Install Utility

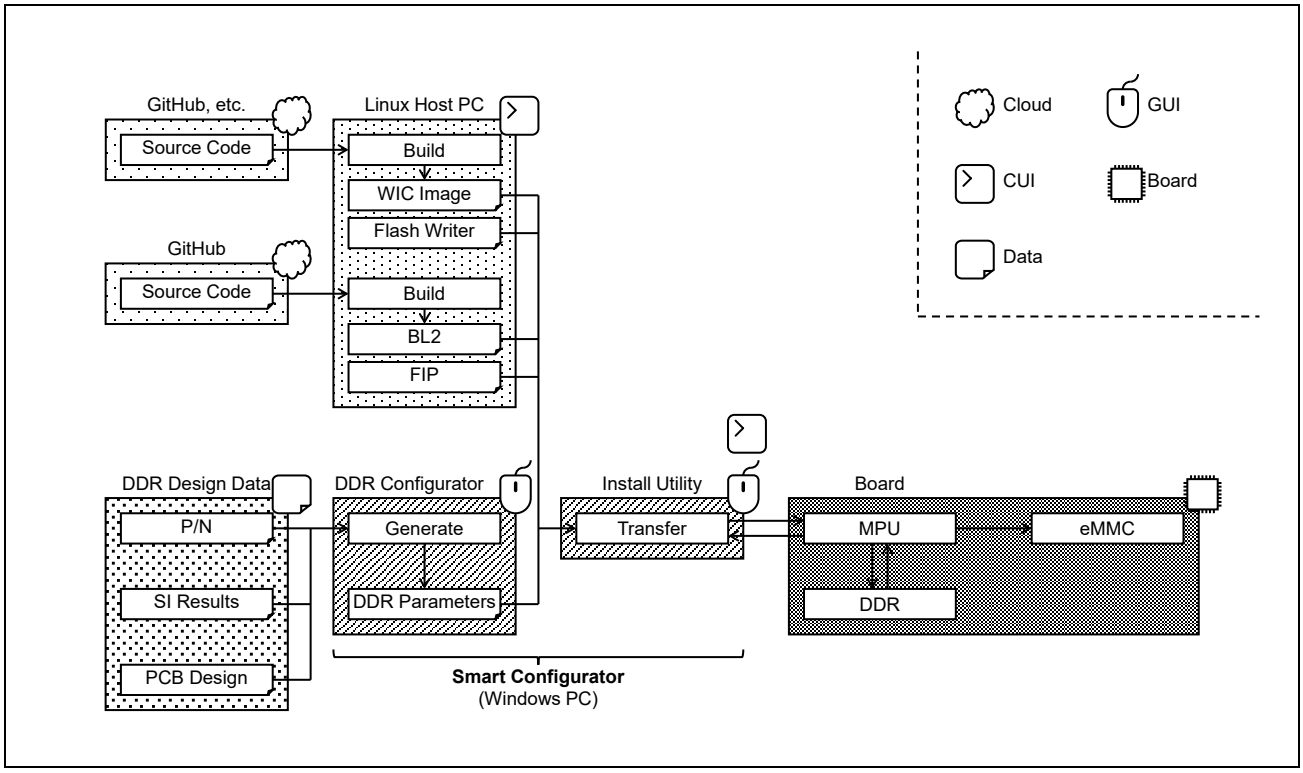
RZ Family Install Utility Start-up Guide

Introduction

This document shows how to prepare software for "Install Utility" and easy usage of "Install Utility" on Windows PC.

"Install Utility" is a GUI / CUI tools to install Linux to eMMC.

- Install Linux with less steps
- Install Linux with USB cable
- Install Linux with GUI and CUI



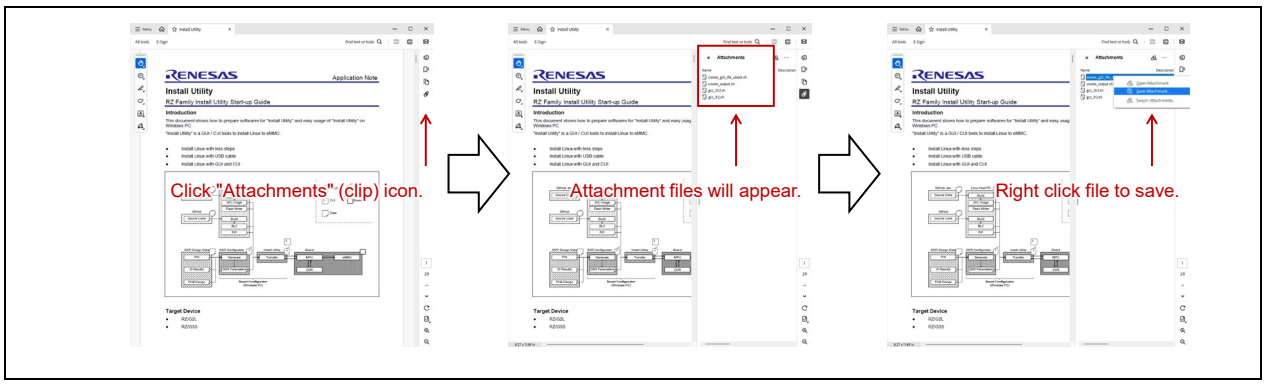
Target Device

- RZ/G2L
- RZ/G3S

Note

How to extract files in this PDF file

1. Open this PDF file using "Adobe Acrobat Reader" or "Adobe Acrobat".
2. Click "Attachments" (clip) icon.
3. Attachment files will appear.
4. Right click file to save.



Contents

1. Requirements

"DDR Tools" can be used for RZ/G2L-EVKIT and RZ/G3S-EVKIT in the current situation.

- EVKIT
  - RZ/G2L-EVKIT (RTK9744L23S01000BE)
  - RZ/G3S-EVKIT (RTK9845S33S01000BE)
- Linux Host PC and Windows PC
  - See RZ/G2L, RZ/G2LC and RZ/G2UL-EVKIT Linux Start-up Guide.
  - See SMARC EVK of RZ/G3S Linux Start-up Guide.

2. Supported Linux Versions

Supported Linux versions are shown below.

Table 2.1 Supported Linux Versions

MPU	Linux Version
RZ/G2L	VLP 4.0.0
RZ/G3S	VLP 3.0.7 update 3

3. Preparation

Please prepare "Install Utility" for the target device.

It is recommended to build "Install Utility" for the target device on Linux Host PC.

3.1 Common Items

3.1.1 Download Android SDK Platform Tools

It is necessary to install "Android SDK Platform Tools" and "Google USB Driver" on Windows PC.

- SDK Platform Tools
- Google USB Driver

3.1.2 Install Android SDK Platform Tools

In this document, "Android SDK Platform Tools" is extracted in `C:\tmp` folder. So, executable files are included in `C:\tmp\platform-tools`.

In this document, "Google USB Driver" is extracted in `C:\tmp` folder. To install "Google USB Driver", you should right click `android_winusb.inf` and select "Install".

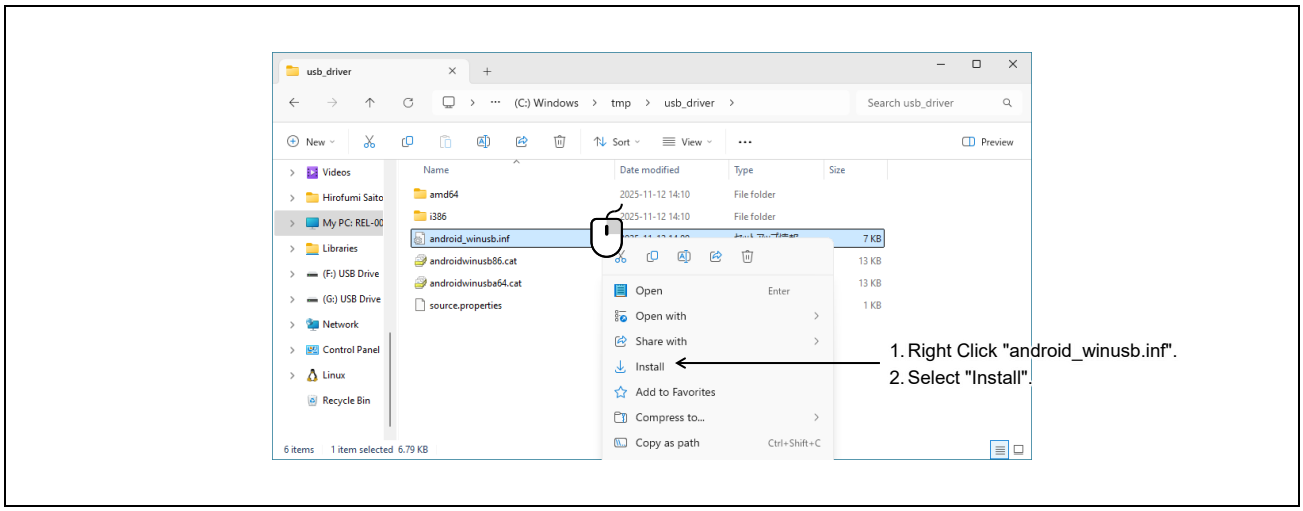


Figure 3.1 Install Driver

3.2 RZ/G2L

3.2.1 Prepare SDK for VLP 4.0.0

Please see SMARC EVK of RZ/G2L, RZ/G2LC Linux Start-up Guide and install SDK on Linux host PC.

3.2.2 Prepare U-Boot and TF-A

3.2.2.1 Download U-Boot

Download source code of "U-Boot" from GitHub.

Code 3.1 Set Environments on Linux Host PC and Download U-Boot

```
$ export MPU_NAME=rzg2l
$ export COMPILE_DIR=$(pwd)/install_utility_${MPU_NAME}
$ mkdir -pv ${COMPILE_DIR}
$ cd ${COMPILE_DIR}

$ git clone https://github.com/renesas-rz/rz_tool_u-boot.git
$ cd ${COMPILE_DIR}/rz_tool_u-boot
$ git checkout -f v1.0
```

3.2.2.2 Download TF-A

Download source code of "U-Boot" from GitHub.

Code 3.2 Download TF-A

```
$ cd ${COMPILE_DIR}

$ git clone https://github.com/renesas-rz/rzg_trusted-firmware-a.git
$ cd ${COMPILE_DIR}/rzg_trusted-firmware-a
$ git checkout v2.9/rz
```

3.2.2.3 Build U-Boot

In this document, it is assumed that the SDK is installed in the `/opt/poky/3.1.31/environment-setup-aarch64-poky-linux` directory.

Code 3.3 Build U-Boot

```
$ cd ${COMPILE_DIR}/rz_tool_u-boot

$ source /opt/poky/3.1.31/environment-setup-aarch64-poky-linux
$ make smarc-rzg2l_defconfig
$ make
```

3.2.2.4 Build TF-A

The create_output.sh and gcc_9.2.sh scripts are attached to this PDF. Please copy create_output.sh and gcc_9.2.sh to ${COMPILE_DIR} and use it.

Code 3.4 Download Cross Compiler and Build TF-A

```
$ cd ${COMPILE_DIR}


$ wget 'https://developer.arm.com/-/media/Files/downloads/gnu-a/9.2-
2019.12/binrel/gcc-arm-9.2-2019.12-x86_64-aarch64-none-linux-gnu.tar.xz'
$ tar -xvf gcc-arm-9.2-2019.12-x86_64-aarch64-none-linux-gnu.tar.xz


$ export TOOLCHAIN_DIR=${COMPILE_DIR}/gcc-arm-9.2-2019.12-x86_64-aarch64-none-
linux-gnu
$ bash -x ./create_output.sh all
```

3.2.2.5 Copy Files to PC

After building TF-A and U-Boot, SREC files are created in ${COMPILE_DIR}/output/binary_packed directory.

Copy these files below to Windows PC.

Table 3.1 Files

File Names	Descriptions
fip-smarc-rzg2l_pmic.srec	FIP (Firmware Image Package)
bl2_bp-smarc-rzg2l_pmic.srec	BL2 (Boot Loader Stage 2)

3.2.3 Prepare "Flash Writer" and WIC Image

3.2.3.1 Build VLP

It is necessary to build Linux to build "Flash Writer" in the current situation.

1. Generate DDR parameters using "PCB Design Guideline for DDR4/DDR3L (Including DDR config generation tool)".
2. Check DDR parameters using "DDR Tools" in "RZ Smart Configurator".
3. After passing DDR check items, please use these DDR parameters.
4. Build VLP first.
5. Copy them to directories.

- Copy `param_mc.c` to the directories `build/tmp/work/smarc_rzg2l-poky-linux/trusted-firmware-a/2.9+git/git/plat/renesas/rz/soc/g2l/drivers/ddr` and `build/tmp/work/smarc_rzg2l-poky-linux/flash-writer/1.06+git/git/ddr/g2l`.
- Copy `param_swizzle.c` to the directories `build/tmp/work/smarc_rzg2l-poky-linux/trusted-firmware-a/2.9+git/git/plat/renesas/rz/common/drivers/ddr` and `build/tmp/work/smarc_rzg2l-poky-linux/flash-writer/1.06+git/git/ddr/common`.

6. Update `rz_board.mk` in `build/tmp/work/smarc_rzg2l-poky-linux/trusted-firmware-a/2.9+git/git/plat/renesas/rz/board/smarc_pmic_2/rz_board.mk`.

Code 3.5 rz_board.mk

```
DDR_SOURCES += plat/renesas/rz/soc/${PLAT}/drivers/ddr/param_mc.c \
                  plat/renesas/rz/common/drivers/ddr/param_swizzle.c
```

7. After that, please build VLP again.

Code 3.6 Build VLP Again

```
$ MACHINE=smarc-rzg2l bitbake flash-writer -c compile -f
$ MACHINE=smarc-rzg2l bitbake flash-writer -c deploy
$ MACHINE=smarc-rzg2l bitbake trusted-firmware-a -c compile -f
$ MACHINE=smarc-rzg2l bitbake trusted-firmware-a -c deploy
$ MACHINE=smarc-rzg2l bitbake firmware-pack -c cleansstate
$ MACHINE=smarc-rzg2l bitbake firmware-pack
```

3.2.3.2 Copy Files to PC

After building VLP, WIC file and MOT file are created in `build/tmp/deploy/images/smarc-rzg2l` directory. Please transfer these files to Windows PC.

Table 3.2 Files

File Names	Descriptions
`Flash_Writer_SCIF_RZG2L_SMARC_PMIC_DDR4_2GB_1PCS.mot`	Flash Writer
`core-image-minimal-smarc-rzg2l.rootfs.wic.gz`	rootfs

3.2.4 Prepare Files

Finally, please prepare these files below on Windows PC.

Table 3.3 Files

File Names	Descriptions
fip-smarc-rzg2l_pmic.srec	FIP (Firmware Image Package)
bl2_bp-smarc-rzg2l_pmic.srec	BL2 (Boot Loader Stage 2)
Flash_Writer_SCIF_RZG2L_SMARC_PMIC_DDR4_2GB_1PCS.mot	Flash Writer
core-image-minimal-smarc-rzg2l.rootfs.wic.gz	rootfs

3.3 RZ/G3S

3.3.1 Prepare SDK for VLP 3.0.7

Please see "SMARC EVK of RZ/G3S Linux Start-up Guide" and install SDK on Linux host PC.

3.3.2 Prepare U-Boot and TF-A

3.3.2.1 Download U-Boot

Download source code of "U-Boot" from GitHub.

Code 3.7 Set Environments on Linux Host PC and Download U-Boot

```
$ export MPU_NAME=rzg3s
$ export COMPILE_DIR=$(pwd)/install_utility_${MPU_NAME}
$ mkdir -pv ${COMPILE_DIR}
$ cd ${COMPILE_DIR}

$ git clone https://github.com/renesas-rz/rz_tool_u-boot.git
$ cd ${COMPILE_DIR}/rz_tool_u-boot
$ git checkout -f v1.1_RZ/G3S
```

3.3.2.2 Download TF-A

Download source code of "TF-A" from GitHub.

Code 3.8 Download TF-A

```
$ cd ${COMPILE_DIR}

$ git clone https://github.com/renesas-rz/rzg_trusted-firmware-a.git
$ cd ${COMPILE_DIR}/rzg_trusted-firmware-a
$ git checkout -f 2.7.0/g3s_1.0.2
```

3.3.2.3 Build U-Boot

In this document, it is assumed that the SDK is installed in the `/opt/poky/3.1.31/environment-setup-aarch64-poky-linux` directory.

Code 3.9 Build U-Boot

```
$ cd ${COMPILE_DIR}/rz_tool_u-boot

$ source /opt/poky/3.1.31/environment-setup-aarch64-poky-linux
$ make smarc-rzg3s_defconfig
$ make
```

3.3.2.4 Build TF-A

The `create_g3s_tfa_uboot.sh` and `gcc_10.3.sh` scripts are attached to this PDF. Please copy `create_g3s_tfa_uboot.sh` and `gcc_10.3.sh` to `${COMPILE_DIR}` and use it.

Code 3.10 Download Cross Compiler and Build TF-A

```
$ cd ${COMPILE_DIR}


$ wget 'https://developer.arm.com/-/media/Files/downloads/gnu-a/10.3-
2021.07/binrel/gcc-arm-10.3-2021.07-x86_64-aarch64-none-elf.tar.xz'
$ tar -xvf gcc-arm-10.3-2021.07-x86_64-aarch64-none-elf.tar.xz


$ bash -x ./create_g3s_tfa_uboot.sh
```

3.3.2.5 Copy Files to PC

After building TF-A and U-Boot, SREC files are created in `${COMPILE_DIR}/output` directory.

Copy these files below to Windows PC.

Table 3.4 Files

File Names	Descriptions
`fip-smarc-rzg3s.srec`	FIP (Firmware Image Package)
`bl2_bp_emmc-smarc-rzg3s.srec`	BL2 (Boot Loader Stage 2)

3.3.3 Prepare "Flash Writer" and WIC Image

3.3.3.1 Download "Flash Writer"

Please prepare the cross-compile environment on Linux host PC.

Download and extract cross compile environment.

Code 3.11 Set Environments on Linux Host PC and Download "Flash Writer"

```
$ export MPU_NAME=rzg3s
$ export COMPILE_DIR=$(pwd)/install_utility_${MPU_NAME}
$ mkdir -pv ${COMPILE_DIR}
$ cd ${COMPILE_DIR}


$ git clone https://github.com/renesas-rz/rz_tool_flash_writer.git
$ cd ${COMPILE_DIR}/rz_tool_flash_writer
$ git checkout -f v1.1_RZ/G3S
```

3.3.3.2 Download Cross Compiler

Code 3.12 Download Cross Compiler

```
$ cd ${COMPILE_DIR}


$ wget 'https://developer.arm.com/-/media/Files/downloads/gnu-a/10.2-
2020.11/binrel/gcc-arm-10.2-2020.11-x86_64-aarch64-none-elf.tar.xz'
$ tar -xvf gcc-arm-10.2-2020.11-x86_64-aarch64-none-elf.tar.xz
```

3.3.3.3 Build "Flash Writer"

Code 3.13 Build "Flash Writer"

```
$ cd ${COMPILE_DIR}


$ export ARCH=arm64
$ export CROSS_COMPILE=$(pwd)/gcc-arm-10.2-2020.11-x86_64-aarch64-none-
elf/bin/aarch64-none-elf-
$ export CC=${CROSS_COMPILE}gcc
$ export AS=${CROSS_COMPILE}as
$ export LD=${CROSS_COMPILE}ld
$ export AR=${CROSS_COMPILE}ar
$ export OBJDUMP=${CROSS_COMPILE}objdump
$ export OBJCOPY=${CROSS_COMPILE}objcopy


$ cd ${COMPILE_DIR}/rz_tool_flash_writer


$ make -f makefile-g3s.gcc-arm clean
$ make -f makefile-g3s.gcc-arm BOARD=RZG3S_SMARC
FILE_NAME=AArch64_output/DDR_Tool_SCIF_RZG3S_SMARC_LPDDR4
```

3.3.3.4 Copy File to PC

After building "Flash Writer", MOT file is created in `${COMPILE_DIR}/rz_tool_flash_writer/AArch64_output` directory.

Copy the file below to Windows PC.

Table 3.5 Files

File Names	Descriptions
`Flash_Writer_SCIF_RZG3S_SMARC_LPDDR4.mot`	Flash Writer

3.3.3.5 Build VLP

It is necessary to build Linux to build "Flash Writer" in the current situation.

1. Generate DDR parameters using "DDR Configurator" in "RZ Smart Configurator".
2. Check DDR parameters using "DDR Tools" in "RZ Smart Configurator".
3. After passing DDR check items, please use these DDR parameters.
4. Build VLP first.
5. Copy DDR parameter file to directory.
   - Copy `ddr_param_def_lpddr4.c` to the directories `build/tmp/work/smarc_rzg3s-poky-linux/trusted-firmware-a/v2.7+git-r0/git/plat/renesas/rz/soc/g3s/drivers/ddr`.
6. Update `rz_board.mk` in `build/tmp/work/smarc_rzg3s-poky-linux/trusted-firmware-a/v2.7+git-r0/git/plat/renesas/rz/board/g3s_smarc/rz_board.mk`.

Code 3.14 rz_board.mk

```
DDR_SOURCES += plat/renesas/rz/soc/g3s/drivers/ddr/ddr_setup_lpddr4.c      \
               plat/renesas/rz/soc/g3s/drivers/ddr/ddr_param_def_lpddr4.c \
               plat/renesas/rz/soc/g3s/drivers/ddr/ddr_retcsr_lpddr4.c     \
               plat/renesas/rz/soc/g3s/drivers/ddr/decode_streaming_message_lpddr4.c
```

7. After that, please build VLP again.

Code 3.15 Build VLP Again

```
$ MACHINE=smarc-rzg3s bitbake flash-writer -c compile -f
$ MACHINE=smarc-rzg3s bitbake flash-writer -c deploy
$ MACHINE=smarc-rzg3s bitbake trusted-firmware-a -c compile -f
$ MACHINE=smarc-rzg3s bitbake trusted-firmware-a -c deploy
$ MACHINE=smarc-rzg3s bitbake firmware-pack -c cleansstate
$ MACHINE=smarc-rzg3s bitbake firmware-pack
```

3.3.3.6 Copy File to PC

After building VLP, WIC file is created in `build/tmp/deploy/images/smarc-rzg3s` directory. Please transfer the file to Windows PC.

Table 3.6 Files

File Names	Descriptions
`core-image-minimal-smarc-rzg3s.wic.gz`	rootfs

3.3.4 Prepare Files

Finally, please prepare these files below on Windows PC.

Table 3.7 Files

File Names	Descriptions
fip-smarc-rzg3s.srec	FIP (Firmware Image Package)
bl2_bp_emmc-smarc-rzg3s.srec	BL2 (Boot Loader Stage 2)
Flash_Writer_SCIF_RZG3S_SMARC_LPDDR4.mot	Flash Writer
core-image-minimal-smarc-rzg3s.wic.gz	rootfs
ddr_param_def_lpddr4.c	DDR Parameters

4. Install Linux using "Install Utility"

4.1 Initial Settings

First, please set the directory of "Android SDK Platform Tools".

1. Execute "Smart Configurator".
2. Select "Window" and select "Preferences".
3. Click "DDR" on the left tree menu.
4. Click "Browse" button of "SDK platform" and select folder which "Android SDK Platform Tools" is included.
   - In this document, please set `C:\tmp\platform-tools`.
5. Click "Apply" button and "Apply and Close" button.
6. Close "Smart Configurator".

This procedure is only needed when launching "Smart Configurator" for the first time.

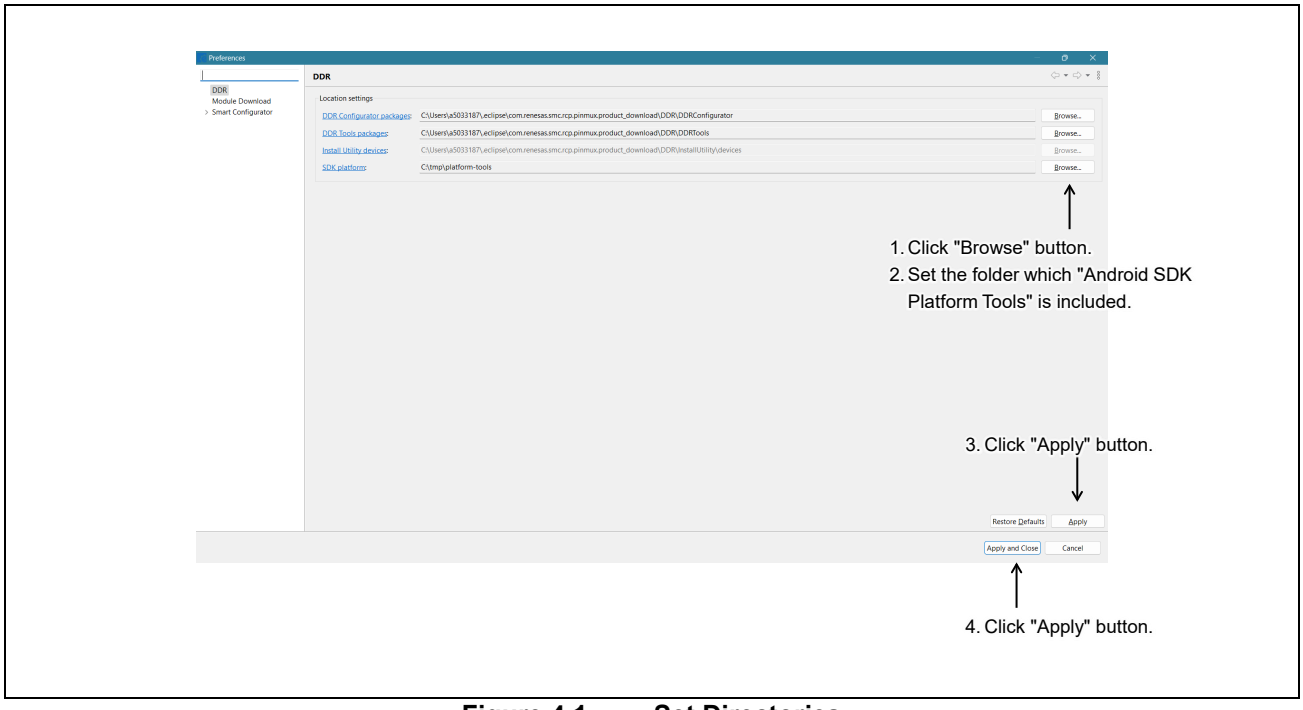


Figure 4.1 Set Directories

4.2 RZ/G2L

This section shows how to install Linux to eMMC using "Smart Configurator".

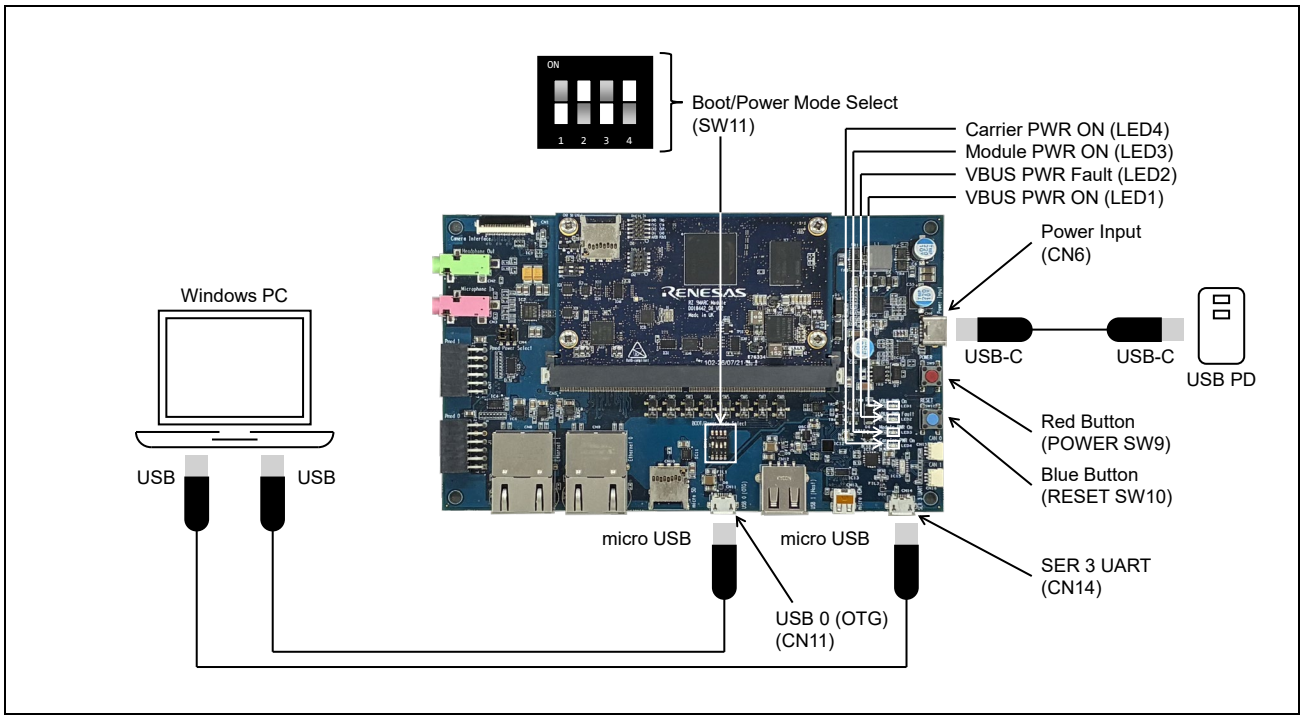
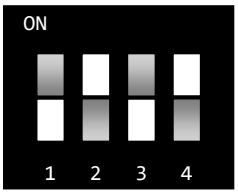


Figure 4.2 Connections for RZ/G2L

1. Before installing Linux using "Smart Configurator", please confirm that BL2, FIP, "Flash Writer" and rootfs files are prepared.
2. Switch "Boot/Power Mode Select (SW11)" to SCIF download mode.

Table 4.1 SCIF Download Mode

Mode	Switch Image	Switch Image
SCIF Download Mode		1: OFF, 2: ON, 3: OFF, 4: ON

3. Connect "Power Input (CN6)" and USB PD power supply.
   - Please confirm that the "VBUS PWR ON (LED1)" and "Module PWR ON (LED3)" are on.
4. Connect "SER 3 UART (CN14)" and USB on Windows PC.
5. Connect "USB 0 (OTG) (CN11)" and USB on Windows PC.
6. Press and hold "Red Button (POWER SW9)".
   - Confirm that the "VBUS PWR ON (LED1)", "Module PWR ON (LED3)" and "Carrier PWR ON (LED4)" are on.
7. Execute "Smart Configurator".
8. Select "Tools" and select "Install Utility".

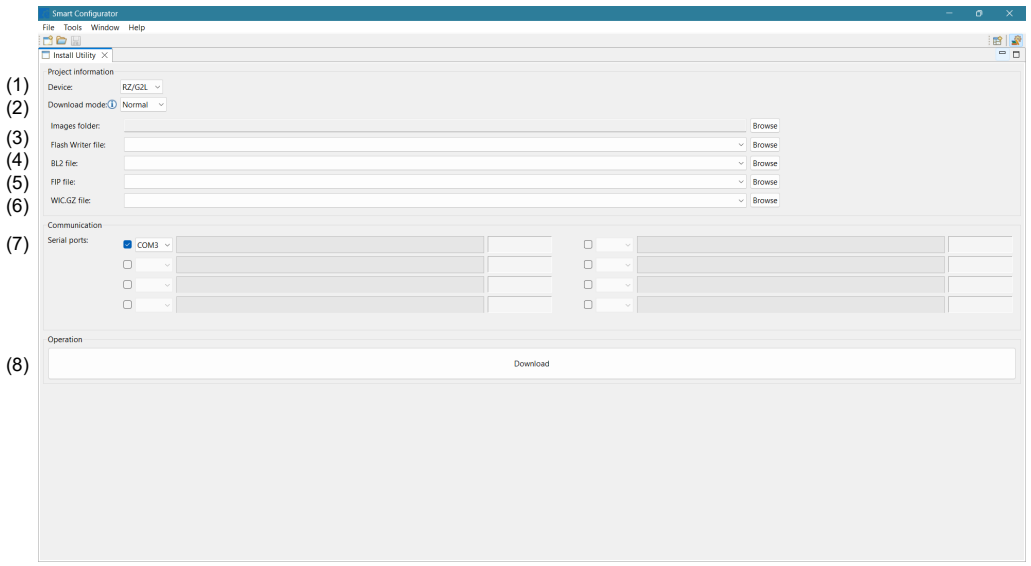


Figure 4.3 Install Utility for RZ/G2L

9. Select "Device" to "RZ/G2L" (1).
10. Select "Normal" (2).
11. Browse "Flash Writer file" which is built according to this document (3).
12. Browse "BL2 file" which is built according to this document (4).
13. Browse "FIP file" which is built according to this document (5).
14. Browse "WIC.GZ file" which is built according to "SMARC EVK of RZ/G2L, RZ/G2LC Linux Start-up Guide" (6).

15. Select "Serial ports" (7).
    - Please check serial port number using "Device Manager".

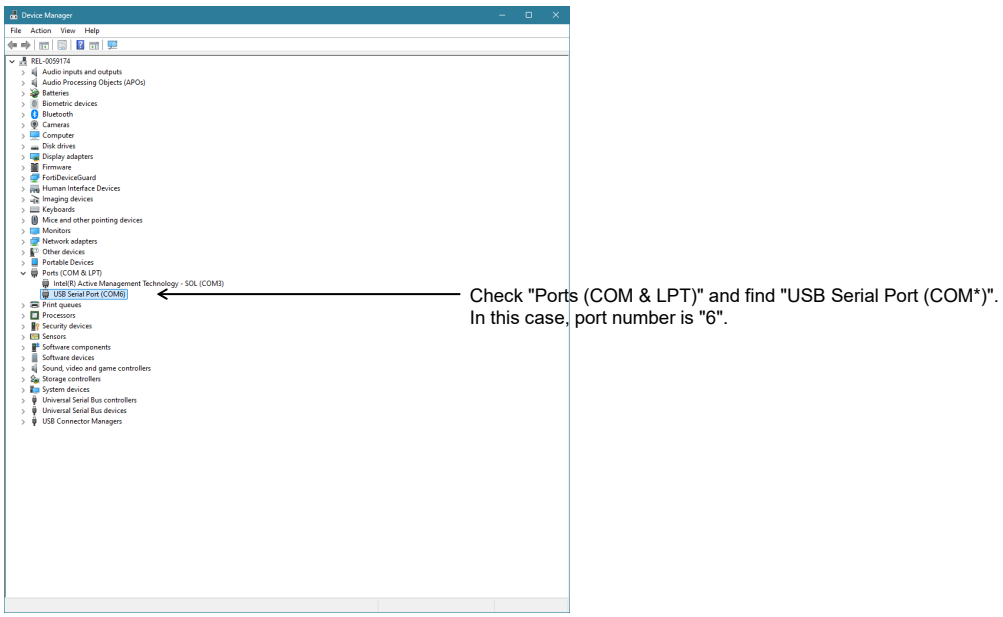


Figure 4.4 Device Manager

16. Clock "Download" button to start installing Linux (8).
17. Pop-up menu appears. Please confirm "Boot/Power Mode Select (SW11)" is SCIF download mode, and press "Blue Button (RESET SW10)" on EVK.
    - Log can be seen in "Console" tab.

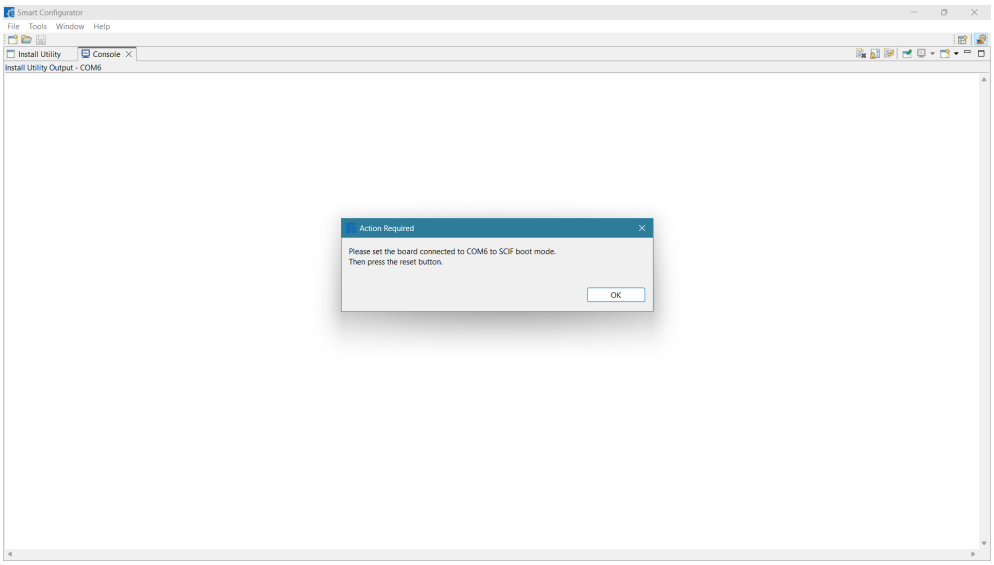
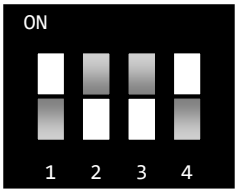


Figure 4.5 Pop-up Message - 1

18. Pop-up menu appears again. Then, please change "Boot/Power Mode Select (SW11)" to eMMC boot mode.

Table 4.2 eMMC Boot Mode

Mode	Switch Image	Switch Image
eMMC Boot Mode		1: ON, 2: OFF, 3: OFF, 4: ON

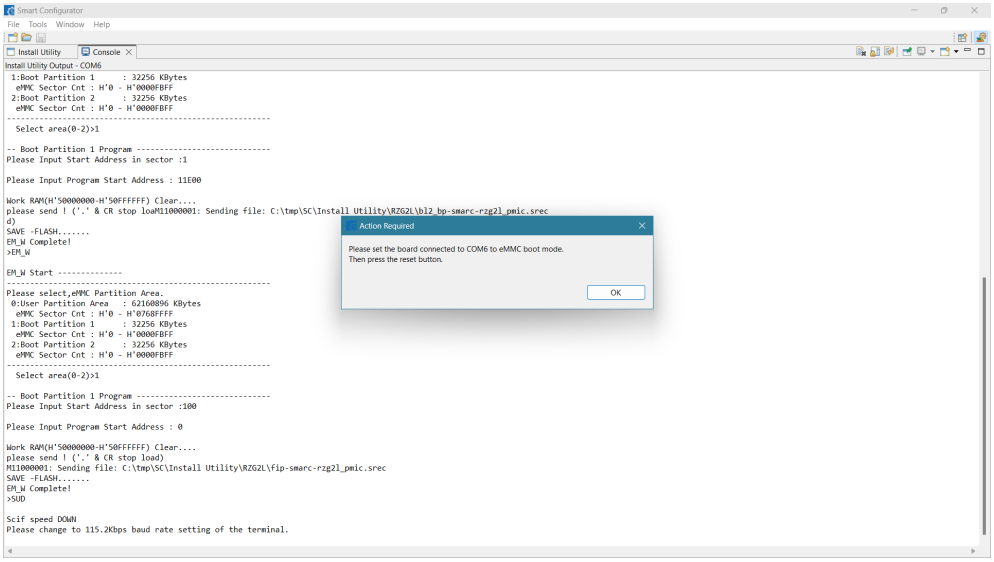


Figure 4.6 Pop-up Message – 2

19. After installing U-Boot, Linux boots from eMMC.

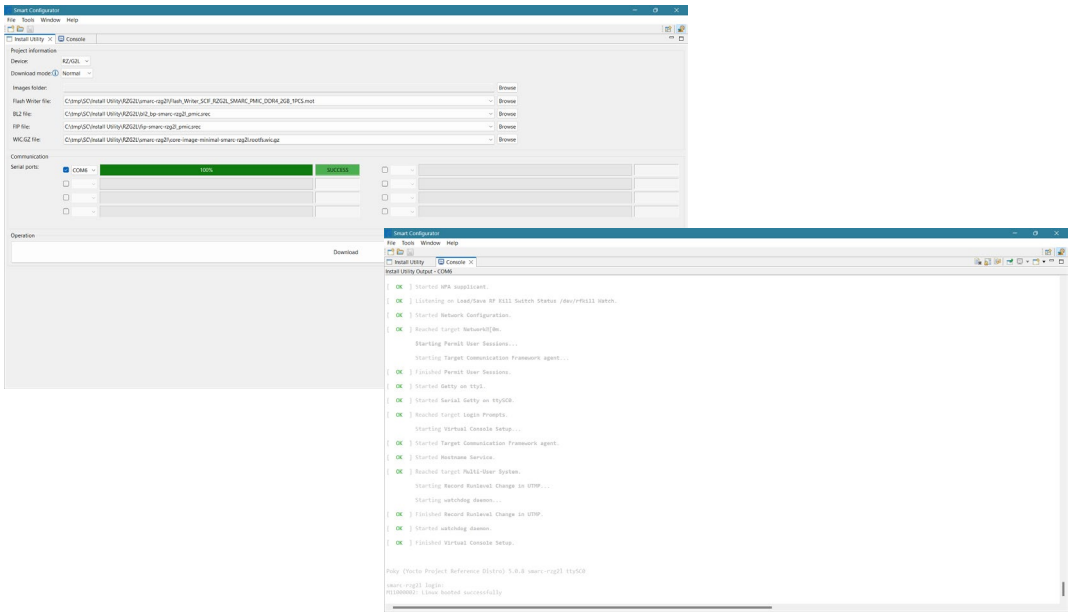


Figure 4.7 Success

4.3 RZ/G3S

This section shows how to install Linux to eMMC using "Smart Configurator".

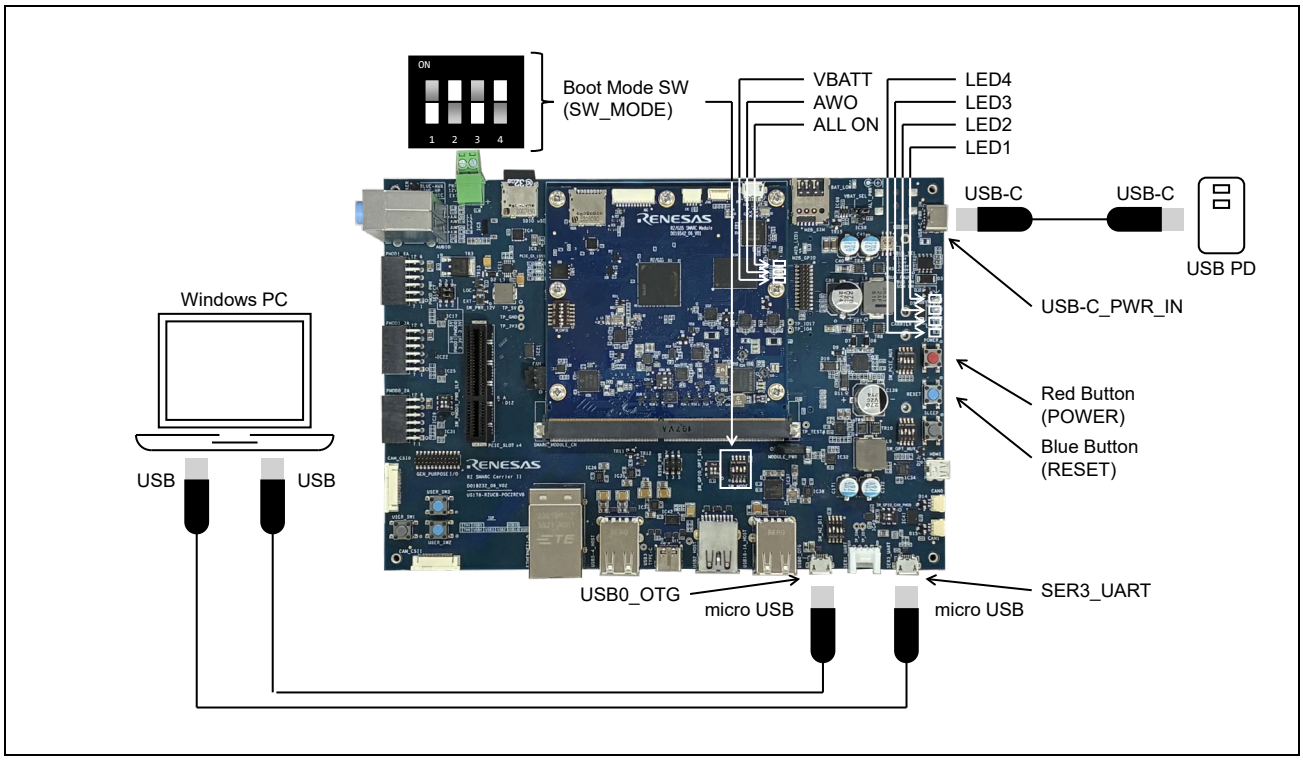
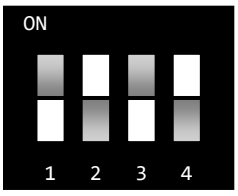


Figure 4.8 Connections for RZ/G3S

1. Before installing Linux using "Smart Configurator", please confirm that BL2, FIP, "Flash Writer" and rootfs files are prepared.
2. Switch " Boot Mode SW (SW_MODE)" to SCIF download mode.

Table 4.3 SCIF Download Mode

Mode	Switch Image	Switch Image
SCIF Download Mode		1: OFF, 2: ON, 3: OFF, 4: ON

3. Connect "USB-C_PWR_IN" and USB PD power supply.
   - Please confirm that the "LED2" and "LED3" are on.
4. Connect "SER3_UART" and USB on Windows PC.
5. Connect "USB0_OTG" and USB on Windows PC.
6. Press and hold "Red Button (POWER)".
   - Confirm that "LED4", "ALL ON" and "AWO" are on.

7. Execute "Smart Configurator".
8. Select "Tools" and select "Install Utility".

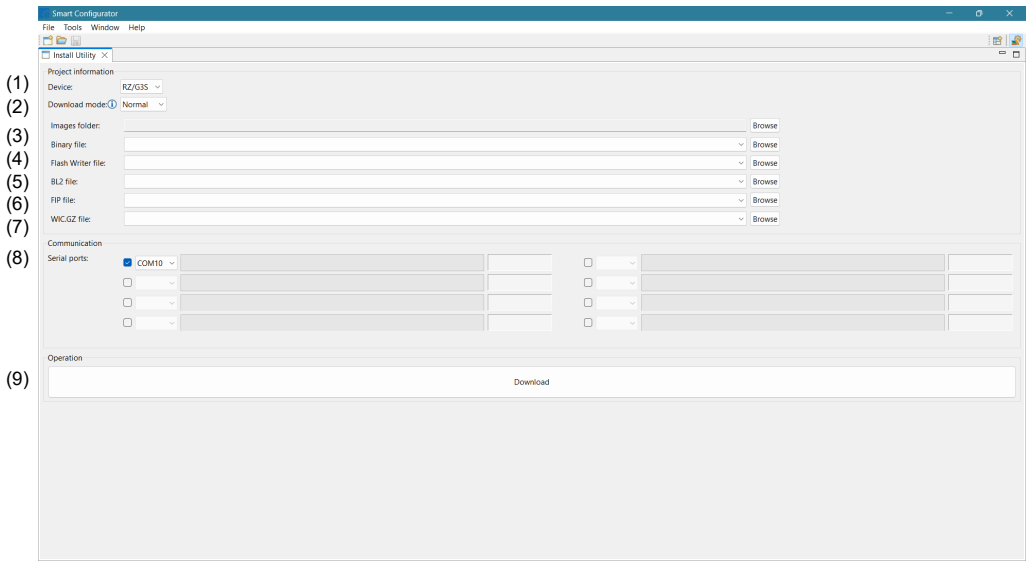


Figure 4.9 Install Utility for RZ/G3S

9. Select "Device" to "RZ/G3S" (1).
10. Select "Normal" (2).
11. Browse "DRAM Parameters file" which is generated by "DDR Configurator" (3).
12. Browse "Flash Writer file" which is built according to this document (4).
13. Browse "BL2 file" which is built according to this document (5).
14. Browse "FIP file" which is built according to this document (6).
15. Browse "WIC.GZ file" which is built according to "SMARC EVK of RZ/G3S Linux Start-up Guide" (7).

16. Select "Serial ports". (8)
    - Please check serial port number using "Device Manager".

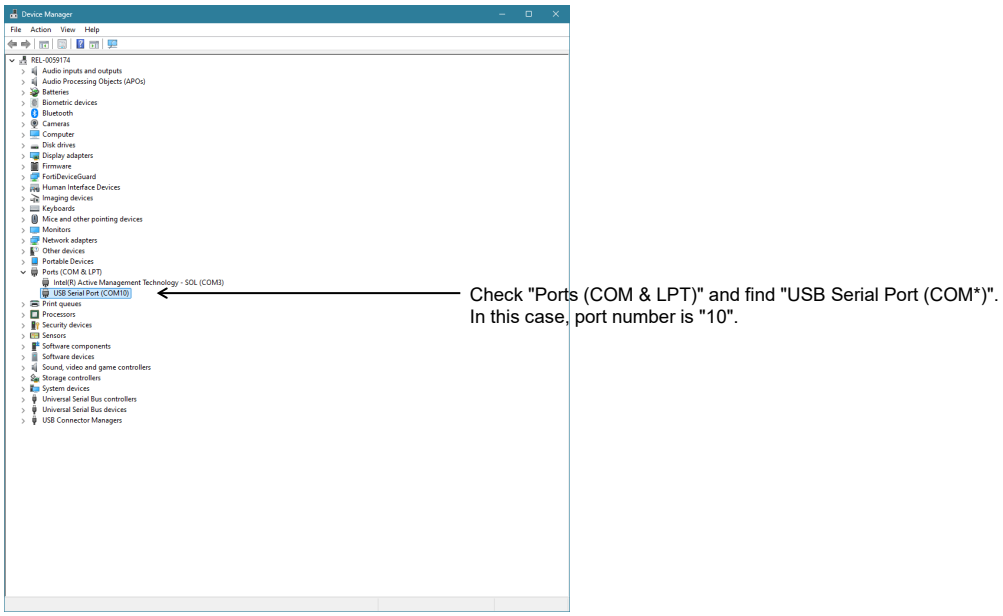


Figure 4.10 Device Manager

17. Clock "Download" button to start installing Linux (9).
18. Pop-up menu appears. Please confirm "Boot/Power Mode Select (SW11)" is SCIF download mode, and press "Blue Button (RESET)" on EVK.
    - Log can be seen in "Console" tab.

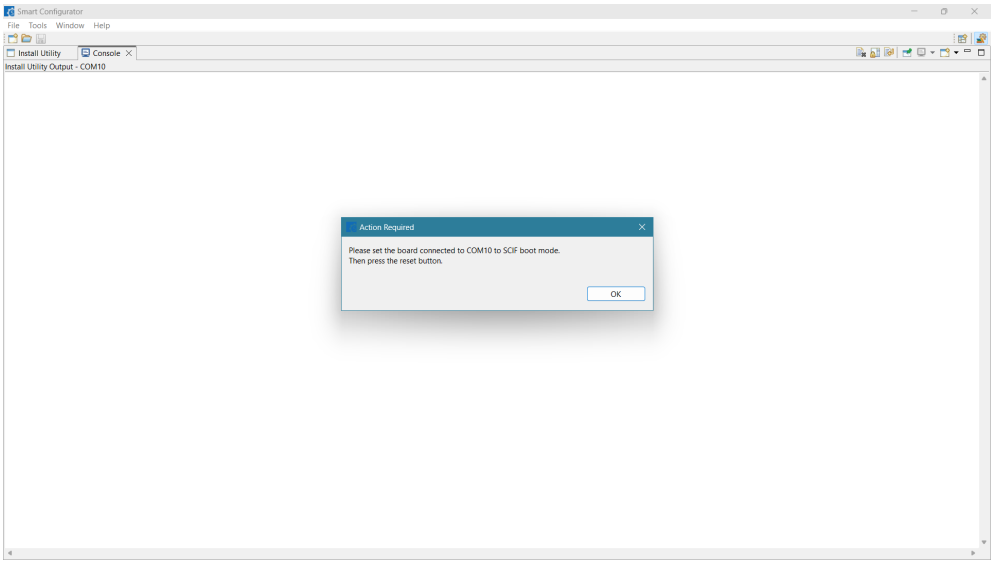
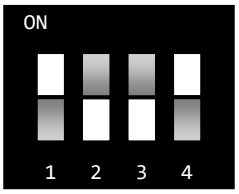


Figure 4.11 Pop-up Message – 1

19. Pop-up menu appears again. Then, please change "Boot/Power Mode Select (SW11)" to eMMC boot mode.

Table 4.4 eMMC Boot Mode

Mode	Switch Image	Switch Image
eMMC Boot Mode		1: ON, 2: OFF, 3: OFF, 4: ON

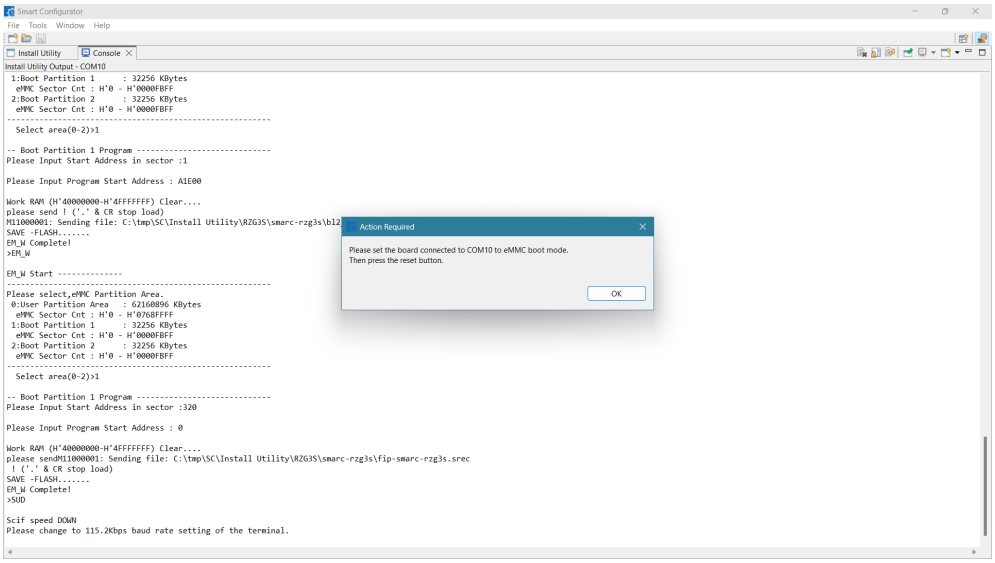


Figure 4.12 Pop-up Message - 2

20. After installing U-Boot, Linux boots from eMMC.

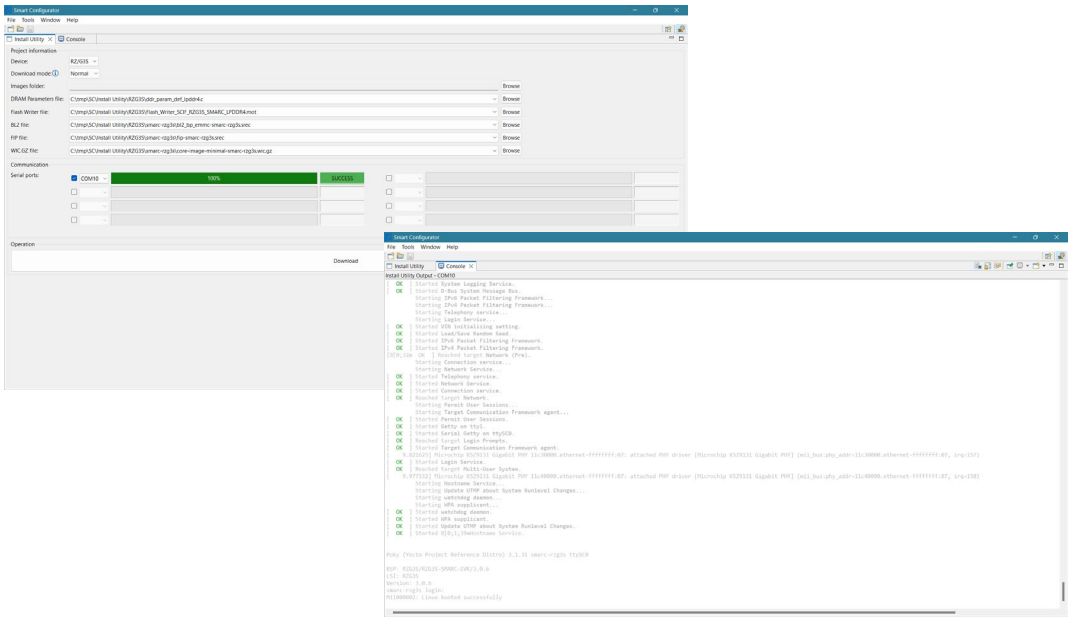


Figure 4.13 Success

Revision History

Rev.	Date	Description	
		Page	Summary
1.00	Dec. 20, 2025	-	First edition issued
1.01	Jan. 23, 2026	2	Add notes to extract files.
		All	Update code and figure captions and fix typos.

General Precautions in the Handling of Microprocessing Unit and Microcontroller Unit Products

The following usage notes are applicable to all Microprocessing unit and Microcontroller unit products from Renesas. For detailed usage notes on the products covered by this document, refer to the relevant sections of the document as well as any technical updates that have been issued for the products.

1. Precaution against Electrostatic Discharge (ESD)
   A strong electrical field, when exposed to a CMOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop the generation of static electricity as much as possible, and quickly dissipate it when it occurs. Environmental control must be adequate. When it is dry, a humidifier should be used. This is recommended to avoid using insulators that can easily build up static electricity. Semiconductor devices must be stored and transported in an anti-static container, static shielding bag or conductive material. All test and measurement tools including work benches and floors must be grounded. The operator must also be grounded using a wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions must be taken for printed circuit boards with mounted semiconductor devices.
2. Processing at power-on
   The state of the product is undefined at the time when power is supplied. The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the time when power is supplied. In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the time when power is supplied until the reset process is completed. In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the time when power is supplied until the power reaches the level at which resetting is specified.
3. Input of signal during power-off state
   Do not input signals or an I/O pull-up power supply while the device is powered off. The current injection that results from input of such a signal or I/O pull-up power supply may cause malfunction and the abnormal current that passes in the device at this time may cause degradation of internal elements. Follow the guideline for input signal during power-off state as described in your product documentation.
4. Handling of unused pins
   Handle unused pins in accordance with the directions given under handling of unused pins in the manual. The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of the LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible.
5. Clock signals
   After applying a reset, only release the reset line after the operating clock signal becomes stable. When switching the clock signal during program execution, wait until the target clock signal is stabilized. When the clock signal is generated with an external resonator or from an external oscillator during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Additionally, when switching to a clock signal produced with an external resonator or by an external oscillator while program execution is in progress, wait until the target clock signal is stable.
6. Voltage application waveform at input pin
   Waveform distortion due to input noise or a reflected wave may cause malfunction. If the input of the CMOS device stays in the area between V_{IL} (Max.) and V_{IH} (Min.) due to noise, for example, the device may malfunction. Take care to prevent chattering noise from entering the device when the input level is fixed, and also in the transition period when the input level passes through the area between V_{IL} (Max.) and V_{IH} (Min.).
7. Prohibition of access to reserved addresses
   Access to reserved addresses is prohibited. The reserved addresses are provided for possible future expansion of functions. Do not access these addresses as the correct operation of the LSI is not guaranteed.
8. Differences between products
   Before changing from one product to another, for example to a product with a different part number, confirm that the change will not lead to problems. The characteristics of a microprocessing unit or microcontroller unit products in the same group but having a different part number might differ in terms of internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.

Notice

1. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation or any other use of the circuits, software, and information in the design of your product or system. Renesas Electronics disclaims any and all liability for any losses and damages incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics hereby expressly disclaims any warranties against and liability for infringement or any other claims involving patents, copyrights, or other intellectual property rights of third parties, by or arising from the use of Renesas Electronics products or technical information described in this document, including but not limited to, the product data, drawings, charts, programs, algorithms, and application examples.
3. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
4. You shall be responsible for determining what licenses are required from any third parties, and obtaining such licenses for the lawful import, export, manufacture, sales, utilization, distribution or other disposal of any products incorporating Renesas Electronics products, if required.
5. You shall not alter, modify, copy, or reverse engineer any Renesas Electronics product, whether in whole or in part. Renesas Electronics disclaims any and all liability for any losses or damages incurred by you or third parties arising from such alteration, modification, copying or reverse engineering.
6. Renesas Electronics products are classified according to the following two quality grades: “Standard” and “High Quality”. The intended applications for each Renesas Electronics product depends on the product’s quality grade, as indicated below.
   "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; industrial robots; etc.
   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control (traffic lights); large-scale communication equipment; key financial terminal systems; safety control equipment; etc.

   Unless expressly designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not intended or authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems; surgical implantations; etc.), or may cause serious property damage (space system; undersea repeaters; nuclear power control systems; aircraft control systems; key plant systems; military equipment; etc.). Renesas Electronics disclaims any and all liability for any damages or losses incurred by you or any third parties arising from the use of any Renesas Electronics product that is inconsistent with any Renesas Electronics data sheet, user’s manual or other Renesas Electronics document.
7. No semiconductor product is absolutely secure. Notwithstanding any security measures or features that may be implemented in Renesas Electronics hardware or software products, Renesas Electronics shall have absolutely no liability arising out of any vulnerability or security breach, including but not limited to any unauthorized access to or use of a Renesas Electronics product or a system that uses a Renesas Electronics product. RENESAS ELECTRONICS DOES NOT WARRANT OR GUARANTEE THAT RENESAS ELECTRONICS PRODUCTS, OR ANY SYSTEMS CREATED USING RENESAS ELECTRONICS PRODUCTS WILL BE INVULNERABLE OR FREE FROM CORRUPTION, ATTACK, VIRUSES, INTERFERENCE, HACKING, DATA LOSS OR THEFT, OR OTHER SECURITY INTRUSION (“Vulnerability Issues”). RENESAS ELECTRONICS DISCLAIMS ANY AND ALL RESPONSIBILITY OR LIABILITY ARISING FROM OR RELATED TO ANY VULNERABILITY ISSUES. FURTHERMORE, TO THE EXTENT PERMITTED BY APPLICABLE LAW, RENESAS ELECTRONICS DISCLAIMS ANY AND ALL WARRANTIES, EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT AND ANY RELATED OR ACCOMPANYING SOFTWARE OR HARDWARE, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE.
8. When using Renesas Electronics products, refer to the latest product information (data sheets, user’s manuals, application notes, “General Notes for Handling and Using Semiconductor Devices” in the reliability handbook, etc.), and ensure that usage conditions are within the ranges specified by Renesas Electronics with respect to maximum ratings, operating power supply voltage range, heat dissipation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions, failure or accident arising out of the use of Renesas Electronics products outside of such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of Renesas Electronics products, semiconductor products have specific characteristics, such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Unless designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not subject to radiation resistance design. You are responsible for implementing safety measures to guard against the possibility of bodily injury, injury or damage caused by fire, and/or danger to the public in the event of a failure or malfunction of Renesas Electronics products, such as safety design for hardware and software, including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult and impractical, you are responsible for evaluating the safety of the final products or systems manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. You are responsible for carefully and sufficiently investigating applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive, and using Renesas Electronics products in compliance with all these applicable laws and regulations. Renesas Electronics disclaims any and all liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. Renesas Electronics products and technologies shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You shall comply with any applicable export control laws and regulations promulgated and administered by the governments of any countries asserting jurisdiction over the parties or transactions.
12. It is the responsibility of the buyer or distributor of Renesas Electronics products, or any other party who distributes, disposes of, or otherwise sells or transfers the product to a third party, to notify such third party in advance of the contents and conditions set forth in this document.
13. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
14. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

(Note1) “Renesas Electronics” as used in this document means Renesas Electronics Corporation and also includes its directly or indirectly controlled subsidiaries.
(Note2) “Renesas Electronics product(s)” means any product developed or manufactured by or for Renesas Electronics.

(Rev.5.0-1 October 2020)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan
www.renesas.com

Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.

Contact information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit:
www.renesas.com/contact/.

Trademarks (continued)

USB®: USB is a registered trademark of USB Implementers Forum, Inc.

Google®: Google is a registered trademark of Google LLC.

Android®: Android is a registered trademark of Google LLC.

Linux®: Linux is a registered trademark of Linus Torvalds in the U.S. and other countries.

Windows®: Windows is a registered trademarks or trademarks of Microsoft Corporation in the United States and other countries.

Adobe Acrobat Reader® and Adobe Acrobat®:

Adobe, Adobe Acrobat Reader and Adobe Acrobat are registered trademarks or trademarks of Adobe in the United States and other countries.

Note that in each section of this document, trademark notation of ® and ™ may be omitted.

All other trademarks and registered trademarks are the property of their respective owners.