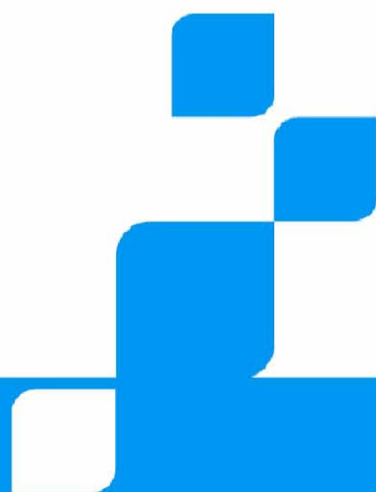




ZTE Linux 操作系统驱动安装与加载



法律声明

若接收中兴通讯股份有限公司（以下称为“中兴通讯”）的此份文档，即表示您已同意以下条款。若不同意以下条款，请停止使用本文档。

本文档版权所有中兴通讯股份有限公司。保留任何未在本文档中明示授予的权利。文档中涉及中兴通讯的专有信息。未经中兴通讯事先书面许可，任何单位和个人不得复制、传递、分发、使用和泄漏该文档以及该文档包含的任何图片、表格、数据及其他信息。

ZTE和**ZTE中兴**是中兴通讯的注册商标。中兴通讯产品的名称和标志是中兴通讯的商标或注册商标。在本文档中提及的其他产品或公司名称可能是其各自所有者的商标或注册商标。在未经中兴通讯或第三方权利人事先书面同意的情况下，阅读本文档并不表示以默示、不可反言或其他方式授予阅读者任何使用本文档中出现的任何标记的权利。

本产品符合有关环境保护和人身安全方面的设计要求，产品的存放、使用和弃置应遵照产品手册、相关合同或相关国法律、法规的要求进行。

本文档按“现状”和“仅此状态”提供。本文档中的信息随着中兴通讯产品和技术的进步将不断更新，中兴通讯不再通知此类信息的更新。

意见反馈表

为提高中兴通讯用户资料的质量，更好地为您服务，希望您在百忙之中提出您的建议和意见，并请传真至：0755-26772236，或邮寄至：深圳市高新技术产业园科技南路中兴通讯大厦中兴通讯学院文档开发部收，邮编：518057，邮箱：doc@zte.com.cn。

资料名称	ZTE Linux 操作系统驱动安装与加载					
产品版本		资料版本	20130111-R1.0			
您单位选用该产品的时间						
为了能够及时与您联系，请填写以下有关您的信息						
姓名		单位名称				
邮编		单位地址				
电话			E-mail			
您对本资料的评价		好	较好	一般	较差	差
	总体满意					
	工作指导					
	查阅方便					
	内容正确					
	内容完整					
	结构合理					
	图表说明					
	通俗易懂					
您对本资料的改进建议		详细说明				
	内容结构					
	内容详细					
	内容深度					
	表达简洁					
	增加图形					
	增加实例					
	增加 FAQ					
	其 他					
您对中兴通讯用户资料的其他建议						

前言

手册说明

本文档介绍了 Linux 操作系统驱动安装与加载的信息和方法。

版本更新说明

资料版本	资料更新说明
20130111-R1.0	手册第一次发行

目录

1	驱动安装与加载.....	1
1.1	项目信息.....	1
1.2	USB驱动添加系统组件.....	1
1.2.1	USB串口驱动添加.....	1
1.2.2	增加具体设备驱动.....	1
1.2.3	USB串口驱动过滤NDIS接口.....	2
1.3	NDIS驱动加载方法.....	2
1.3.1	NDIS驱动添加系统组件.....	3
1.3.2	NDIS驱动加载.....	3
1.3.3	NDIS驱动口使用.....	3

1 驱动安装与加载

驱动安装包括添加系统驱动以及集成 ZTE 提供的 ndis 口驱动。

1.1 项目信息

0x0199	LTE	0	AT	ZTE CMCC AT Interface
		1	NDIS	ZTE CMCC NDIS Interface
		2	Modem	ZTE CMCC MODEM
		3	LOG	ZTE CMCC LOG Interface

1.2 Usb 驱动添加系统组件

1.2.1 Usb 串口驱动添加

普通的 3G 数据卡（modem）采用 usb 接口，对上层的功能接口是串口，在 linux 系统中通常使用 usb 转串口的驱动。

驱动添加需要配置系统 linux 系统内核，配置方法如下：

- cd kernel
- make menuconfig
- device drivers--->usb support--->usb serial converter support

选中如下组件：

USB driver for GSM and CDMA modems

选中后保存配置，重新编译内核即可。

1.2.2 增加具体设备驱动

本文介绍的 ZM8623 模块驱动添加步骤如下：

- 找到内核源码文件 option.c(一般情况下，路径在..\drivers\usb\serial\option.c)
- 在源码中查找 option_ids 表，请参照上面 1.1 节中项目信息添加 ZTE 产品的 VID（0x19d2）和 PID（0x0199）。

- 添加完成后，跟系统一起编译，烧入目标设备。

选中后保存配置，重新编译内核即可。

说明：在比较早的 linux 版本中，对应的驱动文件为：usb-serial.c 文件，路径在..\drivers\usb\serial\usb-serial.c

1.2.3 Usb 串口驱动过滤 ndis 接口

由于 usb 串口跟 ndis 都属于非标准 cdc 设备，因此需要在 option.c（低版本为 usb-serial.c）中的 probe 函数内增加 blacklist 进行过滤，以防止 ndis 口被 usb 串口驱动加载导致 ndis 驱动无法顺利完成加载。参考代码如下：

```
#define ZTE_PRODUCT_2_6G          0x0199  /* 2.6G模块 */
option_probe中添加以下代码
/* ZTE 2.6G模块 */
if (serial->dev->descriptor.idVendor == ZTE_VENDOR_ID &&
    serial->dev->descriptor.idProduct == ZTE_PRODUCT_2_6G &&
    serial->interface->cur_altsetting->desc.bInterfaceNumber == 1)
    return -ENODEV;
```

经过上面的操作，正常情况下看系统的 log，开机后检测到模块的枚举过程应该如下，

```
[ 1004.143310] usb 1-1: new high speed USB device using ehci-omap and address 4
[ 1004.306396] usb 1-1: New USB device found, idVendor=19d2, idProduct=0196
[ 1004.306427] usb 1-1: New USB device strings: Mfr=1, Product=2, SerialNumber=0
[ 1004.306427] usb 1-1: Product: ZTE BootLoader
[ 1004.306457] usb 1-1: Manufacturer: ZTE Co,Ltd.
[ 1009.095306] usb 1-1: USB disconnect, address 4
[ 1016.249481] usb 1-1: new high speed USB device using ehci-omap and address 5
[ 1016.416687] usb 1-1: New USB device found, idVendor=19d2, idProduct=0199
[ 1016.416931] usb 1-1: New USB device strings: Mfr=11, Product=12, SerialNumber=13
[ 1016.416961] usb 1-1: Product: ZTE-Modem
[ 1016.416992] usb 1-1: Manufacturer: ZTE
[ 1016.416992] usb 1-1: SerialNumber: ZTE-Modem
[ 1016.434356] option 1-1:1.0: GSM modem (1-port) converter detected
[ 1016.450866] usb 1-1: GSM modem (1-port) converter now attached to ttyUSB0
[ 1016.477081] option 1-1:1.2: GSM modem (1-port) converter detected
[ 1016.495025] usb 1-1: GSM modem (1-port) converter now attached to ttyUSB1
[ 1016.511016] option 1-1:1.3: GSM modem (1-port) converter detected
[ 1016.525421] usb 1-1: GSM modem (1-port) converter now attached to ttyUSB2
```

也就是接口 1 不会被加载搭配 option 驱动上。到此，串口驱动加载 ok。

如果到此有什么问题，可以联系 ZTE 这边的接口同事协助分析。

1.3 Ndis 驱动加载方法

由于 ndis 驱动是以库的形式提供的，主要包括两个文件：cdc_encap.ko 和 cdc_ether.ko

下面主要介绍系统组件的添加以及库的加载方式。

1.3.1 ndis 驱动添加系统组件

由于 ndis 驱动需要系统的 usbnet 驱动作支持，因此需要配置系统 linux 系统内核，配置方法如下：

- cd kernel
- make menuconfig
- device drivers--->Network device support--->usb Network Adapters

选中如下组件：

<M>Mutil-purpose USB Networking Framework

选中后保存配置，重新编译内核即可。

1.3.2 ndis 驱动加载

编译后把 1.3.1 中生成的 usbnet.ko 连同 ndis 驱动一同加入到嵌入式系统启动项中，依次加载如下所示：

```
insmod usbnet.ko
```

```
insmod cdc_encap.ko
```

```
insmod cdc_ether.ko
```

带设备枚举完成后，正常加载的系统 log 如下：

```
|cdc_ether 1-1.1:1.1: ZTE Ethernet Device: claiming interface 1
|usb 1-1.1: bad CDC descriptors
|usb0: register 'cdc_ether' at usb-platform-1.1, ZTE Ethernet Device, 5a:a8:1a:d0:50:e1
```

至此，驱动加载全部完成。

1.3.3 ndis 驱动口使用

驱动编译完成后，需要手动创建设备节点，由于 ndis 驱动是动态分配的，名字为 cdcecm_usb0 设备，因此需要在嵌入式系统的 terminal 中输入

```
cat /proc/devices
```


查找其对应的主辅设备号，然后在驱动加载完成后创建该设备节点，如下例子：

```
~ # cat /proc/devices | grep cdcecm_usb0  
253 cdcecm_usb0
```

本例中主设备号为 253（不同嵌入式系统设备这里可能不同，以本地看到的为主）

创建设备节点：

```
mknod /dev/cdcecm_usb0 c 253 0
```

之后可以通过

```
ifconfig -a
```

查到其创建的网络接口

```
usb0      Link encap:Ethernet  HWaddr C2:F3:BA:09:FB:3E  
          BROADCAST MULTICAST  MTU:1500  Metric:1  
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0  
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0  
          collisions:0 txqueuelen:1000  
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)
```

至此，驱动工作全部完成。