RZ/Five Verified Linux Package
Version 3.0.2-update1

R01US0608EJ0101
Rev. 1.01
Feb 10, 2023

Release Note

Introduction

This release note describes the contents, building procedures and important points of the RZ/Five Verified Linux Package (hereinafter referred to as "VLP/F").

Contents

1.   Release Items

- **Name and version**
  RZ/Five Verified Linux Package
  Version 3.0.2-update1 (hereinafter referred to as "VLP/F v3.0.2-update1")

- **Distribution method**
  Please visit the site below and create an account to download the packages. Basic packages of VLP/F v3.0.2-update1 which are listed in **Table 1** can be downloaded.

  RZ/Five product page:
  https://www.renesas.com/us/en/products/microcontrollers-microprocessors/rz-mpus/rzfive-risc-v-general-purpose-microprocessors-risc-v-cpu-core-andes-ax45mp-single-10-ghz-2ch-gigabit-ethernet

  RZ/Five Verified Linux Package [5.10-CIP]:
  https://www.renesas.com/us/en/software-tool/rzfive-verified-linux-package-510-cip

- **Target board**
  RZ/Five reference board

  - RZ/Five Evaluation board Kit (smarc-rzfive) (*)
    - o RZ/Five SMARC Module Board (P/N: RTK9743F01C01000BE)
    - o RZ SMARC Series Carrier Board (P/N: RTK97X4XXXB00000BE)

(*)   "RZ/Five Evaluation board Kit" includes the RZ/Five SMARC Module Board and the RZ SMARC Series Carrier Board.
      The "Evaluation board Kit for RZ/Five MPU" will be called "RZ/Five Evaluation Kit" in the next section.

- **Functions**
  Linux BSP
  - Linux Kernel
  - Linux Drivers

- **File contents**
  BSP is delivered by the files listed in **Table 1**.

Table 1. RZ/Five Verified Linux Package

Basic packages

File	Description
RTK0EF0045Z0025AZJ-v3.0.2-update1.zip(*1)	RZ/Five Verified Linux Package. This file includes the **Yocto recipe packages** and the necessary documents.
rzfive_vlp_v3.0.2.tar.gz(*1)	**Yocto recipe packages**
oss_pkg_rzfive_v3.0.2.7z(*1)	Open source software packages See the Note below before download
r01us0608ej0101-rz-five(Release Note).pdf	This document
r01us0609ej0101-rz-five(Component List).pdf	Component list
r01us0556ej0103-rz-g(Board_StartUp_Guide_smarcEVK).pdf	Documents describing booting method and the required settings of bootloader for RZ/G2L, RZ/G2LC, RZ/G2UL, and **RZ/Five**.

Additional file of VLP/F v3.0.2-update1

File	Description
rzfive_v302-to-v302update1.patch	This is the patch to update BSP from VLP/F v3.0.2 to VLP/F v3.0.2-update1. VLP/F v3.0.2-update1 includes the bug fixe. Please refer to Appendix.

(*1) These packages are provided "AS IS" with no warranty and the license which is described in the source code. Please check the contents of the license, then consider the applicability to the product carefully.

Note) Open source software packages contain all source codes of OSSs. These are the same versions of OSSs used when VLP/F was verified.
   If you are just evaluating VLP/F and RZ/Five series, open source software 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 software packages are required for an "offline" environment. The word "offline" means an isolated environment which does not connect to any network. VLP/F 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 software 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/F supports can be built on an "offline" environment. Please refer to **2. Build environment**.

RENESAS

2.   Build environment

The environment for building the BSP is listed in Table 2. Please refer to the below documents for details about setting up the environment:

SMARC EVK of RZ/G2L, RZ/G2LC, RZ/G2UL, RZ/V2L, and RZ/Five Start-up Guide

A Linux PC is required for building the software.

A Windows PC can be used as the serial terminal interface with software such as TeraTerm.

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

Equipment		Description
Linux Host PC		Used as build/debug environment 100GB free space on HDD is necessary
	OS	Ubuntu 20.04 LTS 64 bit OS must be used.
Windows Host PC		Used as debug environment, controlling with terminal software
	OS	Windows 10
	Terminal software	Used for controling serial console of the target board Tera Term (latest version) is recommended Available at https://ttssh2.osdn.jp/index.html.en
	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: ● http://www.ftdichip.com/Drivers/VCP.htm

Most bootable images VLP/F supports can be built on an "offline" environment.
The word "offline" means an isolated environment which does not connect to any network. Since VLP/F includes all necessary source codes of OSS except for the Linux kernel, VLP/F can always build images in this "offline" environment without affected from changes of repositories of OSS. Also, this "offline" environment reproduces the same images as the images which were verified by Renesas.

Below images can be built "offline".
- core-image-minimal
- core-image-bsp

Below are not available in the "offline" environment. Please connect your Linux Host PC to the internet.
- Preparing a Linux Host PC

3.   Building Instructions of a BSP

3.1   Building images to run on the board

This section describes the instructions to build the Board Support Package (hereinafter referred to as "BSP"). Before starting the build, run the command below on the Linux Host PC to install packages used for building the BSP.

```
$ 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 libsdl1.2-dev xterm p7zip-full libyaml-dev \
libssl-dev
```

Please refer to the URL below for detailed information:

- https://docs.yoctoproject.org/3.1.5/brief-yoctoprojectqs/brief-yoctoprojectqs.html

Run the command**s** below and set the user name and email address before starting the 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"
```


Copy all files obtained from Renesas into your Linux Host PC prior to the steps below. The directory which you put the files in is described as <package download directory> in the build instructions. .

(1)   Create a working directory at your home directory, and decompress Yocto recipe package

Run the commands below. The name and the place of the working directory can be changed as necessary. Copy the compressed Yocto recipe package file (rzfive_vlp_v3.0.2.tar.gz) into your home directory prior to this step.

If you need update patch, please see Section7.

```
$ mkdir ~/rzfive_vlp_v3.0.2
$ cd ~/rzfive_vlp_v3.0.2
$ cp ../<package download directory>/*.zip .
$ unzip ./RTK0EF0045Z0025AZJ-v3.0.2-update1.zip
$ tar zxvf ./RTK0EF0045Z0025AZJ-v3.0.2-update1/rzfive_vlp_v3.0.2.tar.gz
```


(2)   Setup a build environment

Run the commands below. The environment to build is set by the source command.

```
$ source poky/oe-init-build-env
```


(3)   Prepare the default configuration files for the target board

Run the commands below.

```
$ cp ../meta-renesas/docs/template/conf/smarc-rzfive/*.conf  ./conf/
```

(4)   Decompress OSS files to "build" directory (Optional)

Run the commands below. This step is not mandatory and able to go to the step (5) in case the "offline" environment is not required. All OSS packages will be decompressed with this '7z' command.

```
$ cp ../../<package download directory>/*.7z .
$ 7z x ~/oss_pkg_rzfive_v3.0.2.7z
```

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.

After the above procedure is finished, the "offline" environment is ready. If you want to prevent network access, please change the line in the "`~/rzfive_vlp_v3.0.2/build/conf/local.conf`" as below:

```
BB_NO_NETWORK = "1"
```

Change BB_NO_NETWORK from "0" to "1".
When you applying the update1 patch which is described in the section 7 Appendix, set BB_NO_NETWORK = "0".

(5)   Start a build

Run the commands below to start a build. Building a minimal image can take up to a few hours depending on the user's host system performance.

```
$ bitbake core-image-minimal
```

Run the commands below to start a build. Building a bsp image can take up to a few hours depending on the user's host system performance.

```
$ bitbake core-image-bsp
```

After the build is successfully completed, a similar output will be seen, and the command prompt will return.

```
NOTE: Tasks Summary: Attempted 3788 tasks of which 265 didn't need to be rerun and all succeeded.
```

All necessary files listed in **Table 4** will be generated by the bitbake command and will be located in the **build/tmp/deploy/images** directory.

VLP/F can build a few types of images listed in **Table 3**. Please refer to the "Component list" for details about components of each image.

For the booting method and the required settings, please refer to the "SMARC EVK of RZ/G2L, RZ/G2LC, RZ/G2UL, RZ/V2L, and RZ/Five Start-up Guide (R01US0556)".

The loader files written at the time of mass production may not be the latest. When using EVK for the first time, be sure to write Boot loader files generated by building latest BSP to your board.

Table 3. Supported images of VLP/F

Image name	Description
core-image-minimal	Image which includes minimal set of components
core-image-bsp	Image which includes minimal set of components, audio components, and some useful tools

Table 4. Image files for RZ/Five

RZ/Five	Linux kernel	Image-smarc-rzfive.bin
	Device tree file	Image-r9a07g043f01-smarc.dtb
	root filesystem	*<image name>*-smarc-rzfive.tar.bz2
	Boot loader	• fit-smarc-rzfive.srec • spl-smarc-rzfive.srec
	Flash Writer	Flash_Writer_SCIF_RZFIVE_SMARC.mot

3.2   Building SDK

To build Software Development Kit (SDK), run the commands below after the steps (1) – (5) of section **3.1** are finished.

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 ~/rzfive_vlp_v3.0.2/build
$ bitbake core-image-minimal -c populate_sdk
```

For building bsp applications:

```
$ cd ~/rzfive_vlp_v3.0.2/build
$ bitbake core-image-bsp -c populate_sdk
```

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 sh poky-glibc-x86_64-core-image-minimal-riscv64-smarc-rzfive-toolchain-3.1.1
7.sh
```

Or

```
$ sudo sh poky-glibc-x86_64-core-image-bsp-riscv64-smarc-rzfive-toolchain-3.1.17.sh
```

Note)   The SDK build may fail depending on the build environment. At that time, please run the build again after a period of time. Or build it again from scratch with the below commands.

```
$ cd ~/rzfive_vlp_v3.0.2/build
$ bitbake core-image-minimal -c cleanall
$ bitbake core-image-minimal
```

For building general applications:

```
$ bitbake core-image-minimal -c populate_sdk
```

For building bsp applications:

```
$ bitbake core-image-bsp -c populate_sdk
```

RENESAS

4. Components

The components which are commonly used in this release are listed in Table 5. Please also refer to the "Component list" for details.

Table 5. Versions of commonly used components

Components	RZ/Five BSP v1.0-update1	VLP/F v3.0.2	VLP/F v3.0.2-update1
Linux kernel	5.10.83-cip1	5.10.145-cip17	5.10.145-cip17
GCC	8.3.0 (RISC-V GCC 8.3-2019.03)	8.3.0 (RISC-V GCC 8.3-2019.03)	8.3.0 (RISC-V GCC 8.3-2019.03)
glibc	2.28	2.28	2.28
busybox	1.31.1	1.30.1	1.30.1
openssl	1.1.1d	1.1.1n	1.1.1n

5.   Restrictions

(1) Docker

Docker container is not supported.

(2) EMMC boot

EMMC boot will be supported in the next release.

6. Notes

(1) GPLv3 packages

In this release, the GPLv3 packages are disabled as default in *build/conf/local.conf*:

```
INCOMPATIBLE_LICENSE = "GPLv3 GPLv3+"
```

If you want to use GPLv3, just hide this line:

```
#INCOMPATIBLE_LICENSE = "GPLv3 GPLv3+"
```

(2) CIP Core Packages

VLP/F includes Debian 10 (Buster) based CIP Core Packages and is enabled by the default settings. These packages can be changed.

Note that network access is required to start the build process when you enable these packages except for Buster which is set as the default setting.

CIP Core Packages are going to be maintained by the Civil Infrastructure Platform project. For more technical information, please contact Renesas.

1. Buster (default):

The following lines are added as default in the `local.conf`:

```
# Select CIP Core packages
CIP_CORE = "1"
```

2. Bullseye:

Please change "`CIP_MODE`" in the **local.conf** to change from Buster to Bullseye:

```
# Select CIP Core packages by switching betwwen Buster and Bullseye.
#  - Buster (default)   : build all supported Debian 10 Buster recipes
#  - Bullseye           : build all supported Debian 11 Bullseye recipes
#  - Not set (or different with above): not use CIP Core, use default packages
version in Yocto


CIP_MODE = "Bullseye"
```

3. No CIP Core Packages:

If the CIP Core Packages are unnecessary, comment out and add the following lines to disable CIP Core Packages in the `local.conf`:

```
# Select CIP Core packages
#CIP_CORE = "1"
```

Note)   The above 2 settings disable GPLv3 packages as default. In case the GPLv3 packages are required, please comment out the following line in the `local.conf`.

```
# INCOMPATIBLE_LICENSE = "GPLv3 GPLv3+"
```

By building the BSP, the packages will be replaced as below in the table.

Table 6. Versions of all Buster Debian packages

Package	Buster Debian	Bullseye Debian
Attr	2.4.48	2.4.48
busybox	1.30.1	1.30.1
coreutils	6.9	6.9
GCC	8.3.0	-
glib-2.0	2.58.3	2.62.2
glibc	2.28	2.31
kbd	2.0.4	2.2.0
libgcrypt	1.8.4	1.8.5
openssh	7.9p1	8.2p1
Perl	5.30.1	5.30.1
pkgconfig	0.29	0.29.2
Quilt	0.65	0.66

7. Appendix

- **Update patch for VLP/F v3.0.2**

  Renesas released VLP/F v3.0.2 in the end of December 2022. After that, the patch file is released to update VLP/F v3.0.2 to VLP/F v3.0.2-update1.

  Update points of VLP/F v3.0.2-update1

  - Update the OpenSBI to set the Instruction local memory and Data local memory (ILM & DLM) region available for M-mode only to avoid unexpected access from S/U-modes.

  Copy the update package file (RTK0EF0045Z0025AZJ-v3.0.2-update1.zip) obtained from Renesas into your home directory prior to the steps below.

- **How to use the patch file**

  Please run the commands below instead of the step (1) in the section 3.1.

```
$ mkdir ~/rzfive_vlp_v3.0.2
$ cd ~/rzfive_vlp_v3.0.2
$ cp ../<package download directory>/*.zip .
$ unzip ./RTK0EF0045Z0025AZJ-v3.0.2-update1.zip
$ tar zxvf ./RTK0EF0045Z0025AZJ-v3.0.2-update1/rzfive_vlp_v3.0.2.tar.gz
$ patch -p1 < ./RTK0EF0045Z0025AZJ-v3.0.2-update1/rzfive_v302-to-v302update1.patch
```

RENESAS

8. Revision History

		Description	
Rev.	**Date**	**Page**	**Summary**
1.00	Dec. 27, 2022	-	First edition for VLP/F v3.0.2
1.01	Feb. 10, 2023	13	Add the appendix section for VLP/F v3.0.2-update1.

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.