Quick Start Guide

Renesas System Release Package

RZ/G2L-SBC, Single Board Computer

Quick Start Guide

Renesas RZ Family
RZ/G Series

Renesas Electronics
www.renesas.com

Rev.1.10 Jun.04.2025

Corporate Headquarters

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

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)

For the “Cortex” notation, it is used as follows;

— Arm® Cortex®-A55

— Arm® Cortex®-M33

Note that after this page, they may be noted as Cortex-A55 and Cortex-M33 respectively.

Examples of trademark or registered trademark used in the RZ/G2L SMARC Module Board RTK9744L23C01000BE User’s Manual: Hardware;

CoreSight™: CoreSight is a trademark of Arm Limited.

MIPI®: MIPI is a registered trademark of MIPI Alliance, Inc.

eMMC™: eMMC is a trademark of MultiMediaCard Association.

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

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

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.

Renesas RZ Family

Renesas System Release Package

Introduction

This user manual describes the unified system release package. The system release package contains supported hardware and software.

The result is a consistent experience across the different platforms. This streamlines the development effort for user applications.

Package Contents

The system release package contains the following:

- Multiple Images that are geared to general baseline use cases.
- Yocto build scripts.
- Host side tools.
- Environmental files.
- SDKs for all images
- Documentation, which includes a user manual and copyright & license information

Features

The following are the general features of the system release package.

- Architected to support multiple platforms with the same image and tools over time.
- Common frameworks
- Open-source packages using GPLv2 packages
- Carefully considered base images that allow for a quick starting point to build a product.
- Complete set of features working out of the box.
- Seamless out-of-box experience.
- Automated Yocto build scripts that can rebuild the entire package with only a few commands.
- Host tools to flash the firmware in multiple processes.
- Tools supporting both Linux and Windows workflows.
- Docker-friendly build scripts.
- Extensive documentation covering the hardware, software, and application development and deployment.

Contents

Glossary

Terms	Description
802.11 - Wi-Fi	The technical name of the standard specification for Wi-Fi is 802.11. This is also the working group that develops and maintains the standards for Wi-Fi that everyone conforms to.
ADC – Analog to digital converter	A hardware unit that converts an input analog signal to a digital value by measuring its immediate voltage at a fixed resolution.
BSP – Board Support Package	BSP is an essential software package that has bootloaders, Linux kernel, a minimal user space and programming tools, allowing the device to boot. This core software allows the system to boot into an operating system, enables all the features and allows application development.
CAN – Controller area network	This is a standardized communication protocol used widely on automotive and aerospace systems. It connects various ECU's known as nodes and uses two wires / lines as a pair carrying differential signals. This method of signaling allows long length cables to interface different systems on the machine with reliable signals. The CAN protocol has multiple specifications and is an ISO standard. It supports flexible data rates reaching as high as 8Mbps. Most automobiles have CAN networks in them, and it is a part of OBD-2 specification which is mandatory law in most of the world for automotive machines like cars.
DAC – Digital to analog converter	A hardware unit that takes digital value and exerts a corresponding analog voltage on an output line.
Firmware	For the scope of this document, the term 'firmware' refers to the low-level software that runs before an OS takes over. This includes arm trust zone, optee & u-boot at the very least. It also refers to the standalone binaries that run on the embedded real-time core like the CM33.
I2C - Inter Integrated circuit protocol:	This is a communication protocol used to implement digital communication between two devices (chips / board) using only two wires. It is a standardized specification and is used widely to implement low to medium data rate data transfers both among devices on the same circuit board as well as external add on peripheral boards. I2C can be implemented across a few meters in distance. I2C is half duplex meaning only one device can communicate at a time. Speeds range from 100 Kbps to 3Mbps while 100 / 400 Kbps are the typical operating mode. The other major advantage of this protocol is that it allows many devices to be on the same two lines reducing the cost of the interfacing. This is ideal when there are many devices like sensors that transfer limited amounts of data periodically. I2C can support up to 127 independent directly addressable devices on the same channel.
IEEE- Institute of Electrical and Electronics Engineers	IEEE is the world's largest technical professional organization dedicated to advancing technology for the benefit of humanity. It is a major technical organization covering vast fields of engineering and a major standards organization.
MCU – Micro controller unit	A micro controller unit is a self-contained unit that has the core processing as well as core memory within the same device. It often contains the core software programmed into the chip itself. This allows the device to start executing with minimal external devices / circuitry. Some microcontrollers can be powered on a mere breadboard.
MPU – Micro processing unit	An MPU is a processing unit: a CPU that contains only the processing core and interfaces for external peripherals. A microprocessor is usually a powerful CPU in its class. However, it requires a very large number of external circuitries to achieve its functionality like external memory, disk drives, etc.
PMIC – Power management IC	This is a specific chip on the board that manages multiple power supply lines at various levels. It manages the respective supplies along with sequences which control power on and power off cycles.
SBC – Single board computer	It is a standard term that means a tiny computer in the form factor of a single circuit board usually just inches in area. This board is self-sufficient in every way and can give you a usable computer with just a power supply, keyboard, mouse, and display.

SiP – System in Package	SiP is a device where multiple silicon IP's are combined to form a single device. It is one of the densest chips where the external devices like flash memory, DDR RAM and even Wi-Fi module are all packaged into a single chip. These are used in very niche application that require ultra small size and low thermal requirement.
SoC- System on Chip	A system on chip is a complete hardware platform packaged on to a single chip. It contains the CPU, internal fast memory, interrupt controllers, pin controllers, ROM memory, and a few other peripherals and sensors; all packaged into the same IC. An SoC despite the high level of integration does not necessarily power on and run by itself. Microcontrollers are often independent SoC's that can work on their own. However, SoC's often combine MPU's and MCU's into the same chip. This allows very powerful systems to be built in a compact form factor but requires external supporting peripherals like DDR RAM and flash memory and power management IC's.
SPI - Serial Peripheral interface	SPI is another standard interface used to interface other devices on the board or attaching peripheral boards. It specifies 3 wires / lines to achieve fast full duplex data transfer. Two devices can send / receive data at the same time in this protocol. The protocol is also a high-speed protocol where typical operating speeds start at 5Mbps and go over 50Mbps. This high speed allows interfacing high speed devices like memory, Wi-Fi, subsystems made of independent microcontrollers, etc. While only 3 lines are needed to interface two devices, a fourth line is used as a device selector allowing multiple devices to share the same interface. However, only two devices may communicate at a time.

1. Overview

The Renesas System Release Package is a unified software package that aims to provide an easy-to-use yet comprehensive software platform for the Renesas RZ series of SoC-based boards. It aims to provide fully functional base images for supported reference designs, along with easy-to-use development and programming tools that allow the user to quickly get started on their application development. This package aims to provide a standardized and familiar workflow for a similar experience across a variety of Renesas RZ SoC-based product platforms.

This package provides comprehensive documentation, Quick Start Guides, multiple Linux-based distribution images, automated tools and scripts, and an ongoing expansion of supported products.

1.1 Supported Distributions

The System Release package supports a set of both Yocto images and custom images to enable the user to start quickly on their embedded end application. The large collection of images in prebuilt format provides a wide set of capabilities. This release focuses on Yocto images.

1.1.1 Yocto Images

This section lists the standard Yocto images, offering a variety of configurations that cater to different embedded use cases. From a minimal bootable environment to fully graphical systems, these images provide the essential building blocks for embedded Linux development.

Table 1. Yocto images

Distribution	Image file	Version	Description
Yocto minimal	core-image-minimal	dunfell-23.0.26	A basic image that contains the minimal set of components required to boot the device. It focuses on essential system functions without extra tools or features.
Yocto BSP	core-image-bsp	dunfell-23.0.26	Extends core-image-minimal with additional utilities and tools, providing a lightweight environment for system validation, hardware diagnostics, and basic development.
Yocto weston	core-image-weston	dunfell-23.0.26	A standard graphical image with Wayland and Weston support for embedded GUI applications.
Yocto Qt	core-image-qt	dunfell-23.0.26	Extends core-image-weston with the Qt LGPL version (Qt5 framework) enabled, allowing for Qt-based application development and execution in a Wayland-based environment. Additionally, this image provides some example applications (demo apps) to quickly deploy Qt applications on the development platform.

1.1.2 Renesas Custom Images

This section presents Renesas-specific custom images, which are customized and optimized for Renesas products. These images offer specialized features, including fast booting and tailored environments for both graphical and CLI-based applications.

Table 2. Renesas custom images

Distribution	Image file	Version	Description
Renesas CLI Base	renesas-core-image-cli	dunfell-23.0.26	Based on core-image-bsp, this image offers a CLI environment for Renesas hardware development without graphical interfaces. Besides the useful tools inherited from the core-image-bsp, this image also contains new packages for SBC (Single Board Computer) development. For example, package managers (apt, dpgk), network utilities for Bluetooth, Wi-Fi.
Renesas Quickboot CLI	renesas-quickboot-cli	dunfell-23.0.26	This image has the same system functionality as the renesas-core-image-cli but with Quickboot enabled, allowing for faster boot times and efficient system validation on a CLI environment.
Renesas Weston (Qt5)	renesas-core-image-weston	dunfell-23.0.26	Renesas customized core image based on the core-image-weston, with Qt5 framework support (no QT demo apps included). This image offers a full graphical environment for Renesas hardware development and all the useful tools from the renesas-core-image-cli.
Renesas Quickboot Wayland	renesas-quickboot-wayland	dunfell-23.0.26	This image has the same system functionality as the renesas-core-image-weston but with Quickboot enabled, allowing for faster boot times and efficient system validation on a graphical environment.

Note: Quickboot is a trade term that refers to the specific optimizations that are performed to achieve ultra-low start-up times in specific images. Depending on the board architecture, the startup time can be as below 2s. While there is no assurance of the startup time in these images for every platform, these images are the most optimized on our platforms.

1.1.3 Ubuntu Images

This section presents custom Ubuntu-based images tailored for embedded systems, offering a variety of configurations to suit both headless and graphical environments. These images are optimized for performance and ease of use, providing a solid foundation for deploying embedded applications on Renesas platforms.

Distribution	Image file	Version	Description
Ubuntu Core	ubuntu-core-image-qt	ubuntu-base-22.04	A minimal, headless Ubuntu image tailored for embedded systems. It includes Qt framework support for developing Qt-based applications in a resource-efficient environment.
Ubuntu LXDE	ubuntu-lxde-image-qt	ubuntu-base-22.04-base	A lightweight Ubuntu image featuring the LXDE desktop environment, providing a graphical interface while maintaining low resource consumption. This image also includes Qt framework support for GUI development.

1.2 Supported Platforms

Platform	SoC	OPN	Description
RZ/G2L-SBC	RZ/G2L	US157-G2LSBCPOCZ	RZ/G2L-based Pi-compatible SBC.

2. Quick Start

This section describes how to quickly get setup and start running the supported platforms with this release. The following are the essential steps for an SD-MMC card-based boot:

1. Select an image from the list of available images listed in section 1.1 Supported Distributions.
2. Prepare an SD MMC card that has the image programmed onto it.
3. Prepare the hardware with power and debug UART interface. Displaying the connection to one of the HDMI interfaces is highly recommended but not essential.
4. Program the firmware using the appropriate scripts and process in the 'host/tools' directory of the package.
5. Boot normally with the SD MMC card.

2.1 SD-MMC Card Flashing

The Linux bootable SD card creation is a very simple process. The idea is to use any filesystem imaging tool (etcher) to burn the required image's '.wic' file (core-image-qt.wic for a qt demo image) located in the 'target/images' directory of the release to the sd-mmc card. We recommend you install Balena etcher, which is available for Linux, MacOS, and Windows.

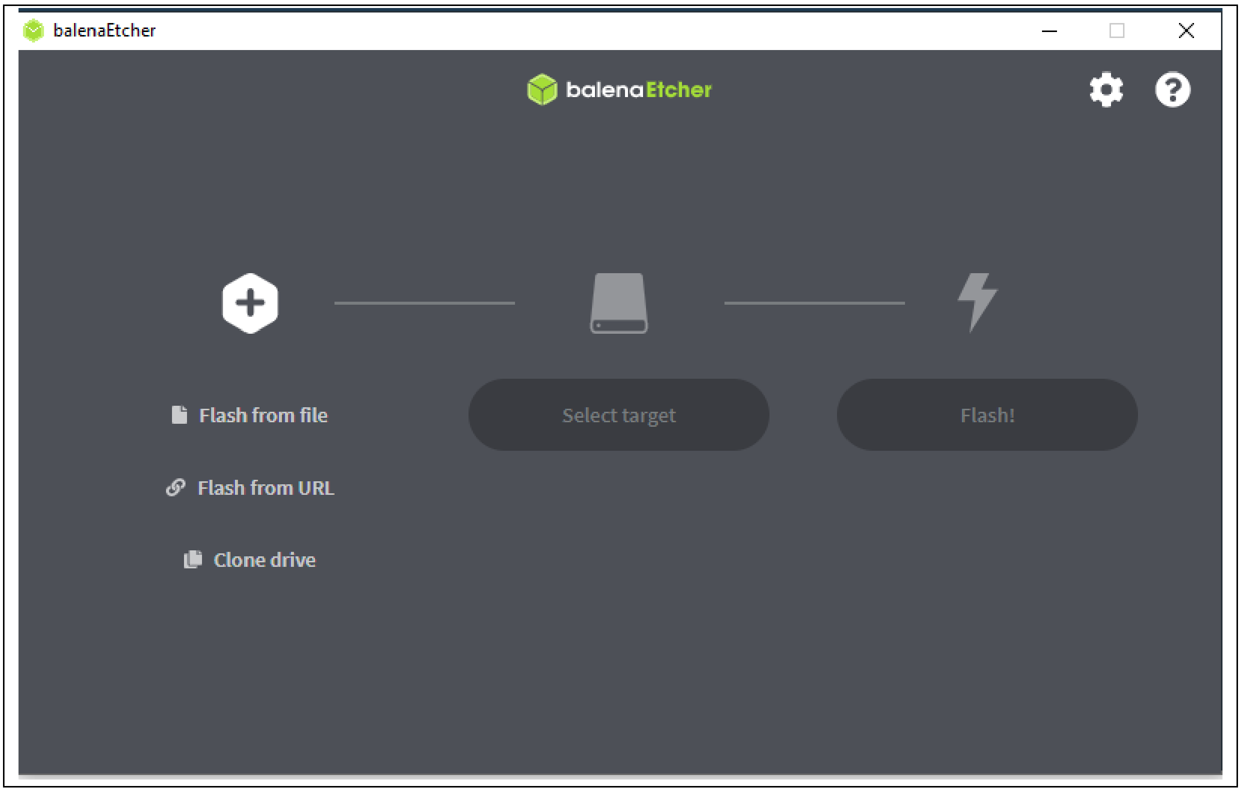

Figure 1. Balena etcher UI

Steps:

1. Select "Flash from File".
2. In the popup window, navigate to your release and select one of the chosen image files (core-image-qt.wic).
3. Then click on 'Select target,' and it will list all available devices.
4. Select your SD MMC card.
   Be mindful not to select your primary laptop/desktop hard drive.
5. Select 'Flash'.
6. When flashing is completed, it will automatically dismount the SD MMC card device.

7. Insert the SD MMC card into the RZ/G2L-SBC bottom SD MMC card connector.

2.2 RZ/G2L-SBC

This section describes the hardware-specific processes for the RZ/G2L-SBC single-board computer.

Note:

- The release consists of images that have desktops and display support.
- At least one basic display, like a 1080p HDMI monitor, must be available.
- You can also use the DSI touch panel described in the MIPI DSI Display Touch Panel.
- It is recommended to use an FTDI cable for the UART and not any other converter chip.

2.2.1 Hardware Requirements

The basic hardware setup consists of the following:

1. RZ/G2L-SBC
2. FTDI RS232 UART cable
3. USB-C 5V 3A+ power supply
4. SD-MMC card (minimum 8 GB)
5. 1080p HDMI display/Waveshare 5” MIPI DSI display touch panel
6. Ethernet cables.
7. OV5640 MIPI CSI camera
8. USB keyboard and mouse
9. 3.5mm Headphone with microphone

2.2.2 Essential Hardware Setup

Figure 2. Essential minimum interfaces show the basic essential hardware setup. We expect a UART cable and an HDMI display to be available.

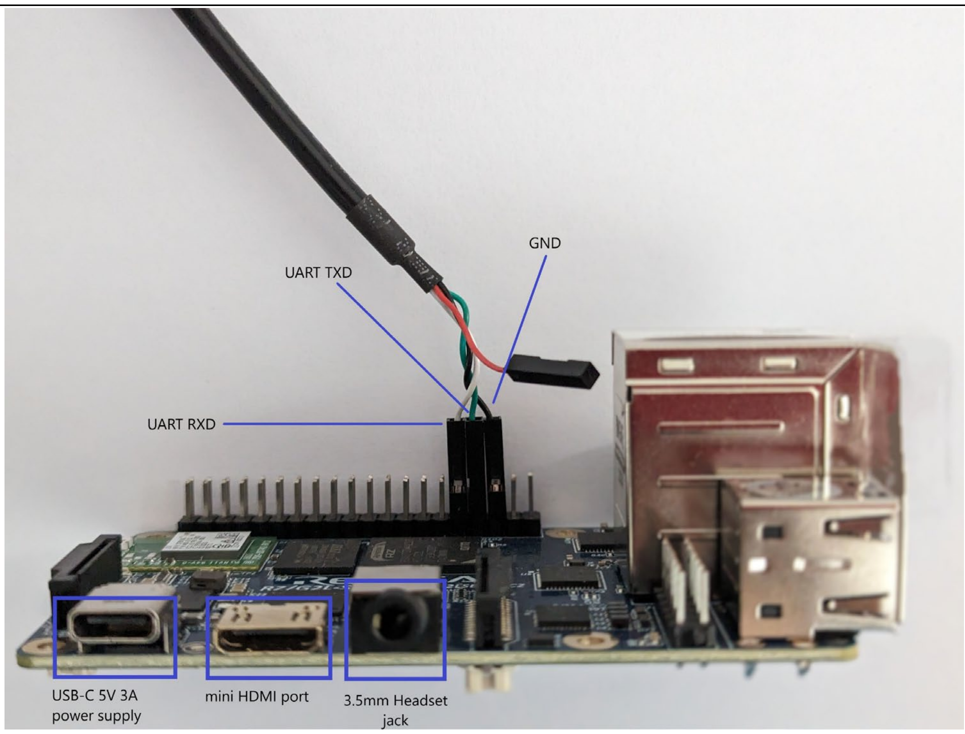

Figure 2. Essential minimum interfaces

2.2.3 Complete Hardware Setup

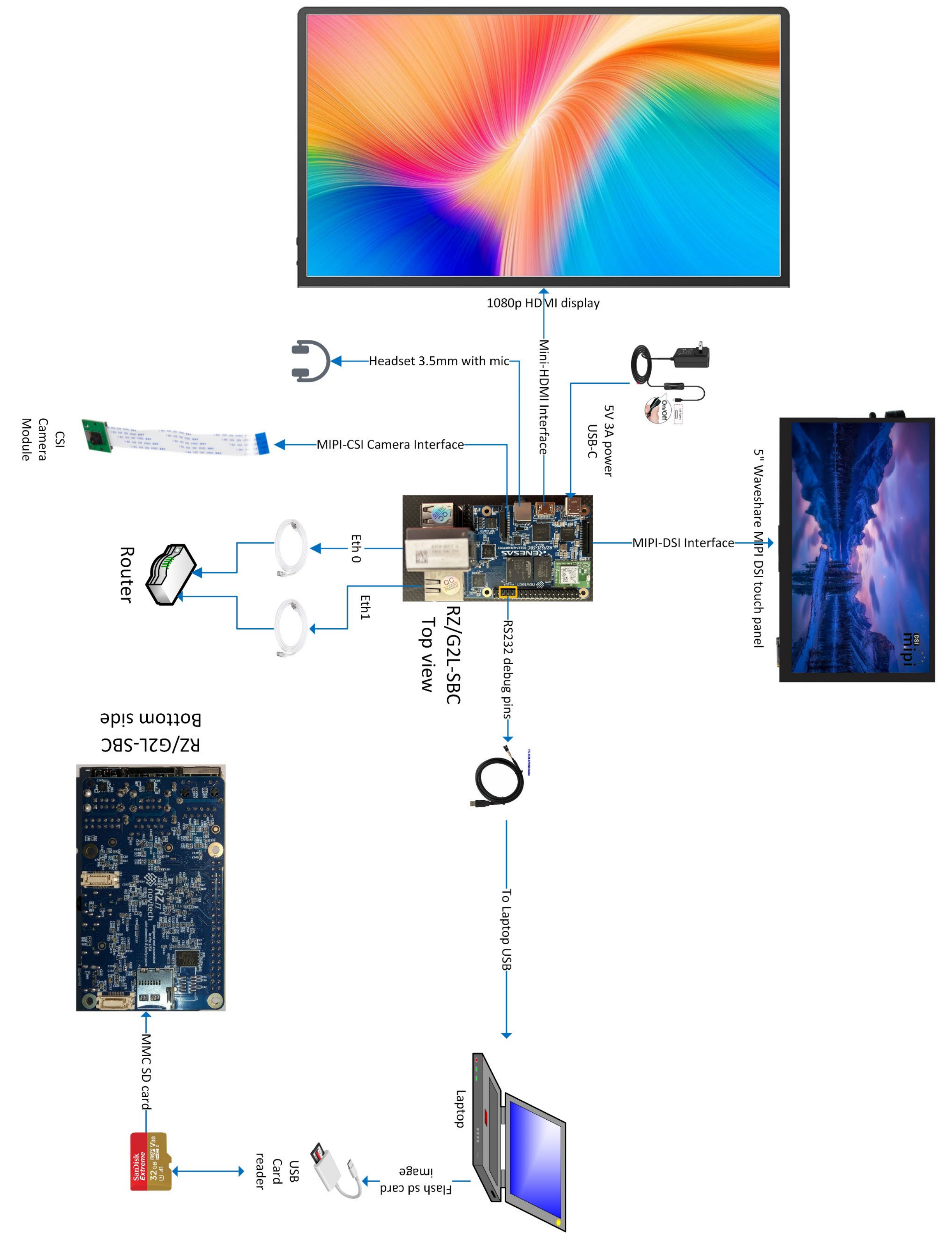

Figure 3. Complete setup

2.2.4 Booting

The booting is straightforward.

1. Insert the MMC card into the MMC port on the bottom side of the RZ/G2L-SBC.
2. Connect the keyboard, mouse, and HDMI display; then insert the USB-C power supply and turn the power on.
3. You should see the boot log on the UART console and the Weston desktop with qt apps on the HDMI screen.
4. Click on any of the applications and interact with them.

The image is fully featured and has powerful desktop-grade features. Read further to learn more about the features packed into the Linux image.

2.2.5 Overview of Connectors

Given below is the basic positioning of the top-level connectors.

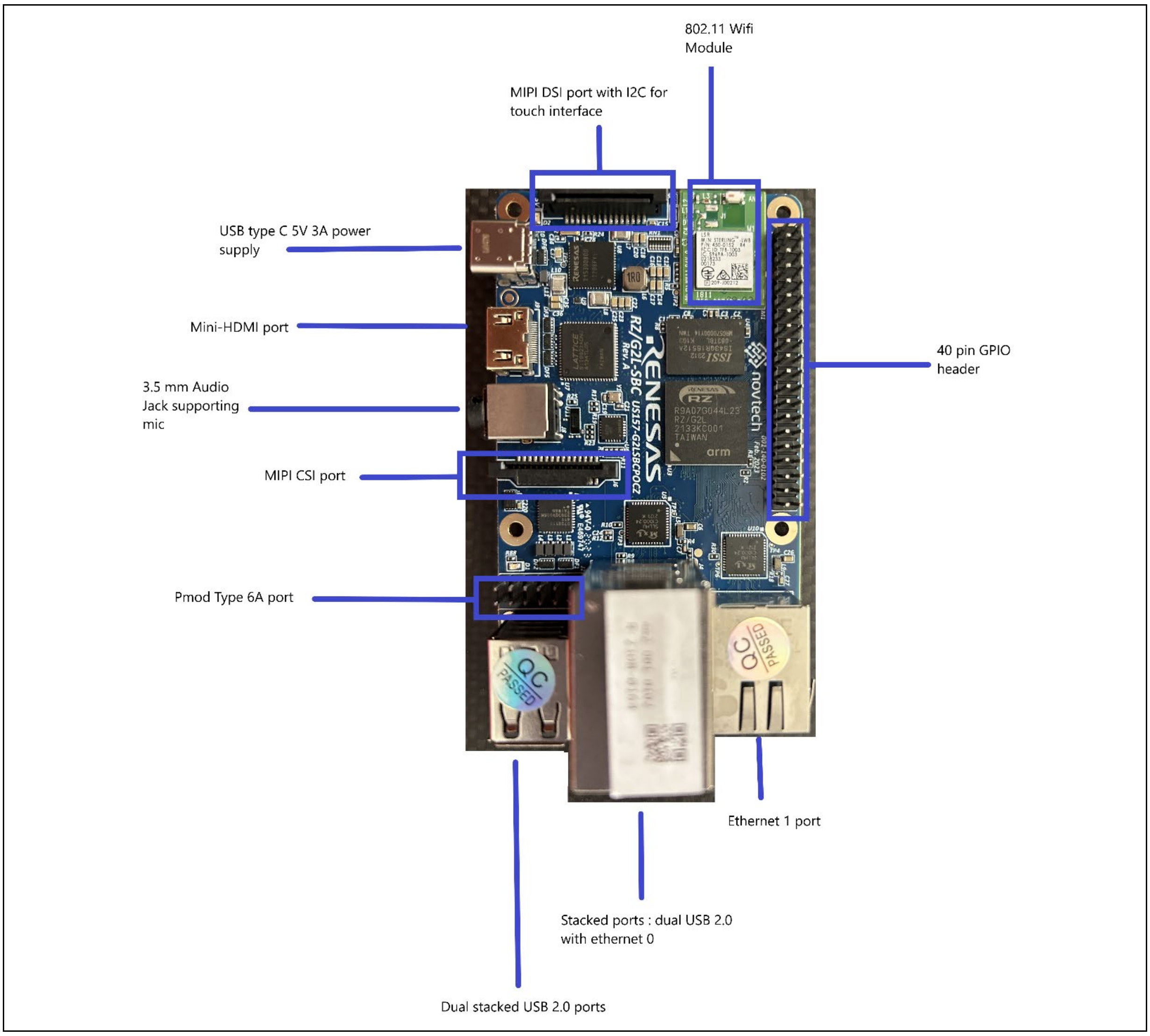

Figure 4. RZ/G2L-SBC top side connectors.

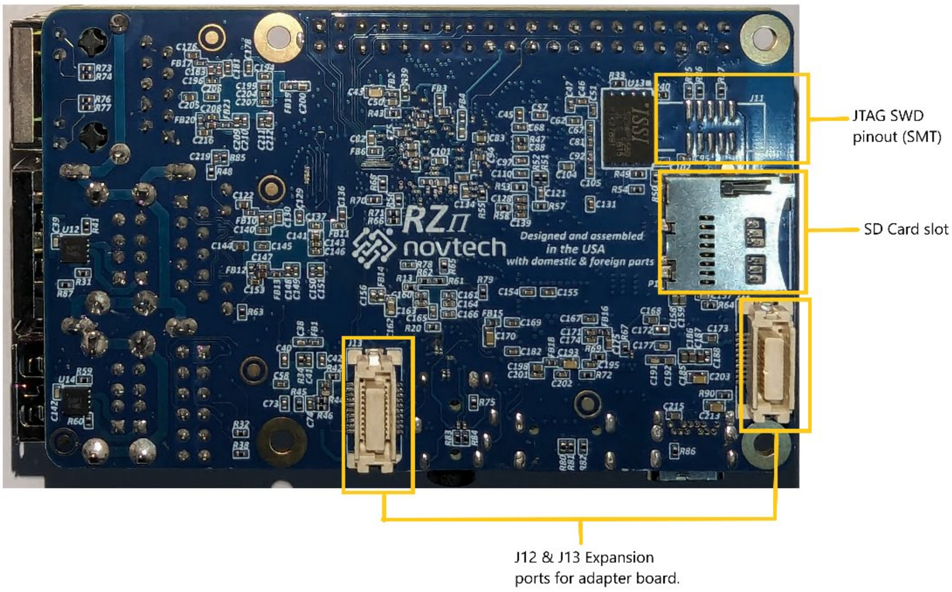

Figure 5. RZ/G2L-SBC Bottom view connectors.

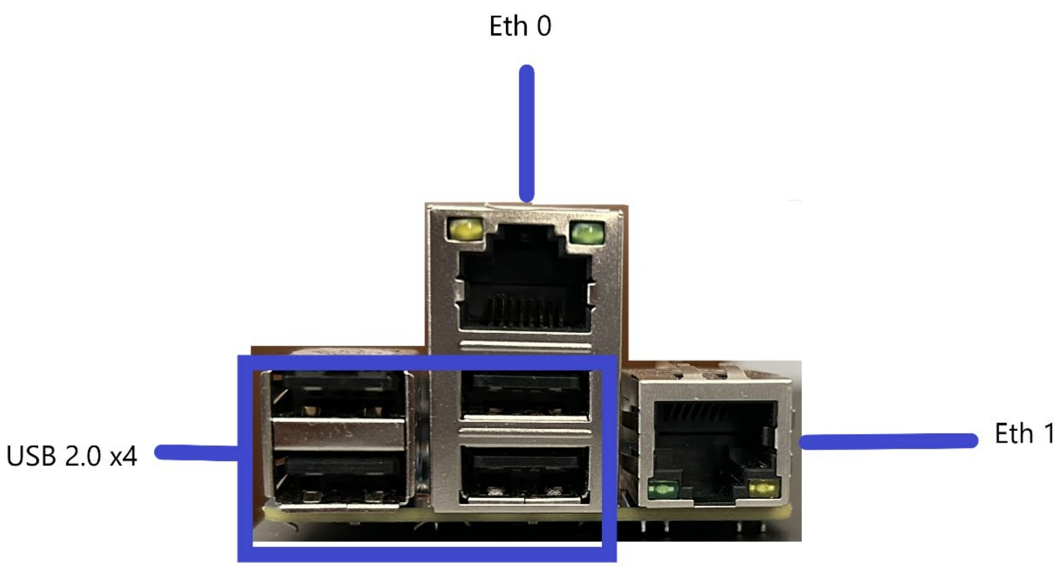

Figure 6. RZ/G2L-SBC side view I/O ports.

2.2.5.1 Power Supply

This section delves into the RZ/G2L-SBC's power supply architecture. The RZ/G2L-SBC uses a simple design with a 5V supply as the single external power source.

(1) USB TYPE-C POWER

This board has one USB Type-C receptacle for power input with USB chargers. The USB Type–C power connector is meant to connect to a 5V power supply. The RZ/G2L-SBC requires a minimum of 3A power to prevent brownouts. However, we recommend a 4.5 -5A power supply as several ports support peripherals that consume substantial power.

2.2.5.2 Peripheral Interface

(1) 40-PIN I/O HEADER

The RZ/G2L-SBC comes with a 40-pin GPIO interface, which is broadly compliant with the Raspberry Pi 3 40-pin GPIO interface and provides additional interfaces like two CAN ports. The diagram below shows the pin configuration along with the marking of the bottom I/O ports for reference to the orientation of the board.

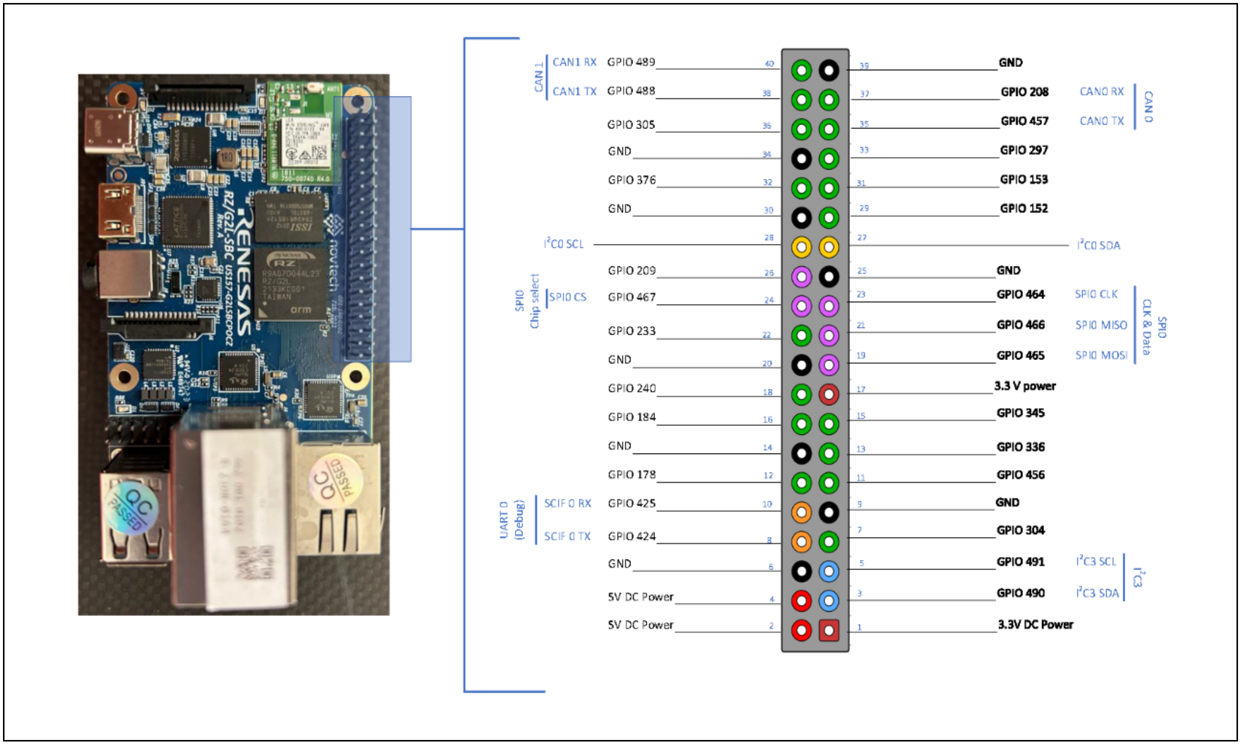

Figure 7. 40 PIN GPIO map with orientation details.

(2) PMOD TYPE 6A STANDARD INTERFACE

The RZ/G2L-SBC is equipped with a 2x6-pin header routed to the PMOD Type-6A interface, conforming to the 1.3.0 specification of PMOD. It includes the alternate pin functions from the specification.

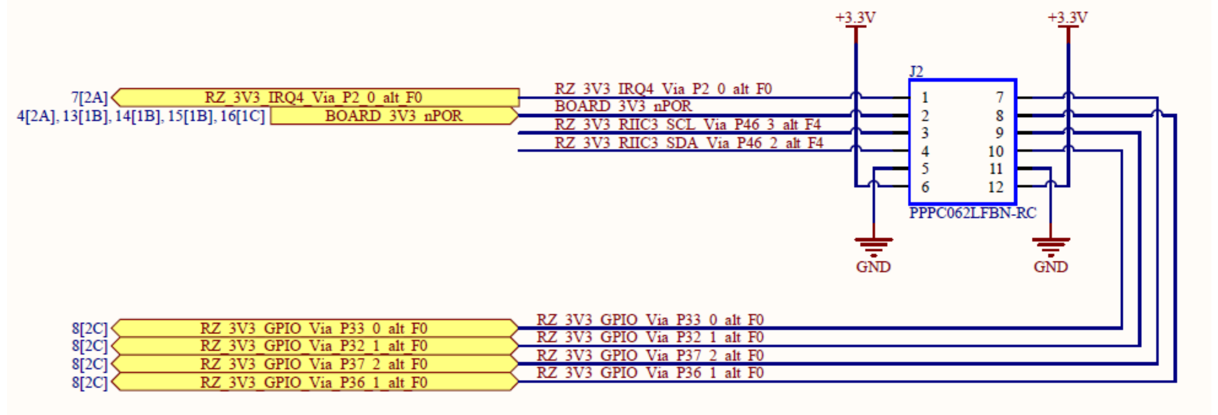

Figure 8. Schematic of PMOD Type 6 A pin header J2.

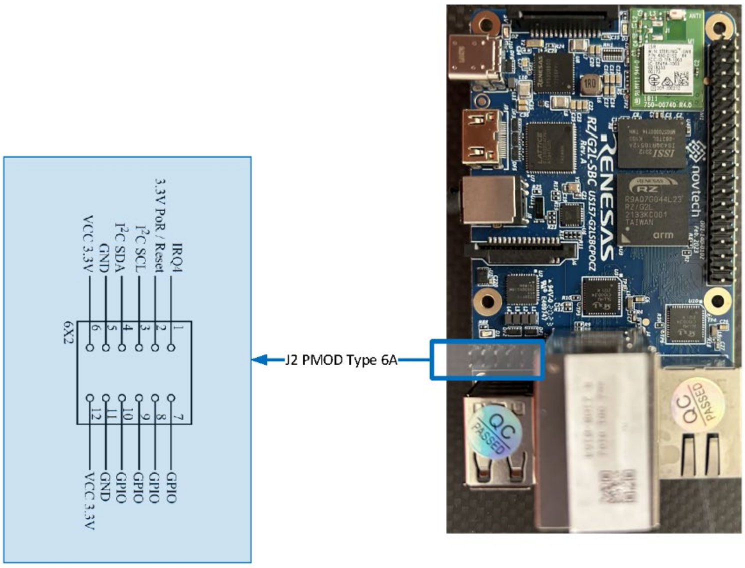

Figure 9. PMOD Type 6A 2x6 0.1mm pin out with orientation details.

(3) USD-CARD INTERFACE

The RZ/G2L-SBC comes with a spring-loaded micro-SD card slot. This is intended to be the primary storage as well as the OS boot device. The SD card is connected to channel 0 of the RZ/G2L SoC SD/MMC interface. The SoC SDIO interface is compliant with memory card standard version 3.0 and supports UHS-1 mode of 50 MB/s (SDR50) and 104 MB/s (SDR104).

Revision History

Rev.	Date	Description	
		Page	Summary
1.00	Mar.03.25	—	Initial release
1.10	Jun.04.25	—	Ubuntu release

System Release Package, RZ Series – User Manual

Publication Date: Jun.04.25

Published by: Renesas Electronics Corporation

RZ Family/ RZ/G Series

Renesas Electronics Corporation