Python常见问题 1篇
python应用 1篇
4G杂项 2篇
博客目录 1篇
USB协议学习 2篇
8910芯片USB描述符 5篇
Linux常用技巧 5篇
RDA8910 CSD2二次开发... 20篇
乐鑫ESP系列环境搭建 3篇
elua解析器开源文档 10篇
Linux学习笔记 9篇
合宙Air模块AirLua开发... 12篇


```
28
29 sudo ls -l /sys/class/tty/ttyUSB0 /sys/class/tty/ttyUSB1 > ./device_usb.txt
30 fi
31
32 if [ ! -n "$tty1" ]; then # "! -n" shows blank var #非空检测
33 echo "tty1 is empty"
34 fi
35 #delay 0.01s to make sure the device_usb.txt complete
36 sudo sleep 0.01s
37 #remove the old USB device shortcut
38
39 if [ ! -e "/dev/ttydata" ]; then # 如果/dev/ttydata本身不存在
40 echo "-----/dev/ttydata not found"
41 else
42 #如果存在，则需删除之，然后重新创建之
43 echo "/dev/ttydata is exist"
44 sudo rm /dev/ttydata
45 fi
46 #next Python Language to get the right USB interface
47 ./getUSB.py #调用当前路径下的getUSB.py这个Python语言，明确此次是哪个，ttyUSB0,或者ttyUSB1挂载在端口3-1.1上
48
49 usbdev=$(cat ./usbdev) #获取到这个设备
50 echo "the device is : "
51 echo $usbdev
52
53 sudo ln -s /dev/$usbdev /dev/ttydata #将这个设备软连接到/dev/ttydata以后每次打开这个ttydata即可
54
55
56
```

getUSB.py是通过device_usb.txt中的信息，获取到当前挂着的端口3-1.1上的是ttyUSB0还是ttyUSB1并保存在usbdev 中。

getUSB.py如下：

```
1 #!/usr/bin/python
2 #coding:utf-8
3 import re #正在表达式
4 sss = open("../device_usb.txt","rb") #打开device_usb.txt设备，并读取内容
5 www = open("../usbdev","wb") #当前路径下创建usbdev文件，后续会写入内容
6 s_read = sss.read() usb3/3-1/3-1.1/
7 r = r"usb3/3-1/3-1.1.+(ttyUSB[0-9])"
8 #正则中，“需要转义，所以使用\"表示。”
9 #这个规则是找到usb3/3-1/3-1.1/这个字符串后面紧跟的是此次上电生成的ttyUSB0或者ttyUSB1
10 output = re.findall(r,s_read)
11 www.write(output[0]) #将结果写到usbdev中
12 www.close()
13 sss.close()
```

完成之后设置开机项目，将文件夹当道一个位置，然后设置开机启动cmd.sh（在/etc/rc.local中设置）则每次开机之后，会从/dev/ttydata获取到固定端口的数据。

4. udev

这两天在ubuntu中开发跟串口有关程序时，发现来回拔插串口线或者插多个串口线时总是出现串口名为ttyUSB*不固定的问题，给应用程序带来不少麻烦，遂google解决。

linux中设备号一般按先后顺序一次向后增大，udev规则文件可以解决这个问题。udev是一种Linux 2.6内核采用的/dev 目录的管理系统（可以把它认为是windows中的设备管理器），它通过从sysfs获得的信息，可以提供对特定设备的固定的设备名。sysfs是linux 2.6内核的一种新型文件系统，它提供了当前设备的基本信息。

udev的重要功能就是为设备提供固定的设备名，根据Wirting udev rules的详细介绍，udev有如下功能：

Rename a device node from the default name to something else Provide an alternative/persistent name for a device node by creating a symbolic link to the default device node Name a device node based on the output of a program Change permissions and ownership of a device node Launch a script when a device node is created or deleted (typically when a device is attached or unplugged) Rename network interfaces

简单阅读之后创建文件/etc/udev/rules.d/10-local.rule，内容如下

```
1 KERNEL=="ttyUSB*", ATTRS{idVendor}=="067b", ATTRS{idProduct}=="2303", MODE="0777", SYMLINK+="user_uart"
2 KERNEL=="ttyUSB*", ATTRS{idVendor}=="1a86", ATTRS{idProduct}=="7523", MODE="0777", SYMLINK+="mcu_uart"
```

意思就是匹配sys中内核名为ttyUSB*的设备，属性匹配依据生产商编号idVendor和产品号idProduct，设定读写权限为0777。符号链接名为：

- 1、user_uart-----PL2303串口转USB，
- 2、mcu_uart-----CH340串口转USB。

idVendor和idProduct由 lsusb -vvv 命令查看。

保存退出后udev规则就生效了，重新拔插两个串口设备，就可以看到/dev/user_uart指向/dev/ttyUSB0，/dev/mcu_uart指向/dev/ttyUSB1。这样以来，我只要在程序里打开/dev/user_uart或/dev/mcu_uart就可以一直准确的打开指定的串口设备了。

PS：如果您的两个串口用的都是PL2303或者都是CH340/CP2102，那就真的无能为力了。。。

常用的匹配类型：

- BUS：匹配总路类型，比如PCI USB等
 - KERNEL：- 匹配Kernel设备名，比如hda hdb。
- DRIVER：- 匹配Kernel的驱动程序名
- SUBSYSTEM：- 匹配子系统名。
- ID：- 匹配总路系统的ID（e.g. PCI bus ID）。
- PLACE：- 匹配物理位置（对USB很有用）

资料链接：

<http://www.cnblogs.com/jymxf/p/6379449.html>

<http://biancheng.dnbcw.net/linux/295328.html>

linux USB端口生产固定的设备节点名 感谢的专栏 440

1.将串口设备插入USB口，先查看生成的默认设备节点：/dev/ttyUSB0 2.通过指令查看节点的sys信息 udevadm info --attribute-walk --name=/dev/ttyUSB0 ...

1024程序节你知道老师上课随机点名是怎么实现的吗 最新发布 m0_67388094的博客 2321

身为程序员的你，在老师使用工具随机点名的时候，你是心里紧张的默念“抽不到我，抽不到我”，还是在琢磨老师的随机点名是如何实现的呢？今天这篇... 3-4

零基础学Linux内核之设备驱动篇(10)_设备树1_添加设备节点

8. 设备模型 9. 平台设备驱动 10. 设备树1_添加设备节点 文章目录

uart驱动分析_小K仔的博客_怎么查看uart生成的节点

一个uart_driver通常会注册一段设备号,即在用户空间会看到uart_driver对应多个设备节点。例如:/dev/ttyS0/dev/ttyS1 每个设备节点是对应一个具体硬... 2-28

