



SIM7600 Series_Open Linux UART&SPI_Application Note

LTE 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:	SIM7600 Series_Open Linux UART&SPI_Application Note
Version:	2.00
Date:	2020.8.6
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

THIS DOCUMENT CONTAINS PROPRIETARY TECHNICAL INFORMATION WHICH IS THE PROPERTY OF SIMCOM WIRELESS SOLUTIONS LIMITED COPYING, TO OTHERS AND USING THIS DOCUMENT, ARE FORBIDDEN WITHOUT EXPRESS AUTHORITY BY SIMCOM. OFFENDERS ARE LIABLE TO THE PAYMENT OF INDEMNIFICATIONS. ALL RIGHTS RESERVED BY SIMCOM IN THE PROPRIETARY TECHNICAL INFORMATION , INCLUDING BUT NOT LIMITED TO REGISTRATION GRANTING OF A PATENT , A UTILITY MODEL OR DESIGN. ALL SPECIFICATION SUPPLIED HEREIN ARE SUBJECT TO CHANGE WITHOUT NOTICE AT ANY TIME.

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

Copyright © 2020 SIMCom Wireless Solutions Limited All Rights Reserved.

About Document

Version History

Version	Date	Owner	What is new
V2.00	2020.8.6	Zhichong Huang	Update document format

SIMCom
Confidential

Contents

About Document.....	3
Version History.....	3
Scope.....	错误! 未定义书签。
Contents.....	4
1. Document Overview.....	5
2. Hardware interface.....	6
2.1 Description of related PIN.....	6
3. UART.....	7
3.1 High speed UART.....	7
3.1.1 PIN6-9 Configure high speed UART.....	7
3.1.2 PIN6-9 Configures BT.....	9
3.1.3 PIN66-71 configures high speed UART.....	10
3.2 Low speed UART.....	11
3.2.1 PIN7,8 configures low speed UART.....	11
3.2.2 PIN68,71 configures low speed UART.....	13
4. SPI.....	14
4.1 PIN6-9 configures SPI.....	14
4.2 PIN66-71 configures SPI.....	17

1. Document Overview

This document explains the compilation and configuration of the module's UART interface and SPI interface.

SIMCom
Confidential

2. Hardware interface

2.1 Description of related PIN

PIN No.	PIN name	Default Function	Function1	Function2
6	SPI_CLK	UART_RTS	SPI_CLK	UART_RTS
7	SPI_MISO	UART_RX	SPI_MISO	UART_RX
8	SPI_MOSI	UART_TX	SPI_MOSI	UART_TX
9	SPI_CS	UART_CTS	SPI_CS	UART_CTS
66	UART_RTS	UART_RTS	UART_RTS	SPI_CLK
67	UART_CTS	UART_CTS	UART_CTS	SPI_CS
68	UART_RX	UART_RX	UART_RX	SPI_MISO
71	UART_TX	UART_TX	UART_TX	SPI_MOSI

3. UART

3.1 High speed UART

3.1.1 PIN6-9 Configure high speed UART

```
.\kernel\arch\arm\boot\dts\qcom\mdm9607.dtsi:

blsp1_uart3: uart@78b1000 {
    compatible = "qcom,msm-hsuart-v14";
    reg = <0x78b1000 0x200>,
        <0x7884000 0x2b000>;
    reg-names = "core_mem", "bam_mem";
    interrupt-names = "core_irq", "bam_irq";
    #address-cells = <0>;
    interrupt-parent = <&blsp1_uart3>;
    interrupts = <0 1>;
    #interrupt-cells = <1>;
    interrupt-map-mask = <0xffffffff>;
    interrupt-map = <0 &intc 0 119 0
        1 &intc 0 238 0>;

    qcom,bam-tx-ep-pipe-index = <4>;
    qcom,bam-rx-ep-pipe-index = <5>;
    qcom,master-id = <86>;
    qcom,dev-id = <0>;
    clock-names = "core_clk", "iface_clk";
    clocks = <&clock_gcc clk_gcc_blsp1_uart3_apps_clk>,
        <&clock_gcc clk_gcc_blsp1_ahb_clk>;
    pinctrl-names = "sleep", "default";
    pinctrl-0 = <&blsp1_uart3_sleep>;
    pinctrl-1 = <&blsp1_uart3_active>;
    qcom,msm-bus,name = "blsp1_uart3";
    qcom,msm-bus,num-cases = <2>;
    qcom,msm-bus,num-paths = <1>;
```

```

qcom,msm-bus,vectors-KBps =
    <86 512 0 0>,
    <86 512 500 800>;
status = "disabled";
};

```

```

.\kernel\arch\arm\boot\dts\qcom\mdm9607-mtp.dtsi:

```

```

&blsp1_uart3 {
    status = "ok";
};

```

```

/* remove spi_2 config

```

```

&spi_2 {
    status = "ok";
};
*/

```

Modify as shown above, PIN6,7,8,9 are configured as UART function, and the system generates /dev/ttyHS0 device node after system start up.

qcom,dev-id = <0>; Determine the generated device node serial number

The above configuration does not support RX wake-up. Applications are required to manage device power, and configuration can also be modified to support RX wake-up sleep.

```

blsp1_uart3: uart@78b1000 {
    ...
    interrupt-names = "core_irq", "bam_irq", "wakeup_irq";
    #address-cells = <0>;
    interrupt-parent = <&blsp1_uart3>;
    interrupts = <0 1 2>;
    #interrupt-cells = <1>;
    interrupt-map-mask = <0xffffffff>;
    interrupt-map = <0 &intc 0 119 0
                    1 &intc 0 238 0
                    2 &tlmm_pinmux 1 0>;

    qcom,inject-rx-on-wakeup;
    qcom,rx-char-to-inject = <0xFD>;
    qcom,bam-tx-ep-pipe-index = <4>;
    qcom,bam-rx-ep-pipe-index = <5>;
    ...
    status = "disabled";
}

```



```
};
```

3.1.2 PIN6-9 Configures BT

Modify the file:

```
.\kernel\arch\arm\boot\dts\qcom\mdm9607.dtsi:
    blsp1_uart3: uart@78b1000 {
        compatible = "qcom,msm-hsuart-v14";
        reg = <0x78b1000 0x200>,
            <0x7884000 0x2b000>;
        reg-names = "core_mem", "bam_mem";
        interrupt-names = "core_irq", "bam_irq", "wakeup_irq";
        #address-cells = <0>;
        interrupt-parent = <&blsp1_uart3>;
        interrupts = <0 1 2>;
        #interrupt-cells = <1>;
        interrupt-map-mask = <0xffffffff>;
        interrupt-map = <0 &intc 0 119 0
            1 &intc 0 238 0
            2 &tlmm_pinmux 1 0>;

        qcom,inject-rx-on-wakeup;
        qcom,rx-char-to-inject = <0xFD>;
        qcom,bam-tx-ep-pipe-index = <4>;
        qcom,bam-rx-ep-pipe-index = <5>;
        qcom,master-id = <86>;
        qcom,dev-id = <0>;
        clock-names = "core_clk", "iface_clk";
        clocks = <&clock_gcc clk_gcc_blsp1_uart3_apps_clk>,
            <&clock_gcc clk_gcc_blsp1_ahb_clk>;
        pinctrl-names = "sleep", "default";
        pinctrl-0 = <&blsp1_uart3_sleep>;
        pinctrl-1 = <&blsp1_uart3_active>;
        qcom,msm-bus,name = "blsp1_uart3";
        qcom,msm-bus,num-cases = <2>;
        qcom,msm-bus,num-paths = <1>;
        qcom,msm-bus,vectors-KBps =
            <86 512 0 0>,
```

```

        <86 512 500 800>;
        status = "disabled";
    };
    .\kernel\arch\arm\boot\dts\qcom\ mdm9607-mtp.dtsi:
    &blsp1_uart3 {
        status = "ok";
    };

    /* remove spi_2 config
    &spi_2 {
        status = "ok";
    };
    */

```

Modify as shown above, the Bluetooth application can use ttyHS0 to communicate with the Bluetooth chip. After the Bluetooth application and the Bluetooth chip stop communicating for 3 seconds, ttyHS0 automatically goes to sleep.

3.1.3 PIN66-71 configures high speed UART

Modify the file:

```

.\kernel\arch\arm\boot\dts\qcom\ mdm9607.dtsi:
    blsp1_uart2: uart@78b0000 {
        compatible = "qcom,msm-hsuart-v14";
        reg = <0x78b0000 0x200>,
            <0x7884000 0x2b000>;
        reg-names = "core_mem", "bam_mem";
        interrupt-names = "core_irq", "bam_irq";
        #address-cells = <0>;
        interrupt-parent = <&blsp1_uart2>;
        interrupts = <0 1>;
        #interrupt-cells = <1>;
        interrupt-map-mask = <0xffffffff>;
        interrupt-map = <0 &intc 0 108 0
            1 &intc 0 238 0>;

        qcom,bam-tx-ep-pipe-index = <2>;
        qcom,bam-rx-ep-pipe-index = <3>;
    };

```

```
qcom,master-id = <86>;
qcom,dev-id = <1>;
clock-names = "core_clk", "iface_clk";
clocks = <&clock_gcc clk_gcc_bls1_uart2_apps_clk>,
        <&clock_gcc clk_gcc_bls1_ahb_clk>;
pinctrl-names = "sleep", "default";
pinctrl-0 = <&blsp1_uart2_sleep>;
pinctrl-1 = <&blsp1_uart2_active>;
qcom,msm-bus,name = "blsp1_uart2";
qcom,msm-bus,num-cases = <2>;
qcom,msm-bus,num-paths = <1>;
qcom,msm-bus,vectors-KBps =
        <86 512 0 0>,
        <86 512 500 800>;
status = "disabled";
};
```

```
.\kernel\arch\arm\boot\dts\qcom\mdm9607-mtp.dtsi:
```

```
&blsp1_uart2 {
    status = "ok";
};
```

```
/* remove spi_1 config
```

```
&spi_1 {
    status = "ok";
};
*/
```

Modify as shown above, PIN66, 67, 68, 71 are configured as UART function. After the system starts up, the /dev/ttyHS1 device node is generated. The UART is generally controlled by using the DTR (PIN72) to wake up the device.

3.2 Low speed UART

3.2.1 PIN7,8 configures low speed UART

The low speed UART does not support flow control and DMA. Large amount of data communication

will certainly lose data, generally used for module console.

Modify the file:

```
.\kernel\arch\arm\boot\dts\qcom\ mdm9607.dtsi:
    blsp1_uart3: serial@78b1000 { /* BLSP1 UART3 */
        compatible = "qcom,msm-lsuart-v14";
        reg = <0x78b3000 0x200>;
        interrupts = <0 119 0>;
        clocks = <&clock_gcc clk_gcc_blsp1_uart3_apps_clk>,
                <&clock_gcc clk_gcc_blsp1_ahb_clk>;
        clock-names = "core_clk", "iface_clk";
        status = "disabled";
    };

.\kernel\arch\arm\boot\dts\qcom\ mdm9607-mtp.dtsi:
&blsp1_uart3 {
    status = "ok";
    pinctrl-names = "default";
    pinctrl-0 = <&uart3_console_sleep>;
};

.\kernel\arch\arm\boot\dts\qcom\ mdm9607-pinctrl.dtsi
    uart3_console_sleep: uart3_console_sleep {
        mux {
            pins = "gpio0", "gpio1";
            function = "blsp_uart3";
        };
        config {
            pins = "gpio0", "gpio1";
            drive-strength = <2>;
            bias-pull-down;
        };
    };
};
```

Modify as shown above, PIN 7,8 is configured as a low-speed UART function. After the system starts up, the /dev/ttyHSL1 device node is generated. In this way, PIN 6, 9 can be configured for other functions. If the UART is required for the console, then execute /sbin/getty - L ttyHSL1 115200 console or add S1:2345:respawn:/sbin/getty -L ttyHSL1 115200 console to /etc/inittab.

3.2.2 PIN68,71 configures low speed UART

Modify the file:

```

.\kernel\arch\arm\boot\dts\qcom\mdm9607.dtsi:
    blsp1_uart2: serial@78b0000 { /* BLSP1 UART2 */
        compatible = "qcom,msm-lsuart-v14";
        reg = <0x78b0000 0x200>;
        interrupts = <0 108 0>;
        clocks = <&clock_gcc clk_gcc_blsp1_uart2_apps_clk>,
                <&clock_gcc clk_gcc_blsp1_ahb_clk>;
        clock-names = "core_clk", "iface_clk";
        status = "disabled";
    };

.\kernel\arch\arm\boot\dts\qcom\mdm9607-mtp.dtsi:
&blsp1_uart2 {
    status = "ok";
    pinctrl-names = "default";
    pinctrl-0 = <&uart2_console_sleep>;
};

.\kernel\arch\arm\boot\dts\qcom\mdm9607-pinctrl.dtsi
    uart2_console_sleep: uart2_console_sleep {
        mux {
            pins = "gpio4", "gpio5";
            function = "blsp_uart2";
        };
        config {
            pins = "gpio4", "gpio5";
            drive-strength = <2>;
            bias-pull-down;
        };
    };
};

```

Modify as shown above, The PIN 68, 71 is configured as a low-speed UART function. After the system is booted, the /dev/ttyHSL1 device node is generated so that the PINs 66, 67 can be configured for other functions. If the UART is required for the console, then execute /sbin/getty -L ttyHSL1 115200 console or add S1:2345:respawn:/sbin/getty -L ttyHSL1 115200 console to /etc/inittab.

4.SPI

4.1 PIN6-9 configures SPI

Modify the file:

```
.\kernel\arch\arm\boot\dts\qcom\mdm9607.dtsi:
aliases {
    /* smdttty devices */
    smd7 = &smdttty_data1;
    smd8 = &smdttty_data4;
    smd11 = &smdttty_data11;
    smd21 = &smdttty_data21;
    smd36 = &smdttty_loopback;
    smd38 = &smdttty_data38;
    /* spi device */
    spi1 = &spi_1;
    spi2 = &spi_2;
    i2c4 = &i2c_4;
    sdhc2 = &sdhc_2; /* SDC2 SD card slot */
    i2c5 = &i2c_5; /* Add by sunxiang */
};
...

spi_2: spi@78b7000 { /* BLSP1 QUP2 */
    compatible = "qcom,spi-qup-v2";
    #address-cells = <1>;
    #size-cells = <0>;
    reg-names = "spi_physical", "spi_bam_physical";
    reg = <0x78b7000 0x600>,
        <0x7884000 0x2b000>;
    interrupt-names = "spi_irq", "spi_bam_irq";
    interrupts = <0 97 0>, <0 238 0>;
    spi-max-frequency = <19200000>;
    pinctrl-names = "spi_default", "spi_sleep";
    pinctrl-0 = <&spi2_default &spi2_cs0_active>;
    pinctrl-1 = <&spi2_sleep &spi2_cs0_sleep>;
    clocks = <&clock_gcc clk_gcc_blsp1_ahb_clk>
```

```

        <&clock_gcc clk_gcc_blspl1_qup3_spi_apps_clk>;
        clock-names = "iface_clk", "core_clk";
        qcom,infinite-mode = <0>;
        qcom,use-bam;
        qcom,use-pinctrl;
        qcom,ver-reg-exists;
        qcom,bam-consumer-pipe-index = <16>;
        qcom,bam-producer-pipe-index = <17>;
        qcom,master-id = <86>;
        status = "disabled";
};

```

.\kernel\arch\arm\boot\dts\qcom\ mdm9607-mtp.dtsi:

```

&spi_2 {
    status = "ok";
};

```

/* remove uart3 config

```

&blsp1_uart3 {
    status = "ok";
};
*/

```

.\kernel\arch\arm\boot\dts\qcom\ mdm9607-pinctrl.dtsi

```

spi2 {
    spi2_default: spi2_default {
        /* active state */
        mux {
            /* MOSI, MISO, CLK */
            pins = "gpio0", "gpio1", "gpio3";
            function = "blsp_spi3";
        };

        config {
            pins = "gpio0", "gpio1", "gpio3";
            drive-strength = <12>; /* 12 MA */
            bias-pull-up; /* PULL UP */
        };
    };

    spi2_sleep: spi2_sleep {

```

```
/* suspended state */
mux {
    /* MOSI, MISO, CLK */
    pins = "gpio0", "gpio1", "gpio3";
    function = "gpio";
};

config {
    pins = "gpio0", "gpio1", "gpio3";
    drive-strength = <2>; /* 2 MA */
    bias-pull-down; /* PULL Down */
};

spi2_cs0_active: spi2_cs0_active {
    /* CS */
    mux {
        pins = "gpio2";
        function = "blsp_spi3";
    };

    config {
        pins = "gpio2";
        drive-strength = <12>;
        bias-pull-up;
    };
};

spi2_cs0_sleep: spi2_cs0_sleep {
    /* CS */
    mux {
        pins = "gpio2";
        function = "gpio";
    };

    config {
        pins = "gpio2";
        drive-strength = <12>;
        bias-pull-up;
    };
};
};
```


4.2 PIN66-71 configures SPI

```
.\kernel\arch\arm\boot\dts\qcom\mdm9607-mtp.dtsi:  
&spi_1 {  
    status = "ok";  
};  
  
/* remove uart2 config  
&blsp1_uart2 {  
    status = "ok";  
};  
*/
```

SIMCOM
Confidential