MEIG
股票代码:002881

全球领先的物联网终端及无线数据方案提供商

Meige module Linux serial port driver loading and GOBINET dial-up guide

Controlled file name: MeiG module Linux serial port driver loading and Gobinet dial-up guide

Controlled version number:

V2.1 Publishing organization: MeiG Intelligent Technology

Co., Ltd. Release date: January 26, 2018

美格智能技术股份有限公司
MEIG SMART TECHNOLOGY CO.,LTD
地址：中国·深圳·宝安区福永街道凤凰第四工业区岭下路5号方格科技园
电话：86-755-61163458
传真：86-755-61163452
邮箱：m2msales@meigsmart.com
网址：www.meigsmart.com
扫一扫，微信关注美格智能

Important Notice

Copyright Notice

Disclaimer

MeiG Intelligent Technology Co., Ltd. makes no representations or warranties, express or implied, regarding any content in this document, and makes no representations or warranties regarding any content in this document, nor does it warrant any warranty of any kind for any particular purpose.

merchantability or fitness for purpose or any indirect, special or consequential damages.

Confidentiality Statement

The information contained in this document (including any attachments) is confidential information. The recipient understands that the document is confidential and shall not be used for any purpose other than the specified purpose, nor shall it be disclosed to any third party.

Disclaimer

The company does not assume any responsibility for property or personal injury caused by abnormal operation of the customer. Please follow the technical specifications and reference equipment in the manual.

The company has the right to modify the contents of this manual according to the needs of technological development before any declaration is made.

Without further notice.

Modification Record

Version Number	date	Revisions
V1.0	2016-11-14	Initial establishment
V1.1	2016-05-05	Update NDIS dial-up
V1.2	2016-05-09	Update NDIS dialing chart
V1.3	2016-05-12	Update ECM dial
V1.4	2016-05-25	Update ECM dial chart
V1.5	2016-06-19	Update gobinet to support linux 2.6.21
V1.6	2016-08-23	Update NDIS to support Linux kernel 2.6.32 and above
V1.7	2017-05-25	Updated NDIS to support 4.4 kernel
V1.8	2017-07-23	Update NDIS QC$RMCALL command
V1.9	2017-09-05	Update DNS configuration
V2.0	2018-01-15	Updated gobinet network test chart
V2.1	2018-01-26	Update firmware V15 version support

sheet

picture

1. Manual Description

This document introduces the serial port driver loading and GobiNet dialing steps of the Meige LTE module in the Linux system.

2.USB port information

The Mega Module USB has 6 ports, as shown in the table:

Table 1: MeiG module USB 6 ports

Vendor ID:0x05C6 Product ID:0xF601	
Port Number	**Function**
0	Diagnostic port for upgrading and debugging
1	Modem port, used for PPP dial-up
2	AT port, AT command interaction
3	NMEA port, get GPS data
4	ADB port, used for upgrading and debugging
5	NDIS network card for GobiNet dial-up

For MeiG firmware version V15 and above, the port combination is:

Table 2: 6 USB ports on the Mega module

Vendor ID:0x05C6 Product ID:0xF615	
Port Number	**Function**
0	Diagnostic port for upgrading and debugging
1	Modem port, used for PPP dial-up
2	AT port, AT command interaction
3	NMEA port, get GPS data
4	NDIS network card for GobiNet dial-up

Use AT+SGSW to distinguish the versions.

AT+SGSW

SoftwareVersion: SLM730_2.0.9_EQ133

InnerVersion: SLM730_EQ133_00B.B8E980.B8A1B0_171215_209_C00_V15

Build_date: [Dec 19 2017 10:15:16]

OK

3. Add USB serial port

Due to differences in kernel versions, the modifications to add serial port support before and after Linux 2.6.30 are different.

Linux Kernel >2.6.30

1. Modify option.c to add PID and VID

2. Block ndis and adb ports in option.c

2.6.30> Linux Kernle > 2.6.21

1. Modify option.c to add PID and VID

2.ÿÿ option.c -> option_send_setup -> if(port->number!=0) return 0;

3. Modify the port number restriction option.c -> option_send_setup ->

return usb_control_msg(serial->dev,usb_rcvctrlpipe(serial->dev, 0),

0x22,0x21,val,0,NULL,0,USB_CTRL_SET_TIMEOUT);

Change to return usb_control_msg(serial->dev,

usb_rcvctrlpipe(serial->dev,0),
0x22,0x21,val,serial->interface->cur_altsetting->desc.bInterfaceNumber,NULL,0,USB_CTRL_SET_TIMEOUT);

4. Shield the ndis and adb ports in usb-serial.c, in the usb_serial_probe function.

3.1 Add system support

First, configure the kernel to support USB to serial port. The reference configuration method is as follows:

Execute make menuconfig

Open the following components:

device drivers->usb support->usb serial converter support->USB driver for GSM and CDMA modems

3.2 Add serial port ID

Open the kernel source file kernel/drivers/usb/serial/option.c and add the product's VID and PID in option_ids.

/* Vendor and product IDs */

```
//wangbo add


#define MEIG_VENDOR_ID                                  0x05C6
#define MEIG_PRODUCT_730                                0xF601
#define MEIG_PRODUCT_730V15                             0xF615


static const struct usb_device_id option_ids[] = {
//wangbo add
//              { USB_DEVICE(0x05C6, 0xF601) },
                { USB_DEVICE(MEIG_VENDOR_ID,MEIG_PRODUCT_730) },
                { USB_DEVICE(MEIG_VENDOR_ID,MEIG_PRODUCT_730V15) },
```

3.3 Deleting NDIS and ADB ports

NDIS and ADB ports are not serial devices, so you need to remove these two ports in the serial driver. Use option driver and usb-serial The location of the driver modification is different.

Step 1: Use option driver

Modify kernel/driver/usb/serial/option.c, option_probe function

```
static int option_probe(struct usb_serial *serial,
                        const struct usb_device_id *id)
{
        struct usb_wwan_intf_private *data;
//wangbo add for 730
        if (serial->dev->descriptor.idVendor == MEIG_VENDOR_ID &&
            (serial->dev->descriptor.idProduct == MEIG_PRODUCT_730||
serial->dev->descriptor.idProduct == MEIG_PRODUCT_730V15)&&
            serial->interface->cur_altsetting->desc.bInterfaceNumber >= 4)
            return -ENODEV;
```

If the kernel version is 2.6 series, the serial port driver needs to comment out the port number limit in the option_send_setup function in option.c. In addition, the usb_control_msg function needs to be changed as follows:

```
static int option_send_setup(struct usb_serial_port *port)
```

```
{
        struct usb_serial *serial = port->serial;
        struct option_port_private *portdata;
        dbg("%s", __FUNCTION__);
//wangbo@meigsmart.com remove for SLM730 start
//      if (port->number != 0)
//              return 0;
//end
        portdata = usb_get_serial_port_data(port);

        if (port->tty) {
                int val = 0;
                if (portdata->dtr_state)
                        val |= 0x01;
                if (portdata->rts_state)
                        val |= 0x02;
//wangbo@meigsmart.com remove for SLM730 start
//
//              return usb_control_msg(serial->dev,
//                      usb_rcvctrlpipe(serial->dev, 0),
//                      0x22,0x21,val,0,NULL,0,USB_CTRL_SET_TIMEOUT);
                return usb_control_msg(serial->dev,
                        usb_rcvctrlpipe(serial->dev, 0),
                        0x22,0x21,val,serial->interface->cur_altsetting->desc.bInterNumber,
NULL,0,USB_CTRL_SET_TIMEOUT);
        }
        return 0;
}
```

Step 2: Use the usb-serial driver

Modify kernel/driver/usb/serial/usb-serial.c, usb_serial_probe function

```
                serial = create_serial (dev, interface, type);
                if (!serial) {
                        unlock_kernel();
                        dev_err(&interface->dev, "%s - out of memory\n", __FUNCTION__);
                        return -ENOMEM;
                }
        //wangbo add
                if ( (serial->dev->descriptor.idVendor == MEIG_VENDOR_ID &&
                        (serial->dev->descriptor.idProduct == MEIG_PRODUCT_730||
serial->dev->descriptor.idProduct == MEIG_PRODUCT_730V15) )&&
                        serial->interface->cur_altsetting->desc.bInterfaceNumber >=4 )
                        return -ENOMEM;
```

4.NDIS port driver loading method

4.1 NDIS driver adds system components

The NDIS driver needs to configure the Linux kernel to support usbnet. The reference configuration method is as follows:

Execute make menuconfig and select the following components:

device drivers->Network device support->usb Network Adapters->Multi-purpose USB Networking Framework

4.2 Driver Compilation

NDIS drivers are provided in source code form and compiled by users on their own systems.

You can choose to compile it separately or compile it in the kernel.

4.2.1 Separate compilation

Execute make in the driver directory to generate the GobiNet.ko driver.

4.2.2 Compiling with the kernel

Copy the code folder to the kernel/drivers/net/ directory and add the following content to the Makefile:

```
obj-m += gobinet/
```

If you want to compile the driver into the kernel, change obj-m to obj-y. The Makefile example is as follows:

```
# Makefile for USB Network drivers

obj-m += GobiNet.o

GobiNet-objs := GobiUSBNet.o QMIDevice.o QMI.o usbnet_3_12_xx.o

KDIR := /lib/modules/$(shell uname -r)/build

PWD := $(shell pwd)

all:

	$(MAKE) -C $(KDIR) M=$(PWD) modules

install: all

	mkdir -p $(OUTPUTDIR)

	cp -f GobiNet.ko $(OUTPUTDIR)

	depmod
```

```
clean:

    rm -rf *.o *~ core .depend .*.cmd *.ko *.mod.c .tmp_versions Module.* modules.order
```

5. Driver loading method

Suitable for manual loading and testing when the driver is compiled as a module. If compiled into the kernel, it will be loaded automatically.

5.1 Loading USB serial port driver

Execute modprobe option and use dmesg to view the kernel log. The four ttyUSBs correspond to the USB diagnostic (ttyUSB0), Modem (ttyUSB1), AT (ttyUSB2), and NMEA (ttyUSB3) ports.

```
[root@bogon MeiG-CM]# lsusb -t
/: Bus 02.Port 1: Dev 1, Class=root_hub, Driver=uhci_hcd/2p, 12M
    |__ Port 1: Dev 2, If 0, Class=Human Interface Device, Driver=usbhid, 12M
    |__ Port 2: Dev 3, If 0, Class=Hub, Driver=hub/7p, 12M
/: Bus 01.Port 1: Dev 1, Class=root_hub, Driver=ehci_hcd/6p, 480M
    |__ Port 1: Dev 2, If 0, Class=Vendor Specific Class, Driver=option, 480M
    |__ Port 1: Dev 2, If 1, Class=Vendor Specific Class, Driver=option, 480M
    |__ Port 1: Dev 2, If 2, Class=Vendor Specific Class, Driver=option, 480M
    |__ Port 1: Dev 2, If 3, Class=Vendor Specific Class, Driver=option, 480M
    |__ Port 1: Dev 2, If 4, Class=Vendor Specific Class, Driver=, 480M
 |__ Port 1: Dev 2, If 5, Class=Vendor Specific Class, Driver=GobiNet, 480M
```

5.2 Loading NDIS Driver

Execute modeprobe gobinet.

Note: Different devices will have different network card names, and the devices used for dialing will be different. You need to modify the dial-up code main.c +371 line of code. If the device is 1, change the following 0 to 1.

```
if (kernel_version >= KVERSION( 2,6,39 ))
        sprintf(net_path, "/sys/class/net/eth%d", 0);
else
sprintf(net_path, "/sys/class/net/usb%d", 0);
```

6. Gobinet dial-up test

MeiG provides two dialing methods

1. QMI dial

2. Speed Dial

6.1 QMI dial

Demonstrate the basic GobiNet dial-up process on Ubuntu PC. The dial-up script provides source code, and the customer compiles the executable file. After compilation, Execute ./MeiG-CM –s [apn] [user] [password] [auth]

[root@bogon MeiG-CM]# ./MeiG-CM -s cmnet

Mobile APN is cmnet

China Unicom APN is 3gnet

Telecom APN is ctnet

Use cmda 1x to gobinet dial, that is, at+mododr=8, and you need to set the APN.

For example: AT+CGDCONT=1,"IP","CTNET" , dial ./MeiG-CM –s ctnet card card.

Use ifconfig to view the IP address

[root@bogon MeiG-CM]# ifconfig ens35u1i5

ens35u1i5: flags=4291<UP,BROADCAST,RUNNING,NOARP,MULTICAST> mtu 1500

inet 10.71.167.4 netmask 255.255.255.248 broadcast 10.71.167.7

inet6 fe80::5c1a:3dff:fe64:9288 prefixlen 64 scopeid 0x20<link>

ether 5e:1a:3d:64:92:88 txqueuelen 1000 (Ethernet)

RX packets 4 bytes 1254 (1.2 KiB)

RX errors 0 dropped 0 overruns 0 frame 0

TX packets 4 bytes 1312 (1.2 KiB)

TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

Use ping to test network connection

[root@bogon MeiG-CM]# ping -I ens35u1i5 www.baidu.com

PING www.a.shifen.com (61.135.169.121) from 10.71.167.4 ens35u1i5: 56(84) bytes of data.

64 bytes from 61.135.169.121: icmp_seq=1 ttl=53 time=82.3 ms

64 bytes from 61.135.169.121: icmp_seq=2 ttl=53 time=59.9 ms

64 bytes from 61.135.169.121: icmp_seq=3 ttl=53 time=59.6 ms

64 bytes from 61.135.169.121: icmp_seq=4 ttl=53 time=71.2 ms

64 bytes from 61.135.169.121: icmp_seq=5 ttl=53 time=74.0 ms

64 bytes from 61.135.169.121: icmp_seq=6 ttl=53 time=73.2 ms

^C

--- www.a.shifen.com ping statistics ---

6 packets transmitted, 6 received, 0% packet loss, time 5008ms

rtt min/avg/max/mdev = 59.630/70.088/82.373/8.051 ms

6.1.1 PC environment test

In a PC environment, get the device name using ifconfig.

Run dhclient eth1 to obtain an IP address.

6.1.2 Embedded Environment

In an embedded environment, the udhcpc command of busybox is usually used to obtain the IP address.

There is also a ported version of dhclient that does this.

Line 126 of MeiG's CM code main.c is implemented through the udhcpc_start() function.

```
125     if (link)
126         udhcpc_start(profile);
127     else
128         udhcpc_stop(profile);
129
```

Figure 1: udhcpc_start() function diagram

Customers can re-implement this part according to their own needs.

like:

```
if(link)
        system("pkill dhclient");
        system("dhclient eth1");
    else
        system("pkill dhclient");
```

Figure 2: Example of customer implementation based on requirements

6.1.3 Exception handling

Cannot ping the domain name (www.baidu.com), but can ping the DNS (8.8.8.8).

Check DNS file configuration /etc/resolv.conf

like:

nameserver 202.102.192.98

nameserver 202.102.192.69

6.2 Speed Dial

6.2.1 Configuring APN

Set APN via AT+CGDCONT. For details, please refer to the AT manual.

For example: AT+CGDCONT=1,"IP","cmnet"

```
at+cgdcont=1,"ip","cmnet"
OK
at+cgdcont?
+CGDCONT: 1,"IP","cmnet","0.0.0.0",0,0,0,0
+CGDCONT: 2,"IPV4V6","IMS","0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0",0,0,0,0
+CGDCONT: 3,"IPV4V6","cmnet","0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0",0,0,0,0
+CGDCONT: 4,"IPV4V6","CMWAP","0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0",0,0,0,0
+CGDCONT: 5,"IPV4V6","","0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0",0,0,0,1

OK
```

Figure 3: AT+CGDCONT=1,"IP","cmnet"

6.2.2 Dial

The dialing commands are as follows. Please refer to the AT manual for details.

Method 1:

AT^NDISDUP=1,1,”apn”,”user”,”password”,0

Method 2:

AT$QCRMCALL=<Action>,<Instance> [,<IP Type> [,<Tech Pref > [,<umts profile number> [,<cdma profile number > [,<APN>]]]]]

AT^NDISDUP Command

For example:

AT^NDISDUP=1,1,"3gnet","","",0 China Unicom card

AT^NDISDUP=1,1,"ctnet","card","card",0 Telecom card

AT^NDISDUP=1,1,"cmnet","","",0 Mobile card

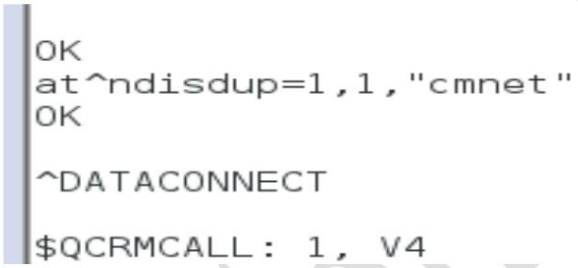

Figure 4: AT^NDISDUP command

AT$QCRMCALL command

At$QCRMCALL=1,1,1,1,,1 Telecom 3G

At$QCRMCALL=1,1,1,2,,1 China Mobile, China Unicom, China Telecom 4G

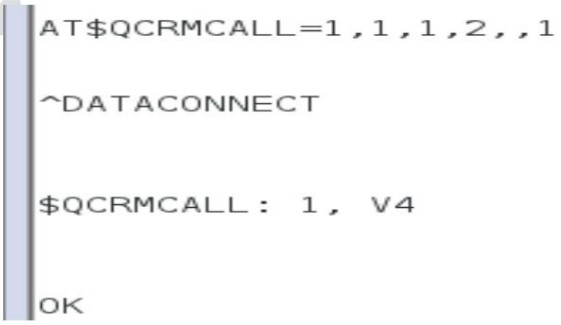

Figure 5: AT$QCRMCALL command

6.2.3 Disconnection

There are two AT commands for disconnection. Please refer to the AT manual for details.

Method 1:

AT^NDISDUP=1,0

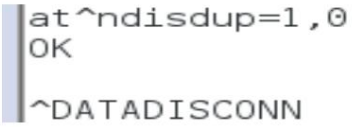

Figure 6: AT^NDISDUP=1,0

Method 2:

AT$QCRMCALL=0,1 ÿ

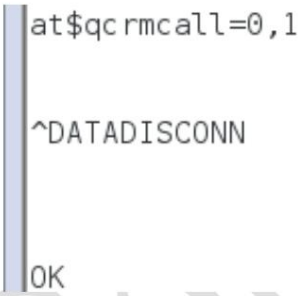

Figure 7: AT$QCRMCALL=0,1

Use AT+ AT^NDISSTATQRY? to query the data link status. See the AT manual for details.

^NDISSTATQRY:Conversation indicates that the data connection has been established

^NDISSTATQRY:Idle means not connected

Here, the status of successful dialing is:

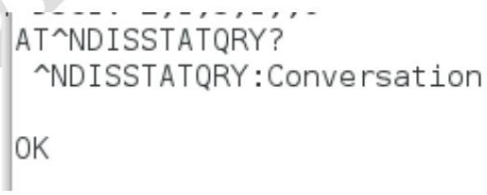

Figure 8: Example of successful dialing status

The dial-up failure status is:

```
AT^NDISSTATQRY?
 ^NDISSTATQRY:Idle

OK
```

Figure 9: Example of dial failure status

7.ECM dialing

7.1 Switch to ECM port

Use AT command to switch to ECM mode: AT+SER=2,1, and then restart the module.

After powering on, make sure the module is in ECM mode.

TO+SEE?

Make sure the return value is: +ser:2

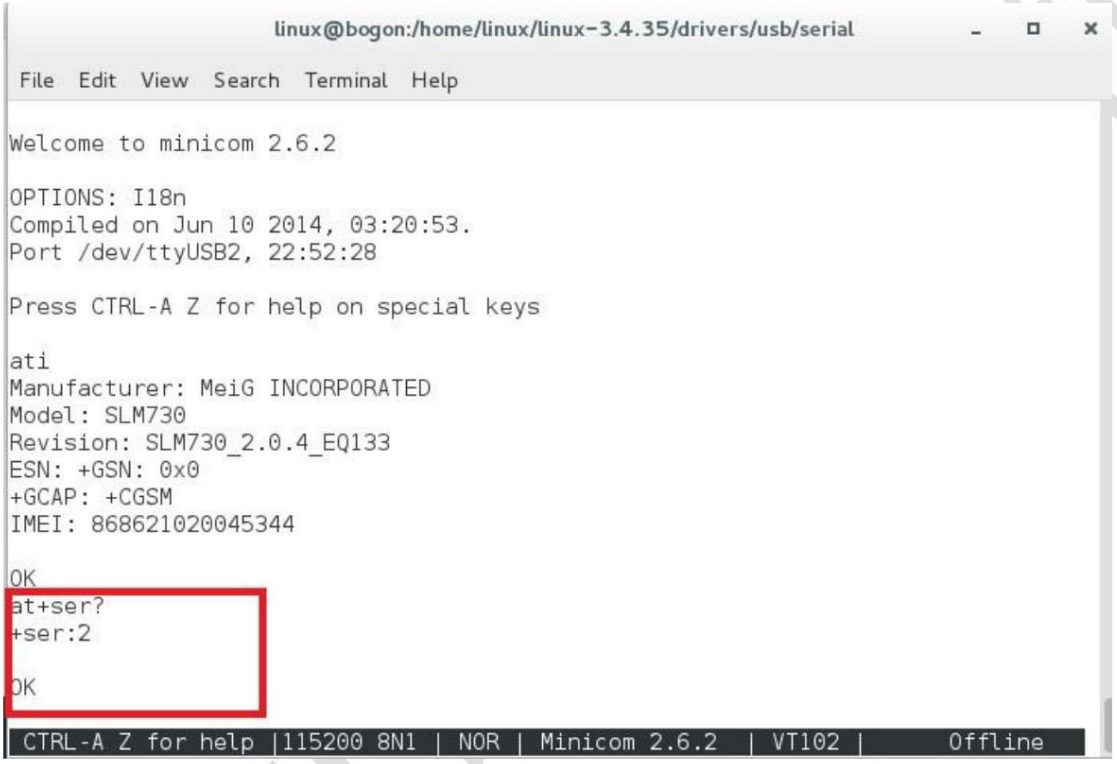

Figure 10: AT+SER? return value confirmation

Meige firmware version V15 and above:

Use AT+SER=14,1 to switch to ECM mode.

7.2 ECM Driver Installation

7.2.1 Check whether the driver code supports ECM mode

ECM is a standard NIC port. If you use the ECM device, you must load the cdc_ether driver in the system.

Therefore, the following items in the products[] array must not be deleted in the system:

```
{ USB_INTERFACE_INFO(USB_CLASS_COMM, USB_CDC_SUBCLASS_ETHERNET, USB_CDC_PROTO_NONE), .driver_info
        = (unsigned long) &cdc_info,
}
```

7.2.2 Installing the cdc_ether driver module

Linux distributions usually come with the cdc_ether.ko driver.

The following is how to load the driver in PC linux3.4.35 OS (the method is similar in embedded Linux, or the kernel is compiled and loaded by default):

insmod cdc_ether.ko //Load CDC_ether driver;

lsusb -t //Check whether the driver is loaded successfully.

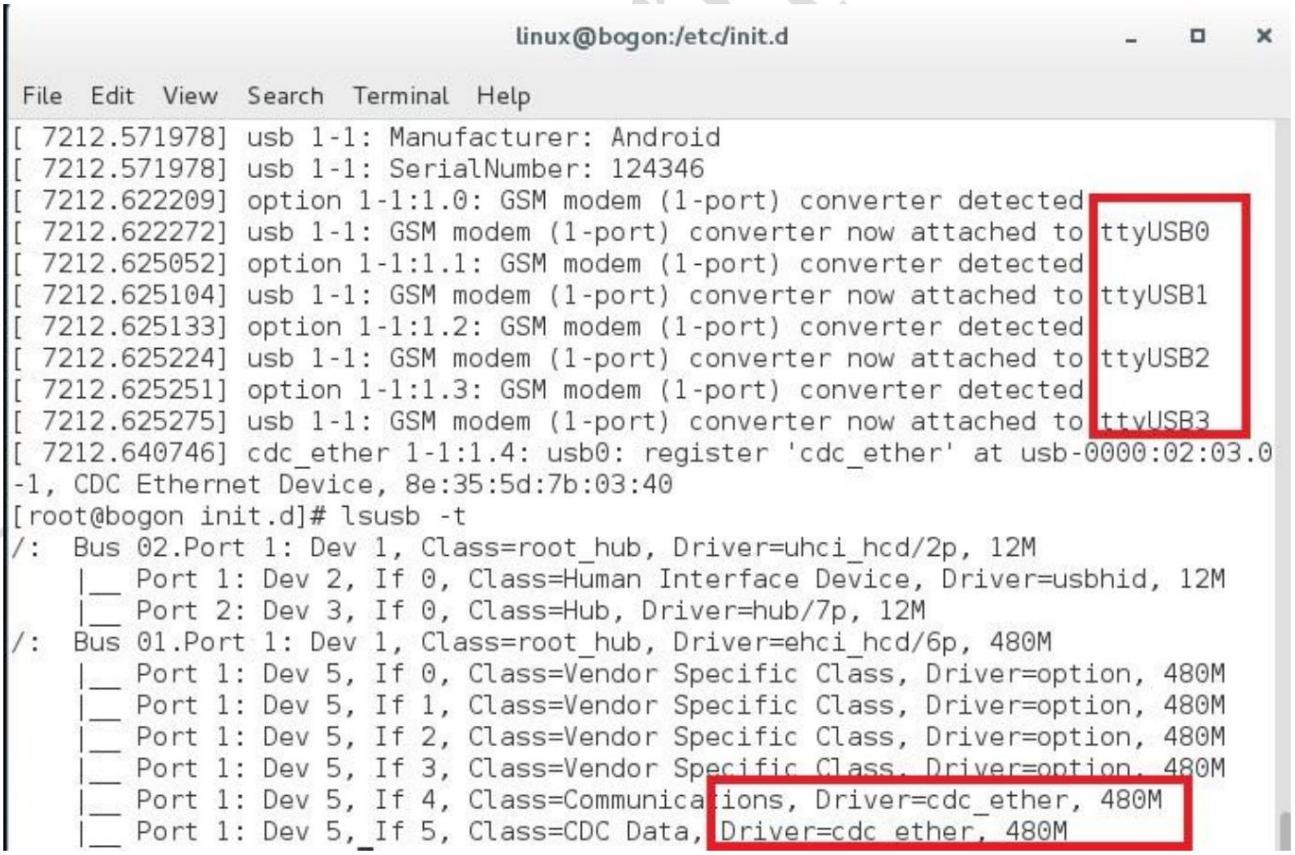

Figure 11: Check if the driver is loaded successfully

ifconfig -a //Check whether the network card node is generated.

7.3 ECM dial test

7.3.1 Setting dial-up APN

Use AT+CGDCONT to set the dial-up APN. See the AT manual for details.

```
at+cgdcont=1,"IP","CMNET"
OK
AT+CGDCONT?
+CGDCONT: 1,"IP","CMNET","0.0.0.0",0,0,0,0
+CGDCONT: 2,"IPV4V6","3GWAP","0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0",0,0,0,0
+CGDCONT: 3,"IPV4V6","WONET","0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0",0,0,0,0
```

Figure 12: Use AT+CGDCONT to set the dial-up APN

7.3.2 Using ECM dialing

Use AT+ECMDUP command to perform ECM dialing. For details, refer to AT manual.

```
AT+ECMDUP=1,1,1
^DATACONNECT

OK

^DSCI: 2,1,3,1,,0
```

Figure 13: AT+ECMDUP command

7.3.3 Using DHCP Command to Obtain IP and DNS

Use dhclinet ens35u1i4 (the name of the local network card) to obtain the IP address.

For embedded systems, you can also use udhcpc + ens35u1i4 to obtain an IP address.

```
[root@bogon init.d]# dhclient ens35u1i4 -v
Internet Systems Consortium DHCP Client 4.2.5
Copyright 2004-2013 Internet Systems Consortium.
All rights reserved.
For info, please visit https://www.isc.org/software/dhcp/

Listening on LPF/ens35u1i4/8e:35:5d:7b:03:40
Sending on   LPF/ens35u1i4/8e:35:5d:7b:03:40
Sending on   Socket/fallback
DHCPDISCOVER on ens35u1i4 to 255.255.255.255 port 67 interval 5 (xid=0xe0ff926)
DHCPREQUEST on ens35u1i4 to 255.255.255.255 port 67 (xid=0xe0ff926)
DHCPOFFER from 192.168.225.1
DHCPACK from 192.168.225.1 (xid=0xe0ff926)
bound to 192.168.225.33 -- renewal in 19855 seconds.
```

Figure 14: Use dhclinet ens35u1i4 (name of the local network card) to obtain the IP address

7.3.4 Use "ifconfig" to check network status

Use ifconfig to check if the network IP is obtained. ECM will obtain an IP in the 192.168.225.x domain.

```
[root@bogon init.d]# ifconfig
ens35u1i4: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.225.33  netmask 255.255.255.0  broadcast 192.168.225.255
        inet6 fe80::8c35:5dff:fe7b:340  prefixlen 64  scopeid 0x20<link>
        ether 8e:35:5d:7b:03:40  txqueuelen 1000  (Ethernet)
        RX packets 58  bytes 4557 (4.4 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 33  bytes 3722 (3.6 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 16436
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 0  (Local Loopback)
        RX packets 16  bytes 1360 (1.3 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 16  bytes 1360 (1.3 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Figure 15: Use ifconfig to check whether the network IP is obtained

7.3.5 Disconnecting the ECM network

Use AT+ECMDUP=1,0,1 to disconnect from the network.

```
AT+ECMDUP=1,0,1
^DATADISCONN

OK
```

Figure 16: Disconnecting from the network

7.4 Other issues

Using the ECM dial-up method is manual network management and conflicts with the NetworkManager that comes with the Ubuntu/Centos system.

NetworkManager and ECM dial-up are two different network management methods, and you can only choose one.

Otherwise it will cause ttyUSB0 ttyUSB1 ttyUSB2 ttyUSB3 to restart continuously.

When testing ECM on Centos, please use administrator privileges to disable NetworkManager. For example:

/etc/init.d/NetwrokManager stop

To test ECM on Ubuntu, use administrator privileges and execute the following command:

stop network-manager