RPI 4B Model Reference Platform

- Host Setup
- Repo Setup
- Yocto Build Steps
- Flashing Procedure
- Validated Functionalities

Introduction

This manual describes the rdk-generic-broadband-image build procedure for Yocto 3.1 builds. This will upgrade the default build system used in RDKB, from morty(2.2) to dunfell(3.1). In addition to upgrades to opensource packages, Yocto 3.1 also contain Long term Support release feature. The current version lacks the variety of features available in the du';po-0=nfell. The upgraded build tools will be used to generate rdk-generic-broadband-image for Raspberrypi reference platform.

Yocto 3.1 Upgradation support the following:

- Yocto BSP layer for meta-raspberrypi.
- OpenEmbedded and Yocto Dunfell.
- Linux kernel 5.4.72.
- Version upgrades for bitbake, and other oe/python recipes.

Each component in RDK is a standalone repository with its own individual build tools producing a library or set of binaries. When we upgrade the OE layers to the newer versions, we need to make necessary changes in the RDK layers which use these components, to avoid build failures.

Host Setup

The OpenEmbedded build system should be able to run on Ubuntu 18.04 distribution/other compatible linux distribution with the following versions for Git, tar, and Python.

- Git 1.8.3.1 or greater
- tar 1.27 or greater
- Python 3.4.0 or greater
- Coreutils(E.g realpath)

Note: You should also have about 50 Gbytes of free disk space for building images.

The essential packages you need for a supported Ubuntu or Debian distribution are shown in the following command:

```
$ 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 python3-git python3-jinja2 libegl1-mesa libsdl1.2-dev \
  pylint3 xterm bmap-tools

$ sudo apt-get install git cmake autoconf texinfo openjdk-8-jdk openjdk-8-jre \
  m4 libtool libtool-bin curl pkg-config lib32z1 doxygen
```

Repo Setup

In order to use Yocto build system, the repo tool must be properly installed on the machine.

To install Repo make sure you have a /bin directory in your home directory and that it is included in your path

Repo Setup Steps

```
$ mkdir ~/bin
$ PATH=~/bin:$PATH
Download the repo tool and ensure that it is executable
$ curl http://commondatastorage.googleapis.com/git-repo-downloads/repo > ~/bin/repo
$ chmod a+x ~/bin/repo
```

Note: it is also recommended to put credentials in ~/.netrc when interacting with repo.

A sample **~/.netrc** file is illustrated below

.netrc

```
machine code.rdkcentral.com
login YOUR_USERNAME
password YOUR_PASSWORD
```

Yocto Build Steps

To build, follow below instructions

Build Steps

```
$ mkdir <workspace dir>
$ cd <workspace dir>
$ repo init -u https://code.rdkcentral.com/r/manifests -b dunfell -m rdkb-extsrc.xml
$ repo sync -j`nproc` --no-clone-bundle
$ MACHINE=raspberrypi4-rdk-broadband source meta-cmf-raspberrypi/setup-environment
$ bitbake rdk-generic-broadband-image
```

Flashing Procedure

Following command can be used to flash the RPI image to sd card using linux machine . bmap tool should be available in linux

Flash command

```
bzip2 -d <path to ImageName.wic.bz2>
sudo -E bmaptool copy --nobmap <path to ImageName.wic> <path to SD card space>

Example:

$ bzip2 -d rdk-generic-broadband-image-raspberrypi-rdk-broadband.wic.bz2
$ sudo -E bmaptool copy --nobmap rdk-generic-broadband-image-raspberrypi-rdk-broadband.wic.bz2 /dev/sdb
```

Validated Functionalities

No	Feature	Supported
1	LAN Connected Devices-Ethernet	✅
2	WAN Connected Devices-Wi-Fi	✅
3	Parental Control	✅
4	Firewall settings	✅
5	Advanced Config: Port Triggering	❌
6	Advanced Config: Port Forwarding	✅
7	Advanced Config: Remote Management	✅
8	Advanced Config: DMZ	✅
9	Xfinity Wi-Fi 2.4/5 GHz – Public Hotspot	❌
10	Test and Diagnostics	✅
11	Local WebUI Configuration	✅
12	Factory Reset	✅

13	DHCP /Reserved IP	✓
14	EthWan	✓
15	Eth Agent	✓
16	2.4 GHz Band Support	✓
17	5 GHz Band Support	✓
18	Bridge Mode Support	✓
19	Persistent Storage Management	✓
20	WebPA for Comcast,community	✓
21	Lost and Found	✗
22	Bluetooth	✗
22	Harvester Support	✗
23	TR-69	✗
24	SNMP	✓
27	Boot time data measurement	✗
28	Wireless Protection Setup(WPS - 2G support)	✓
29	Captive Portal	✓
30	Wi-Fi MAC Filtering	✗
31	Log Rotation Support	✓
32	Firmware Upgrade Support	✓
33	Multiboot Support	✓
34	Telemetry Support (T2 single profile support)	✓
35	IPV6	✗
36	Log level control through CcspLogAgent	✓
37	Band Steering using RSSI	✗
38	Backup and Restore	✗
39	SelfHeal	✓
40	Password-Reset	✓
41	WebUI JST	✓