

RG500U-CN Linux USB

驱动用户指导

5G 模块系列

版本：1.0

日期：2021-01-28

状态：受控文件

上海移远通信技术股份有限公司始终以为客户提供最及时、最全面的服务为宗旨。如需任何帮助，请随时联系我司上海总部，联系方式如下：

上海移远通信技术股份有限公司

上海市闵行区田林路 1016 号科技绿洲 3 期（B 区）5 号楼 邮编：200233

电话：+86 21 51086236 邮箱：info@quectel.com

或联系我司当地办事处，详情请登录：<http://www.quectel.com/cn/support/sales.htm>。

如需技术支持或反馈我司技术文档中的问题，可随时登陆如下网址：

<http://www.quectel.com/cn/support/technical.htm> 或发送邮件至：support@quectel.com。

前言

上海移远通信技术股份有限公司提供该文档内容用以支持其客户的产品设计。客户须按照文档中提供的规范、参数来设计其产品。因未能遵守有关操作或设计规范而造成的损害，上海移远通信技术股份有限公司不承担任何责任。在未声明前，上海移远通信技术股份有限公司有权对该文档进行更新。

免责声明

上海移远通信技术股份有限公司尽力确保开发中功能的完整性、准确性、及时性或效用，但不排除上述功能错误或遗漏的可能。除非其他有效协议另有规定，否则上海移远通信技术股份有限公司对开发中功能的使用不做任何暗示或明示的保证。在适用法律允许的最大范围内，上海移远通信技术股份有限公司不对任何因使用开发中功能而遭受的损失或损害承担责任，无论此类损失或损害是否可以预见。

保密义务

除非上海移远通信技术股份有限公司特别授权，否则我司所提供文档和信息的接收方须对接收的文档和信息保密，不得将其用于除本项目的实施与开展以外的任何其他目的。未经上海移远通信技术股份有限公司书面同意，不得获取、使用或向第三方泄露我司所提供的文档和信息。对于任何违反保密义务、未经授权使用或以其他非法形式恶意使用所述文档和信息的违法侵权行为，上海移远通信技术股份有限公司有权追究法律责任。

版权申明

本文档版权属于上海移远通信技术股份有限公司，任何人未经我司允许而复制转载该文档将承担法律责任。

版权所有 ©上海移远通信技术股份有限公司 2021，保留一切权利。

Copyright © Quectel Wireless Solutions Co., Ltd. 2021.

文档历史

修订记录

版本	日期	作者	变更表述
-	2021-01-11	Lee LI	文档创建
1.0	2021-01-28	Lee LI	受控版本

目录

文档历史	2
目录	3
表格索引	4
图片索引	5
1 引言	6
2 Linux USB 接口概述	7
3 Linux USB 驱动移植	8
3.1. USB 转串口驱动	8
3.1.1. 添加 VID 和 PID	8
3.1.2. 添加 USB 零包机制	8
3.1.3. 增加复位恢复机制	9
3.1.4. 增加批量输出 URB 的数量和容量 (Linux 内核 2.6.29 及以下版本)	10
3.1.5. 使能内核配置项	11
3.2. USB 网卡驱动	11
3.3. 使能 PPP 拨号 (不推荐)	12
4 AT 和 USB 网卡拨号功能测试	14
4.1. AT 功能测试	14
4.2. USB 网卡拨号测试	14
5 常见问题	15
5.1. 如何检查 Linux 系统中是否移植了所需 USB 驱动	15
5.2. 如何检查模块是否正确加载 Linux USB 驱动	15
6 附录 A 参考文档和术语缩写	16

表格索引

表 1: Linux USB 接口信息	7
表 2: USB 网卡模式	11
表 3: 参考文档	16
表 4: 术语缩写	16

图片索引

图 1: Kernel 中配置 USB 网卡.....	12
图 2: AT 功能测试结果	14
图 3: 内核日志	15

1 引言

本文档主要介绍如何在 Linux 系统上移植移远通信 5G 模块 RG500U-CN 的 USB 转串口驱动和 USB 网卡驱动、如何测试模块的 AT 功能和 USB 网卡拨号功能以及驱动移植相关常见问题。

2 Linux USB 接口概述

移远通信 RG500U-CN 模块的 USB 驱动包含多个不同的功能接口。下表描述了在 Linux 操作系统下模块 USB 接口的详细信息：

表 1: Linux USB 接口信息

模块的 VID 和 PID	Linux USB 驱动程序	接口
VID: 0x2c7c PID: 0x0900	USB RNDIS/ECM/NCM/MBIM 网卡	接口 0/1: USB 网络适配器
	USB 转串口 option	接口 2: DIAG 命令通信端口
		接口 3: LOG 端口
		接口 4: AT 命令通信端口
		接口 5: Modem 命令通信端口
		接口 6: NMEA 命令通信端口
	USBFS	接口 7: ADB 命令通信端口

3 Linux USB 驱动移植

本章节介绍如何移植 Linux USB 转串口驱动和 USB 网卡驱动。

3.1. USB 转串口驱动

当模块成功加载 USB 转串口 option 驱动后, Linux 会在 `/dev` 目录下创建多个名称如 `ttyUSB0`、`ttyUSB1`、`ttyUSB2` 等的串口设备文件 (Linux 系统下串口设备文件名称非固定, 由系统自动分配可用的名称)。

以下章节介绍如何移植 USB 转串口 option 驱动。

3.1.1. 添加 VID 和 PID

在文件 `[KERNEL]/drivers/usb/serial/option.c` 中添加模块的 VID 和 PID 信息, 如下所示:

```
static const struct usb_device_id option_ids[] = {
    #if 1 //Added by Quectel
    { USB_DEVICE_AND_INTERFACE_INFO(0x2c7c, 0x0900, 0xff, 0x00, 0x00) },
    #endif
}
```

备注

如果用户使用了移远通信提供的驱动文件 (`option.c`), 建议用户检查 `[KERNEL]/drivers/usb/serial/option.c` 中 `option_probe` 函数。依据第 2 章介绍, 模块接口 2/3/4/5/6 是串口, 需要保证 USB 接口号超过 4 个以后的接口不会被过滤掉。

3.1.2. 添加 USB 零包机制

对于 USB Bulk Out 传输模式, 如果发送的数据长度是 USB 数据包长的整数倍, 需要额外再发送一个长度为零的数据包以通知对端数据传送完毕。

- 对于 Linux 内核 2.6.35 及以上版本, 请添加如下语句至文件 `[KERNEL]/drivers/usb/serial/usb_wwan.c`。

```
static struct urb *usb_wwan_setup_urb(struct usb_serial *serial, int endpoint,
                                     int dir, void *ctx, char *buf, int len, void (*callback) (struct urb *))
{
    .....
    usb_fill_bulk_urb(urb, serial->dev,
                     usb_sndbulkpipe(serial->dev, endpoint) | dir,
                     buf, len, callback, ctx);
    #if 1    //Added by Quectel for zero packet
    if (dir == USB_DIR_OUT) {
        struct usb_device_descriptor *desc = &serial->dev->descriptor;

        if (desc->idVendor == cpu_to_le16(0x2C7C))
            urb->transfer_flags |= URB_ZERO_PACKET;
    }
    #endif
    return urb;
}
```

- 对于 Linux 内核 2.6.34 及以下版本，请添加如下语句至文件 `[KERNEL]/drivers/usb/serial/option.c`。

```
/* Helper functions used by option_setup_urbs */
static struct urb *option_setup_urb(struct usb_serial *serial, int endpoint,
                                     int dir, void *ctx, char *buf, int len,
                                     void (*callback)(struct urb *))
{
    .....
    usb_fill_bulk_urb(urb, serial->dev,
                     usb_sndbulkpipe(serial->dev, endpoint) | dir,
                     buf, len, callback, ctx);
    #if 1    //Added by Quectel for zero packet
    if (dir == USB_DIR_OUT) {
        struct usb_device_descriptor *desc = &serial->dev->descriptor;

        if (desc->idVendor == cpu_to_le16(0x2C7C))
            urb->transfer_flags |= URB_ZERO_PACKET;
    }
    #endif
    return urb;
}
```

3.1.3. 增加复位恢复机制

当 MCU 进入挂起或睡眠模式时，某些 USB 主机控制器或 USB 集线器可能会断电或复位，且 MCU 退出挂起或睡眠模式后无法自动恢复 USB 设备。请添加以下语句以使能复位恢复流程。

- 对于 Linux 内核 3.5 及以上版本，请添加如下语句至文件[*KERNEL*]/drivers/usb/serial/option.c。

```
static struct usb_serial_driver option_1port_device = {
.....
#ifdef CONFIG_PM
    .suspend          = usb_wwan_suspend,
    .resume           = usb_wwan_resume,
    #if 1 //Added by Quectel
        .reset_resume = usb_wwan_resume,
    #endif
#endif
};
```

- 对于 Linux 内核 3.4 及以下版本，请添加如下语句至文件[*KERNEL*]/drivers/usb/serial/usb-serial.c。

```
/* Driver structure we register with the USB core */
static struct usb_driver usb_serial_driver = {
    .name =          "usbserial",
    .probe =         usb_serial_probe,
    .disconnect =    usb_serial_disconnect,
    .suspend =       usb_serial_suspend,
    .resume =        usb_serial_resume,
    #if 1 //Added by Quectel
        .reset_resume = usb_serial_resume,
    #endif
    .no_dynamic_id = 1,
    .supports_autosuspend = 1,
};
```

3.1.4. 增加批量输出 URB 的数量和容量（Linux 内核 2.6.29 及以下版本）

对于 Linux 内核 2.6.29 及以下版本，需增加批量输出 URB 的数量和容量以获取更快的上行速率。请添加如下语句至文件[*KERNEL*]/drivers/usb/serial/option.c。

```
#define N_IN_URB 4
#define N_OUT_URB 4 //Increase the quantity of the bulk out URBs to 4
#define IN_BUFLEN 4096
#define OUT_BUFLEN 4096 //Increase the capacity of the bulk out URBs to 4096
```

3.1.5. 使能内核配置项

为了使用 USB 转串口 option 驱动，必须使能以下 Linux 内核配置项：

- CONFIG_USB_SERIAL
- CONFIG_USB_SERIAL_WWAN
- CONFIG_USB_SERIAL_OPTION

3.2. USB 网卡驱动

模块支持 MBIM/RNDIS/ECM/NCM 四种网卡功能，Linux 系统默认支持这些 USB 网卡功能，系统内置驱动模块，不需要对 Linux 系统的驱动文件做任何修改，驱动源码由 GNU Linux 维护。

模块连接到 Linux Host，并成功加载 USB 对应的网卡驱动后，将会在 Host 生成一个网卡。cdc_mbim 还会生成一个 cdc-wdm 字符设备用于命令交互。模块的网卡模式可由 AT 命令配置，如下表所示：

表 2：USB 网卡模式

USB 网卡模式	内核驱动	AT 命令
ECM	cdc_ether	通过 AT+QCFG="usbnet",1 配置 ECM 模式
MBIM	cdc_mbim (内核 3.18 及以上版本)	通过 AT+QCFG="usbnet",2 配置 MBIM 模式
RNDIS	rndis_host	通过 AT+QCFG="usbnet",3 配置 RNDIS 模式
NCM	cdc_ncm	通过 AT+QCFG="usbnet",5 配置 NCM 模式

有关如上 AT 命令的详细信息，请参考[文档 \[1\]](#)。

如需使用 USB 网卡功能，请按照下面的步骤配置 kernel。

第 1 步：执行如下命令切换到 kernel 目录。

```
CD <kernel 目录>
```

第 2 步：执行如下命令设置环境变量并导出用户设备操作系统中的 defconfig 文件。

```
export ARCH=arm
export CROSS_COMPILE=arm-none-linux-gnueabi-
make bcmrpi_defconfig
```

第 3 步：执行如下命令编译 kernel。

```
make menuconfig
```

第 4 步：执行如下命令通过如下图所示选项启用 USB 网卡功能。

```
> Device Drivers > Network device support > USB Network Adapters
```

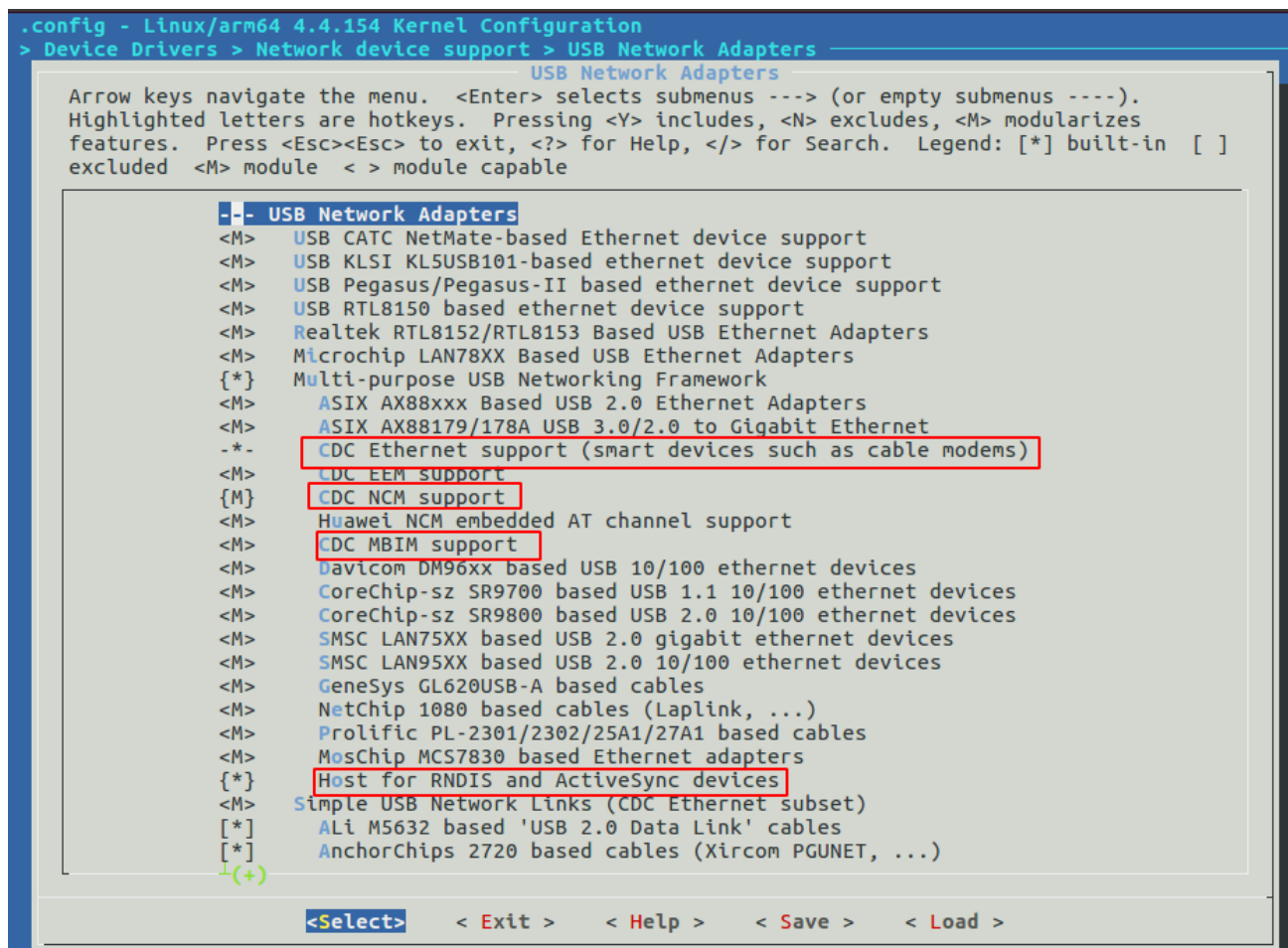


图 1：Kernel 中配置 USB 网卡

3.3. 使能 PPP 拨号（不推荐）

PPP 拨号相对于 USB 网卡上网方式，有如下缺点：

- 使用方式更复杂
- 相同网速下会消耗更高的 CPU
- 数据传输达不到理论速率

因此，不推荐使用 PPP 拨号。如果确实需要，则必须使能以下 Linux 内核配置项：


- CONFIG_PPP
- CONFIG_PPP_ASYNC
- CONFIG_PPP_SYNC_TTY
- CONFIG_PPP_DEFLATE

4 AT 和 USB 网卡拨号功能测试

4.1. AT 功能测试

当模块成功加载 USB 转串口 option 驱动后, Linux 会在 `/dev` 目录下创建多个名称如 `ttyUSB0`、`ttyUSB1`、`ttyUSB2` 等串口设备文件 (Linux 系统下串口设备文件名称非固定, 由系统自动分配可用的名称), 其中第三个串口为模块的 AT 命令端口。可使用串口工具, 如 `minicom` 或 `busybox microcom`, 来测试 AT 功能。

下图所示为使用 `busybox microcom` 工具测试的 AT 功能结果, 示例中系统分配的第三个串口名称为 `/dev/ttyUSB2`。



```
→ ~ busybox microcom /dev/ttyUSB2 -s115200
Quectel
RG500U_CNAA
Revision: RG500UCNAAR01A01M2G_GW_BETA1117
OK
```

图 2: AT 功能测试结果

4.2. USB 网卡拨号测试

USB 网卡拨号详情, 参见《Quectel_RG500U-CN_网卡拨号应用指导》。

5 常见问题

5.1. 如何检查 Linux 系统中是否移植了所需 USB 驱动

目录 `/sys/bus/usb/drivers` 下的文件列表可用来查看 Linux 系统移植了哪些 USB 驱动程序。例如：

```
carl@carl-OptiPlex-7010:~$ ls /sys/bus/usb/drivers
hub option usb usbfs usbhid usbserial usbserial_generic rndis_host cdc_ether cdc_ncm cdc_mbim
```

如需移植 USB 转串口驱动，请确保 `option` 存在。如需移植 USB NCM 驱动，请确保 `cdc_ncm` 存在；如需移植 USB ECM 驱动，请确保 `cdc_ether` 存在；如需移植 USB MBIM 驱动，请确保 `cdc_mbim` 存在；如需移植 USB RNDIS 驱动，请确保 `rndis_host` 存在。

5.2. 如何检查模块是否正确加载 Linux USB 驱动

本章节展示了当模块正确加载 USB 驱动后，Linux 系统会打印的相应日志信息。用户可以通过对比本章节里的日志与实际获取的日志，来查看模块是否正确地加载了 USB 驱动。

```
[591946.774786] usb 2-1: new SuperSpeed Gen 1 USB device number 16 using xhci_hcd
[591946.792507] usb 2-1: New USB device found, idVendor=2c7c, idProduct=0900, bcdDevice= 4.04
[591946.792510] usb 2-1: New USB device strings: Mfr=1, Product=2, SerialNumber=3
[591946.792512] usb 2-1: Product: RG500U
[591946.792513] usb 2-1: Manufacturer: Quectel
[591946.792515] usb 2-1: SerialNumber: 90611891533345
[591946.808004] rndis_host 2-1:1.0 usb0: register 'rndis_host' at usb-0000:00:14.0-1, RNDIS device, d2:c7:9a:95:22:a5
[591947.431604] rndis_host 2-1:1.0 enp0s20f0u1: renamed from usb0
[591949.566326] audit: type=1130 audit(1606446190.021:3372): pid=1 uid=0 auid=4294967295 ses=4294967295 subj==unconfined msg=
[591959.629167] audit: type=1131 audit(1606446200.084:3373): pid=1 uid=0 auid=4294967295 ses=4294967295 subj==unconfined msg=
[592664.966237] option 2-1:1.2: GSM modem (1-port) converter detected
[592664.966453] usb 2-1: GSM modem (1-port) converter now attached to ttyUSB2
[592664.966569] option 2-1:1.3: GSM modem (1-port) converter detected
[592664.966731] usb 2-1: GSM modem (1-port) converter now attached to ttyUSB3
[592664.966818] option 2-1:1.4: GSM modem (1-port) converter detected
[592664.966936] usb 2-1: GSM modem (1-port) converter now attached to ttyUSB4
[592664.967015] option 2-1:1.5: GSM modem (1-port) converter detected
[592664.967133] usb 2-1: GSM modem (1-port) converter now attached to ttyUSB5
[592664.967211] option 2-1:1.6: GSM modem (1-port) converter detected
[592664.967327] usb 2-1: GSM modem (1-port) converter now attached to ttyUSB6
[592951.833328] audit: type=1325 audit(1606447192.285:3374): table=filter family=7 entries=0 op=register pid=76768 subj==unco
```

图 3：内核日志

6 附录 A 参考文档和术语缩写

表 3: 参考文档

序号	文档名	描述
[1]	Quectel_RG500U-CN_AT 命令手册	RG500U-CN 模块 AT 命令手册
[2]	Quectel_RG500U-CN_网卡拨号应用指导	RG500U-CN 网卡拨号应用指导

表 4: 术语缩写

术语	英文全称	中文全称
APN	Access Point Name	接入点名称
ADB	Android Debug Bridge	安卓调试桥
CDC	Communications Device Class	通信设备类
CPU	Central Processing Unit	中央处理器
DNS	Domain Name System	域名系统
ECM	Ethernet Control Model	以太网控制模式
IP	Internet Protocol	网际互连协议
MCU	Microcontroller Unit	微控制器
MBIM	Mobile Broadband Interface Model	移动宽带接口模型
NCM	Network Control Model	网络控制模式
NMEA	NMEA (National Marine Electronics Association) 0183 Interface Standard	NMEA (美国国家海洋电子协会) 0183 接口标准
RNDIS	Remote Network Driver Interface Specification	远程网络驱动程序接口规范
PID	Product ID	产品标识
PPP	Point-to-Point Protocol	点对点协议

VID	Vendor ID	厂家标识
URB	USB Request Block	USB 请求块
USB	Universal Serial Bus	通用串行总线