

SDX55 PCIe BringUP 手册

5G Module Series

Our aim is to provide customers with timely and comprehensive service. For any assistance, please contact our company headquarters:

Quectel Wireless Solutions Co., Ltd.

Building 5, Shanghai Business Park Phase III (Area B), No.1016 Tianlin Road, Minhang District, Shanghai, China 200233

Tel: +86 21 5108 6236

Email: info@quectel.com

Or our local office. For more information, please visit:

<http://www.quectel.com/support/sales.htm>

For technical support, or to report documentation errors, please visit:

<http://www.quectel.com/support/technical.htm>

Or email to: support@quectel.com

GENERAL NOTES

QUECTEL OFFERS THE INFORMATION AS A SERVICE TO ITS CUSTOMERS. THE INFORMATION PROVIDED IS BASED UPON CUSTOMERS' REQUIREMENTS. QUECTEL MAKES EVERY EFFORT TO ENSURE THE QUALITY OF THE INFORMATION IT MAKES AVAILABLE. QUECTEL DOES NOT MAKE ANY WARRANTY AS TO THE INFORMATION CONTAINED HEREIN, AND DOES NOT ACCEPT ANY LIABILITY FOR ANY INJURY, LOSS OR DAMAGE OF ANY KIND INCURRED BY USE OF OR RELIANCE UPON THE INFORMATION. ALL INFORMATION SUPPLIED HEREIN IS SUBJECT TO CHANGE WITHOUT PRIOR NOTICE.

COPYRIGHT

THE INFORMATION CONTAINED HERE IS PROPRIETARY TECHNICAL INFORMATION OF QUECTEL WIRELESS SOLUTIONS CO., LTD. TRANSMITTING, REPRODUCTION, DISSEMINATION AND EDITING OF THIS DOCUMENT AS WELL AS UTILIZATION OF THE CONTENT ARE FORBIDDEN WITHOUT PERMISSION. OFFENDERS WILL BE HELD LIABLE FOR PAYMENT OF DAMAGES. ALL RIGHTS ARE RESERVED IN THE EVENT OF A PATENT GRANT OR REGISTRATION OF A UTILITY MODEL OR DESIGN.

Copyright © Quectel Wireless Solutions Co., Ltd. 2019. All rights reserved.

About the Document

History

Revision	Date	Author	Description
1.0	2019-12-17	Simon.Xiao Carl.YinSimon.	Initial

Contents

About the Document	2
Contents	3
1. AT 使能没有 efuse 模块的 PCIE 功能	4
2. 时序图说明	5
3. IPQ PCIE 识别成功的 LOG	6
4. IPQ PCIE 识别代码流程	7
4.1 IPQ 拉低 PCIE_RESET	7
4.2 IPQ 拉高 PCIE_RESET	7
4.3 IPQ 开始 PCIE 识别, 默认识别 10 次, 间隔 100ms. 如果在这段时间内, SDX55 的 PCIE 没有 READY, 会导致识别不成功, 可适当加大识别次数	7
5. SDX55 上 log 查询 PCIE 是否识别成功 PCIE 识别代码流程	8
6. SDX55 代码 PCIE 识别代码流程	9
6.1 X55 的 PCIE 初始化, 拉低 PCIE_WAKEUP 通知 IPQ 可以开始识别	9
6.2 X55 等待 IPQ 拉高 PCIE_RESET, 每隔 1ms 检查下 PCIE_RESET 电平, 检查 30 秒	9
6.3 X55 检测到 PCIE_RESET 变高, 开始 PCIE 识别, 每隔 1ms 识别一次, 识别 30 秒。	9

1. AT 使能没有 efuse 模块的 PCIE 功能

```
root@OpenWrt:/# busybox microcom /dev/ttyUSB2
```

```
at+qcfg="data_interface",1,0
```

```
OK
```

```
+QCFG: "data_interface",1,0
```

```
OK
```

AT+QCFG="data_interface"[,<network>,<diag>]

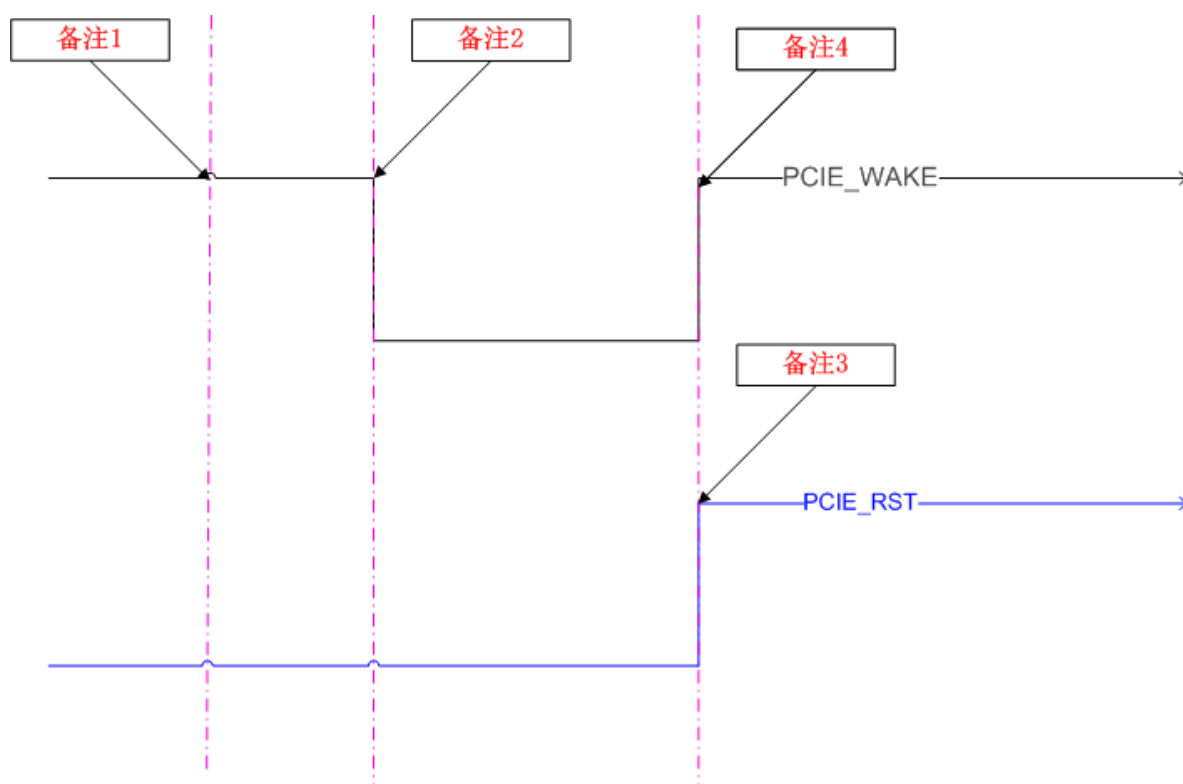
<network> 0 network passing through USB.

 1 network passing through PCIE.

<diag> 0 diagnostic passing through USB.

 1 diagnostic passing through PCIE.

2. 时序图说明



备注 1: 模块开机

备注 2: 模块 PCIE 开机初始化, 拉低 wakeup 通知 AP

备注 3: AP 拉高 Reset, AP 开始 PCIE 枚举

备注 4: 模块收到 reset 变高, 也拉高 wakeup, 也开始 PCIE 枚举

3. IPQ PCIE 识别成功的 LOG

```
root@OpenWrt:/# lspci
```

```
00:00.0 Class 0604: 17cb:0302
```

```
01:00.0 Class ff00: 17cb:0306
```

```
root@OpenWrt:/# dmesg | grep pci
```

```
[ 0.314350] qcom-pcie 20000000.pci: GPIO lookup for consumer perst
[ 0.314362] qcom-pcie 20000000.pci: using device tree for GPIO lookup
[ 0.314375] of_get_named_gpiod_flags: can't parse 'perst-gpios' property of node '/soc/pci@20000000[0]'
[ 0.314390] of_get_named_gpiod_flags: parsed 'perst-gpio' property of node '/soc/pci@20000000[0]' - status
(0)
[ 0.314490] 20000000.pci supply vdda not found, using dummy regulator
[ 0.314617] 20000000.pci supply vdda_phy not found, using dummy regulator
[ 0.314661] 20000000.pci supply vdda_refclk not found, using dummy regulator
[ 0.315184] PCI host bridge /soc/pci@20000000 ranges:
```

//下面是识别成功 log, 可以标红色加粗, 以醒目作用

```
[ 0.450166] qcom-pcie 20000000.pci: link up. retries 1
[ 0.450336] qcom-pcie 20000000.pci: PCI host bridge to bus 0000:00
[ 0.450350] pci_bus 0000:00: root bus resource [bus 00-ff]
[ 0.450364] pci_bus 0000:00: root bus resource [io 0x0000-0xffff] (bus address [0x20200000-0x202fffff])
[ 0.450375] pci_bus 0000:00: root bus resource [mem 0x20300000-0x20ffffff]
[ 0.450405] pci 0000:00:00.0: [17cb:0302] type 01 class 0x060400
[ 0.450436] pci 0000:00:00.0: reg 0x10: [mem 0x00000000-0x00000fff]
[ 0.450493] pci 0000:00:00.0: PME# supported from D0 D3hot D3cold
```

//下面是 17cb 0306 是 X55 的 vid 和 pid, 可以标红色加粗, 以醒目作用

```
[ 0.450937] pci 0000:01:00.0: [17cb:0306] type 00 class 0xff0000
```

4. IPQ PCIE 识别代码流程

4.1 IPQ 拉低 PCIE_RESET

drivers/pci/host/pcie-qcom.c

```
static void qcom_ep_reset_assert(struct qcom_pcie *pcie)
{
    //gpiod_set_value(pcie->reset, 1); //carl.yin delete this, will cause x55 link fail, i donot know why
    usleep_range(PERST_DELAY_US, PERST_DELAY_US + 500);
}
```

4.2 IPQ 拉高 PCIE_RESET

drivers/pci/host/pcie-qcom.c

```
static void qcom_ep_reset_deassert(struct qcom_pcie *pcie)
{
    gpiod_set_value(pcie->reset, 0);
    usleep_range(PERST_DELAY_US, PERST_DELAY_US + 500);
}
```

4.3 IPQ 开始 PCIE 识别, 默认识别 10 次, 间隔 100ms. 如果在这段时间内, SDX55 的 PCIE 没有 READY, 会导致识别不成功, 可适当加大识别次数

drivers/pci/host/pcie-designware.c

```
int dw_pcie_wait_for_link(struct pcie_port *pp)
{
    int retries;

    /* check if the link is up or not */
    for (retries = 0; retries < LINK_WAIT_MAX_RETRIES; retries++) {
        if (dw_pcie_link_up(pp)) {
            //如果识别成功, 会打印如下代码的 LOG
            dev_info(pp->dev, "link up. retries %d\n", retries);
            return 0;
        }
        usleep_range(LINK_WAIT_USLEEP_MIN, LINK_WAIT_USLEEP_MAX);
        if (strstr(dev_name(pp->dev), "20000000")) {
            //usleep_range(LINK_WAIT_USLEEP_MIN, LINK_WAIT_USLEEP_MAX);
            //usleep_range(LINK_WAIT_USLEEP_MIN, LINK_WAIT_USLEEP_MAX);
        }
    }
    //如果识别不成功, 会打印如下代码的 LOG
    dev_err(pp->dev, "phy link never came up. retries %d\n", retries);

    return -ETIMEDOUT;
}
```


5. SDX55 上 log 查询 PCIE 是否识别成功

PCIE 识别代码流程

sdxprairie login: root

Password:

~ # dmesg | grep pci

```
[ 0.233897] pcie-ep 40002000.qcom,pcie: 40002000.qcom,pcie supply vreg-cx not found, using dummy
regulator
[ 0.234209] ep_pcie_get_resources: Clock pcie_cfg_ahb_clk isn't available:-517
[ 0.234230] ep_pcie_probe: PCIe V1711211: failed to get resources
[ 0.234247] ep_pcie_probe: PCIe V1711211: Driver probe failed:-517
[ 0.242480] pcie:pcie_init.
[ 0.263417] pcie-ep 40002000.qcom,pcie: 40002000.qcom,pcie supply vreg-cx not found, using dummy
regulator
[ 0.265027] register_client_adhoc:Client handle 6 pcie-ep
[ 0.267869] ep_pcie_reset_init: After Reset assert pcie_core_reset
[ 0.269101] ep_pcie_reset_init: After Reset de-assert pcie_core_reset
[ 0.269148] ep_pcie_reset_init: After Reset assert pcie_phy_reset
[ 0.270264] ep_pcie_reset_init: After Reset de-assert pcie_phy_reset
//下面 LOG 表示 X55 拉低 PCIE_WAKEUP. IPQ 可以开始 PCIE 识别 X55
[ 0.270300] ep_pcie_core_enable_endpoint: PCIe V1711211: assert PCIe WAKE#
[ 1.072786] ep_pcie_phy_init: PCIe V1711211: Unexpected phy version 2100 is caught
[ 1.073229] ep_pcie_core_enable_endpoint: PCIe V1711211: PCIe PHY is ready
[ 1.131830] ep_pcie_core_enable_endpoint: PCIe V1711211: link initialized for LE PCIe endpoint
[ 1.131917] ep_pcie_core_enable_endpoint: PCIe V1711211: request to turn on the power when link is already
powered on
//下面 LOG 表示 PCIE 识别成功
[ 1.131941] ep_pcie_enumeration: PCIe V1711211: PCIe link training is successful with host side. Waiting for
enumeration to complete
[ 1.131972] ep_pcie_enumeration: PCIe V1711211: PCIe link training is successful with host side. Waiting for
enumeration to complete
[ 3.441152] ehci-pci: EHCI PCI platform driver
~ #
```

6. SDX55 代码 PCIE 识别代码流程

6.1 X55 的 PCIE 初始化, 拉低 PCIE_WAKEUP 通知 IPQ 可以开始识别

drivers\platform\msm\ep_pcie_core.c

```
/* assert PCIe WAKE# */
```

```
EP_PCIE_INFO(dev, "PCIe V%d: assert PCIe WAKE#\n", dev->rev);
```

对应的模块 LOG 如下

```
[ 0.270300] ep_pcie_core_enable_endpoint: PCIe V1711211: assert PCIe WAKE#
```

6.2 X55 等待 IPQ 拉高 PCIE_RESET, 每隔 1ms 检查下 PCIE_RESET 电平, 检查 30 秒

```
/* wait for host side to deassert PERST */
```

```
retries = 0;
```

```
do {
```

```
    if (gpio_get_value(dev->gpio[EP_PCIE_GPIO_PERST].num) == 1)
```

```
        break;
```

```
    retries++;
```

```
    usleep_range(PERST_TIMEOUT_US_MIN, PERST_TIMEOUT_US_MAX);
```

```
} while (retries < PERST_CHECK_MAX_COUNT);
```

如果 30 秒没有检测到 PCIE_RESET 变高, 会打印如下 LOG

```
EP_PCIE_ERR(dev, "PCIe V%d: PERST is not de-asserted by host\n", dev->rev);
```

```
[ 0.272276] ep_pcie_core_enable_endpoint: PCIe V1711211: assert PCIe WAKE#
```

```
[ 30.916856] ep_pcie_core_enable_endpoint: PCIe V1711211: PERST is not de-asserted by host
```

6.3 X55 检测到 PCIE_RESET 变高, 开始 PCIE 识别, 每隔 1ms 识别一次, 识别 30 秒。

如果识别失败, 则打印如下错误 LOG

```
EP_PCIE_ERR(dev, "PCIe V%d: PCIe PHY failed to come up\n", dev->rev);
```