HiHope RZ_G2M Board User Guide

ISSUE: 00B01

DATE: 4th July, 2019

Trademarks and Permissions

Hoperun icons are trademarks of Hoperun Technologies Co., Ltd. All other trademarks and trade names mentioned in this document are the property of their respective holders.

Notice

The purchased products, services and features are stipulated by the contract made between hoperun and the customer. All or part of the products, services and features described in this document may not be within the purchase scope or the usage scope. Unless otherwise specified in the contract, all statements, information, and recommendations in this document are provided "AS IS" without warranties, guarantees or representations of any kind, either express or implied.

The information in this document is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute a warranty of any kind, express or implied.

Jiangsu HopeRun Software Co., Ltd
Address： Building B, HopeRun•Innovation & Intelligence Center,
No.168 Software Avenue, Yuhuatai District, Nanjing, China
Website: http: //www.hoperun.com
E-mail： webadmin@hoperun.com

About This Document

Purpose

This document describes the basic functions, hardware features, User's interface and hardware configurations of the HiHope RZ_G2M Board. This document also provides the software debugging methods.

Related Version

The following table lists the product version related to this document.

Product Name	Version
HiHope RZ_G2M Board	V001

Change History

Changes between document issues are cumulative. The latest document issue contains all the changes made in earlier issues.

Issue 00B01 (2019-07-04)

This issue is the first draft release.

Contents

1. Overview

1.1 Introduction

The HiHope RZ_G2M Board User Guide is an evaluation board developed based on the Renesas media processing chip RZ_G2M. This board is used to demonstrate powerful multimedia functions as well as various peripheral interfaces of RZ_G2M. The reference hardware design based on RZ_G2M is also provided to help customers shorten the product development cycle.

The HiHope RZ_G2M Board hardware and interface configuration are designed based on the 96Boards Consumer Edition Specification.

The HiHope RZ_G2M Board can be connected to a PC by using Micro USB cables and network port cables, forming a basic development system.

1.2 Features

The HiHope RZ_G2M Board has the following features:

- RZ_G2M SoC: Two 1.5-GHz Arm® Cortex®-A57 MP Core™ cores, Four 1.3-GHz Arm® Cortex®-A53 MP Core™ cores
- DDR: LPDDR4, 4GB
- EMMC: eMMC5.0 base, 32GB
- Flash: SPI Flash, 64MB
- WiFi/BT: IEEE 802.11a/b/g/n, 2.4G/5G, 2X2 MIMO antenna; Bluetooth 4.1, CSA2
- One USB 3.0 port
- One USB 2.0 Host Type A port
- One Micro USB port for debug
- One Micro USB port for USB OTG
- One Micro SD card port
- Two PCIe x1 interfaces
- One HDMI interface
- One SATA interface

- One LVDS LCM interface
- One RJ45 port for 100M/1000MHz Ethernet interface
- Two CAN Bus interfaces
- One Low Speed Expansion Connector, 40 pin (20x2), low profile female 2mm receptacle
- One High Speed Expansion Connector, 60 pin 0.8mm high speed Board to Board low profile receptacle connector
- One DC Jack,12V/2A
- One External Fan

2. Hardware

2.1 Structure and Interface

The HiHope RZ_G2 Board is composed of a motherboard and a sub board, connected by a pair of B2B connectors.

Figure 2.1 Interface Structure of Mother Board (Top View)

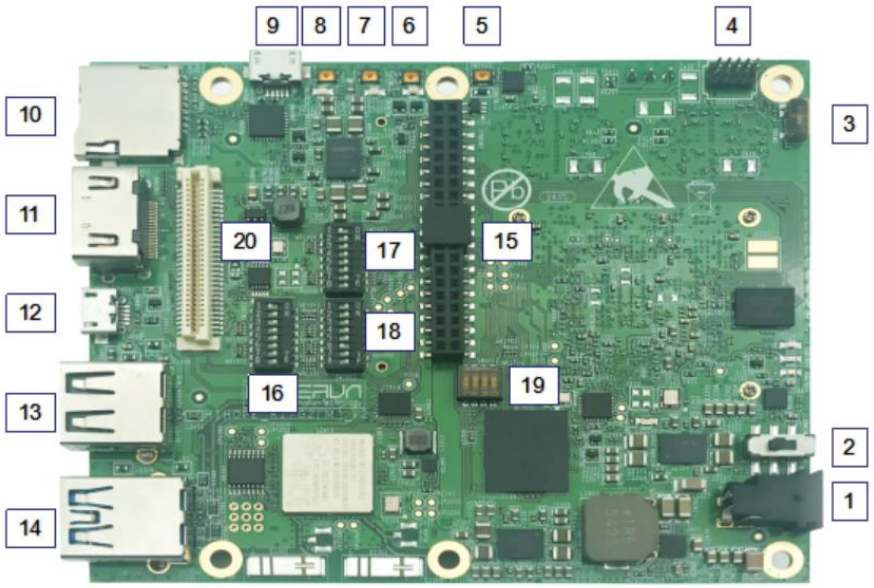


Figure 2.2 Interface Structure of Mother Board (Bottom View)

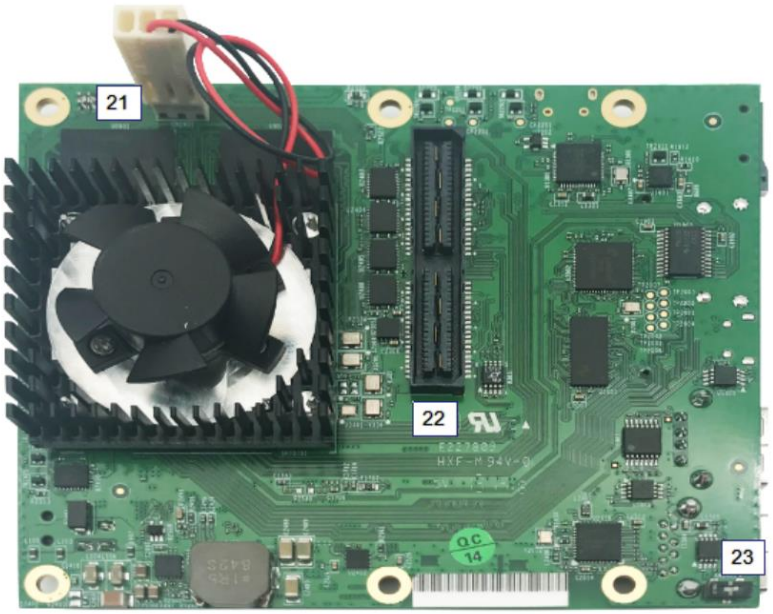


Figure 2.3 Interface Structure of Sub Board (Top View)

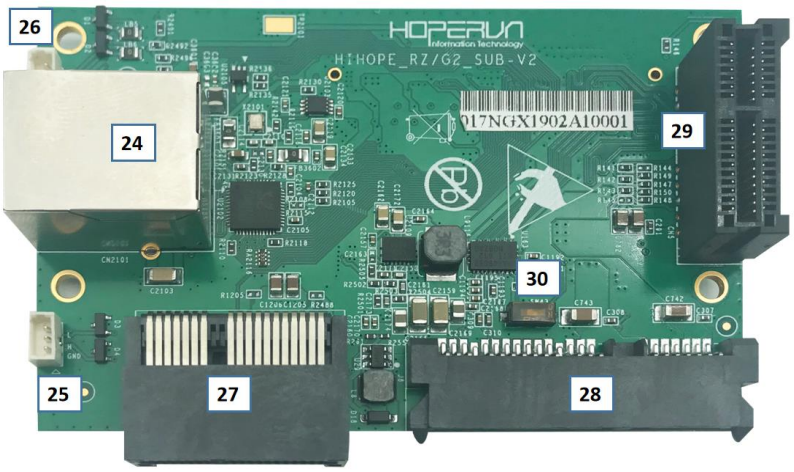


Figure 2.4 Interface Structure of Sub Board (Bottom View)

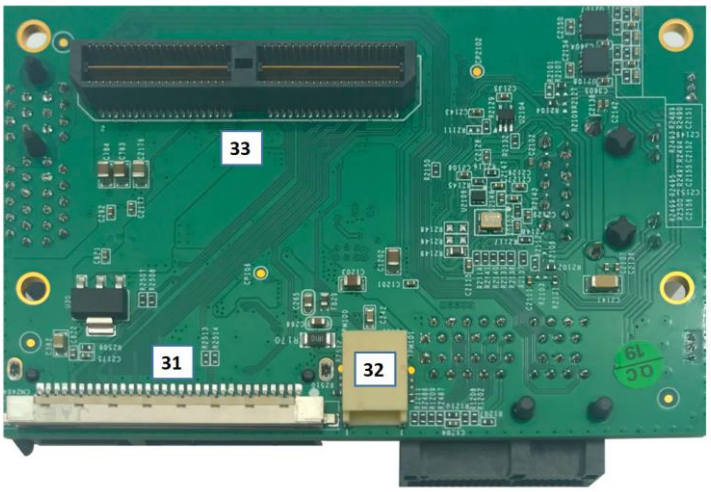


Table 2.1 Structure Interface Signal List

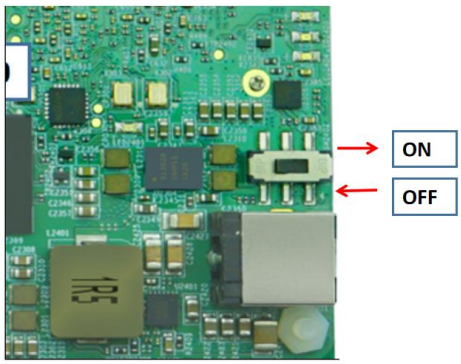
No.	Description
1	Board power interface, supporting 12V/2A or higher performance--J2401
2	DC12V Power Switch--SW2402
3	JTAG Reset Switch--SW305
4	JTAG Connector--CN0301
5	External Reset Button-SW2403 ON: Reset OFF: Reset release
6	User Button1--SW2203, which is connected to GPIO6_13 ON: Input L, LED2203 light off OFF: Input H, LED2203 light on
7	User Button2--SW2201, which is connected to GPIO6_12 ON: Input L, LED2202 light off. OFF: Input H,LED2202 light on.
8	User Button3--SW2202, which is connected to GPIO6_11 ON: Input L, LED2201 light off. OFF: Input H, LED2201 light on.
9	USB for download or debug, Micro USB Connector--CN2201
10	Micro SD Card, support SD3.0--CN1601
11	HDMI output--CN1901
12	USB2.0, support OTG, Micro USB Connector--CN1401
13	USB2.0 host, USB Type A Connector--CN10
14	USB3.0 Port, which is downward compatible with USB 2.0--CN1501
15	Low Speed Expansion Connector: 2 UART, 1 SPI(4 line), 2 I2C, 1 I2S/PCM--CN1801
16	Dip-Switch 1--SW1001

17	Dip-Switch 2--SW1003
18	Dip-Switch 3--SW1002
19	Dip-Switch 4--SW0201 Default settings,reserved for future use
20	High Speed Expansion Connector: 1 SDIO, 1 MIPI DSI, 1 MIPI CSI, 2 I2C--CN1701
21	Power Supply Connector for CPU Fan--CN2401
22	Motherboard to Sub Board Connector(B2B), Male Head--CN2403
23	USB2.0 Overcurrent Protection IC and OTG IC Selector Switch--SW1401
24	Gigabit Ethernet Port(Include Network Transformer)--CN2101
25	CAN BUS1 Interface--J2403
26	CAN BUS0 Interface--J0201
27	PCIe x1 Channal0 Interface--CN2403
28	SATA Interface--J8
29	PCIe x1 Channal1 Interface--CN5
30	SATA/PCIe Switch--SW43
31	LVDS LCD Interface--CN2404
32	LCD Backlight Interface--CN19
33	Sub Board to Motherboard Connector(B2B),Female Socket--J2402

2.2 Interface or Function Description

2.2.1 Power Switch (SW2402)

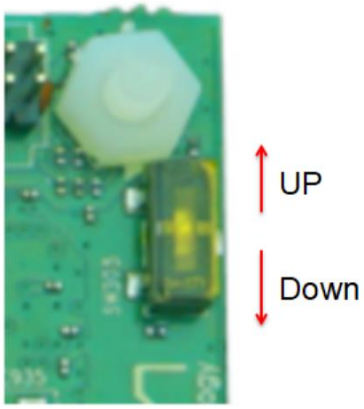
Figure 2.5 Power Switch



ON: Power on the board
OFF: Power off the board

2.2.2 JTAG Reset Switch (SW305)

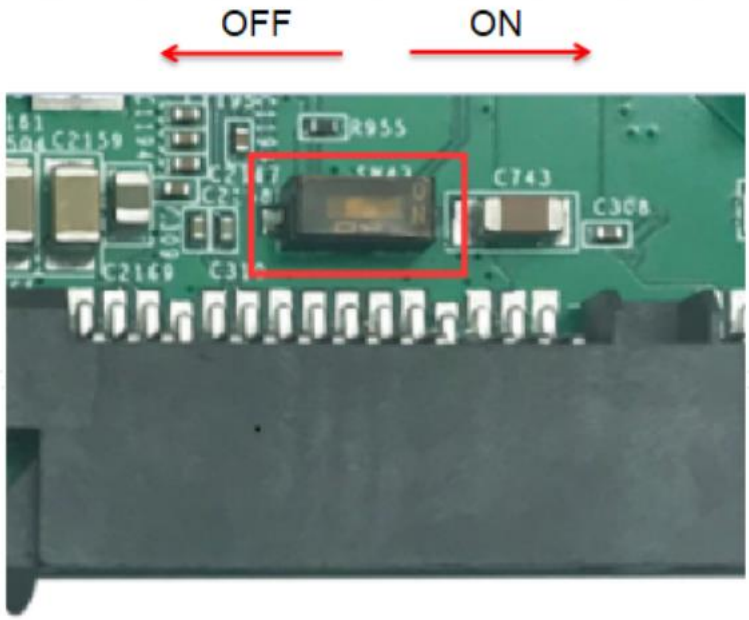
Figure 2.6 JTAG Reset Switch



Up: Reset release
Down: Reset

2.2.3 SATA/PCIe Switch (SW43)

Figure 2.7 SATA/PCIe Switch



ON(L): Select SATA (J8)
OFF(H): Select PCIe (CN5)

2.2.4 DC in Jack

DC Power is provided via the DC jack at J2401. An 8V up to 18V power supply at a minimum of 2A rating can be used to provide sufficient board power for on system requirements as well as external devices.

NOTE: Power should not be supplied simultaneously from multiple sources.

2.2.5 Low Speed Expansion Connector

The Low Speed Expansion brings out 1.8V level SoC signals such as UART0 and UART1, I2C0 and I2C1, GPIO signals as well as SPI, PCM, Reset, 1.8V and 5V power supply. The complete list of signals is shown in Table 2-2 below:

Table 2.2 The Low Speed Expansion Interface Signal List

Signal	Pin	Pin	Signal
GND	1	2	GND
UART0_CTS	3	4	PWR_BTN_N
UART0_TxD	5	6	RST_BTN_N
UART0_RxD	7	8	SPI0_SCLK
UART0_RTS	9	10	SPI0_DIN
UART1_TxD	11	12	SPI0_CS
UART1_RxD	13	14	SPI0_DOUT

I2C0_SCL	15	16	PCM_FS
I2C0_SDA	17	18	PCM_CLK
I2C1_SCL	19	20	PCM_DO
I2C1_SDA	21	22	PCM_DI
GPIO-A	23	24	GPIO-B
GPIO-C	25	26	GPIO-D
GPIO-E	27	28	GPIO-F
GPIO-G	29	30	GPIO-H
GPIO-I	31	32	GPIO-J
GPIO-K	33	34	GPIO-L
+1V8	35	36	SYS_DCIN1
+5V	37	38	SYS_DCIN2
GND	39	40	GND

2.2.6 High Speed Expansion Connector

The High Speed Expansion Connector is a board to board low profile 60 pin receptacle connector, brings out 1.8V level, Include 1 SDIO, 1 MIPI DSI, 1 MIPI CSI, 2 I2C. The complete list of signals is shown in Table 2-3 below:

Table 2.3 The Low Speed Expansion Interface Signal List

Signal	Pin	Pin	Signal
SD_DAT0	1	2	CSI0_CLK+
SD_DAT1	3	4	CSI0_CLK-
SD_DAT2	5	6	GND
SD_DAT3	7	8	CSI0_D0+
SD_SCLK	9	10	CSI0_D0-
SD_CMD	11	12	GND
GND	13	14	CSI0_D1+
CLK0/CSI0_MCLK	15	16	CSI0_D1-
CLK1/CSI1_MCLK	17	18	GND
GND	19	20	CSI0_D2+
DSI_CLK+	21	22	CSI0_D2-
DSI_CLK-	23	24	GND
GND	25	26	CSI0_D3+
DSI_D0+	27	28	CSI0_D3-

DSI_D0-	29	30	GND
GND	31	32	I2C2_SCL
DSI_D1+	33	34	I2C2_SDA
DSI_D1-	35	36	I2C5_SCL
GND	37	38	I2C5_SDA
DSI_D2+	39	40	GND
DSI_D2-	41	42	CSI1_D0+
GND	43	44	CSI1_D0-
DSI_D3+	45	46	GND
DSI_D3-	47	48	CSI1_D1+
GND	49	50	CSI1_D1-
USB_D+	51	52	GND
USB_D-	53	54	CSI1_CLK+
GND	55	56	CSI1_CLK-
RESERVED	57	58	GND
RESERVED	59	60	1.8V Pull Up（100K Resistor）

2.2.7 Display Interfaces

A HDMI Type-A port mounted at CN1901.

A LVDS connector is provided on the sub-board at CN2404.

2.2.8 USB Ports

There are a total of 3 USB ports on the board.

A Type-A USB 3.0 port mounted at CN1501. It supports Host\Function mode.

A Type-A USB 2.0 port mounted at CN10. It supports Host mode.

A Micro USB port mounted at CN1401. It is an USB OTG port, and connected with a charge pump so that it can charge slave device more quickly. The charge pump is enabled\disabled by Charge Pump Switch (SW1401).

Another Micro USB port mounted at CN2201 is Debug Serial interface.

2.2.9 System and User LEDs

There are 8 LEDs on the board.

Figure 2.8 System and User LEDs

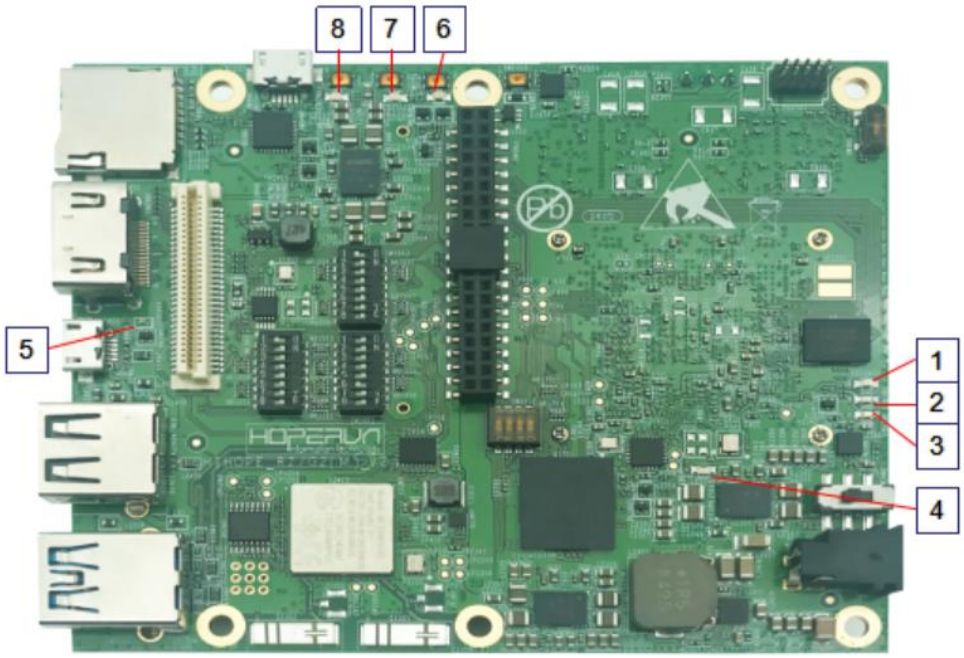


Table 2.4 System and User LEDs List

No.	Description
1	LED2402, be controlled by SoC Pin AJ4(D0) High: LED on Low: LED off
2	LED2403, be controlled by SoC Pin AK3(GPIO GP7_01) High: LED on Low: LED off
3	LED2404, be controlled by SoC Pin AJ5(GPIO GP7_00) High: LED on Low: LED off
4	LED2401, the system reset LED Reset insert: LED on Reset release: LED off
5	LED1901, the HDMI detection indicator LED Insert: LED on Pull out: LED off
6	LED2203, be controlled by User button3(SW2203) Push: LED off Off: LED on

7	LED2202, be controlled by User button1(SW2201) Push: LED off Off: LED on
8	LED2201, be controlled by User button2(SW2202) Push: LED off Off: LED on

2.3.0 JTAG Header

The board includes the option for soldering a 10 pin header that brings out the SoC signals for JTAG debug. A FTSH-105-01-F-DK header can be populated at CN0301.

2.3.1 UART Debug

There is a Micro USB port mounted at CN2201 for debugging. This is normally used by the first stage bootloader developers, and is connected to the UART0 interface of the SoC.

2.3.2 PCIe and SATA Connectors

There are 2 PCIe x1 connectors and 1 SATA connector on the sub board.

At the same time, only one of PCIe connector (CN5) and SATA connector (J8) can work. Which one can work is decided by the Switch (SW43).

2.3.3 RZ_G2M Overall installation diagram

Figure 2.9 RZ_G2M Overall Installation Diagram(Top View)

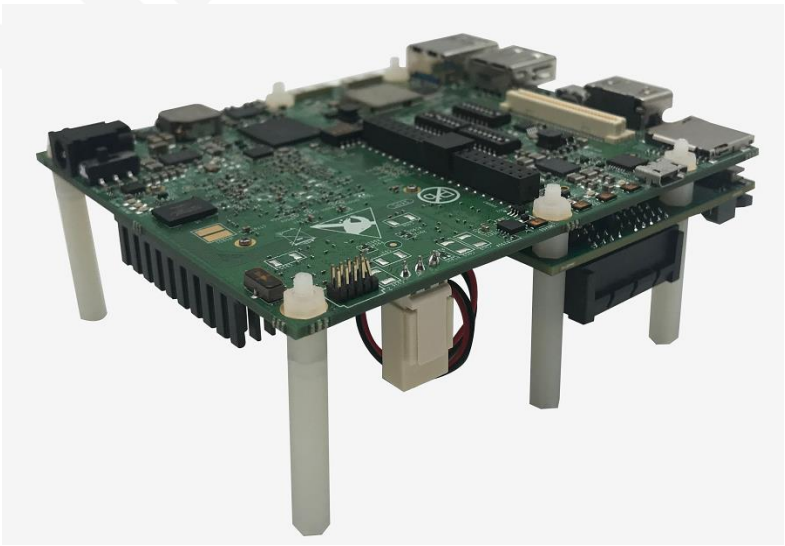
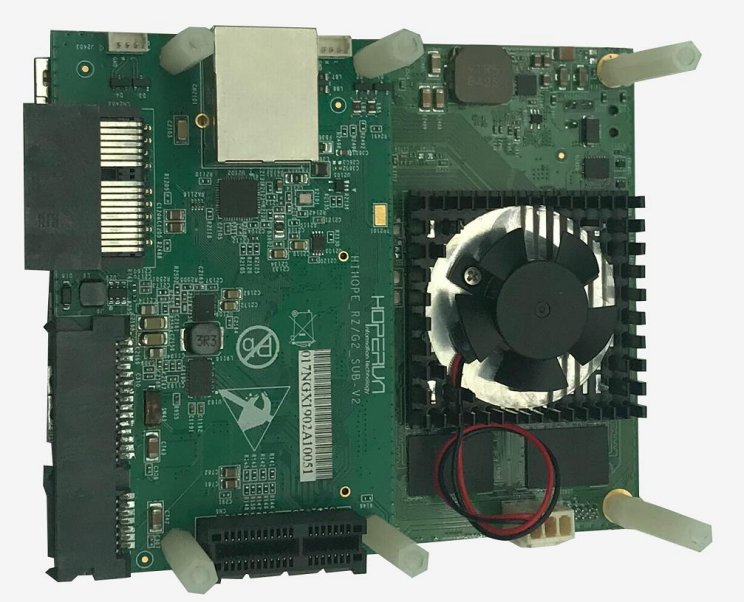


Figure 2.10 RZ_G2M Overall Installation Diagram(Bottom View)



2.3.4 External Dimensions and Hole Locations

Figure 2.11 External Dimensions and Hole Locations(Mother Board)

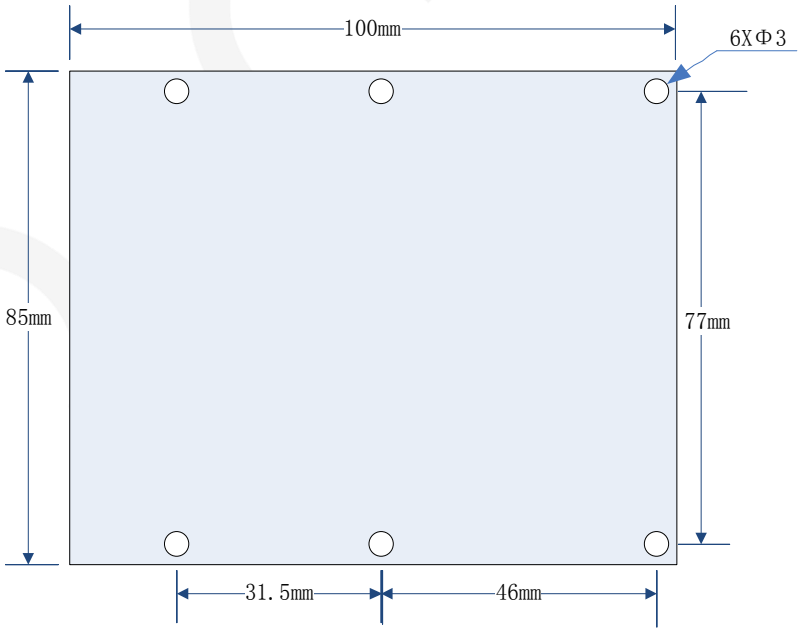
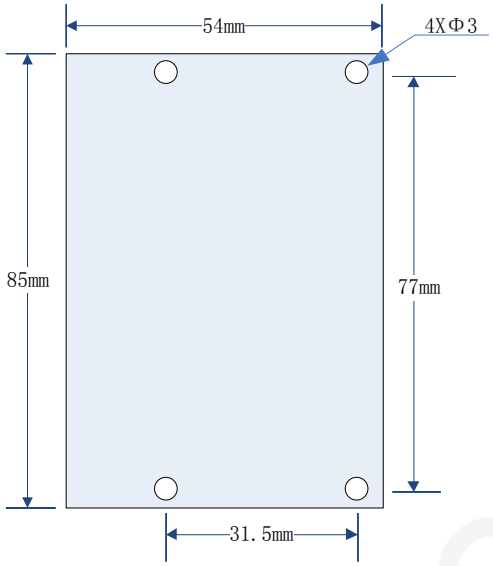


Figure 2.12 External Dimensions and Hole Locations(Sub Board)



3. Operation Guide

3.1 Precautions

The board applies to the laboratory or engineering development environment. Take the following precautions before performing operations:

- Never perform the hot-swap operation on the board.
- Before unpacking the board package or installing the board, take antistatic measures to protect the board hardware from being damaged by the electrostatic discharge (ESD).
- Hold the edges of the board and do not touch the exposed metal on the board. Otherwise, the component parts on the board may be damaged by the static electricity.
- Place the board on a dry plane and keep them away from heat sources, electromagnetic interference sources, radiant sources, and electromagnetic susceptibility equipment (such as the medical equipment).
- Ensure that you can identify the components such as the power supplies, connectors, and indicators and know their positions.

3.2 Configuring the Board

The configuration of the Download Mode Enable Bit involves the settings of 3 DIP switches, these Dials to the right is on, left is off. Don't leave the dial DIP in the middle position.

Figure 3.1 Download Mode Configuration DIP Switches

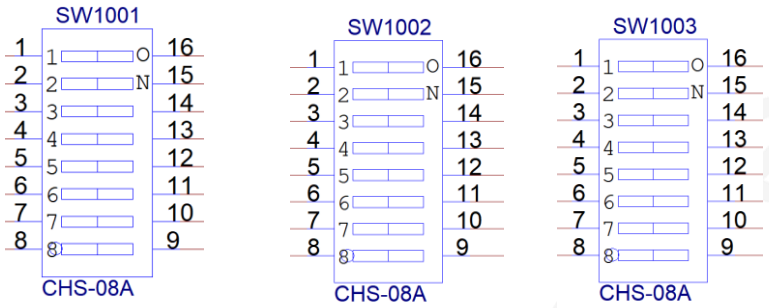


Table 3.1 Describe the Serial Download Mode States of the DIP Switches

No	SW1001	SW1002	SW1003
1	Off	On	Off
2	On	On	On
3	On	On	On
4	On	On	On
5	On	Off	On
6	On	Off	On
7	On	Off	On
8	On	Off	On

Table 3.2 Describe SPI Flash Download Mode States of the DIP Switches

No	SW1001	SW1002	SW1003
1	Off	On	Off
2	On	On	On
3	On	On	On
4	On	On	On
5	On	On	On
6	On	Off	On
7	On	On	On

8	On	On	On

3.3 Getting Start

3.3.1 Prerequisites

Before you power up your RZ_G2M board for the first time, you will need the following:

- RZ_G2M board
- A power supply output DC 12V/2A
- A HDMI LCD Monitor that supports a resolution of 1080P/60Hz or 4K/30Hz
- HDMI cable used to connect the board with the monitor
- A computer keyboard with USB port
- A computer mouse with USB port

3.3.2 Protocol Setting for Debug Serial Interface

3.3.2.1 Install USB to UART Bridge Driver

Before using debug UART, follow these steps.

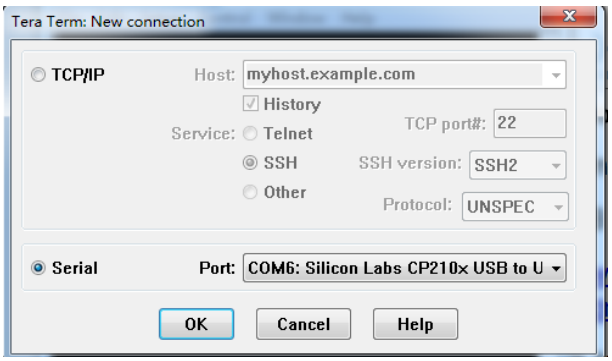
1) Install USB to UART bridge driver to your PC.

   https://www.silabs.com/products/development-tools/software/usb-to-uart-bridge-vcp-drivers

2) Install the terminal soft “Tera Term” to your PC.

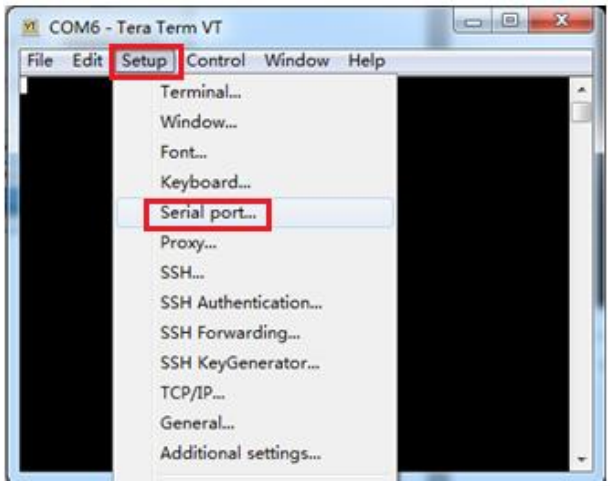
   https://ttssh2.osdn.jp/index.html.en

3) Plug debug cable into computer and board. Open the “Tera term” on PC.

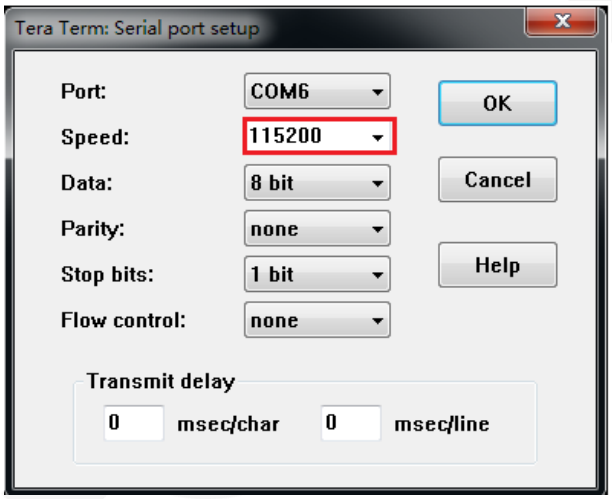


Choose the COM Port

4) Setup the “Tera term”.



Choose “Serial port…”



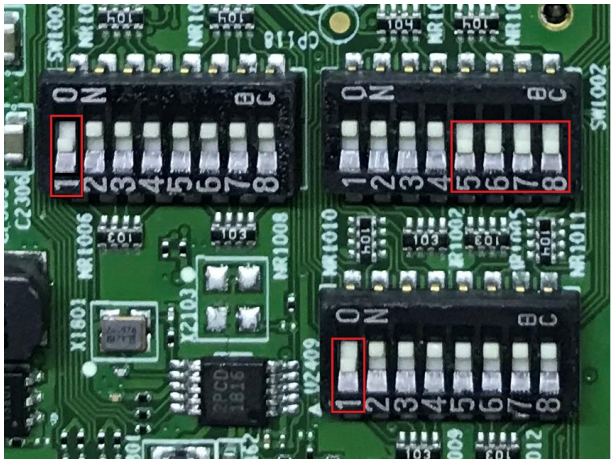
Setup Serial Port

3.3.2.2 Software Installation

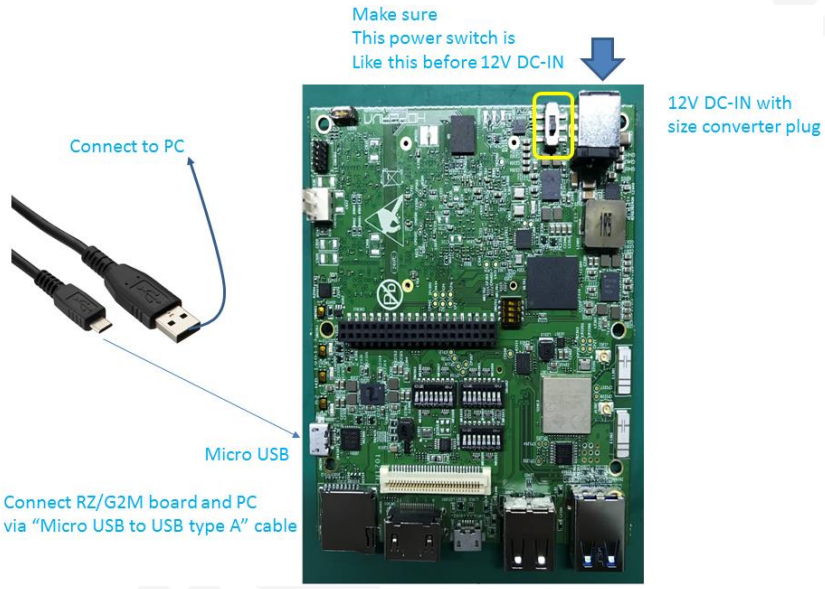
Download IPL and u-boot to the board, follow these steps:

1) Set red DIP switches to “off” to enter serial download mode.

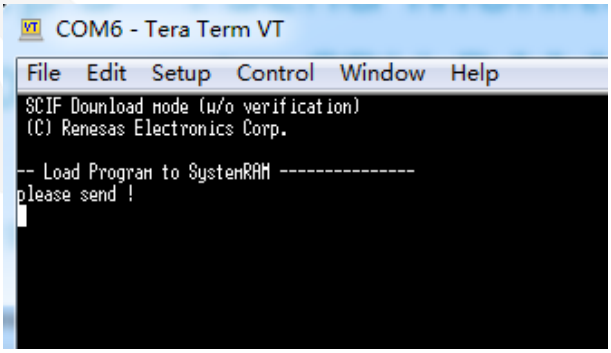
NOTE：Don't leave the dial DIP in the middle position.



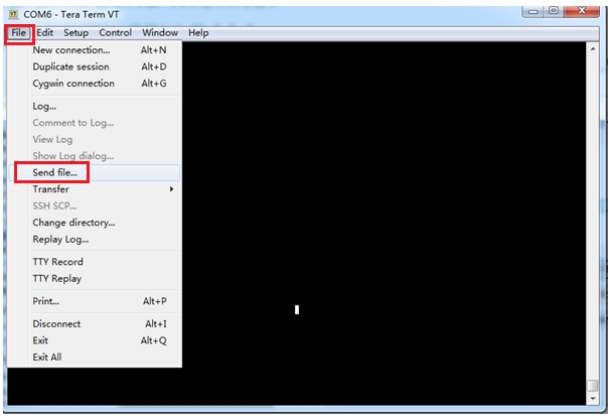
2) Plug in debug cable and power supply.



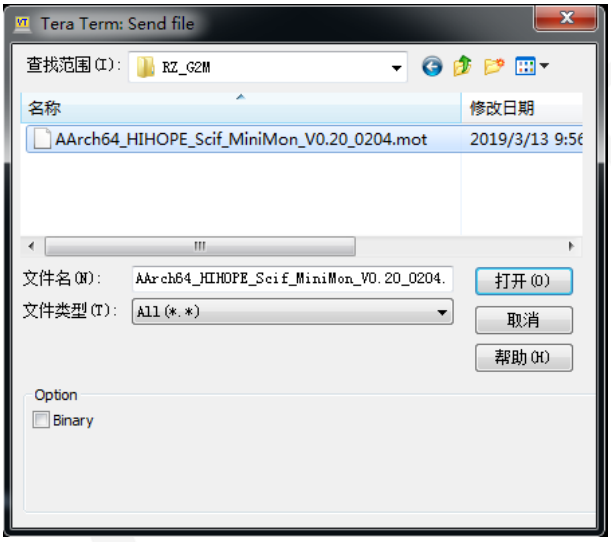
3) Power on and check the output information.



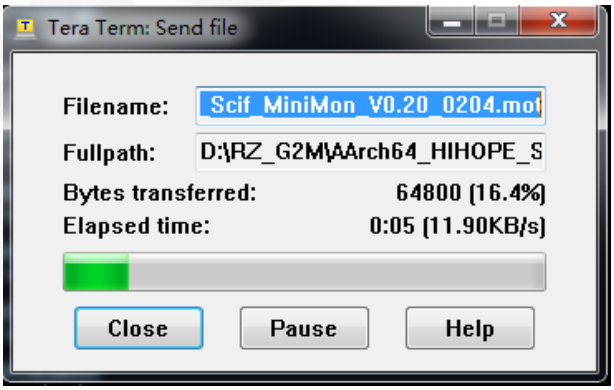
4) Send Monitor Program to CPU RAM.



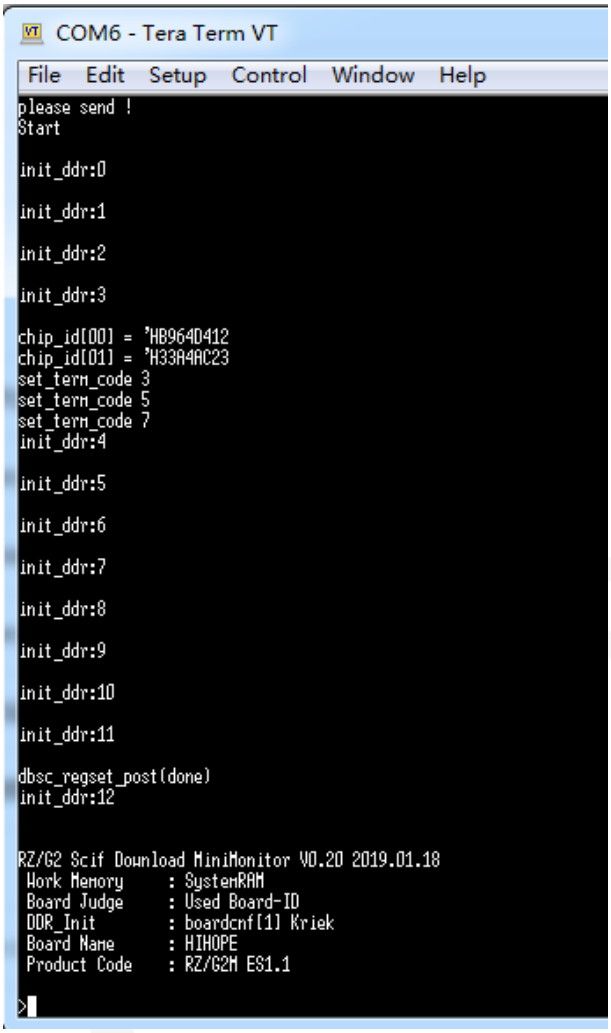
Select “Send file…”



Open “AArch64_HiHope_Scif_MiniMon_V0.20_0204.mot”

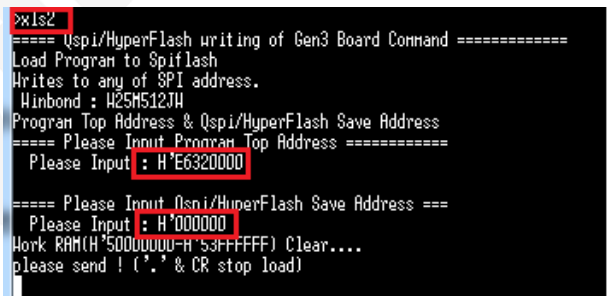


Send Monitor Program to CPU RAM

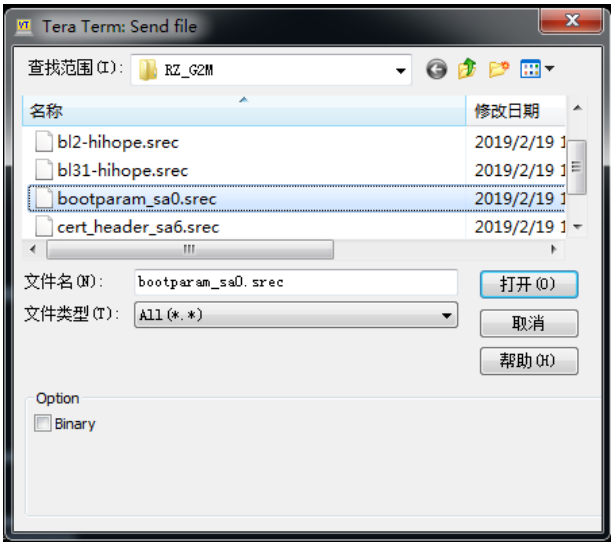


Download MiniMon successfully and output startup information

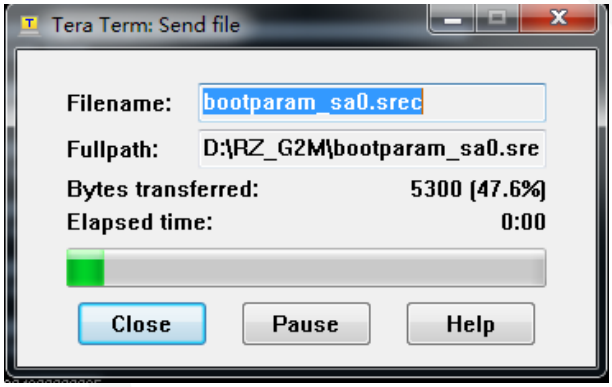
5) Write “bootparam_sa0.srec” to QSPI Flash via SCIF.



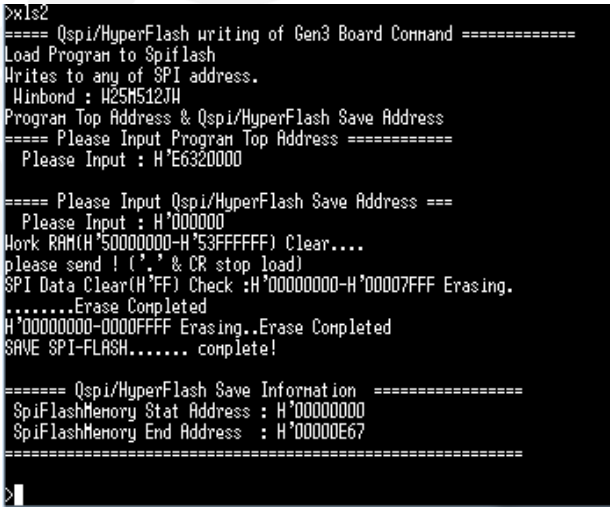
Input cmd “xls2” and address



Open “bootparam_sa0.srec”

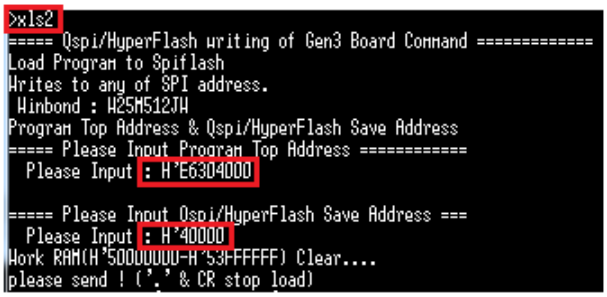


After download “bootparam_sa0.srec”, input “y”

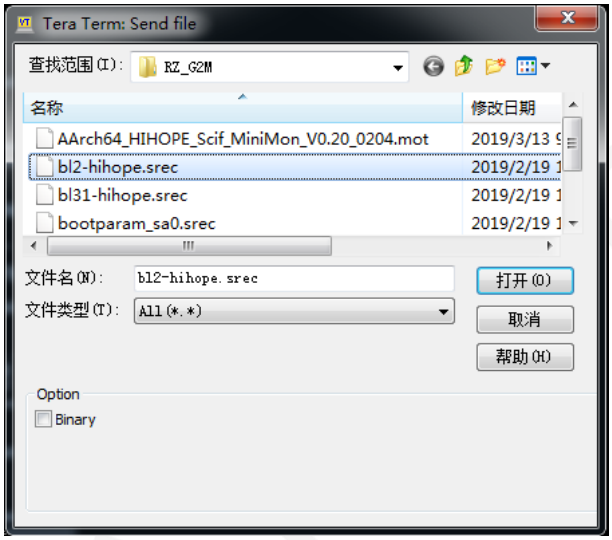


Download “bootparam_sa0.srec” successfully

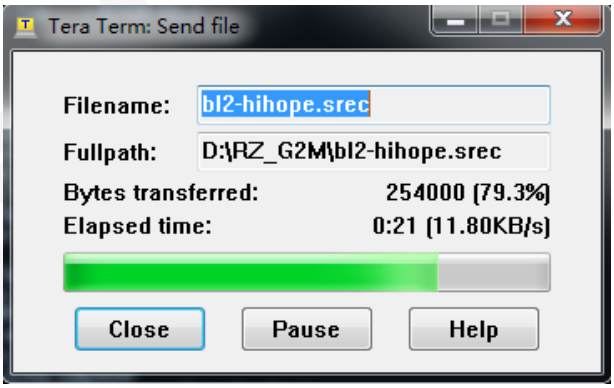
6) Write “bl2-hihope.srec” to QSPI Flash via SCIF.



Input cmd "xls2" and address



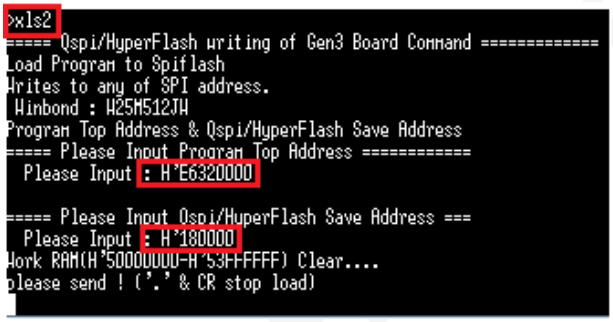
Open "bl2-hihope.srec"



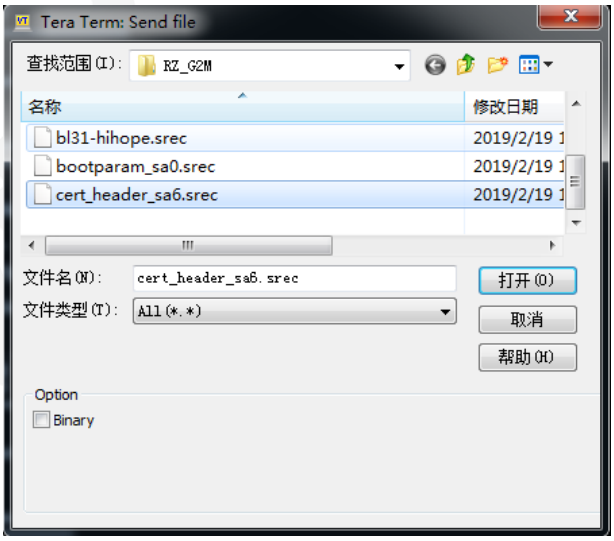
After download "bl2-hihope.srec" , input "y"

Download “bl2-hihope.srec” successfully

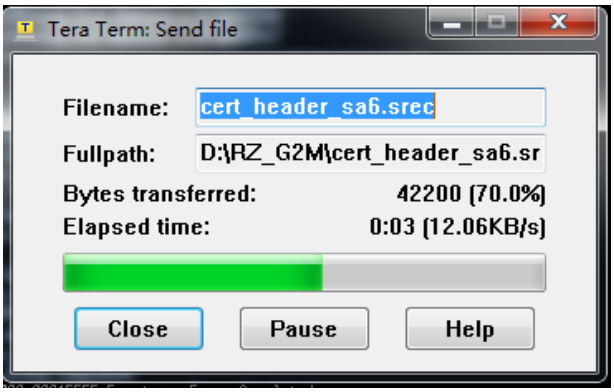
7) Write “cert_header_sa6.srec” to QSPI Flash via SCIF.



Input cmd “xls2” and address



Open “cert_header_sa6.srec”



After download “cert_header_sa6.srec” , input “y”

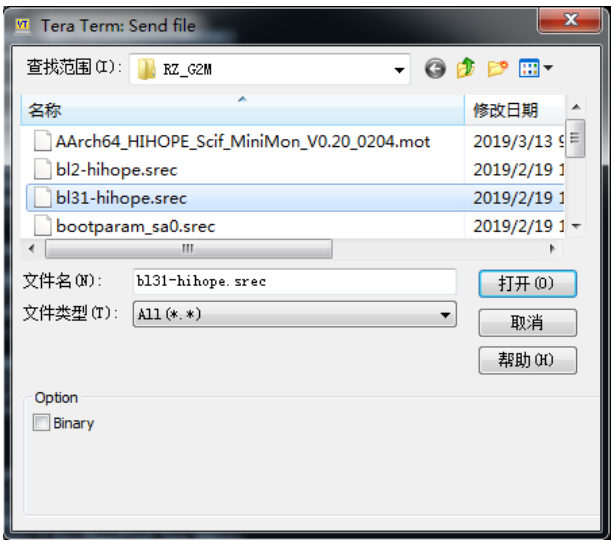
```
>xls2
===== Qspi/HyperFlash writing of Gen3 Board Command ============
Load Program to Spiflash
Writes to any of SPI address.
 Winbond : W25M512JW
Program Top Address & Qspi/HyperFlash Save Address
===== Please Input Program Top Address ============
  Please Input : H'E6320000

===== Please Input Qspi/HyperFlash Save Address ===
  Please Input : H'180000
Work RAM(H'50000000-H'53FFFFFF) Clear....
please send ! ('.' & CR stop load)
SPI Data Clear(H'FF) Check :H'00180000-0018FFFF Erasing..Erase Completed
SAVE SPI-FLASH..... complete!

======= Qspi/HyperFlash Save Information  ================
 SpiFlashMemory Stat Address : H'00180000
 SpiFlashMemory End Address  : H'00184E67
============================================================
>
```

Download “cert_header_sa6.srec” successfully

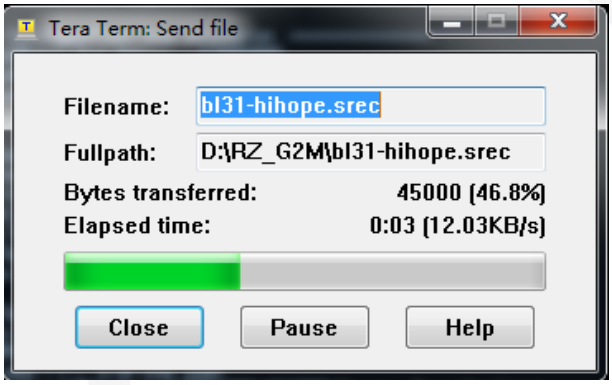
8) Write “bl31-hihope.srec” to QSPI Flash via SCIF.

```
>xls2
===== Qspi/HyperFlash writing of Gen3 Board Command ============
Load Program to Spiflash
Writes to any of SPI address.
 Winbond : W25M512JW
Program Top Address & Qspi/HyperFlash Save Address
===== Please Input Program Top Address ============
  Please Input : H'44000000

===== Please Input Qspi/HyperFlash Save Address ===
  Please Input : H'1C0000
Work RAM(H'50000000-H'53FFFFFF) Clear....
please send ! ('.' & CR stop load)
```

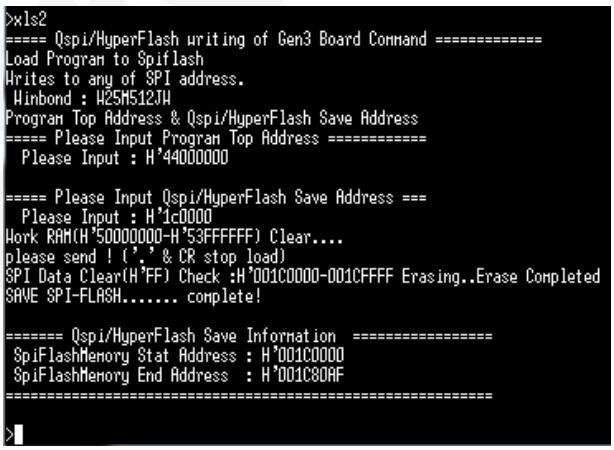
Input cmd “xls2” and address



Open “bl31-hihope.srec”

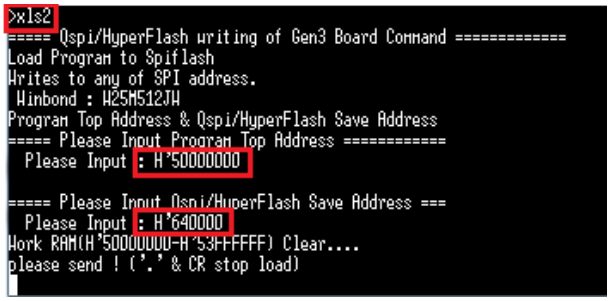


After download “bl31-hihope.srec” , input “y”

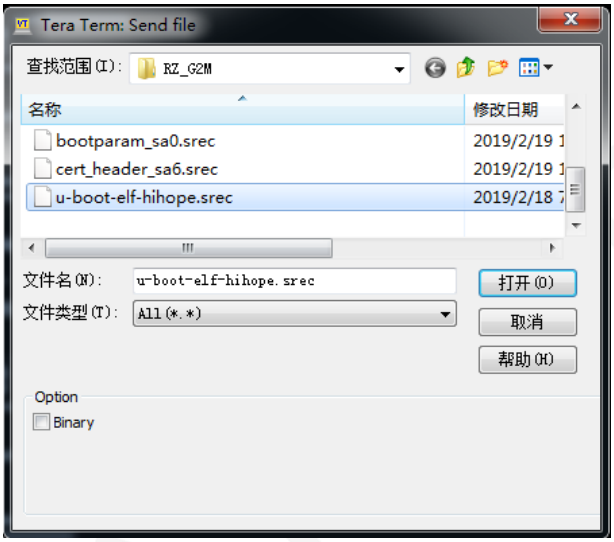


Download “bl31-hihope.srec” successfully

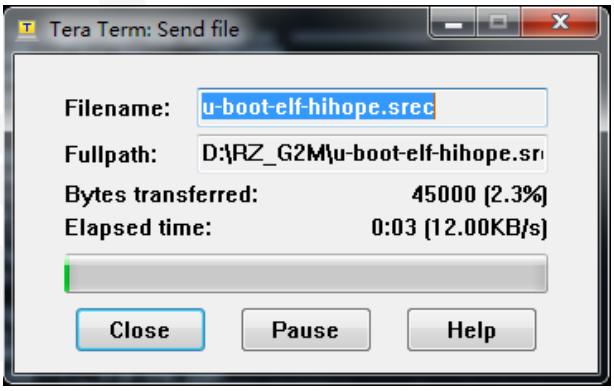
9) Write “u-boot-elf-hihope.srec” to QSPI Flash via SCIF.



Input cmd "xls2" and address



Open "u-boot-elf-hihope.srec"

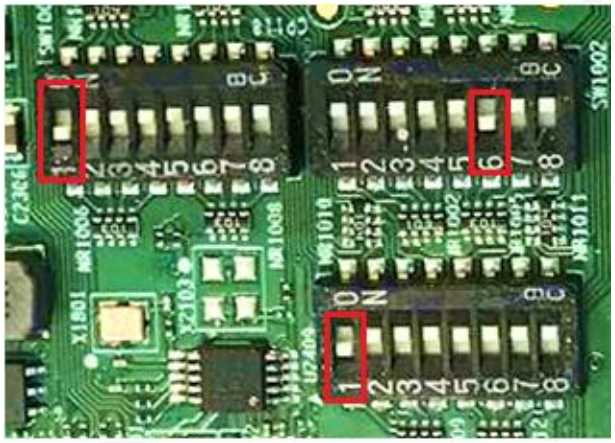


After download "u-boot-elf-hihope.srec" , input "y"

```
>xls2
===== Qspi/HyperFlash writing of Gen3 Board Command ============
Load Program to Spiflash
Writes to any of SPI address.
 Winbond : W25M512JW
Program Top Address & Qspi/HyperFlash Save Address
===== Please Input Program Top Address ============
  Please Input : H'50000000

===== Please Input Qspi/HyperFlash Save Address ===
  Please Input : H'640000
Work RAM(H'50000000-H'53FFFFFF) Clear....
please send ! ('.' & CR stop load)
SPI Data Clear(H'FF) Check :H'00640000-006EFFFF Erasing.....Erase Completed
SAVE SPI-FLASH..... complete!

======= Qspi/HyperFlash Save Information  ================
 SpiFlashMemory Stat Address : H'00640000
 SpiFlashMemory End Address  : H'006E2233
=========================================================

>
```

Download “u-boot-elf-hihope.srec” successfully

After completing steps 1 to 9, IPL and u-boot were successfully downloaded. Turn off board and set switches to SPI Boot mode.



Set red switches to “off” to enter SPI Boot mode

Turn on board and u-boot can run normally on board:

The log of BL2

```
[    0.004568] NOTICE:  BL2: Initial Program Loader(Rev.1.0.23)
[    0.010183] NOTICE:  BL2: PRR is R-Car M3 Ver.1.1 / Ver.1.2
[    0.015719] NOTICE:  BL2: Board is HiHope Rev.1.0
[    0.020380] NOTICE:  BL2: Boot device is QSPI Flash(40MHz)
[    0.025820] NOTICE:  BL2: LCM state is CM
[    0.029802] NOTICE:  BL2: DDR3200(rev.0.34)
[    0.041309] NOTICE:  BL2: [COLD_BOOT]
[    0.047407] NOTICE:  BL2: DRAM Split is 2ch
[    0.050102] NOTICE:  BL2: QoS is default setting(rev.0.17)
[    0.055545] NOTICE:  BL2: DRAM refresh interval 1.95 usec
[    0.060961] NOTICE:  BL2: Lossy Decomp areas
[    0.065136] NOTICE:       Entry 0: DCMPAREACRAx:0x80000540 DCMPAREACRBx:0x570
[    0.072220] NOTICE:       Entry 1: DCMPAREACRAx:0x40000000 DCMPAREACRBx:0x0
[    0.079132] NOTICE:       Entry 2: DCMPAREACRAx:0x20000000 DCMPAREACRBx:0x0
[    0.086047] NOTICE:  BL2: v1.5(release):7f126e2-dirty
[    0.091056] NOTICE:  BL2: Built : 05:30:26, Feb 19 2019
[    0.096243] NOTICE:  BL2: Normal boot
[    0.099888] NOTICE:  BL2: dst=0xe631ed00 src=0x8180000 len=512(0x200)
[    0.106372] NOTICE:  BL2: dst=0x43f00000 src=0x8180400 len=6144(0x1800)
[    0.114100] NOTICE:  BL2: dst=0x44000000 src=0x81c0000 len=65536(0x10000)
[    0.133018] NOTICE:  BL2: dst=0x50000000 src=0x8640000 len=1048576(0x100000)
[    0.353278] NOTICE:  BL2: Booting BL31
```

The log of U-Boot

```
U-Boot 2018.09 (Feb 17 2019 - 22:58:22 +0000)

CPU: Renesas Electronics R8A7796 rev 1.1/rev 1.2
Model: HOPERUN HiHope board based on r8a7796
DRAM:  3.9 GiB
Bank #0: 0x048000000 - 0x0bfffffff, 1.9 GiB
Bank #1: 0x600000000 - 0x67fffffff, 2 GiB

MMC:   sd@ee100000: 0, sd@ee160000: 1
Loading Environment from MMC... communication out of sync
communication out of sync
communication out of sync
communication out of sync
communication out of sync
communication out of sync
communication out of sync
communication out of sync
communication out of sync
communication out of sync
communication out of sync
communication out of sync
OK
In:    serial@e6e88000
Out:   serial@e6e88000
Err:   serial@e6e88000
Net:   eth0: ethernet@e6800000
Hit any key to stop autoboot:  1
```

Make Micro SD Card startup images, follow these steps:

1) Use fdisk create two partitions.

```
Welcome to fdisk (util-linux 2.27.1).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.


Command (m for help): p
Disk /dev/sdc: 62.5 GiB, 67108864000 bytes, 131072000 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x8c0e9c44

Device     Boot  Start       End   Sectors  Size Id Type
/dev/sdc1         2048    133119    131072   64M 83 Linux
/dev/sdc2       133120 131071999 130938880 62.4G 83 Linux
```

2) Copy Image and dtb file to /dev/sd*1.

```
-rw-r--r--  1 run  run  28355072 Jul  3 04:10 Image
-rw-r--r--  1 run  run     30943 Jul  3 04:10 Image-r8a7796-hihope.dtb
```

Build image file path: build/tmp/deploy/images/hihope-rzg2m/

Note: Image and dtb file name should be the same as settings in environment variables.

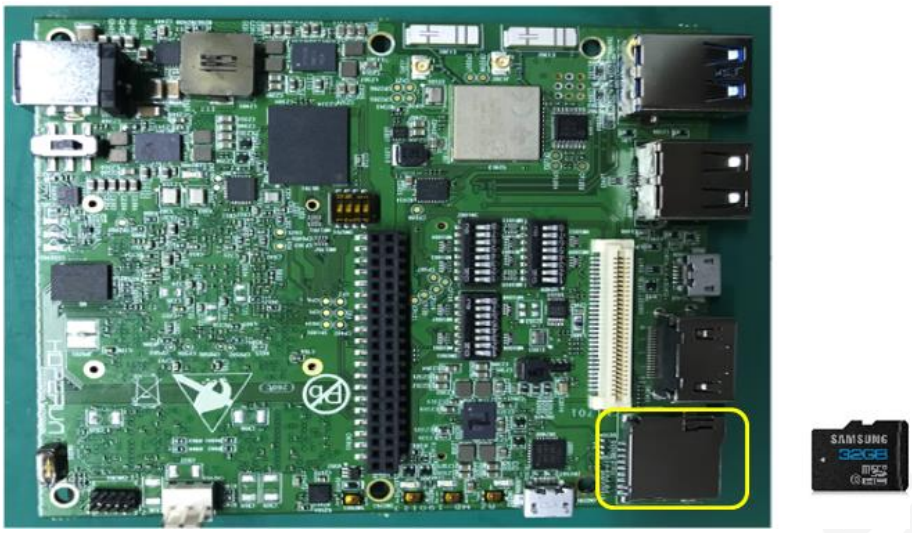
```
Error: ethernet@e6800000 address not set.
eth-1: ethernet@e6800000
Hit any key to stop autoboot:  0
=>
=>
=>
=>
=>
=> print
baudrate=115200
bootargs=root=/dev/mmcblk1p2 rootwait
bootcmd=fatload mmc 0:1 0x48080000 Image; fatload mmc 0:1 0x48000000 Image-r8a7796-hihope.dtb booti 0x48080000 - 0x48000000
bootdelay=2
fdt_high=0xffffffffffffffff
fdtcontroladdr=bfe0bc08
initrd_high=0xffffffffffffffff
loadaddr=0x58000000
stderr=serial@e6e88000
stdin=serial@e6e88000
stdout=serial@e6e88000
usb_pgood_delay=2000
ver=U-Boot 2018.09 (Feb 17 2019 - 22:58:22 +0000)

Environment size: 433/131068 bytes
=>
```

3) Copy file system to /dev/sd*2.

But this step was not tested successfully. At present, the available file system is made by “Win32 diskimager-1.0.0-install.exe” tool.

Download image to the board, follow these steps:

4) Turn off board and insert Micro SD Card (include kernel image、dtb and file system) into Micro SD card slot.



5) Turn on board and u-boot can boot Linux from TF card.

```
Hit any key to stop autoboot:  0
18926080 bytes read in 791 ms (22.8 MiB/s)
70298 bytes read in 3 ms (22.3 MiB/s)
## Flattened Device Tree blob at 48000000
   Booting using the fdt blob at 0x48000000
   Using Device Tree in place at 0000000048000000, end 0000000048014299

Starting kernel ...

[    0.000000] Booting Linux on physical CPU 0x0
[    0.000000] Linux version 4.14.75-ltsi-yocto-standard (oe-user@oe-host) (gcc version 7.2.1 20171011 (Linaro GCC 7.2-2017.11))
#1 SMP PREEMPT Thu Apr 18 00:17:37 UTC 2019
[    0.000000] Boot CPU: AArch64 Processor [411fd073]
[    0.000000] Machine model: Renesas Salvator-X 2nd version board based on r8a7796
[    0.000000] efi: Getting EFI parameters from FDT:
[    0.000000] efi: UEFI not found.
[    0.000000] Reserved memory: created CMA memory pool at 0x0000000057000000, size 16 MiB
[    0.000000] OF: reserved mem: initialized node linux,adsp@57000000, compatible id shared-dma-pool
[    0.000000] Reserved memory: created CMA memory pool at 0x0000000058000000, size 384 MiB
[    0.000000] OF: reserved mem: initialized node linux,cma@58000000, compatible id shared-dma-pool
[    0.000000] Reserved memory: created CMA memory pool at 0x0000000070000000, size 256 MiB
[    0.000000] OF: reserved mem: initialized node linux,multimedia@70000000, compatible id shared-dma-pool
[    0.000000] NUMA: No NUMA configuration found
[    0.000000] NUMA: Faking a node at [mem 0x0000000000000000-0x000000067fffffff]
[    0.000000] NUMA: NODE_DATA [mem 0x67ff91400-0x67ff92eff]
```

6) Waiting for Linux boot to complete.

```
[  OK  ] Started Avahi mDNS/DNS-SD Stack.
[  OK  ] Started Login Service.
[  OK  ] Listening on Load/Save RF Kill Switch Status /dev/rfkill Watch.
         Starting pvrinit.service...
[  OK  ] Reached target Network.
         Starting Permit User Sessions...
         Starting Network Name Resolution...
[  OK  ] Started pvrinit.service.
[  OK  ] Started Permit User Sessions.
[  OK  ] Started Serial Getty on ttySC0.
[  OK  ] Started Getty on tty1.
[  OK  ] Reached target Login Prompts.
[  OK  ] Started Network Name Resolution.
[  OK  ] Reached target Host and Network Name Lookups.
[  OK  ] Reached target Multi-User System.
         Starting Update UTMP about System Runlevel Changes...
[  OK  ] Started Weston Wayland compositor.
[  OK  ] Started Update UTMP about System Runlevel Changes.
[   21.114200] audit: type=1006 audit(1550746921.107:2): pid=3627 uid=0 old-auid=4294967295 auid=0 tty=(none) old-ses=4294967295
ses=1 res=1

Poky (Yocto Project Reference Distro) 2.4.3 hihope ttySC0

hihope login:
```

User name: root, password is empty