



Welink your smart

DRIVER

Wireless Module Driver User Guide for Embedded Linux

Version: V2.4

Date: 2018-05-02

Driver Series



Website: www.gosuncnwelink.com

E-mail: welink@gosuncn.com

REVISION HISTORY

Version	Date	Description
V1.0	2012-07-18	Initial version of document.
V1.1	2013-03-11	Increase reversion, Legal Information. Update version, modify some mistake
V1.2	2013-04-02	Modify the logo of cover and page footer Modify Legal Information Increase the revision history
V1.3	2013-05-28	Add the support for ZM8510 modem.
V1.4	2013-11-08	Add the support for ZM8620 modem Add the contact information
V1.5	2013-12-09	Fix some information for LTE modem.
V1.6	2014-03-07	<ol style="list-style-type: none"> 1. Add the Applicability Table 2. Update the format of this document 3. Update the Legal Information 4. Add the Contact Information 5. Add the detail information in Chapter 2.1 of Enable drivers and PPP feature from kernel 6. Update the content in Chapter 2 of Driver Integration 7. Add figures in Chapter 4 of Setup data call via ECM 8. Add Chapter 6 of Appendix: Ports of GOSUNCN modems
V1.7	2015-08-18	Update the Applicability Table
V1.8	2015-08-26	Add the PID of all the modules in Chapter 6
V1.9	2016-01-06	In Chapter 6: <ul style="list-style-type: none"> • Modify the PID information of ME3620, ZM5202, ZM8620. • Add the PID information of MC2716, MC8332, ME3610 and MC8618 • Add some notices.
V2.0	2016-07-06	Add PID of ME3630
V2.1	2016-07-29	Modify the PID information of ME3630
V2.2	2016-08-31	Add chapter 5 of USB power management Guide Modify the PID information of ME3630 Update the information in chapter 2.2 about how to add PID/VID to module.
V2.3	2017-12-22	Update the port of MW3650,ME3610_V2,MC5635,ME3612 Update the document format
V2.4	2018-05-02	Update the document format

ABOUT THIS DOCUMENT

A. Application Range

Product Model
ZM5202
ZM5202E
ZM2210
ZM8620
ZM5330
MF226
MF206A
MG3732_V2
ME3620
MW3650
MC8635
ME3630
ME3610_V2
MC5635
ME3612

B. Reading Note

The symbols below are the reading notes you should pay attention on:



: Warning or Attention









: Note or Remark



NOTE: Note: Consult our website for up-to-date product descriptions, documentation, application notes, firmware upgrades, troubleshooting tips, and press releases

SAFETY INFORMATION

The following safety precautions must be observed during all phases of the operation, such as usage, service or repair of any cellular terminal or mobile incorporating GOSUNCN module. Manufacturers of the cellular terminal should send the following safety information to users and operating personnel and to incorporate these guidelines into all manuals supplied with the product. If not so, GOSUNCN does not take on any liability for customer failure to comply with these precautions.

	Full attention must be given to driving at all times in order to reduce the risk of an accident. Using a mobile while driving (even with a hands free kit) cause distraction and can lead to an accident. You must comply with laws and regulations restricting the use of wireless devices while driving.
	Switch off the cellular terminal or mobile before boarding an aircraft. Make sure it switched off. The operation of wireless appliances in an aircraft is forbidden to prevent interference with communication systems. Consult the airline staff about the use of wireless devices on boarding the aircraft, if your device offers a Airplane Mode which must be enabled prior to boarding an aircraft.
	Switch off your wireless device when in hospitals or clinics or other health care facilities. These requests are designed to prevent possible interference with sensitive medical equipment.
	GSM cellular terminals or mobiles operate over radio frequency signal and cellular network and cannot be guaranteed to connect in all conditions, for example no mobile fee or an invalid SIM card. While you are in this condition and need emergent help, please remember using emergency call. In order to make or receive call, the cellular terminal or mobile must be switched on and in a service area with adequate cellular signal strength.
	Your cellular terminal or mobile contains a transmitter and receiver. When it is on, it receives and transmits radio frequency energy. RF interference can occur if it is used close to TV set, radio, computer or other electric equipment.
	In locations with potentially explosive atmospheres, obey all posted signs to turn off wireless devices such as your phone or other cellular terminals. Areas with potentially explosive atmospheres including fuelling areas, below decks on boats, fuel or chemical transfer or storage facilities, areas where the air contains chemicals or particles such as grain, dust or metal powders.

CONTENTS

REVISION HISTORY.....	II
ABOUT THIS DOCUMENT.....	III
SAFETY INFORMATION	IV
CONTENTS.....	V
1 INTRODUCTION	6
1.1 Purpose	6
1.2 Scope	6
1.3 Conventions.....	6
2 DRIVER INTEGRATION	7
2.1 Enable drivers and PPP feature from kernel.....	7
2.2 Add VID/PID to USB-modem driver.....	7
2.3 Build the kernel	9
2.4 Verify whether modem driver is loaded.....	10
3 MAKE DATA CONNECTION USING PPPD AND CHAT	11
3.1 Sample script for data connect using WCDMA modem.....	11
3.2 Sample script for data connect using CDMA modem	12
3.3 Start to dial and connect	14
3.4 Disconnect	14
4 SETUP DATA CALL VIA ECM.....	15
5 USB POWER MANAGEMENT GUIDE	17
5.1 Basic Conception of Power Management	17
5.2 Linux Driver Bind	17
5.2.1 THE CONFIGURATION OF LINUX KERNEL DRIVER.....	17
5.2.2 DRIVER BINDING	17
5.3 Settings about Power Management	18
5.3.1 ENABLE SELECTIVE SUSPEND AND REMOTE WAKEUP.....	18
5.3.2 SYSTEM WAKEUP SUSPEND.....	19
5.3.3 SUSPEND DELAY SETUP	19
6 TROUBLESHOOTING.....	20
6.1 Dropped connection.....	20
6.2 DNS issues	20
6.3 How to get log	20
7 APPENDIX: PID&PORTS OF GOSUNCN MODEMS	21

1 INTRODUCTION

1.1 PURPOSE

This document is intended as a guide to describe how to set up and connect to the Internet using GOSUNCN Wireless Modem on the EmbeddedLinux platform using open source software packages. Although this document is provided to guide current Linux users to use GOSUNCN Wireless Modems on any EmbeddedLinux distribution, however, the setup instructions in this document has only been tested with uCLinux. For setup on other Linux distributions, you should still be able to follow the instructions after you download and install the necessary software packages outlined below.

1.2 SCOPE

The readers of this document are assumed to software developers generally familiar with EmbeddedLinux development.

1.3 CONVENTIONS

Function declarations, function names, type declarations, and code samples appear in a different font. For example: *#include*

Red font means the information should be notes, if ignore this may cause some failures. For example: */dev/ttyUSB*

2 DRIVER INTEGRATION

2.1 ENABLE DRIVERS AND PPP FEATURE FROM KERNEL

USB-modem drivers and PPP features may be not enabled in the kernel, so please go to kernel, do “*make menuconfig*” and include the following components into building:

1. Device Drivers --->
 - USB support --->
 - USB Serial Converter support --->
 - USB Generic Serial Driver
 - USB driver for GSM and CDMA modems
2. Device Drivers --->
 - Network device support --->
 - PPP (point-to-point protocol) support
 - PPP multilink support
 - PPP filtering
 - PPP support for async serial ports
 - PPP support for sync tty ports

For modems with ECM or NDIS interface, such as ZM5330/ZM8620/ME3620/ME3630, including the following component additionally

3. Device Drivers --->
 - Network device support --->
 - USB Network Adapters --->
 - Multi-purpose USB Networking Framework

After saving the configuration, check the “*.config*” file, the following lines must exist, if any line doesn’t exist, it means your configuration above is not correct.

```
CONFIG_USB_SERIAL=y
CONFIG_USB_SERIAL_GENERIC=y
CONFIG_USB_SERIAL_OPTION=y
CONFIG_PPP=y
CONFIG_PPP_MULTILINK=y
CONFIG_PPP_FILTER=y
CONFIG_PPP_ASYNC=y
CONFIG_PPP_SYNC_TTY=y
CONFIG_USB_NET_CDCETHER=y (For modem with ECM/NDIS interface)
```

2.2 Add VID/PID TO USB-MODEM DRIVER

USB network adapter driver can automatically identify the GOSUNCN module, therefore, users don't need to add any content in *cdc_ether.c* file. But USB serial port driver cannot automatically identify, users need to add GOSUNCN module device information to the driver file *option.c* which locates in “*\linux-source-2.6.XX\drivers\usb\serial*”. Add modem PID and VID (code in red) in driver *id_table*:

```
static struct usb_device_id option_ids[] = {
    { USB_DEVICE(0x19d2, 0x0117),
```

```
{ USB_DEVICE(0x19d2, 0x0199) },
{ USB_DEVICE(0x19d2, 0x1476) },
```

```
static struct usb_device_id option_ids[] = {
    { USB_DEVICE(ZTE_VENDOR_ID, 0x0117) },
    { USB_DEVICE(OPTION_VENDOR_ID, OPTION_PRODUCT_COLT) },
    { USB_DEVICE(OPTION_VENDOR_ID, OPTION_PRODUCT_RICOLA) },
};
```

Figure 2-1

“0x19d2” is the VID of GOSUNCN corporation, it is appropriate for all GOSUNCN wireless modems. “0x0117” is the PID of the modem, it may vary with the modem you are using, and you can use Linux command “lsusb” to get it. You can also get the PID of module in chapter 7 Appendix. Figure 2-2 shows an example on a virtual machine.

```
> lsusb
Bus 002 Device 002: ID 0e0f:0002 VMware, Inc. Virtual USB Hub
Bus 002 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub
Bus 001 Device 035: ID 19d2:0117 ONDA Communication S.p.A.
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
```

Figure 2-2

As the PID/VID is add to the linux kernel and there is not any additional port information, the USB network port(ECM) is loaded as USB serial port. For modems which have ECM interface, such as ZM5330/ZM8620/ME3620/ME3630, users need to modify the function of option_probe to add the ports corresponding to ECM to blacklist. Only in this way, we can prevent the USB network card is loaded as USB serial port.

For modems which have ECM interface, such as ME3630/ME3620(common firmware with PID 0x1476), the ports corresponding to ECM are port 3&4, ME3630 tower firmware with PID 0x1509,you need to add the following code to function option_probe

```
printk("idVendor=%x, idProduct=%x, bInterfaceNumber =%d\r\n",
    serial->dev->descriptor.idVendor,
    serial->dev->descriptor.idProduct,
    serial->interface->cur_altsetting->desc. bInterfaceNumber);

if (serial->dev->descriptor.idVendor == 0x19d2 &&
    serial->dev->descriptor.idProduct == 0x1476 &&
    serial->interface->cur_altsetting->desc. bInterfaceNumber == 3)
    return -ENODEV;

if (serial->dev->descriptor.idVendor == 0x19d2 &&
    serial->dev->descriptor.idProduct == 0x1476 &&
    serial->interface->cur_altsetting->desc. bInterfaceNumber == 4)
    return -ENODEV;

if (serial->dev->descriptor.idVendor == 0x19d2 &&
    serial->dev->descriptor.idProduct == 0x1509 &&
    serial->interface->cur_altsetting->desc. bInterfaceNumber == 4)
    return -ENODEV;

if (serial->dev->descriptor.idVendor == 0x19d2 &&
    serial->dev->descriptor.idProduct == 0x1509 &&
    serial->interface->cur_altsetting->desc. bInterfaceNumber == 5)
    return -ENODEV;
```




"0x1509/0x1476" is the PID of the modem, it may vary with the modem you are using, you can use Linux command "lsusb" to get it.

The first line of "printk" is used for debugging. "if" conditional judgment is used to add the ECM ports to blacklist, you need to modify the idProduct and bInterfaceNumber parameters according to the module you used. For the detail information of idProduct and bInterfaceNumber, you can refer to chapter 7 Appendix.

2.3 BUILD THE KERNEL

Build kernel, and flash this kernel file to target board, reboot target board.

2.4 VERIFY WHETHER MODEM DRIVER IS LOADED

After reboot target system, to make sure the driver had been installed successfully, just attach the modem to the target board and run the following on the command-line:

`dmesg | grep -i ttyUSB`

You should see something like:


`usb 5-1: generic converter now attached to ttyUSB0`

`usb 5-1: generic converter now attached to ttyUSB1`

`usb 5-1: generic converter now attached to ttyUSB2`

This LOG indicates the USB serial driver has been loaded. In general, each GOSUNCN modem has multi interface. For different GOSUNCN modem, the device node of modem port and AT port may different. Chapter 6 below shows the port information of some GOSUNCN modems. Please contact customer manager of GOSUNCN to get more details.

For modems which have ECM interface, such as ZM5330/ZM8620/ME3620/ME3630, **if the modem is in ECM mode**, run “`ifconfig -a`” command in command line, you should see a new network card appeared which named “`usbx/ecmx/ethx`”, “`x`” can be any number. This indicates the ECM interface driver was loaded successfully. Figure below shows an example.

 **Note:** For how to switch modem to ECM mode, refer to Chapter 4.

```
usb0      Link encap:Ethernet  HWaddr 6a:74:c5:cc:38:a0
          inet6 addr: fe80::6874:c5ff:fecc:38a0/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:8 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:1152 (1.1 KB)
```

Figure 2-3

3 MAKE DATA CONNECTION USING PPPD AND CHAT

This chapter gives an example for how to setup a data connection using pppd and chat program. You should have integrated the driver following chapter 2, now your device can be recognized by the system. This chapter also assumes you have installed pppd and chat on your current Linux distribution.



Note: For LTE modems such as ZM5330/ZM8620/ME3620/ME3630, it also can use pppd and chat to setup a data connection, but the speed may be limited, we recommend to uses ECM to setup data connection on ZM5330/ZM8620/ME3620 modem, reference to chapter 4.

3.1 SAMPLE SCRIPT FOR DATA CONNECT USING WCDMA MODEM

Four files need to be created; their name and content are shown below. All files must be saved in the same path.

File 1: ppp-on

```
#!/bin/sh

clear

OPTION_FILE="gosuncn_options"
DIALER_SCRIPT=$(pwd)/gosuncn_ppp_dialer
exec /usr/sbin/pppd file $OPTION_FILE connect "/usr/sbin/chat -v -f ${DIALER_SCRIPT}"
```


This file is the main script to dial up.

File 2: gosuncn_options

```
/dev/ttyUSB2
115200
crtscts
modem
persist
lock
noauth
noipdefault
debug
nodetach
user Anyname
password Anypassword
ipcp-accept-local
ipcp-accept-remote
replacedefaultroute
defaultroute
usepeerdns
noccp
nobsdcomp
novj
dump
```




Note 1: /dev/ttyUSB2 in the first line is the modem interface of your device, please refer Chapter 6 to use proper modem interface.

 **Note 2:** User and password may be different due to mobile operator. If operator do not " provide username and password, please use *Anyname* and *Anypassword*.

File 3: gosuncn_ppp_dialer

```
ABORT          "NO CARRIER"
ABORT          "ERROR"
TIMEOUT        120
""            ATE
SAY            "ATE"
ECHO           ON
OK            ATH
OK            ATP
OK            AT+CGDCONT=1, \\"IP\\",\\"3GNET\\"
OK            ATD*99#
CONNECT
```

 **Note 1:** The number *"*99#"* may vary with your mobile operator, *"*99#"* is valid if your mobile operator is CHINA UNICOM.

 **Note 2:** The string *"3GNET"* is the APN, it may vary with your mobile operator, *"3GNET"* is valid if your mobile operator is CHINA UNICOM.

File4: disconnect

```
#!/bin/sh
killall pppd
```

3.2 SAMPLE SCRIPT FOR DATA CONNECT USING CDMA MODEM

Four files need to be created; their name and content are shown below. All files must be saved in the same path.

File 1: ppp-on

```
#!/bin/sh
clear
OPTION_FILE="gosuncn_options"
DIALER_SCRIPT=$(pwd)/gosuncn_ppp_dialer
exec /usr/sbin/pppd file $OPTION_FILE connect "/usr/sbin/chat -v -f ${DIALER_SCRIPT}"
```

This file is the main script to dial up.

File 2: gosuncn_options

```
/dev/ttyUSB0

115200

crtcts


modem


persist

lock

noauth
```

```
noipdefault
debug
nodetach
user card
password card
ipcp-accept-local
ipcp-accept-remote
replacedefaultroute
defaultroute
usepeerdns
noccp
nobsdcomp
novj
dump
```


 **Note 1:** `/dev/ttyUSB0` in the first line is the modem interface of your device, please refer Chapter 6 to use proper modem interface.

 **Note 2:** User and password may be different due to mobile operator. If an operator do not provide username and password, please use Anyname and Anypassword. The username and password “card” is valid in CHINA TELECOM.

File 3: gosuncn_ppp_dialer

```
ABORT          "NO CARRIER"
ABORT          "ERROR"
TIMEOUT        120
""            ATE
SAY            "ATE"
ECHO           ON
OK            ATH
OK            ATP
OK            "AT+ZCAPN=card,card"
OK            ATD#777
CONNECT
```

 **Note 1:** The number “#777” may vary with your mobile operator, “#777” is valid if your mobile operator is CHINA TELECOM.

 **Note 2:** The command “AT+ZCAPN=card,card” sets the username and password, it may be different due to mobile operator. For example, if a mobile operator’s username is “user_a” and password is “1234”, then the command should be “AT+ZCAPN=user_a, 1234”. If an mobile operator do not provide username and password, please do not add this command.

File4: disconnect

```
#!/bin/sh
killall pppd
```

3.3 START TO DIAL AND CONNECT

Open a terminal window and run the main script “ppp-on” to start to dial and connect

3.4 DISCONNECT

1. Enter “ctrl+c” at the terminal window;
2. Run the “disconnect” script.

4 SETUP DATA CALL VIA ECM

ECM interface can be used to setup data call on ZM5330/ZM8620/ME3620/ME3630 modem. The data connection can be setup by following steps:

Note: In the following example, we assume the name of the new network card is “usb0”, refer to chapter 2.4 for how to get the name of the new network card after the driver had been installed.

Step0: Switch modem to ECM mode using AT command: `AT+ZSWITCH=L`, then reboot the modem.

Note: If your modem has already been set to ECM mode, then this step is not needed

```
AT+ZSWITCH=L
file write success, fs_result: 1
OK
```

Figure 4-1

Step1: Setup data call parameter using AT command +CGDCONT. For example, can configure APN “CMNET” using command: `AT+CGDCONT=1,"IP","CMNET"`

```
AT+CGDCONT=1,"IP","CMNET"
OK
```

Figure 4-2

Step2: Setup ECM data call using AT command: `AT+ZECMCALL=1`

```
AT+ZECMCALL=1
+ZECMCALL: CONNECT
OK
```

Figure 4-3

Step3: Start DHCP to get IP and DNS. Enter “`sudo dhcpcd usb0`” in the command-line window:

```
> sudo dhcpcd usb0
dhcpcd.sh: interface usb0 has been configured with new IP=1.1.5.1
```

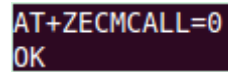
Figure 4-4

Step4: Using “`ifconfig usb0`” check whether network card get IP address. As shown in Figure 4-5, in this example, the IP address has been assigned, so if there were no other errors, users were able to access the internet now.

```
usb0      Link encap:Ethernet  HWaddr d6:5b:a0:6a:85:0a
          inet addr:1.1.5.1  Bcast:1.1.5.3  Mask:255.255.255.252
          inet6 addr: fe80::d45b:a0ff:fe6a:850a/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:6 errors:0 dropped:0 overruns:0 frame:0
          TX packets:34 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:892 (892.0 B)  TX bytes:6472 (6.4 KB)
```

Figure 4-5

Step5: To disconnect the ECM data call, using AT command: **AT+ZECMCALL=0**



```
AT+ZECMCALL=0
OK
```

Figure 4-6

Please refer to AT command document for further more information.

5 USB POWER MANAGEMENT GUIDE

5.1 BASIC CONCEPTION OF POWER MANAGEMENT

Power management is an energy-saving method to reduce the power supply amount by suspending all devices which in idle state. All suspended devices are in low power consumption, and they can be activated again by resuming if necessary.

System suspend: The state when all devices in system (including CPU and others) are all in idle situation.

Selective suspend (dynamic suspend): The state when only one particular device in system is in idle state, but the rest of the others are still in operating.

Normally speaking, device is resumed by computer system. Remote wakeup is the function of a device which can notify system to activate itself during it is suspending.

The module products of GOSUNCN can shift into low power consumption if they are suspended or in system standby and they also can be awakened by some external events, such as receiving SMS, call, network data or remote wakeup function.

5.2 LINUX DRIVER BIND

The process of loading suitable Linux driver for modules is called bind. Especially speaking, this process is needed for supporting selective suspend and remote wakeup functions.

5.2.1 THE CONFIGURATION OF LINUX KERNEL DRIVER

First of all, Linux kernel must available for power management. Please make sure the following config options are valid:

1. CONFIG_SUSPEND or CONFIG_HIBERNATION
2. CONFIG_PM
3. CONFIG_PM_RUNTIME
4. CONFIG_USB_SUSPEND (Linux 3.10 or above is not included)

Secondly, USB driver should be loaded to Linux system (option driver are recommended). Please make sure the following config options are valid:

1. CONFIG_USB_SERIAL
2. CONFIG_USB_SERIAL_WWAN
3. CONFIG_USB_SERIAL_OPTION

If ECM function is required, USB ECM driver is an extra demand. Please make sure the following config options are valid:

1. CONFIG_USB_USBNET
2. CONFIG_USB_NET_CDCETHER

5.2.2 DRIVER BINDING

Linux kernel will bind suitable driver for USB port under normal conditions. Bind state can be checked out by lsusb command:

```
# lsusb -t
/: Bus 05.Port 1: Dev 1, Class=root_hub, Driver=uhci_hcd/2p, 12M
/: Bus 04.Port 1: Dev 1, Class=root_hub, Driver=uhci_hcd/2p, 12M
/: Bus 03.Port 1: Dev 1, Class=root_hub, Driver=uhci_hcd/2p, 12M
/: Bus 02.Port 1: Dev 1, Class=root_hub, Driver=uhci_hcd/2p, 12M
/: Bus 01.Port 1: Dev 1, Class=root_hub, Driver=ehci-pci/8p, 480M
   |__ Port 2: Dev 6, If 0, Class=Vendor Specific Class, Driver=option, 480M
   |__ Port 2: Dev 6, If 1, Class=Vendor Specific Class, Driver=option, 480M
   |__ Port 2: Dev 6, If 2, Class=Vendor Specific Class, Driver=option, 480M
   |__ Port 2: Dev 6, If 3, Class=Communications, Driver=cdc_ether, 480M
   |__ Port 2: Dev 6, If 4, Class=CDC Data, Driver=cdc_ether, 480M
```

Figure 1

From the figure above, you can see the 3 USB ports are bound with option driver, ecm network card is bound with cdc_ether driver. Because of the variety of kernel versions, some Linux kernel may not bind the option driver for USB port correctly. For this situation, driver can be bound in a manual way.

```
# lsusb -t
/: Bus 05.Port 1: Dev 1, Class=root_hub, Driver=uhci_hcd/2p, 12M
/: Bus 04.Port 1: Dev 1, Class=root_hub, Driver=uhci_hcd/2p, 12M
/: Bus 03.Port 1: Dev 1, Class=root_hub, Driver=uhci_hcd/2p, 12M
/: Bus 02.Port 1: Dev 1, Class=root_hub, Driver=uhci_hcd/2p, 12M
/: Bus 01.Port 1: Dev 1, Class=root_hub, Driver=ehci-pci/8p, 480M
   |__ Port 2: Dev 6, If 0, Class=Vendor Specific Class, Driver=, 480M
   |__ Port 2: Dev 6, If 1, Class=Vendor Specific Class, Driver=, 480M
   |__ Port 2: Dev 6, If 2, Class=Vendor Specific Class, Driver=, 480M
   |__ Port 2: Dev 6, If 3, Class=Communications, Driver=cdc_ether, 480M
   |__ Port 2: Dev 6, If 4, Class=CDC Data, Driver=cdc_ether, 480M
```

Figure 2

The figure above shows that no driver for USB port, which means driver bind must be done manually. The precondition is that right option driver module is existed for Linux driver, if not, it can be loaded by modprobe or insmod command.

Input the command below to bind driver after correct option module loaded:

```
echo 0x19d2 0x1476 > /sys/bus/usb-serial/drivers/option1/new_id
```

See the catalog of /sys/bus/usb-serial/drivers/option1, if ttyUSB0, ttyUSB1 and ttyUSB2 are generated successfully, it means the success of USB driver bind.

5.3 SETTINGS ABOUT POWER MANAGEMENT

Under the circumstance of successful driver bind, the power management function is defaulted as close and it can be ON by some settings.

5.3.1 ENABLE SELECTIVE SUSPEND AND REMOTE WAKEUP

The first step is figure out the sysfs path of module USB device. The path format is like the below:

```
/sys/bus/usb/devices/Bus-Port/
```

The bus and port can be identified by lsusb command. In figure1, the module is at Bus 1 and Port 2, therefore the device sysfs path is:

```
/sys/bus/usb/devices/1-2/
```

Input the following command to enable selective suspend and remote wakeup function:

```
echo auto >/sys/bus/usb/devices/1-2/power/control
```

```
echo auto >/sys/bus/usb/devices/1-2/power/level
```

Input the following command to make sure function is available:

```
cat /sys/bus/usb/devices/1-2/power/control
```

```
cat /sys/bus/usb/devices/1-2/power/level
```

If the return is "auto" for both commands, selective suspend function and remote wakeup function are ON now.

5.3.2 SYSTEM WAKEUP SUSPEND

When system is suspended, module can wake it up to operation. The command below can make this function activated:

```
echo enabled >/sys/bus/usb/devices/1-2/power/wakeup
```

The function below can test the state of this function:

```
cat /sys/bus/usb/devices/1-2/power/wakeup
```

If the return value is "enabled", which means the function is operating.

5.3.3 SUSPEND DELAY SETUP

Only after a period of idle time, USB driver will recognize it as suspended. This period of idle time is the delay time and its value can be set via 2 sysfs ports:

```
/sys/bus/usb/devices/1-2/power/autosuspend  
/sys/bus/usb/devices/1-2/power/autosuspend_delay_ms
```

The unit of autosuspend setup is second, the unit of autosuspend_delay_ms is millisecond. In factory value, the delay period is 2s, recommended value is 2-5s. The following is the setting command:

```
echo 2 >/sys/bus/usb/devices/1-2/power/autosuspend  
or  
echo 2000 > /sys/bus/usb/devices/1-2/power/autosuspend_delay_ms
```

6 TROUBLESHOOTING

6.1 DROPPED CONNECTION

In the event that the connection is lost due to any reason, a manual reconnect is required. You can configure your connection to automatically redial on disconnect.

6.2 DNS ISSUES

For DNS-related issues, please verify entries in `/etc/resolv.conf` file. Please refer to your Linux distribution's user manual for more information about fixing this problem.

6.3 HOW TO GET LOG

If some error occurred, please provide us the QXDM LOG. The QXDM LOG can be captured in Linux system by the tools provided by GOSUNCN named *"DiagLog"*, which can be found in directory *"dir/tools/DiagLog"*.

7 APPENDIX: PID&PORTS OF GOSUNCN MODEMS

After the driver has been installed, if a GOSUNCN modem is connected, several ports will appear in the operating system. Refer to the following table for more details.

Note: PID in the first column is the product ID of your modem.

Modules	PID(idProduct)	Interface number(<i>blInterfaceNumber</i>)	Port
MF206A& ZM5202E	0x0117	0	Diag
MG3732_V2/B &V2/C		1	NMEA
ME3610_V1		2	Modem
MC5635			
ZM5330	0x1432(NDIS) 0x1433(ECM) for Linux	0	Diag
		1	AT
		2	Modem
		3	NDIS(ECM)
		4	ADB
ZM5202	0x0144	0	Diag
		1	GPS
		2	NMEA
		3	NMEAext
	0x0117	4	Modem
		0	Diag
		1	NMEA
		2	Modem
ZM2210	0x0152	0	Modem
		1	Diag
		2	AT
		3	Vousb
		4	NMEA
ZM8620	0x0396	0	Diag
		1	AT
		2	Modem
		3	NDIS(ECM)
		4	ADB
MW3650	0xFFEB	0	Service/AT
MG3732_V2/A		1	VoUSB

Wireless Module Driver Guide for EmbeddedLinux

		2	Diag
		3	Modem
MC8635 MC8630	0xFFEE	0	Modem
		1	Service/AT
		2	Vousb
		3	Diag
MC2716	0xFFED	0	Modem
		1	Service/AT
		2	Vousb
		3	Diag
MC8332 MC8618	0xFFEA	0	Diag
		1	Modem
ME3620_C1A	0x1432(NDIS) 0x1433(ECM) 0x1476(ECM)	0	Diag
		1	AT
		2	Modem
		3	NDIS(ECM)
		4	ADB
MF226	0x0144(PRC, Japan)	0	Diag
		1	GPS
		2	AT
		3	ATExt
		4	Modem
	0x0117(South Korea)	0	Diag
		1	NMEA
		2	Modem
	0x2003(Malaysia)	0	Diag
		1	NMEA
		2	VoUSB
		3	Modem
ME3630 ME3610_V2 ME3612	0x1432(NDIS)	0	DIAG
		1	AT
		2	MODEM
		3	NDIS
		4	ADB
ME3630	0x1476(ECM)	0	DIAG

ME3610_V2 ME3612		1	AT
		2	MODEM
		3	ECM
		4	ECM
	0x1509(China Tower) Customization	0	DIAG
		1	AT
		2	MODEM
		3	ADB
		4	ECM
		5	ECM
	0x0016 (Download mode)	0	DIAG
		1	AT
		2	MODEM
	0x0601 (RNDIS)	0	RNDIS
		1	RNDIS
		2	DIAG
		3	AT
		4	MODEM
		5	ADB
	0x0602 (Win8)	0	DIAG
1		AT	
2		MODEM	
3		MBIM	
4		MBIM	
5		ADB	

Note:
ZM5202:

- PID 0x0144 is the one for common use.
- PID 0x0117 is aimed at South Korea.

ME3620_CIA:

- PID 0x1432 is compatible with NDIS port, but this format is only used in windows system.
- PID 0x1433 is specific to China Tower.
- PID 0x1476 is compatible with ECM port, this format is only used in Linux/Android system. it is the general availability under Linux system, or used for SGCC(State Grid Cooperation of China), Japan and South Korea.

ME3630:

- PID 0x1432 is compatible with NDIS port, but this format is only used in windows system.
- PID 0x1509 is specific to China Tower.

- PID 0x1476 is compatible with ECM port, this format is only used in Linux/Android system.

Please pay attention on these points above and choose the correct PID.

GOSUNCN Confidential
doing@doiot.net 47.104.50.27 2019/11/7 20:51:09