SIMCom 5G Modules PCIE Application Note

5G Module

SIMCom Wireless Solutions Limited
Building B, SIM Technology Building, No.633, Jinzhong Road
Changning District, Shanghai P.R. China
Tel: 86-21-31575100
support@simcom.com
www.simcom.com

Document Title:	SIMCom 5G Modules PCIE Application Note
Version:	1.01
Date:	2021.1.17
Status:	Released

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

SIMCom Wireless Solutions Limited
Building B, SIM Technology Building, No.633 Jinzhong Road, Changning District, Shanghai P.R. China
Tel: +86 21 31575100
Email: simcom@simcom.com

For more information, please visit:
https://www.simcom.com/download/list-863-en.html

For technical support, or to report documentation errors, please visit:
https://www.simcom.com/ask/ or email to: support@simcom.com

About Document

Version History

Version	Date	Owner	What is new
1.00	2020-10-21	Zhichong.huang	First release
1.01	2021-1-14	Zhichong.huang	Add support for X86 platform

Scope

This document applies to all SIMCom 5G modules which can support PCIE interface.

Contents

1 Introduction

1.1 Purpose of the document

The PCIE interface from SIMCom 5G modules can support HOST and EP mode, this document will describe how to install PCIE driver on both ARM and X86 platform running Linux.

1.2 Related Documents

[1] SIM8200 Series_AT Command Manual

1.3 Conventions and abbreviations

Abbreviation	Description
PCIE	Peripheral Component Interconnect Express
EP	End Point

2 PCIE Introduction

The PCIE HOST could dial-up Internet access and send AT command over PCIE interface from SIMCom 5G modules.

2.1 PCIE Feature Description

The PCIE from 5G modules support following functions:

Support 1X
Support MSI
Support PCIE Gen 3
Support D0 D3 D3cold

3 Driver Installation on ARM Platform

3.1 5G Module FW Request

SIMCom 5G modules provide AT cmd to configure HOST or EP mode for PCIE interface, please check if following command has already been supported with current FW, if it is not supported please contact SIMCom FAE to upgrade FW.

AT+CPCIEMODE=EP //Configure PCIE to EP mode
AT+CPCIEMODE=HOST // Configure PCIE to HOST mode

3.2 Linux Kernel Request

Suggest to use newer than 4.9.11 Linux kernel version.

3.3 Driver Installation

Please contact SIMCom FAE for PCIE driver source code.

1. Copy qti folder to kernel/driver/
2. Modify kernel/drivers/Makefile, add obj-y += qti/
3. Modify kernel/drivers/Kconfig, add source “drivers/qti/Kconfig”
4. Configure CONFIG_QTI_MHI=y to open Modem Host Interface

```
<*> Modem Host Interface
```

Note:

Inside qti folder there is compiled .o file, customer may not use it if Linux OS is different. If can not use it, please contact SIMCom FAE , provide kernel version\cross compiler\.config file, we will generate valid .o file.

5. Add patch

```
git apply 0002-net-Add-the-get-current-NAPI-context-API.patch
git apply 0003-msm_rmnet-merge-support-for-RAWIP-msm_rmnet-device.patch
git apply 0004-net-rmnet_data-Add-snapshot-of-rmnet_data-driver.patch
git apply 0005-net-ipv6-Generate-random-IID-for-addresses-on-RAWIP-.patch
```

If inside driver package there is additional external-modem-pcie.dtsi, then need to modify device tree inside dts.

```
....
#include "external-modem-pcie.dtsi"
```

At last rebuild kernel, update kernel image.

3.4 PCIE Driver Debugging

1. Check whether PCIE device has been recognized by host.
   Customer can use lspci cmd to list all detected pci devices,17cb:0306 is the ID for SIMCom 5G modules.

```
[root@xxx ~]# lspci
01:00.0 Class ff00: 17cb:0306
```

If lspci cmd is not supported,please try following method.

```
[root@xxx ~]# cat sys/bus/pci/devices/*/vendor
…
0x17cb
[root@xxx ~]# cat /sys/bus/pci/devices/*/device
…
0x0306
```

2. If in step 1,device can not be recognized, please check as following.

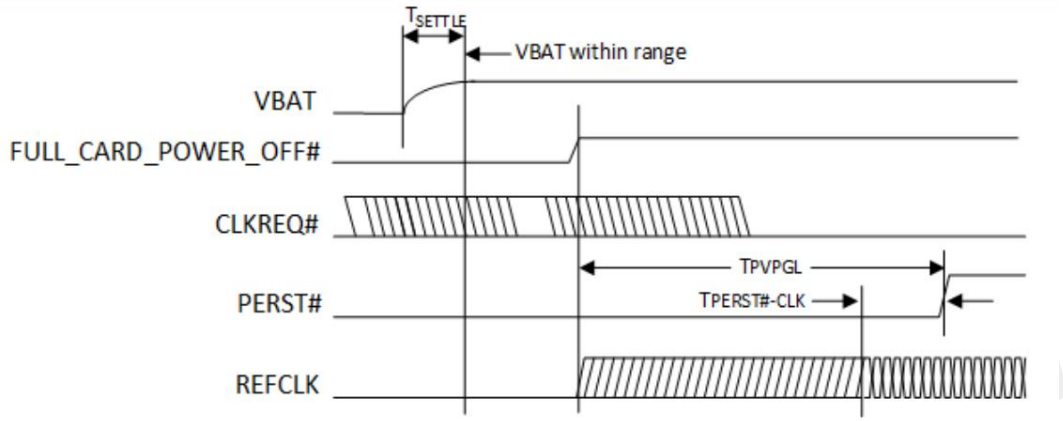
- Check if module has entered into PCIE EP mode.

```
at+cpciemode?
+CPCIEMODE: EP

OK
```

When PCIE is in EP mode, the USB PID is 901F.

- Check if host Linux has opened PCIE HOST controller function, for different SOC the driver is different, please contact SOC vendor for help if it is closed.
- Check if the start up timing for PCIE is correct as following picture shows.



3. After PCIE device get recognized, check MHI state.

```
[root@xxx ~]# ls /dev/mhi*
/dev/mhi_0306_00.01.00_pipe_0    /dev/mhi_0306_00.01.00_pipe_16
/dev/mhi_0306_00.01.00_pipe_10   /dev/mhi_0306_00.01.00_pipe_22
/dev/mhi_0306_00.01.00_pipe_14   /dev/mhi_0306_00.01.00_pipe_32
```

4. Check rmnet_mhi0 state with ifconfig cmd

```
rmnet_mhi0 Link encap:UNSPEC  HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00
          [NO FLAGS]  MTU:16384  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)
```

If rmnet_mhi0 can be recognized, until now PCIE driver on Linux has been successfully installed.

5. Check network registeration status

```
[root@xxx ~]# cat /dev/mhi_0306_00.01.00_pipe_32 &
[root@xxx ~]# echo -e "at+cpsi?\r\n" > /dev/mhi_0306_00.01.00_pipe_32
+CPSI:
LTE,Online,460-01,0x1824,10559509,483,EUTRAN-BAND3,1650,5,5,-81,-805,-505,15
```

6. When network gets registered, compile and run simcom-cm program, then check if there is assigned IP. About source code for simcom-cm, please contact SIMCom support.

```
[root@xxx ~]# ./simcom-cm -d /dev/mhi_0306_00.01.00_pipe_14    -i rmnet_mhi0 &
Build Version: 2020-09-08 10:20:15
.....
udhcpc: sending select for 10.166.183.38
```

If udhcpc can not get IP, please try following manual method.

```
[root@xxx ~]# busybox udhcpc -f -n -q -t 5 -i rmnet_mhi0
```

7. Check network status

```
[root@xxx ~]# ifconfig rmnet_mhi0
rmnet_mhi0 Link encap:UNSPEC  HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00
          inet addr:10.166.183.38  Mask:255.255.255.252
          inet6 addr: fe80::200:ff:fe00:0/64 Scope:Link
          UP RUNNING  MTU:1436  Metric:1
          RX packets:5 errors:0 dropped:0 overruns:0 frame:0
          TX packets:18 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:864 (864.0 B)  TX bytes:1660 (1.6 KiB)
```

8. Verify by ping operation.

```
[root@xxx ~]# ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8): 56 data bytes
64 bytes from 8.8.8.8: seq=0 ttl=114 time=65.025 ms
64 bytes from 8.8.8.8: seq=1 ttl=114 time=77.891 ms
64 bytes from 8.8.8.8: seq=2 ttl=114 time=76.357 ms
```

4 Driver Installation on X86 Platform

For x86 platform usually SIMCom will provide ko module file, customer need to install the ko file when system is powered up.

a) insmod mhi_pcie.ko
b) Refer to chapter 3.4.
c) After verification, add the ko file to auto start option on system.

```
[root@xxx ~]# su root
[root@xxx ~]# mkdir /lib/modules/`uname -r`/kernel/simcom
[root@xxx ~]# cp mhi_pcie.ko /lib/modules/`uname -r`/kernel/simcom
[root@xxx ~]# cd /lib/modules/`uname -r`/kernel/simcom
[root@xxx ~]# depmod
[root@xxx ~]# modinfo mhi_pcie
[root@xxx ~]# echo "mhi_pcie" >> /etc/modules
```