meta-raspberrypi Documentation

Release rocko

meta-raspberrypi contributors

Sep 06, 2018

Contents

Contents:

CHAPTER 1

meta-raspberrypi

Yocto BSP layer for the Raspberry Pi boards - http://www.raspberrypi.org/.

1.1 Quick links

- Git repository web frontend: https://github.com/agherzan/meta-raspberrypi
- Mailing list (yocto mailing list): yocto@yoctoproject.org
- Issues management (Github Issues): https://github.com/agherzan/meta-raspberrypi/issues
- Documentation: http://meta-raspberrypi.readthedocs.io/en/rocko/

1.2 Description

This is the general hardware specific BSP overlay for the RaspberryPi device.

More information can be found at: http://www.raspberrypi.org/ (Official Site)

The core BSP part of meta-raspberrypi should work with different OpenEmbedded/Yocto distributions and layer stacks, such as:

- Distro-less (only with OE-Core).
- Angstrom.
- Yocto/Poky (main focus of testing).

1.3 Dependencies

This layer depends on:

- URI: git://git.yoctoproject.org/poky
  - branch: rocko
  - revision: HEAD
- URI: git://git.openembedded.org/meta-openembedded
  - layers: meta-oe, meta-multimedia, meta-networking, meta-python
  - branch: rocko
  - revision: HEAD

1.4 Quick Start

1. source poky/oe-init-build-env rpi-build
2. Add this layer to bblayers.conf and the dependencies above
3. Set MACHINE in local.conf to one of the supported boards
4. bitbake rpi-hwup-image
5. dd to a SD card the generated sdimg file (use xzcat if rpi-sdimg.xz is used)
6. Boot your RPI.

1.5 Maintainers

- Andrei Gherzan <andrei at gherzan.ro>

CHAPTER 2

Layer Contents

2.1 Supported Machines

- raspberrypi
- raspberrypi0
- raspberrypi0-wifi
- raspberrypi2
- raspberrypi3
- raspberrypi3-64 (64 bit kernel & userspace)
- raspberrypi-cm (dummy alias for raspberrypi)
- raspberrypi-cm3 (dummy alias for raspberrypi2)

Note: The raspberrypi3 machines include support for Raspberry Pi 3B+.

2.2 Images

- rpi-hwup-image
  - Hardware up image
- rpi-basic-image
  - Based on rpi-hwup-image with some added features (ex: splash)
- rpi-test-image
  - Image based on rpi-basic-image which includes most of the packages in this layer and some media samples.

For other uses it's recommended to base images on `core-image-minimal` or `core-image-base` as appropriate. The old image names (`rpi-hwup-image` and `rpi-basic-image`) are deprecated.

2.3 WiFi firmware blobs

Be aware that the WiFi firmwares for the supported boards are not provided by `linux-firmware` but by a custom recipe which only packages the needed blobs for these boards. This is because the upstream `linux-firmware` doesn't support or has outdated files for the blobs we need. The recipe `linux-firmware-raspbian` is based on a fork of `linux-firmware` which includes everything we need in order to fully support the WiFi hardware. All machines define `MACHINE_EXTRA_RRECOMMENDS` accordingly.

CHAPTER 3

Optional build configuration

There are a set of ways in which a user can influence different paramenters of the build. We list here the ones that are closely related to this BSP or specific to it. For the rest please check: http://www.yoctoproject.org/docs/latest/ref-manual/ref-manual.html

3.1 Compressed deployed files

1. Overwrite IMAGE_FSTYPES in local.conf
   - `IMAGE_FSTYPES = "tar.bz2 ext3.xz"`
2. Overwrite SDIMG_ROOTFS_TYPE in local.conf
   - `SDIMG_ROOTFS_TYPE = "ext3.xz"`
3. Overwrite SDIMG_COMPRESSION in local.conf
   - `SDIMG_COMPRESSION = "xz"`

Accommodate the values above to your own needs (ex: ext3 / ext4).

3.2 GPU memory

- `GPU_MEM`: GPU memory in megabyte. Sets the memory split between the ARM and GPU. ARM gets the remaining memory. Min 16. Default 64.
- `GPU_MEM_256`: GPU memory in megabyte for the 256MB Raspberry Pi. Ignored by the 512MB RP. Overrides gpu_mem. Max 192. Default not set.
- `GPU_MEM_512`: GPU memory in megabyte for the 512MB Raspberry Pi. Ignored by the 256MB RP. Overrides gpu_mem. Max 448. Default not set.
- `GPU_MEM_1024`: GPU memory in megabyte for the 1024MB Raspberry Pi. Ignored by the 256MB/512MB RP. Overrides gpu_mem. Max 944. Default not set.

3.3 Add purchased license codecs

To add you own licenses use variables KEY_DECODE_MPG2 and KEY_DECODE_WVC1 in local.conf. Example:

```
KEY_DECODE_MPG2 = "12345678"
KEY_DECODE_WVC1 = "12345678"
```

You can supply more licenses separated by comma. Example:

```
KEY_DECODE_WVC1 = "0x12345678,0xabcdabcd,0x87654321"
```

3.4 Disable overscan

By default the GPU adds a black border around the video output to compensate for TVs which cut off part of the image. To disable this set this variable in local.conf:

```
DISABLE_OVERSCAN = "1"
```

3.5 Set overclocking options

The Raspberry PI can be overclocked. As of now overclocking up to the "Turbo Mode" is officially supported by the raspbery and does not void warranty. Check the config.txt for a detailed description of options and modes. Example turbo mode:

```
ARM_FREQ = "1000"
CORE_FREQ = "500"
SDRAM_FREQ = "500"
OVER_VOLTAGE = "6"
```

3.6 Video camera support with V4L2 drivers

Set this variable to enable support for the video camera (Linux 3.12.4+ required):

```
VIDEO_CAMERA = "1"
```

3.7 Enable offline compositing support

Set this variable to enable support for dispmanx offline compositing:

```
DISPMANX_OFFLINE = "1"
```

This will enable the firmware to fall back to off-line compositing of Dispmanx elements. Normally the compositing is done on-line, during scanout, but cannot handle too many elements. With off-line enabled, an off-screen buffer is allocated for compositing. When scene complexity (number and sizes of elements) is high, compositing will happen off-line into the buffer.

Heavily recommended for Wayland/Weston.

See: http://wayland.freedesktop.org/raspberrypi.html

3.8 Enable kgdb over console support

To add the kdbg over console (kgdboc) parameter to the kernel command line, set this variable in local.conf:

```
ENABLE_KGDB = "1"
```

3.9 Disable rpi boot logo

To disable rpi boot logo, set this variable in local.conf:

```
DISABLE_RPI_BOOT_LOGO = "1"
```

3.10 Boot to U-Boot

To have u-boot load kernel image, set in your local.conf:

```
RPI_USE_U_BOOT = "1"
```

This will select the appropriate image format for use with u-boot automatically. For further customisation the KERNEL_IMAGETYPE and KERNEL_BOOTCMD variables can be overridden to select the exact kernel image type (eg. zImage) and u-boot command (eg. bootz) to be used.

3.11 Image with Initramfs

To build an initramfs image:

- Set this 3 kernel variables (in linux-raspberrypi.inc for example)
  - kernel_configure_variable BLK_DEV_INITRD y
  - kernel_configure_variable INITRAMFS_SOURCE “”
  - kernel_configure_variable RD_GZIP y
- Set the yocto variables (in linux-raspberrypi.inc for example)
  - `INITRAMFS_IMAGE = "<a name for your initramfs image>"`
  - `INITRAMFS_IMAGE_BUNDLE = "1"`
- Set the meta-rasberrypi variable (in raspberrypi.conf for example)
  - `KERNEL_INITRAMFS = "-initramfs"`

3.12 Enable SPI bus

When using device tree kernels, set this variable to enable the SPI bus:

```
ENABLE_SPI_BUS = "1"
```

3.13 Enable I2C

When using device tree kernels, set this variable to enable I2C:

```
ENABLE_I2C = "1"
```

3.14 Enable PiTFT support

Basic support for using PiTFT screens can be enabled by adding below in local.conf:

- `MACHINE_FEATURES += "pitft"`
  - This will enable SPI bus and i2c device-trees, it will also setup framebuffer for console and x server on PiTFT.

NOTE: To get this working the overlay for the PiTFT model must be build, added and specified as well (dtoverlay= in config.txt).

Below is a list of currently supported PiTFT models in meta-raspberrypi, the modelname should be added as a MACHINE_FEATURES in local.conf like below:

```
MACHINE_FEATURES += "pitft <modelname>"
```

List of currently supported models:

- pitft22
- pitft28r
- pitft35r

3.15 Misc. display

If you would like to use the Waveshare "C" 1024×600, 7 inch Capacitive Touch Screen LCD, HDMI interface (http://www.waveshare.com/7inch-HDMI-LCD-C.htm) Rev 2.1, please set the following in your local.conf:

```
WAVESHARE_1024X600_C_2_1 = "1"
```

3.16 Enable UART

RaspberryPi 0, 1, 2 and CM will have UART console enabled by default.

RaspberryPi 0 WiFi and 3 does not have the UART enabled by default because this needs a fixed core frequency and enable_uart wil set it to the minimum. Certain operations - 60fps h264 decode, high quality deinterlace - which aren't performed on the ARM may be affected, and we wouldn't want to do that to users who don't want to use the serial port. Users who want serial console support on RaspberryPi3 will have to explicitely set in local.conf:

```
ENABLE_UART = "1"
```

Ref.:

- https://github.com/raspberrypi/firmware/issues/553
- https://github.com/RPi-Distro/repo/issues/22

CHAPTER 4

Extra apps

4.1 omxplayer

omxplayer depends on libav which has a commercial license. So in order to be able to compile omxplayer you will need to whiteflag the commercial license in your local.conf:

```
LICENSE_FLAGS_WHITELIST = "commercial"
```

CHAPTER 5

Contributing

5.1 Mailing list

The main communication tool we use is a mailing list:

- yocto@yoctoproject.org
- https://lists.yoctoproject.org/listinfo/yocto

Feel free to ask any kind of questions but always prepend your email subject with "[meta-raspberrypi]". This is because we use the 'yocto' mailing list and not a perticular 'meta-raspberrypi' mailing list.

5.2 Patches and pull requests

All the contributions should be compliant with the openembedded patch guidelines: http://www.openembedded.org/wiki/Commit_Patch_Message_Guidelines

To contribute to this project you should send pull requests to the github mirror (https://github.com/agherzan/meta-raspberrypi). **Additionally** you can send the patches for review to the above specified mailing list.

When creating patches for the mailing list, please use something like:

```
git format-patch -s --subject-prefix='meta-raspberrypi][PATCH' origin
```

When sending patches to the mailing list, please use something like:

```
git send-email --to yocto@yoctoproject.org <generated patch>
```

5.3 Github issues

In order to manage and trace the meta-raspberrypi issues, we use github issues: https://github.com/agherzan/meta-raspberrypi/issues

If you push patches which have a github issue associated, please provide the issue number in the commit log just before “Signed-off-by” line(s). Example line for a bug: `[Issue #13]`

CHAPTER 6

Indices and tables

- genindex
- modindex
- search