

# RZ/G Verified Linux Package for 64bit kernel Version 1.0.5-RT

R01TU0311EJ0102  
Rev. 1.02  
Sep 7, 2020

## Release Note for HTML5

### Introduction

This release note describes the contents, building procedures for HTML5 (Gecko) and important points of the RZ/G Verified Linux Package for 64bit kernel (hereinafter referred to as “VLP64”). In this release, Linux packages for HTML5 is preliminary and provided AS IS with no warranty.

If you need information to build Linux BSPs without a GUI Framework of HTML5, please refer to “RZ/G Verified Linux Package for 64bit kernel Version 1.0.5-RT Release Note”.

### Contents

1. Release Items .....	2
2. Build environment .....	4
3. Building Instructions.....	6
3.1 Setup the Linux Host PC to build images.....	6
3.2 Building images to run on the board .....	8
3.3 Building SDK .....	12
4. Components .....	13
5. Restrictions .....	14
6. Notes .....	15
7. Appendix.....	19

## 1. Release Items

- **Name and version**

RZ/G Verified Linux Package for 64bit kernel  
Version 1.0.5-RT

- **Distribution method**

Provided via the RZ/G Marketplace. Please visit the sites below and create an account to download the packages.

**RZ/G Marketplace:**

English: <https://www.renesas.com/us/en/products/rzg-linux-platform/rzg-marketplace.html>

Japanese: <https://www.renesas.com/jp/ja/products/rzg-linux-platform/rzg-marketplace.html>

- **Target board**

Hoperun Technology HiHope RZ/G2H platform (hihope-rzg2h)  
Hoperun Technology HiHope RZ/G2M platform (hihope-rzg2m)  
Hoperun Technology HiHope RZ/G2N platform (hihope-rzg2n)  
Silicon Linux RZ/G2E evaluation kit (ek874)

- **Verified functions**

Linux BSP

- Linux Kernel
- Linux Drivers
- Graphics Libraries

GUI Framework

- Qt (LGPL version)
- Gecko

- **Evaluation vs Product Version**

Because the package that contains the VLP64 Yocto recipes also contains the multimedia packages, there are two release versions: Evaluation (eva) and Product (pro). Please note that both of these packages have the same exact functionality. The only difference is that when you execute an application that uses the evaluation version of the multimedia libraries, operation will automatically be stopped after a few hours. The product version does not have this time limitation. To acquire the product version, please contact Renesas to start the formal process of acquiring those releases.

- **File contents**

VLP64 is delivered by the files listed in **Table 1**.

**Table 1. RZ/G Verified Linux Package for 64bit kernel****Common files**

File	Explanation
rzg2_bsp_eva_v105rt.tar.gz (Evaluation version) rzg2_bsp_pro_v105rt.tar.gz (Product version)	Yocto recipe packages (including multimedia package)
oss_pkg_v105rt.7z.001 (500MB)    oss_pkg_v105rt.7z.009 (500MB) oss_pkg_v105rt.7z.002 (500MB)    oss_pkg_v105rt.7z.010 (500MB) oss_pkg_v105rt.7z.003 (500MB)    oss_pkg_v105rt.7z.011 (500MB) oss_pkg_v105rt.7z.004 (500MB)    oss_pkg_v105rt.7z.012 (500MB) oss_pkg_v105rt.7z.005 (500MB)    oss_pkg_v105rt.7z.013 (500MB) oss_pkg_v105rt.7z.006 (500MB)    oss_pkg_v105rt.7z.014 (500MB) oss_pkg_v105rt.7z.007 (500MB)    oss_pkg_v105rt.7z.015 (245MB) oss_pkg_v105rt.7z.008 (500MB)	Open source packages  <b>See the Note below before download</b>
r01tu0277ej0106-rz-g.pdf	Build instruction without HTML5
r01tu0278ej0105-rz-g.pdf	Component list
r01tu0279ej0104-rz-g.pdf	Documents describing booting method and the required settings of bootloader

**Common files for HTML5 (Gecko) environment**

File	Explanation
rzg2_bsp_gecko_v105rt.tar.gz	Yocto recipe packages for HTML5
oss_pkg_gecko_v105rt.7z.001 (500MB) oss_pkg_gecko_v105rt.7z.002 (500MB) oss_pkg_gecko_v105rt.7z.003 (500MB) oss_pkg_gecko_v105rt.7z.004 (316MB)	Open source packages for HTML5 (Gecko 68)
r01tu0311ej0102-rz-g.pdf	This document

Note) Open source packages contain all source codes of OSSs except for Linux kernel. These are the same versions of OSSs used when VLP was verified.

If you are just evaluating VLP and RZ/G series, open source packages are not mandatory to use. Usually, all the software can be built without using these files if your build machine is connected to the Internet.

Open source packages are required for an “offline” environment. The word “offline” means an isolated environment which does not connect to any network. VLP can always build images in this “offline” environment by using these packages without affected from changes of original repositories of OSSs. Also, this “offline” environment always reproduces the same images as the images which were verified by Renesas. Note that if you build without using open source packages, there are possibilities to use different source codes than Renesas used due to the implicit changes of the repositories of OSSs.

Most bootable images that VLP supports can be built on an “offline” environment. Please refer to **2. Build environment**.

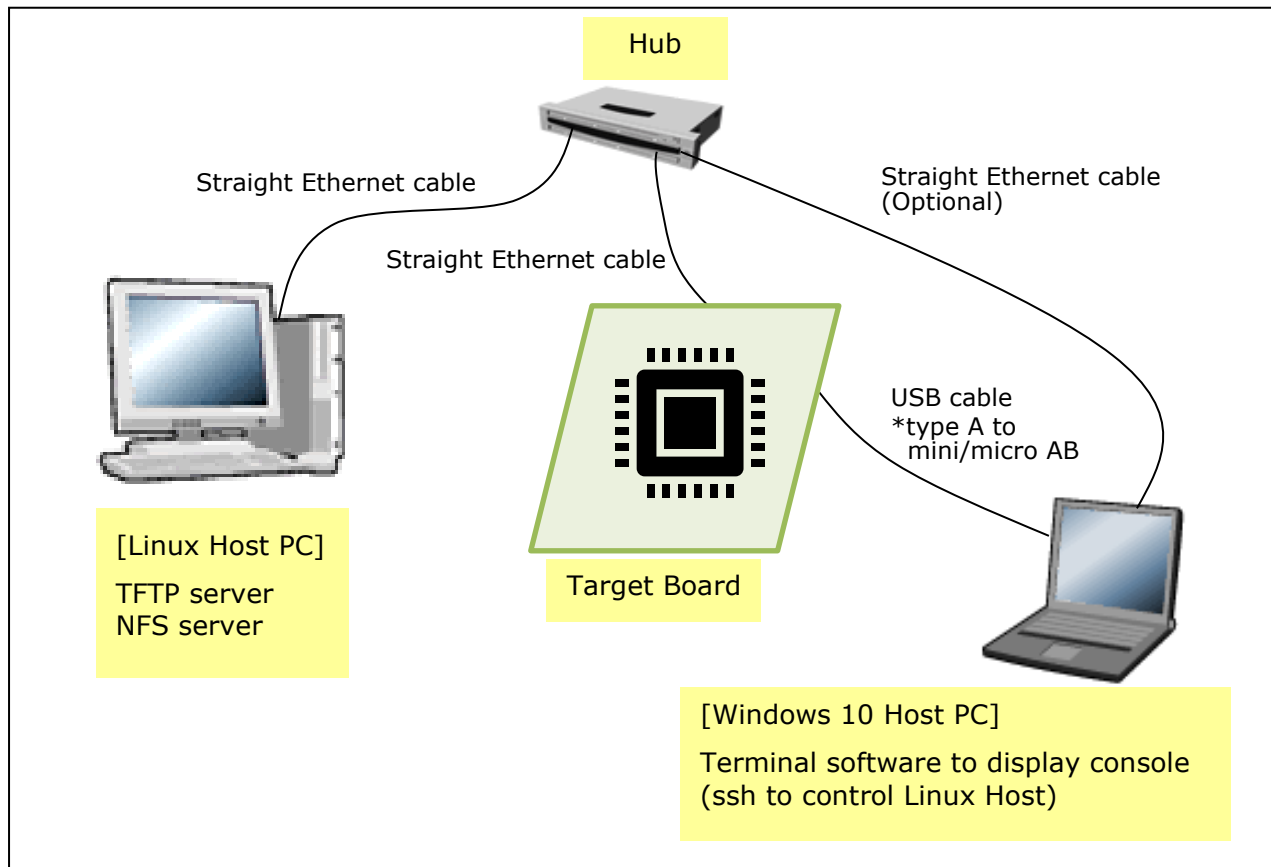
**Optional file to update VLP64 1.0.5-RT**

File	Explanation
v105rt-to-v105rtupdate1.patch.tar.gz	Diff files of Yocto recipes to update VLP64 v1.0.5-RT.

Note) Renesas released VLP64 1.0.5-RT in the end of August 2020. After that, the patch was released to update the packages of VLP64 1.0.5-RT. Please refer to **7. Appendix** for details.

## 2. Build environment

**Figure 1** shows an overall constitution of the recommended environment of VLP64. This environment uses the equipment and the software listed in **Table 2**. Please refer to “RZ/G Verified Linux Package Start-Up Guide” for details about setting up the environment.



**Figure 1. Recommended Environment**

Please connect the Linux Host PC to the internet when downloading Linux kernel source code, or building without using the open source packages files that Renesas provides.

Below images can be built “offline”.

- core-image-weston
- core-image-weston-sdk

Below are not available in the “offline” environment. Please connect your Linux Host PC to the internet.

- Preparing a Linux Host PC

**Table 2. Equipment and Software Necessary for Developing Environments of RZ/G Linux Platform**

Equipment	Description
Linux Host PC	Used as build/debug environment 100GB free space on HDD is necessary
OS	<b>Ubuntu 16.04 LTS</b> 64 bit OS must be used.
TFTP server	Used for downloading the Linux kernel to the board
NFS server	Used for mounting rootfs via NFS
Windows Host PC	Used as debug environment, controlling with terminal software
OS	Windows 10
Terminal software	Used for controlling serial console of the target board Tera Term (latest version) is recommended Available at <a href="https://ttssh2.osdn.jp/index.html.en">https://ttssh2.osdn.jp/index.html.en</a>
VCP Driver	Virtual COM Port driver which enables to communicate Windows Host PC and the target board via USB which is virtually used as serial port. Available at: <ul style="list-style-type: none"> <li>• <a href="http://www.ftdichip.com/Drivers/VCP.htm">http://www.ftdichip.com/Drivers/VCP.htm</a> (for ek874)</li> <li>• <a href="https://www.silabs.com/products/development-tools/software/usb-to-uart-bridge-vcp-drivers">https://www.silabs.com/products/development-tools/software/usb-to-uart-bridge-vcp-drivers</a> (for hihope-rzg2h, hihope-rzg2m, and hihope-rzg2n)</li> </ul> Please install VCP Driver corresponding to the target board.

Note) Build may fail if Ubuntu 14, 18 or 20 is used.

### 3. Building Instructions

#### 3.1 Setup the Linux Host PC to build images

Before starting the build, run the steps below on the Linux Host PC to install packages used for building BSP.

##### (1) Install packages

```
$ sudo apt-get update
$ sudo apt-get install gawk wget git-core diffstat unzip texinfo gcc-multilib \
build-essential chrpath socat cpio python python3 python3-pip python3-pexpect \
xz-utils debianutils iputils-ping libstdc++6-dev xterm p7zip-full autoconf2.13
$ sudo apt install clang llvm clang-3.9 llvm-3.9
$ sudo apt-get install build-essential libasound2-dev libcurl4-openssl-dev \
libdbus-1-dev libdbus-glib-1-dev libgconf2-dev libgtk-3-dev libgtk2.0-dev \
libpulse-dev libx11-xcb-dev libxt-dev nasm nodejs openjdk-8-jdk-headless \
python-dbus python-dev python-pip python-setuptools software-properties-common \
unzip uuid wget xvfb yasm zip
```

Please refer to the URL below for detailed information:

- <https://www.yoctoproject.org/docs/2.4.3/yocto-project-qs/yocto-project-qs.html>

If the Git has not configured yet, set user name and email address before starting build procedure. Without this setting, an error occurs when building procedure runs git command to apply patches.

```
$ git config --global user.email "you@example.com"
$ git config --global user.name "Your Name"
```

##### (2) Install GCC 7 and switch the default GCC

```
$ sudo apt-get install software-properties-common
$ sudo add-apt-repository ppa:ubuntu-toolchain-r/test
$ sudo apt-get update
$ sudo apt-get install gcc-7 g++-7
$ sudo update-alternatives --install /usr/bin/gcc gcc /usr/bin/gcc-5 10 \
--slave /usr/bin/g++ g++ /usr/bin/g++-5
$ sudo update-alternatives --install /usr/bin/gcc gcc /usr/bin/gcc-7 20 \
--slave /usr/bin/g++ g++ /usr/bin/g++-7
```

After the above step, GCC 7 is set as the default GCC compiler. In case to build VLP without Gecko, or necessary to use GCC5 for another reason, run this command to switch the default GCC compiler to GCC 5.

```
$ sudo update-alternatives --config gcc
```

After running the above command, the following text is displayed. Please input a number corresponding to GCC5. In the following example, "1" corresponds to GCC5. In case to switch the default GCC compiler again, run the above command again.

```
There are 2 choices for the alternative gcc (providing /usr/bin/gcc).
```

Selection	Path	Priority	Status
* 0	/usr/bin/gcc-7	20	auto mode
1	/usr/bin/gcc-5	10	manual mode
2	/usr/bin/gcc-7	20	manual mode

```
Press <enter> to keep the current choice[*], or type selection number:
```

### (3) Create a working directory (user\_work), and install packages with bootstrap to build Gecko

```
$ mkdir ~/user_work
$ cd ~/user_work
$ git clone --branch esr68 --depth 1 https://github.com/mozilla/gecko-dev.git
$ export SHELL
$ cd gecko-dev && /bin/echo -e "2\n3\nY\n\n\n" | ./mach bootstrap
```

The name and the place of the working directory can be changed, as necessary. Once this step is finished, gecko-dev directory can be deleted.

## 3.2 Building images to run on the board

**Copy all files obtained from Renesas into your home directory prior to the steps below.**

### (1) Decompress Yocto recipe packages

```
$ cd ~/user_work
$ tar xvf ~/rzg2_bsp_eva_v105rt.tar.gz
$ tar xvf ~/rzg2_bsp_gecko_v105rt.tar.gz
```

Please replace “*eva*” with “*pro*” for the product version.

Note) Please note that your build environment must have 100GB of free hard drive space in order to complete the build. The Yocto BSP build environment is very large. Especially in case you are using a Virtual Machine, please check how much disk space you have allocated for your virtual environment.

Note) VLP64 is set video output to LVDS as default setting. In case using HDMI, please refer to **Section 6 Notes - (2) Video output**.

### (2) Execute the copy script for proprietary software

```
$ cd ~/user_work/meta-rzg2
$ sh docs/sample/copyscript/copy_proprietary_softwares.sh ../proprietary
```

Note) Both product and evaluation version use the same script.

### (3) Setup a build environment

```
$ cd ~/user_work
$ source poky/oe-init-build-env
```

Environment to build are set by the source command.

### (4) Prepare the default configuration files for the target board

```
$ cd ~/user_work/build
$ cp ../meta-rzg2/docs/sample/conf/board/linaro-gcc/*.conf ./conf/
```

Please replace “*board*” by the name below:

HiHope RZ/G2H board:	hihope-rzg2h
HiHope RZ/G2M board:	hihope-rzg2m
HiHope RZ/G2N board:	hihope-rzg2n
EK874 RZ/G2E board:	ek874



**(5) Modify recipe files**

- Enable meta-browser, meta-rust, and meta-clang by adding the line in “~/user\_work /build/conf/bblayers.conf” as below:

```
BBLAYERS += " ${TOPDIR}/../meta-browser "
BBLAYERS += " ${TOPDIR}/../meta-rust "
BBLAYERS += " ${TOPDIR}/../meta-clang "
```

- Enable the browser application by adding the line in “~/user\_work /build/conf/local.conf” as below:

```
IMAGE_INSTALL_append = " firefox "
IMAGE_INSTALL_append = " ttf-sazanami-gothic ttf-sazanami-mincho "
PACKAGECONFIG_append_pn-firefox = " openmax "
PACKAGECONFIG_append_pn-firefox = " webgl "
PACKAGECONFIG_append_pn-firefox = " gpu "
TARGET_CXXFLAGS_remove_toolchain-clang = " --stdlib=libc++"
TUNE_CCARGS_remove_toolchain-clang = " --rtlib=compiler-rt \
--unwindlib=libunwind --stdlib=libc++"
TUNE_FEATURES_remove_toolchain-clang = "cortexa57-cortexa53"
IMAGE_INSTALL_append = " pulseaudio-server "
BBMASK .= "|openssl_debian"
DISTRO_FEATURES_append = " x11"
```

Note) In this release, the following two steps are necessary to build BSP, but these are temporary steps. In next release, HTML5 is going to be built on Debian 10 (Buster) based CIP Core Packages.

- Enable Debian 8 (jessie) based CIP Core Packages indicated as “(CIP)” in **Table 4** by commenting out a line in “~/user\_work /build/conf/local.conf” as below:

```
#MACHINE_FEATURES_append = " Buster-limited"
```

- To remove a new recipe of ca-certificate, please run the below commands.

```
$ cd ~/user_work
$ rm -r ./meta-rzg2/recipes-support/ca-certificates
```

Note) Please skip the above commands in case to use a patch (v105rt-to-v105rtupdate1.patch.tar.gz) to update the packages of VLP64 1.0.5-RT. Please refer to **7. Appendix** for details.

**(6) Decompress OSS files to “build” directory**

This step is not mandatory and able to go to the step (8) in case the “offline” environment is not used.

```
$ cd ~/user_work/build
$ 7z x ~/oss_pkg_v105rt.7z.001 (*1)
$ 7z x ~/oss_pkg_gecko_v105rt.7z.001 (*1)
```

(\*1) All OSS packages will be decompressed at once with this '7z' command. Please place all oss\_pkg\_v105rt.7z and oss\_pkg\_gecko\_v105rt.7z files in the same directory before running this '7z' command. 7zipped files are not able to decompress individually.

Note) If this step is omitted and BB\_NO\_NETWORK is set to “0” in next step, all source codes will be downloaded from the repositories of each OSS via the internet when running bitbake command. Please note that if you do not use an “offline” environment, a build may fail due to the implicit changes of the repositories of OSS.

#### (7) Download Linux kernel source code

```
$ cd ~/user_work/build
$ bitbake linux-renesas -c fetch
```

After the above procedure is finished, the “offline” environment is ready. If you want to prevent network access, please change the line in the “~/user\_work/build/conf/local.conf” as below:

```
BB_NO_NETWORK = "1"
```

To change BB\_NO\_NETWORK from “0” to “1”.

#### (8) Start a build

```
$ bitbake core-image-weston
```

Building an image can take up to a few hours depending on the user’s host system performance.

After the build is successfully completed, a similar output will be seen:

```
NOTE: Tasks Summary: Attempted 7427 tasks of which 16 didn't need to be rerun and all succeeded.
```

and the command prompt will return.

All necessary files listed in **Table 3** will be generated by the bitbake command and will be located in the **build/tmp/deplo**y/images directory.

Table 3. Image files

	Linux kernel	root filesystem	Boot loader
<b>RZ/G2H</b>	Image-hihope-rzg2h.bin	<image name>- hihope- rzg2h.tar.bz2	u-boot-elf-hihope-rzg2h.srec bootparam_sa0.srec bl2-hihope-rzg2h.srec bl31-hihope-rzg2h.srec tee-hihope-rzg2h.srec cert_header_sa6.srec
<b>RZ/G2M</b>	Image-hihope-rzg2m.bin	<image name>- hihope- rzg2m.tar.bz2	u-boot-elf-hihope-rzg2m.srec bootparam_sa0.srec bl2-hihope-rzg2m.srec bl31-hihope-rzg2m.srec tee-hihope-rzg2m.srec cert_header_sa6.srec
<b>RZ/G2N</b>	Image-hihope-rzg2n.bin	<image name>- hihope- rzg2n.tar.bz2	u-boot-elf-hihope-rzg2n.srec bootparam_sa0.srec bl2-hihope-rzg2n.srec bl31-hihope-rzg2n.srec tee-hihope-rzg2n.srec cert_header_sa6.srec
<b>RZ/G2E</b>	Image-ek874.bin	<image name>- ek874.tar.bz2	u-boot-elf-ek874.srec bootparam_sa0.srec bl2-ek874.srec bl31-ek874.srec tee-ek874.srec cert_header_sa6.srec

<image name> will be the name used in the step (8).

#### Device tree files

	Type 1	Type 2	Type 3	Type 4
<b>RZ/G2H</b>	Image-r8a774e1-hihope-rzg2h.dtb	Image-r8a774e1-hihope-rzg2h-ex.dtb	Image-r8a774e1-hihope-rzg2h-ex-idk-1110wr.dtb	Image-r8a774e1-hihope-rzg2h-ex-mipi-2.1.dtb
<b>RZ/G2M</b>	Image-r8a774a1-hihope-rzg2m.dtb	Image-r8a774a1-hihope-rzg2m-ex.dtb	Image-r8a774a1-hihope-rzg2m-ex-idk-1110wr.dtb	Image-r8a774a1-hihope-rzg2m-ex-mipi-2.1.dtb
<b>RZ/G2N</b>	Image-r8a774b1-hihope-rzg2n.dtb	Image-r8a774b1-hihope-rzg2n-ex.dtb	Image-r8a774b1-hihope-rzg2n-ex-idk-1110wr.dtb	Image-r8a774b1-hihope-rzg2n-ex-mipi-2.1.dtb
<b>RZ/G2E</b>	Image-r8a774c0-cat874.dtb	Image-r8a774c0-ek874.dtb	Image-r8a774c0-ek874-idk-2121wr.dtb	Image-r8a774c0-ek874-mipi-2.1.dtb

There are 4 types of the device tree files. Available devices are different depending on them. Please refer to the following description:

- **Type1:** Main board only
- **Type2:** Main board + Sub board
- **Type3:** Main board + Sub board + LVDS panel
- **Type4:** Main board + Sub board + MIPI/CSI2 cameras

Please note that users who use the combination of main and sub boards need to use type2-4 as a device tree file. **If the dtb files of type1 are used, interfaces on the sub board such as Ethernet are not able to be used.**

For the booting method and the required settings, please refer to the “RZ/G2 Reference Boards Linux Start-up Guide”.

Also note that the dtb files listed in the **Table 3 cannot be used for the early version of boards**. See the section **6 Notes**.

### 3.3 Building SDK

To build Software Development Kit (SDK), run the commands below after the steps (1) – (8) of section 3.2 are finished. Before starting to build SDK, please switch the default GCC compiler to GCC 5. Please refer to the step (2) of section 3.1.

The SDK allows you to build custom applications outside of the Yocto environment, even on a completely different PC. The results of the commands below are ‘installer’ that you will use to install the SDK on the same PC, or a completely different PC.

For building general applications:

```
$ cd ~/user_work/build
$ cp ../meta-rzg2/docs/sample/conf/board/linaro-gcc/*.conf ./conf/
$ bitbake core-image-weston-sdk -c populate_sdk
```

Please replace “*board*” by the name below:

HiHope RZ/G2H board: hihope-rzg2h  
HiHope RZ/G2M board: hihope-rzg2m  
HiHope RZ/G2N board: hihope-rzg2n  
EK874 RZ/G2E board: ek874

The resulting SDK installer will be located in `build/tmp/deploy/sdk/`

The SDK installer will have the extension `.sh`

To run the installer, you would execute the following command:

```
$ sudo poky-glibc-x86_64-core-image-weston-sdk-aarch64-toolchain-2.4.3.sh
```

Please refer to “RZ/G2 Group Application Note” and “Verified Linux Package Start-up Guide” for the usage of SDK.

## 4. Components

Compared to VLP for 32bit kernel and previous version of VLP64, this version uses different version of Linux kernel and supports newer HTML5 rendering engine. Please also refer to the “Component list” for details.

**Table 4. Versions of commonly used components**

Components	VLP for 32bit kernel v2.1.6-RT	VLP for 64bit kernel v1.0.3-RT	VLP for 64bit kernel v1.0.5-RT
Linux kernel	4.4.208-cip41-rt27	4.19.72-cip10-rt4	4.19.132-cip30-rt12
GCC	7.2.1 (Linaro GCC 7.2-2017.11)	7.3.1 (Linaro GCC 7.3-2018.05)	7.3.1 (Linaro GCC 7.3-2018.05)
glibc	2.19 (CIP)	2.19 (CIP)	2.19 (CIP)
busybox	1.22.0 (CIP)	1.22.0 (CIP)	1.22.0 (CIP)
openssl	1.0.2o	1.0.2o	1.0.2o
gstreamer1.0	1.12.2	1.12.2	1.12.2
wayland	1.13.0	1.13.0	1.13.0
weston	2.0.0	2.0.0	2.0.0
python3	3.5.3	3.5.3	3.5.3
Qt (LGPL version)	5.6.3	5.6.3	5.6.3
Gecko	60	60	68
Docker	–	17.06	17.06

## 5. Restrictions

### (1) Drivers

In this release, the below drivers can be used but are not fully tested on all MPUs.

- Bluetooth
- Wifi
- IPMMU

### (2) USB camera 3.0

RZ/G2E cannot stream with Full HD and above resolution when you use a camera with USB 3.0.

### (3) CSI40

Disable CSI40 in G2H and G2N by default due to SW limitation.

## 6. Notes

### (1) Bootloader/Linux kernel

The dtb files listed in the **Table 3** cannot be used for the early revision of Hoperun boards. If you are using revision 2 boards, please use below files. These are automatically generated at the same place as the other image files when building a BSP.

#### HiHope RZ/G2M board:

- Image-r8a774a1-hihopec-rzg2m-**rev2**.dtb (main board only)
- Image-r8a774a1-hihopec-rzg2m-ex-**rev2**.dtb (main + sub board)
- Image-r8a774a1-hihopec-rzg2m-ex-**rev2**-mipi-2.1.dtb (main + sub board + MIPI/CSI2 cameras)

#### HiHope RZ/G2N board:

- Image-r8a774b1-hihopec-rzg2m-**rev2**.dtb (main board only)
- Image-r8a774b1-hihopec-rzg2m-ex-**rev2**.dtb (main + sub board)
- Image-r8a774b1-hihopec-rzg2m-ex-**rev2**-mipi-2.1.dtb (main + sub board + MIPI/CSI2 cameras)

#### Silicon Linux RZ/G2E board (EK874):

- Please refer to **(8) GPU** in this section.

Note) Board revision is printed below the Hoperun logo on boards.

### (2) Video output

VLP64 is set video output to LVDS as default setting. In case to use HDMI as a video output, please apply the patch with this command.

```
$ cd ~/user_work/meta-rzg2
$ patch -p1 < ../extra/0001-BSPv1.0.5-Add-HDMI-support-for-RZ-G2.patch
```

After applying above patch, please note to below points.

#### Audio

In case 48kHz audio, please set SW2404 to P1 side.

#### Bluetooth

Firmware of Bluetooth is integrated into the kernel. Therefore, this step is not necessary.

```
hciattach /dev/ttySC1 texas 3000000
```

Instead, this step is required.

```
rfkill unblock bluetooth
```

#### SATA

SATA interface on HiHope RZ/G2N or RZ/G2H board is enabled by setting switches as below.

- SW1001-7 on main board: OFF
- SW43 on sub board: ON

### (3) Weston

Due to the specification of opensource software (Weston 2.0.0), it is not recommended to resize application windows. Please consider designing the application to use fixed sized windows.

### (4) Video playback

Due to the specification of opensource software (GStreamer and others) and drivers, multiple GStreamer pipelines with hardware scale cannot run.

Also, below formats of video are not supported.

- NV61
- YUV420
- YUV422
- YUV444
- H.264, 80Mbps

### (5) ECC

The ECC function for DRAM has two modes: 8bit data/5bit ECC mode and 64bit data/8bit ECC mode. 8bit data/5bit ECC mode can be evaluated by the following method. When applying the ECC function to products or need other details including the method to enable 64bit data/8bit ECC mode, please contact Renesas.

Enable the function by changing the lines below in the `local.conf`.

Disable:

```
# MACHINE_FEATURES_append = " ecc"
# ECC_MODE = "Partial"
```

Enable:

```
MACHINE_FEATURES_append = " ecc"
ECC_MODE = "Full"
```

This sets 8bit data/5bit ECC mode for all DRAM region. After building, please replace all images including boot loaders.

### (6) SDHI

Early revision of EK874 boards cannot detect insertion of SD card. Please plugged in a card before turning on the power.

### (7) VIN

One camera input is enabled in default settings. Two camera inputs can be enabled in RZ/G2N and H. Please refer to the Video Capture Driver User's Manual that is included in the RZ/G2 Group BSP Manual Set for more details.

VIN on early revision of HiHope RZ/G2M and N boards fails to work. Please use newer boards in case VIN is necessary.

### (8) GPU

The dtb files listed in the **Table 3** cannot be used for the early version (ES1.0) of RZ/G2E.

If the board prints the messages below when turn on the power, ES1.0 of RZ/G2E is implemented on the board.



```
[ 0.000096] NOTICE: BL2: RZ G2E Initial Program Loader(CA53)
[ 0.004373] NOTICE: BL2: Initial Program Loader(Rev.1.0.23)
[ 0.009991] NOTICE: BL2: PRR is RZG G2E Ver.1.0
```

```
CPU: Renesas Electronics R8A774C0 rev 1.0
```

In this case, please use below dtb files instead. These are built simultaneously when building normal dtb files.

- Image-r8a774c0-es10-cat874.dtb (main board only)
- Image-r8a774c0-es10-ek874.dtb (main + sub board)
- Image-r8a774c0-es10-ek874-idk-2121wr.dtb (main + sub board + LVDS panel)
- Image-r8a774c0-es10-ek874-mipi-2.1.dtb (main + sub board + MIPI/CSI2 cameras)

In case both old and new RZ/G2E LSIs are used in your laboratory at the same time, dtb files can automatically be selected using the environment variable `cut_ver` which is set by u-boot program according to the LSI's information. Please store multiple dtb files in an SD card and set `bootcmd` of u-boot like this.

```
setenv bootcmd 'fatload mmc 0:1 0x48080000 Image; if test "${cut_ver}" = "10"; then
fatload mmc 0:1 0x48000000 Image-r8a774c0-es10-ek874.dtb; else fatload mmc 0:1 0x480
00000 Image-r8a774c0-ek874.dtb ; fi ; booti 0x48080000 - 0x48000000'
```

## (9) Docker

Docker is disabled in the default settings of VLP. To enable Docker, please uncomment the below line inside the file "`~/user_work/build/conf/local.conf`" before building images.

```
#MACHINE_FEATURES_append = " docker"
```

## (10) Wifi

Wifi is disabled in default settings but modules necessary for Wifi functions are installed into rootfs. In case Wifi is necessary, please enable it from a console as below.

```
$ rfkill list
```

If this command shows "Soft blocked: yes", run "unblock" command like this.

```
$ rfkill unblock wlan
```

Then, continue below.

```
$ connmanctl
connmanctl> scan wifi
connmanctl> services
connmanctl> agent on
connmanctl> connect <network_name>
  <imput password>
connmanctl> quit
```

You may need to retry the "connect" command few times.

Note that some settings relating about radio waves should be adjusted according to the laws of each region. Please refer to general information in books and websites about Linux networking.

## (11) USB Video Class

USB Video Class (UVC) driver is not installed with the default settings of VLP due to its big size.

In case UVC devices such as USB cameras are necessary, please install the driver by adding the line below to `local.conf`.

```
IMAGE_INSTALL_append = " kernel-module-uvcdm300 "
```

## (12) HTML5 (Gecko)

When you launch the browser application on the board, please run the below command on a terminal software or a console on the board.

```
$ GDK_BACKEND=wayland firefox <content>
```

Please replace `<content>` with an URL or a path of a local content like the below examples:

```
$ GDK_BACKEND=wayland firefox https://www.renesas.com/us/en/
$ GDK_BACKEND=wayland firefox /home/root/index.html
```

## 7. Appendix

- **Update patch for VLP64 1.0.5-RT**

Renesas released VLP64 1.0.5-RT in the end of August 2020. After that, a patch (v105rt-to-v105rtupdate1.patch.tar.gz) was released to update the packages of VLP64 1.0.5-RT. Please use the patch if you want to update the packages as below:

- Support new CA certificates in an image of HTML5

Note) The update patch is preliminary and provided AS IS with no warranty.

- **How to apply the patch**

**Copy the patch file obtained from Renesas into your home directory prior to the steps below.**

After the step (1) of the section 3.2, run the commands below to update the build environment.

```
$ cd ~/user_work
$ tar xvf ~/v105rt-to-v105rtupdate1.patch.tar.gz
$ patch -p1 < ./v105rt-to-v105rtupdate1.patch
```

After that, run the steps (2) - (7) of the section 3.2.

Additional OSS packages will be downloaded via the internet when running bitbake command in the updated environment, so please connect the Linux Host PC to the network before starting the build, and change the line in the “~/user\_work/build/conf/local.conf” as below:

```
BB_NO_NETWORK = "0"
```

To change BB\_NO\_NETWORK from “1” to “0”.

Then start the build in the step (8) of the section 3.2.

## Website and Support

Renesas Electronics Website

<http://www.renesas.com/>

Inquiries

<http://www.renesas.com/contact/>

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