

Building Genivi 14 (Pulsar) for the Renesas R-Car SoCs

This page describes building Genivi 14 (Pulsar) platforms using Renesas R-Car SoC BSPs.

Specifically the page was written for the R-Car Generation 3 SoCs: H3, M3, M3N and E3

Contents

- [1 Contents](#)
- [2 General \(read first\)](#)
 - [2.1 Yocto BSP Compatibility](#)
- [3 Specific Genivi Platforms](#)
 - [3.1 Genivi Development Platform \(GDP\)](#)
 - [3.2 Genivi Yocto Baseline \(GYB\)](#)

General (read first)

This section provides information that applies to all platforms. Information for specific platforms, such as the Genivi Yocto Baseline (meta-ivi) and GDP, is covered in the following section. Please read this section first.

Yocto BSP Compatibility

The standard Renesas customer Yocto BSP meets the functional and version requirements of Genivi 14 compliance. The kernel simply needs to be configured for the requirements of the Genivi 14 platform.

At a minimum the following kernel config options are enabled:

- FHANDLE
- EXPERIMENTAL
- ECRYPT_FS
- QUOTA
- BT
- CGROUPS

The Genivi 14 specification does not specify a build system and instead allows the implementer the freedom to choose what fits their needs best. The Genivi implementations of the specification in the shape of the Genivi Yocto Baseline and Genivi Development Platform both use Yocto Project 2.4 (Rocko).

At the time of writing the standard R-Car Yocto BSP from v3.4.0 to v3.9.0 provide Yocto Project 2.4 support. Then as a service to the community Renesas maintains the Yocto layer meta-ivi-renesas which adapts the Yocto BSP to the needs of the Genivi Yocto Baseline. The combination of the two layers can then be used to build Genivi 14 software. Renesas works in the Genivi Baseline Integration Team (BIT) and Genivi Development Platform (GDP) to ensure that the combination can be used to build those images for example.

Here are the details:

| Purpose | Repository location | Branch |
|--|---|-------------|
| Upstream Yocto BSP support | https://github.com/renesas-rcar/meta-renesas.git | rocko |
| Adapt Yocto BSP (meta-renesas) to Genivi Yocto Baseline (meta-ivi) | https://github.com/GENIVI/meta-ivi-renesas.git | genivi-14.x |

Specific Genivi Platforms

This section provides information for specific Genivi platforms such as the Genivi Baseline and GDP.

Genivi Development Platform (GDP)

Instructions for building and deploying GDP on the R-Car H3/M3 Starter Kit and H3/M3 Salvator-X boards are maintained in the GDP wiki pages. At the time of writing GDP is being rebased on the Genivi 14 Yocto Baseline. The current working branch can be found in the [14.0.x-rocko](#) branch of GDP and a snapshot has been merged into the master branch.

Instructions for building and deploying GDP on the R-Car H3/M3 Starter Kit and H/3M3 Salvator-X boards are maintained in the [GDP Master](#) wiki page.

Issues are maintained in the [Genivi JIRA GDP project](#).

Genivi Yocto Baseline (GYB)

For those familiar with Yocto and the GYB you can find the Pulsar Renesas Yocto BSP details for Gen 3 in the table below. At the time of writing they have been successfully tested with meta-ivi 14.0.0 (P-1.0).

Steps:

1) Clone and checkout Genivi Yocto Baseline repositories

See the meta-ivi [Readme.md](#) for details.

2) Clone and checkout the Renesas required layers.

Note: With YP 2.4 meta-renesas switched from the linaro gcc toolchain to using the poky gcc toolchain by default. You therefore no longer need to use the meta-linaro-toolchain layers.

| R-Car Generation | Repository Location | Branch |
|-----------------------------|---|-------------|
| R-Car Gen 3 (H3/M3/E3 SoCs) | https://github.com/renesas-rcar/meta-renesas.git | rocko |
| | https://github.com/GENIVI/meta-ivi-renesas.git | genivi-14.x |
| | https://git.linaro.org/openembedded/meta-linaro.git | rocko |

3) Obtain and install the Renesas R-Car Graphics and Multimedia packages

For R-Car Gen 3 instructions for installing the click-through packages can be found on elinux.org.

Download the R-Car Gen 3 Linux Drivers and Gfx/MMP packages for **YBSP v3.9.0** and **Wayland 1.13 / Weston 2.0** from [here](#).

4) You can now follow the build instructions in the meta-ivi [Readme.md](#) .

4a) Export TEMPLATECONF to pick up correct configuration for the build

```
export TEMPLATECONF=/full/path/to/meta-ivi/meta-ivi/conf
```

4b) Run the following command:

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

4c) Setup your Yocto local.conf and bblayers.conf

For R-Car Gen 3 add the following to bblayers.conf:

```
BBLAYERS += " \
    ${TOPDIR}/../meta-renesas/meta-rcar-gen3 \
    ${TOPDIR}/../meta-openembedded/meta-oe \
    ${TOPDIR}/../meta-openembedded/meta-fileSystems \
    ${TOPDIR}/../meta-linaro/meta-optee \
    ${TOPDIR}/../meta-ivi-renesas \
"
```

Examples of the additions to add to your Yocto local.conf:

| | M3 Starter Kit |
|------------|----------------------|
| Local.conf | link |

At the time of writing meta-ivi does not add the weston-init package required to start weston using systemd to the rootfs by default. If you would like to run gfx applications, such as the wayland ivi-extension mock navigation application, please add the following line to your local.conf so weston-init is added:

```
IMAGE_INSTALL_append = " weston-init"
```

To add the R-Car gfx and mmp packages to the image please add the following line to your local.conf:

```
# Add R-Car Gfx and MMP packages to install
IMAGE_INSTALL_append_rcar-gen3 = " \
    packagegroup-gstreamer1.0-plugins \
    packagegroup-wayland-community \
    packagegroup-graphics-renesas-proprietary \
    packagegroup-graphics-renesas-wayland \
"
```

4d) Build the baseline including the Genivi components

```
> $ bitbake pulsar-image
```