# SIMCOM_A7600_Linux_USB_User_Guide_V1.01.00

LTE Module

**Shanghai SIMCom Wireless Solutions Ltd.**
Building A, SIM Technology Building, No.633, Jinzhong Road
Changning District 200335
Tel:86-21-31575100/31575200
support@simcom.com
www.simcom.com

| | |
|---|---|
| **Document Title:** | A7600_Linux_USB_User_Guide |
| **Version:** | 1.01.00 |
| **Date:** | 2021-11-03 |
| **Status:** | Release |
| **Document ID:** | SIMCOM_A7600_Linux_USB_User_Guide _V1.01.00 |



**General Notes**

SIMCom offers this information as a service to its customers, to support application and engineering efforts that use the products designed by SIMCom. The information provided is based upon requirements specifically provided to SIMCom by the customers. SIMCom has not undertaken any independent search for additional relevant information, including any information that may be in the customer's possession. Furthermore, system validation of this product designed by SIMCom within a larger electronic system remains the responsibility of the customer or the customer's system integrator. All specifications supplied herein are subject to change.

**Copyright**

# Version History

| Version | Date | Chapter | What is new |
|---|---|---|---|
| V1.00.00 | 2019-06-06 | | New version |
| V1.01.00 | 2019-08-07 | 3.1、3.2 | 1. Add +DIALMODE and $MYCONFIG description.<br>2. Replace /DEV/ttyUSB3 with /DEV/ttyUSB2. |
| V1.02.00 | 2020-11-27 | 3.3 | 3. Add +USBNETIP |
| | 2021-08-10 | 2 | 4. Add NMEA port |
| | | | |
| | | | |

# Contents

# 1 Introduction

## 1.1. Scope

This user guide serves the following purpose:

- Short introductions how to customize the USB driver for Simcom A7600 module in Linux OS
- Describes how software developers can use Linux devices for typical use cases.

## 1.2. Related Documents

**1:** A7600 ATC.

# 2 Device Driver Installation

In order to recognize the modem, you must add VID and PID in Linux driver option, after the operating system recognizes the modem, devices named /dev/ttyUSBx are created, for example:

- /dev/ttyUSB0 diag port for output developing messages
- /dev/ttyUSB1 AT port for AT commands
- /dev/ttyUSB2 Modem port for ppp-dial
- /dev/ttyUSB3 NMEA port for GNSS(GNSS must be supported)

## 2.1. Add VID and PID

**1.Edit option.c source file**

a. gedit linux-x.x.x/drivers/usb/serial/option.c

b. Check if the following #define statement exists,If the #define does not exist ,add:

```
#define SIMCOM_VENDOR_ID        0x1e0e
#define SIMCOM_PRODUCT_PID_X9011          0x9011
```

c. Check the following struct statement exists,If the struct does not exist ,add the following struct

```
static const struct option_blacklist_info simcom_pid9011_blacklist = {
    .reserved = BIT(0) | BIT(1),
}
```

d. Add the following line to usb_device_id option_ids[] structure, if not exists

```
    { USB_DEVICE(SIMCOM_VENDOR_ID, SIMCOM_PRODUCT_PID_X9011),
      .driver_info = (kernel_ulong_t)&simcom_pid9011_blacklist }
```

If your kernel code is different from the above, please be careful to modify it as appropriate.

## 2.2. Building a Linux Driver Module

How to compile and install a kernel module in Linux , follow the steps below will guide you along in compiling and install your option driver On Ubuntu operating system.

**Step 1: Enter to kernel directory.**

```
cd <your kernel directory>
```

**Step 2: Build the driver.**

```
sudo make -C /lib/modules/`uname -r`/build M=`pwd`/drivers/usb/serial
obj-m=option.o modules
```

**Step 3: Load the driver and reboot.**

```
sudo cp drivers/usb/serial/option.ko /lib/modules/`uname
-r`/kernel/drivers/usb/serial
sudo depmod
sudo reboot
```

## 2.3. Kernel Compilation Configuration

### 2.3.1. Compilation Configuration for USB Serial Driver

| Configuration | Configuration(Y/N) |
|---|---|
| CONFIG_USB_SERIAL | Y |
| CONFIG_USB_SERIAL_OPTION | Y |

### 2.3.2. Compilation Configuration for RNDIS Driver

| Configuration | Configuration(Y/N) |
|---|---|
| CONFIG_USB_SERIAL | Y |
| CONFIG_USB_SERIAL_OPTION | Y |
| CONFIG_USB_USBNET | Y |
| CONFIG_USB_NET_CDCETHER | Y |

### 2.3.3. Compilation Configuration for PPP Driver

| Configuration | Configuration(Y/N) |
|---|---|
| CONFIG_USB_SERIAL | Y |
| CONFIG_USB_SERIAL_OPTION | Y |
| CONFIG_PPP | Y |
| CONFIG_PPP_FILTER | Y |
| CONFIG_PPP_MULTILINK | Y |
| CONFIG_PPP_BSDCOMP | Y |
| CONFIG_PPP_ASYNC | Y |
| CONFIG_PPP_SYNC_TTY | Y |
| CONFIG_PPP_DEFLATE | Y |

# 3 Modem Usage

This chapter mainly introduces several commonly used dialing methods and their general processes.

USB devices must be recognized before use modem.

## 3.1. Description of AT Command

### 3.1.1 AT+DAILMODE Enable/Disable Auto-Dial

| AT+DAILMODE Enable/Disable Auto-Dial | |
|---|---|
| Test Command<br>**AT+DIALMODE =?** | Response<br>**+DIALMODE: (0-1)**<br><br>**OK** |
| Read Command<br>**AT+ DIALMODE?** | Response<br>**+ DIALMODE: 0**<br>**+ DIALMODE: 1**<br><br>**OK** |
| Write Command<br>**AT+ DIALMODE =<0/1>** | Response<br>a)If successfully:<br>**OK**<br>b)If failed:<br>**ERROR** |

**Defined Values**

| | |
|---|---|
| **<0/1>** | The Auto Dial status Enable/Disable, the default value is 1.<br>0 –Enable Auto Dail<br>1 –Disable Auto Dail<br>The function will take effect immediately. |

### 3.1.2 AT$MYCONFIG Set RNDIS/ECM Mode

| AT$MYCONFIG Set RNDIS/ECM Mode | |
|---|---|
| Write Command<br>**AT$MYCONFIG= " USBNETMODE " , <0/1>** | Response<br>a)If successfully:<br>**OK**<br>b)If failed:<br>**ERROR** |

**Defined Values**

| | |
|---|---|
| **<0/1>** | The RNDIS/ECM mode, the default value is 0.<br>0 –RNDIS<br>1 –ECM<br>The function will reset modem then take effect. |

**NOTE**

this command will auto reboot(take effect after reboot).

### 3.1.3 AT+USBNETIP Change RNDIS/ECM Private IP to Public IP

| AT+USBNETIP Change RNDIS/ECM Private IP to Public IP | |
|---|---|
| Test Command<br>**AT+USBNETIP=?** | Response<br>**+USBNETIP: (0-1)[,(0-255)[,(0-255)[,(1-254)]]]**<br><br>**OK** |
| Write Command<br>**AT+USBNETIP=mode[,tpos[,dhcps[,thcpe]]]** | Response<br>a)If successfully:<br>**OK**<br>b)If failed:<br>**+CME ERROR: unknown error** |
| Read Command<br>**AT+USBNETIP?** | Response<br>a)If successfully:<br>**+USBNETIP: mode**<br><br>**OK**<br>b)If failed:<br>**+CME ERROR: unknown error** |
| Parameter Saving Mode | AUTO_SAVE |

**Defined Values**

| | |
|---|---|
| **<mode>** | 0 –Public Ip<br>1 –Private Ip |
| **<tpos>** | The third position of Gateway Addr |
| **<dhcps>** | Dhcp start value |
| **<dhcpe>** | Dhcp end value |

**Examples:**

**Windows 打开 cmd 输入 ipconfig**

```
以太网适配器 以太网 7:

   连接特定的 DNS 后缀 . . . . . . . :
   IPv6 地址 . . . . . . . . . . . . : 2409:8960:1e58:324c:2934:bda5:f9e8:88d7
   临时 IPv6 地址. . . . . . . . . . : 2409:8960:1e58:324c:65cd:269c:30d1:17f0
   本地链接 IPv6 地址. . . . . . . . : fe80::2934:bda5:f9e8:88d7%18
   IPv4 地址 . . . . . . . . . . . . : 192.168.0.100
   子网掩码 . . . . . . . . . . . . : 255.255.255.0
   默认网关. . . . . . . . . . . . . : fe80::1234%18
                                       192.168.0.1
```

串口输入 AT+USBNETIP=0,10,117，成功后，再次输入 **ipconfig**

```
以太网适配器 以太网 7:

   连接特定的 DNS 后缀 . . . . . . . :
   IPv6 地址 . . . . . . . . . . . . : 2409:8960:1e58:324c:2934:bda5:f9e8:88d7
   临时 IPv6 地址. . . . . . . . . . : 2409:8960:1e58:324c:65cd:269c:30d1:17f0
   本地链接 IPv6 地址. . . . . . . . : fe80::2934:bda5:f9e8:88d7%18
   IPv4 地址 . . . . . . . . . . . . : 192.168.10.117
   子网掩码 . . . . . . . . . . . . : 255.255.255.0
   默认网关. . . . . . . . . . . . . : fe80::1234%18
                                       192.168.10.1
```

串口输入 AT+USBNETIP=0,12,98，成功后，再次输入 **ipconfig**

```
以太网适配器 以太网 7:

   连接特定的 DNS 后缀 . . . . . . . :
   IPv6 地址 . . . . . . . . . . . . : 2409:8960:1e58:324c:2934:bda5:f9e8:88d7
   临时 IPv6 地址. . . . . . . . . . : 2409:8960:1e58:324c:65cd:269c:30d1:17f0
   本地链接 IPv6 地址. . . . . . . . : fe80::2934:bda5:f9e8:88d7%18
   IPv4 地址 . . . . . . . . . . . . : 192.168.12.98
   子网掩码 . . . . . . . . . . . . : 255.255.255.0
   默认网关. . . . . . . . . . . . . : fe80::1234%18
                                       192.168.12.1
```

## 3.2. Test AT Commands

```
#cat /dev/ttyUSB2 &
#echo -e "at\r\n">/dev/ttyUSB2
#
OK
```

## 3.3. Use PPP Data connection

### 3.3.1. How Does a PPP Dial-Up Connection Work?

You will need the right software and a couple of pieces of information before you start.First, check the pppd. If the programs do not exist, you can download the source code from https://ppp.samba.org/download.html and port them to your embedded development environment.
Next you must write configuration file for pppd.

## 3.3.1.1. Chat Scription

```
#named simcom-connect-chat and place in /etc/ppp/peers
ABORT "BUSY"
ABORT "NO CARRIER"
ABORT "NO DIALTONE"
ABORT "ERROR"
ABORT "NO ANSWER"
TIMEOUT 30
"" AT
OK ATE0
OK ATI;+CSUB;+CSQ;+CPIN?;+COPS?;+CGREG?;&D2
# Insert the APN provided by your network operator, default apn is 3gnet
OK AT+CGDCONT=1,"IP","3gnet",,0,0
OK ATD*99#
CONNECT
```

```
#named simcom-disconnect-chat and place in /etc/ppp/peers
ABORT "ERROR"
```

```
ABORT "NO DIALTONE"
SAY "\nSending break to the modem\n"
""  +++
""  +++
""  +++
SAY "\nGoodbay\n"
```

### 3.3.1.2. Configure dialing and AT port

```
# named simcom-pppd and place in /etc/ppp/peers
/dev/ttyUSB2 115200
#Insert the username and password for authentication, default user and password are test
user "test" password "test"
# The chat script, customize your APN in this file
connect 'chat -s -v -f /etc/ppp/peers/simcom-connect-chat'
# The close script
disconnect 'chat -s -v -f /etc/ppp/peers/simcom-disconnect-chat'
# Hide password in debug messages
hide-password
# The phone is not required to authenticate
noauth
# Debug info from pppd
debug
# If you want to use the HSDPA link as your gateway
defaultroute
# pppd must not propose any IP address to the peer
noipdefault
# No ppp compression
novj
novjccomp
noccp
ipcp-accept-local
```

```
ipcp-accept-remote
local
# For sanity, keep a lock on the serial line
lock
modem
dump
nodetach
# Hardware flow control
nocrtscts
remotename 3gppp
ipparam 3gppp
ipcp-max-failure 30
# Ask the peer for up to 2 DNS server addresses
usepeerdns
```

### 3.3.1.3. Dial-Up Connection

```
# pppd call simcom-pppd &
```

When you see the output below,it shows that dial-up succeeded.

```
Connect: ppp0 <--> /dev/ttyUSB2
sent [LCP ConfReq id=0x1 <asyncmap 0x0> <magic 0x5107d141> <pcomp> <accomp>]
rcvd [LCP ConfReq id=0x0 <asyncmap 0x0> <auth chap MD5> <magic 0x9a5c1936> <pcomp>
<accomp>]
sent [LCP ConfAck id=0x0 <asyncmap 0x0> <auth chap MD5> <magic 0x9a5c1936> <pcomp>
<accomp>]
rcvd [LCP ConfAck id=0x1 <asyncmap 0x0> <magic 0x5107d141> <pcomp> <accomp>]
sent [LCP EchoReq id=0x0 magic=0x5107d141]
rcvd [LCP DiscReq id=0x1 magic=0x9a5c1936]
rcvd [CHAP Challenge id=0x1 <dd93b9f04d75e2bbba3786f6d24df3d7>, name =
"UMTS_CHAP_SRVR"]
sent [CHAP Response id=0x1 <498d4d7cf3b59dacfc07a45ce6eb7e26>, name = "test"]
rcvd [LCP EchoRep id=0x0 magic=0x9a5c1936 51 07 d1 41]
rcvd [CHAP Success id=0x1 ""]
```

```
CHAP authentication succeeded
CHAP authentication succeeded
sent [IPCP ConfReq id=0x1 <addr 0.0.0.0> <ms-dns1 0.0.0.0> <ms-dns2 0.0.0.0>]
rcvd [IPCP ConfReq id=0x0]
sent [IPCP ConfNak id=0x0 <addr 0.0.0.0>]
rcvd [IPCP ConfNak id=0x1 <addr 10.51.68.23> <ms-dns1 222.66.251.8> <ms-dns2
116.236.159.8>]
sent [IPCP ConfReq id=0x2 <addr 10.51.68.23> <ms-dns1 222.66.251.8> <ms-dns2
116.236.159.8>]
rcvd [IPCP ConfReq id=0x1]
sent [IPCP ConfAck id=0x1]
rcvd [IPCP ConfAck id=0x2 <addr 10.51.68.23> <ms-dns1 222.66.251.8> <ms-dns2
116.236.159.8>]
Could not determine remote IP address: defaulting to 10.64.64.64
local  IP address 10.51.68.23
remote IP address 10.64.64.64
primary   DNS address 222.66.251.8
secondary DNS address 116.236.159.8
Script /etc/ppp/ip-up started (pid 6616)
Script /etc/ppp/ip-up finished (pid 6616), status = 0x0
```

Now PPP call is set up successfully. Please use following commands to check IP/DNS/Route.

```
# ifconfig ppp0
ppp0      Link encap:Point-to-Point Protocol
          inet addr:10.216.159.39  P-t-P:10.64.64.64  Mask:255.255.255.255
          UP POINTOPOINT RUNNING NOARP MULTICAST  MTU:1500  Metric:1
          RX packets:9 errors:0 dropped:0 overruns:0 frame:0
          TX packets:9 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:3
          RX bytes:362 (362.0 B)  TX bytes:316 (316.0 B)

# cat /etc/resolv.conf
nameserver 221.180.132.108

# route -n
```

```
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
0.0.0.0         0.0.0.0         0.0.0.0         U     0      0        0 ppp0
10.64.64.64     0.0.0.0         255.255.255.255 UH    0      0        0 ppp0

# ping baidu.com
PING baidu.com (220.181.57.216) 56(84) bytes of data.
64 bytes from 220.181.57.216: icmp_seq=1 ttl=50 time=84.0 ms
64 bytes from 220.181.57.216: icmp_seq=2 ttl=50 time=34.2 ms
```

Following commands can be used to terminate PPPD process to disconnect a PPP call:

```
# killall pppd
```

## 3.4. Use ECM Data connection

Enable ECM

```
# cat /dev/ttyUSB2 &
# echo -e "AT+DIALMODE=0\r\n">/dev/ttyUSB2
#
# OK
# echo -e "AT\$MYCONFIG=\"usbnetmode\",1 ">/dev/ttyUSB2
#
# OK
```

Please use following commands to check IP/DNS/Route.

```
# ifconfig usb0
usb0      Link encap:Ethernet  HWaddr ae:68:46:d6:b2:80
          inet addr:192.168.0.100  Bcast:192.168.0.255  Mask:255.255.255.0
          inet6 addr: fe80::ac68:46ff:fed6:b280/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:45 errors:0 dropped:0 overruns:0 frame:0
          TX packets:104 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:4237 (4.2 KB)  TX bytes:13148 (13.1 KB)
```

```
# ping baidu.com
PING baidu.com (123.125.114.144) 56(84) bytes of data.
64 bytes from 123.125.114.144: icmp_seq=1 ttl=56 time=114 ms
64 bytes from 123.125.114.144: icmp_seq=2 ttl=56 time=58.6 ms
64 bytes from 123.125.114.144: icmp_seq=3 ttl=56 time=45.1 ms
```

## 3.5. Use RNDIS Data connection

```
# cat /dev/ttyUSB2 &
# echo -e "AT+DIALMODE=0\r\n">/dev/ttyUSB2
#
# OK
# echo -e "AT\$MYCONFIG=\"usbnetmode\",0\r\n">/dev/ttyUSB2
#
# OK
```

Please use following commands to check IP/DNS/Route.

```
# ifconfig usb0
usb0      Link encap:Ethernet  HWaddr ae:68:46:d6:b2:80
          inet addr:192.168.0.100  Bcast:192.168.0.255  Mask:255.255.255.0
          inet6 addr: fe80::ac68:46ff:fed6:b280/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:45 errors:0 dropped:0 overruns:0 frame:0
          TX packets:104 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:4237 (4.2 KB)  TX bytes:13148 (13.1 KB)

# ping baidu.com
PING baidu.com (220.181.38.148) 56(84) bytes of data.
64 bytes from 220.181.38.148: icmp_seq=1 ttl=50 time=94.8 ms
64 bytes from 220.181.38.148: icmp_seq=2 ttl=50 time=135 ms
64 bytes from 220.181.38.148: icmp_seq=3 ttl=50 time=61.9 ms
```

# 4 Troubleshooting

If Linux does not create devices, check for the kernel module:

```
# lsmod | grep option
```

If entries aren't found, load the kernel module with root privileges:

```
# modprobe option
```

Check dmesg output to see that the radio was detected:

```
# dmesg | grep option
```

Check dmesg output to see that the radio was detected:

```
# dmesg | grep option
[   16.672003] usbcore: registered new interface driver option
[   16.672105] option 2-1.2:1.0: GSM modem (1-port) converter detected
[   16.672216] option 2-1.2:1.1: GSM modem (1-port) converter detected
[   16.672292] option 2-1.2:1.2: GSM modem (1-port) converter detected
[   16.672365] option 2-1.2:1.3: GSM modem (1-port) converter detected
[   16.672438] option 2-1.2:1.4: GSM modem (1-port) converter detected
```

If this returns an error response, the kernel module is not on your system. You will need to build the driver

# 5 Appendix A Abbreviations

Table 1: Terms and Abbreviations

| Abbreviation | Description |
|---|---|
| USB | Universal Serial Bus |
| PPP | Point-to-Point Protocol. The Point-to-Point Protocol is designed for simple links which transport packets between two ports. These links provide full-duplex simultaneous bi-directional operation, and are assumed to deliver packets in order. It is intended that PPP provides a common solution for easy connection of a wide variety of hosts, bridges and routers. |
| IPCP | IP Control Protocol |
| IP | Internet Protocol |
| DNS | Domain Name Server |