

**+44 1590 610184 (tel:+44 1590 610184)**

**Embecosm** (<https://www.embecosm.com>) > **Blog** (<https://www.embecosm.com/blog/>) > **Embedded OS** ([/category/embedded\\_os/](/category/embedded_os/)) > Adding RISC-V 64-bit Support to Buildroot

---

# Adding RISC-V 64-bit Support to Buildroot

On September 19, 2018 By [Mark Corbin](https://www.embecosm.com/author/mark-corbin/)

## Topics

[Buildroot](/tag/buildroot/) [Embedded Linux](/tag/embedded-linux/)

[Embedded OS](/category/embedded_os/) [Embedded OS](/tag/embedded-os/)

[open source](/tag/open-source/) [RISC\\_V](/category/risc_v/) [RISC-V](/tag/risc-v/)

## Share this article



(<https://www.facebook.com/sharer.php?u=https://www.embecosm.com/2018/09/19/adding-risc-v-64-bit-support-to-buildroot/>)



(<https://www.linkedin.com/shareArticle?url=&title=>



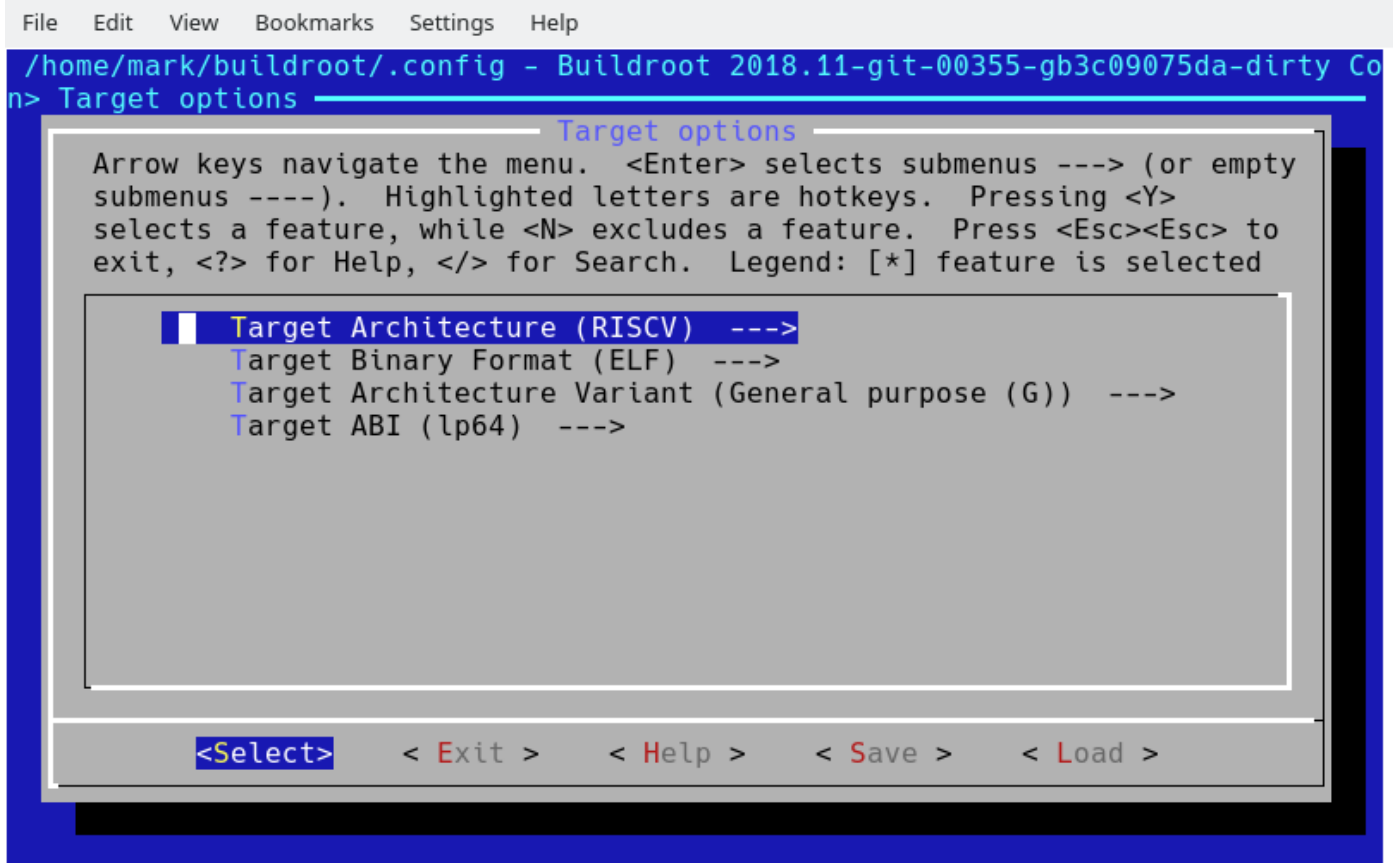
[https://twitter.com/intent/tweet?url=https://www.embecosm.com/2018/09/19/adding-risc-v-64-bit-support-to-buildroot/&text=Adding RISC-V 64-bit Support to Buildroot&via=embecosm](https://twitter.com/intent/tweet?url=https://www.embecosm.com/2018/09/19/adding-risc-v-64-bit-support-to-buildroot/&text=Adding%20RISC-V%2064-bit%20Support%20to%20Buildroot&via=embecosm))



(<https://plus.google.com/share?url={post.link}>)



([mailto:?subject=Adding RISC-V 64-bit Support to Buildroot&body=https://www.embecosm.com/2018/09/19/adding-risc-v-64-bit-support-to-buildroot/](mailto:?subject=Adding%20RISC-V%2064-bit%20Support%20to%20Buildroot&body=https://www.embecosm.com/2018/09/19/adding-risc-v-64-bit-support-to-buildroot/))



## Background

I have been working on adding RISC-V 64-bit architecture support to [Buildroot](https://buildroot.org/) (<https://buildroot.org/>).

Buildroot is an embedded Linux build system that generates complete system images from source for a wide range of boards and processors. Buildroot uses makefiles and is configured using the [kernel kconfig language](https://www.kernel.org/doc/Documentation/kbuild/kconfig-language.txt) (<https://www.kernel.org/doc/Documentation/kbuild/kconfig-language.txt>). If you have ever configured and built a Linux kernel from source you will find the interface very familiar.

## Status of RISC-V Software

As with any embedded Linux system there are a number of choices to be made regarding which component 'parts' to use; tool chain, C library, bootloader, kernel, ... etc. Sometimes these choices are made for you based on existing platform or feature support, sometimes choice is down to personal preference and sometimes the choice is partially dictated by which components or versions 'play nicely' together. In the case of RISC-V a number of the component parts are still evolving and have not yet been upstreamed, so this seriously restricts the range of options to choose from.

Support for RISC-V has been available in [gcc since version 7.1 \(https://gcc.gnu.org/gcc-7/changes.html\)](https://gcc.gnu.org/gcc-7/changes.html), so the Buildroot default 7.x or 8.x gcc versions can be used without any modification. However GNU binutils needs to be 2.30 or greater in order for the kernel to build correctly.

Buildroot offers support for three C libraries (uclibc-ng, musl and glibc), however only glibc is available upstream with RISC-V support...so no choice to be made here.

RISC-V kernels currently use the [Berkeley Boot Loader \(https://github.com/riscv/riscv-pk\)](https://github.com/riscv/riscv-pk) (BBL) to load and boot the image. The kernel vmlinux file needs to be merged with the BBL code during the BBL build process. BBL is not currently available under Buildroot, so I added it as a bootloader package.

The RISC-V architecture has been supported in the mainline kernel from v4.15 onwards, however the upstream kernels won't yet boot under QEMU. The latest updates in the [riscv-linux \(https://github.com/riscv/riscv-linux\)](https://github.com/riscv/riscv-linux) repository to support QEMU are expected to be included in the 4.19 mainline kernel release. Buildroot offers flexible options for building custom kernels, so configuring a build from the riscv-linux git repository was straightforward. Building the riscv-linux-4.15 branch worked without any problems. I also added the ability for Buildroot to build tool chains against custom kernel headers. This allows a tool chain to be built without having to build a kernel.

## Buildroot Modifications

The modifications made to Buildroot fall into three areas:

1. Add RISC-V configuration options

These changes allow the user to select and configure the appropriate RISC-V settings which are then used to set the correct internal Buildroot build variables.

2. Add riscv-pk (BBL) package

The RISC-V PK (Proxy Kernel) package was added as a bootloader to provide BBL. When selected, this will build a bootable 'bbl' image using the kernel vmlinux file as a payload.

3. Add a QEMU default configuration

A default configuration was added that will build a bootable system for riscv64 on a QEMU virtual machine. This configuration has been tested under QEMU version 2.12.1.

A patch set has been submitted to the [Buildroot mailing list](http://lists.busybox.net/pipermail/buildroot/2018-September/230689.html) (<http://lists.busybox.net/pipermail/buildroot/2018-September/230689.html>). In the meantime, a fully patched version based on Buildroot 2018.08-rc1 is available from the [riscv-start branch](https://github.com/riscv/riscv-buildroot/tree/riscv-start) (<https://github.com/riscv/riscv-buildroot/tree/riscv-start>) of the RISC-V repository on GitHub.

## Using Buildroot for RISC-V

Check the [Buildroot system requirements](https://buildroot.org/downloads/manual/manual.html#requirement) (<https://buildroot.org/downloads/manual/manual.html#requirement>) to ensure that you have the necessary host packages installed.

### Get the riscv-buildroot repository

```
cd ~  
git clone https://github.com/riscv/riscv-buildroot  
cd riscv-buildroot  
git checkout riscv-start
```

### Set the default configuration

```
make qemu_riscv64_virt_defconfig
```

### Configure any additional options or packages (if required)

```
make menuconfig
```

### Build the system

```
make
```

### Boot the image under QEMU

```
qemu-system-riscv64 -M virt -kernel output/images/bbl -append "root=/dev/vda ro console=ttyS0"
```

```
File Edit View Bookmarks Settings Help
one is ready
bootconsole [early0] disabled
bootconsole [early0] disabled
EXT4-fs (vda): couldn't mount as ext3 due to feature incompatibilities
EXT4-fs (vda): mounted filesystem without journal. Opts: (null)
VFS: Mounted root (ext4 filesystem) readonly on device 254:0.
devtmpfs: mounted
Freeing unused kernel memory: 136K
This architecture does not have kernel memory protection.
EXT4-fs (vda): warning: mounting unchecked fs, running e2fsck is recommended
EXT4-fs (vda): re-mounted. Opts: block_validity,delalloc,barrier,user_xattr
Starting logging: OK
Initializing random number generator... done.
Starting network: IPv6: ADDRCONF(NETDEV_UP): eth0: link is not ready
udhcpd: started, v1.29.2
IPv6: ADDRCONF(NETDEV_CHANGE): eth0: link becomes ready
udhcpd: sending discover
udhcpd: sending select for 10.0.2.15
udhcpd: lease of 10.0.2.15 obtained, lease time 86400
deleting routers
adding dns 10.0.2.3
OK

Welcome to Buildroot
buildroot login: █
```

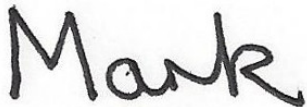
Booting under QEMU

The login username is 'root' with no password.

## Summary

RISC-V 64-bit support has been added to Buildroot and can be evaluated with QEMU by following the instructions above. The system can also be used to build a standalone tool chain for other development work.

Please get in touch if you have any questions regarding RISC-V support for Buildroot, or if you would like to know more about our range of Embedded OS services and world-class compiler technology.



Share this article



([https://www.facebook.com/sharer.php?](https://www.facebook.com/sharer.php?u=https://www.embecosm.com/2018/09/19/adding-risc-v-64-bit-support-to-buildroot/)

[u=https://www.embecosm.com/2018/09/19/adding-risc-v-64-bit-support-to-buildroot/](https://www.embecosm.com/2018/09/19/adding-risc-v-64-bit-support-to-buildroot/))



(<https://www.linkedin.com/shareArticle?url=&title=>



[https://twitter.com/intent/tweet?url=https://www.embecosm.com/2018/09/19/adding-](https://twitter.com/intent/tweet?url=https://www.embecosm.com/2018/09/19/adding-risc-v-64-bit-support-to-buildroot/&text=Adding%20RISC-V%2064-bit%20Support%20to%20Buildroot&via=embecosm)

[risc-v-64-bit-support-to-buildroot/&text=Adding RISC-V 64-bit Support to](https://twitter.com/intent/tweet?url=https://www.embecosm.com/2018/09/19/adding-risc-v-64-bit-support-to-buildroot/&text=Adding RISC-V 64-bit Support to Buildroot&via=embecosm)

[Buildroot&via=embecosm](https://twitter.com/intent/tweet?url=https://www.embecosm.com/2018/09/19/adding-risc-v-64-bit-support-to-buildroot/&text=Adding RISC-V 64-bit Support to Buildroot&via=embecosm))



(<https://plus.google.com/share?url={post.link}>)



([mailto:?subject=Adding RISC-V 64-bit Support to](mailto:?subject=Adding%20RISC-V%2064-bit%20Support%20to%20Buildroot&body=https://www.embecosm.com/2018/09/19/adding-risc-v-64-bit-support-to-buildroot/)

[Buildroot&body=https://www.embecosm.com/2018/09/19/adding-risc-v-64-bit-support-to-buildroot/](mailto:?subject=Adding RISC-V 64-bit Support to Buildroot&body=https://www.embecosm.com/2018/09/19/adding-risc-v-64-bit-support-to-buildroot/))

## Blog topics

[AAP \(/tag/aap/\)](#) [ARM \(/tag/arm/\)](#) [Bletchley Park \(/tag/bletchley-park/\)](#) [Brussels \(/tag/brussels/\)](#)

[Buildroot \(/tag/buildroot/\)](#) [changelog \(/tag/changelog/\)](#) [Comino Foundation \(/tag/comino-foundation/\)](#)

[Community \(/tag/community/\)](#) [Conference \(/tag/conference/\)](#) [cosf \(/tag/cosf/\)](#) [dvcs \(/tag/dvcs/\)](#)

[EdgeTPU \(/tag/edgetpu/\)](#) [Embedded \(/tag/embedded/\)](#) [Embedded Linux \(/tag/embedded-linux/\)](#)

[Embedded OS \(/tag/embedded-os/\)](#) [events \(/tag/events/\)](#)

[floating point arithmetic \(/tag/floating-point-arithmetic/\)](#) [FOSDEM \(/tag/fosdem/\)](#)

[FutureMaker \(/tag/futuremaker/\)](#) [GCC \(/tag/gcc/\)](#) [git \(/tag/git/\)](#) [HackSoton \(/tag/hacksoton/\)](#)

[HPC \(/tag/hpc/\)](#) [Iann Barron \(/tag/iann-barron/\)](#) [LLVM \(/tag/llvm/\)](#) [London \(/tag/london/\)](#)

[MAGEEC \(/tag/mageec/\)](#) [maker \(/tag/maker/\)](#) [Numba \(/tag/numba/\)](#) [open source \(/tag/open-source/\)](#)

[open source hardware \(/tag/open-source-hardware/\)](#) [OSHUG \(/tag/oshug/\)](#) [Parallella \(/tag/parallella/\)](#)

[PyPy \(/tag/pypy/\)](#) [Python \(/tag/python/\)](#) [Recruitment \(/tag/recruitment/\)](#) [RISC-V \(/tag/risc-v/\)](#)

[RSA \(/tag/rsa/\)](#) [Seahorse \(/tag/seahorse/\)](#) [sinf \(/tag/sinf/\)](#) [Somerset House \(/tag/somerset-house/\)](#)

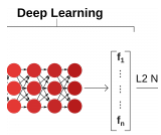
[superoptimization \(/tag/superoptimization/\)](#) [Tensorflow \(/tag/tensorflow/\)](#) [TNMOC \(/tag/tnmoc/\)](#)

[vcs \(/tag/vcs/\)](#)

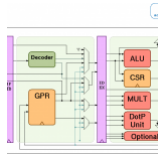
---

## Latest blog posts

[A FaceNet-Style Approach to Facial Recognition on the Google Coral Development Board \(/2019/09/02/a-facenet-style-approach-to-facial-recognition-on-the-google-coral-\)](#)



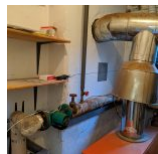
## [development-board/](#)



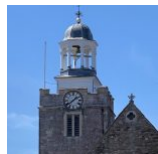
## [A dive into RI5CY core internals \(/2019/08/13/a-dive-into-ri5cy-core-internals/\)](#)



## [Facial Recognition on the New Google Coral Development Board \(/2019/08/09/facial-recognition-on-the-new-google-coral-development-board/\)](#)



## [Monitoring Environmental Noise with a Raspberry Pi \(/2019/07/29/monitoring-environmental-noise-with-a-raspberry-pi/\)](#)



## [Measuring the Accuracy of the Church Clock and Pendulum \(/2019/07/24/measuring-the-accuracy-of-the-church-clock-and-pendulum/\)](#)

---

## Blog Archives

Select Month

---

## Embecosm Twitter Feed

Tweets by [@embecosm](https://twitter.com/embecosm) (<https://twitter.com/embecosm>)

---

[Top of page](#) ↑

## **What we do (<https://www.embecosm.com/services/>)**

**Compiler Tool Chain Development (<https://www.embecosm.com/services/porting/>)**

**Hardware Modeling (<https://www.embecosm.com/services/modeling/>)**

**Open Source Tool Support (<https://www.embecosm.com/services/support/>)**

**Machine Learning Optimization (<https://www.embecosm.com/services/machine-learning-optimization/>)**

**Optimization for Energy Efficiency (<https://www.embecosm.com/services/energy-efficient-compilation/>)**

**Superoptimization (<https://www.embecosm.com/services/superoptimization/>)**

**Embedded Operating Systems (<https://www.embecosm.com/services/embedded-operating-systems/>)**

**Research (<https://www.embecosm.com/research/>)**

## **About us (<https://www.embecosm.com/about/>)**

**Meet The Team (<https://www.embecosm.com/about/meet-the-team/>)**

**Our Customers (<https://www.embecosm.com/about/customers/>)**

**Our Partners (<https://www.embecosm.com/about/our-partners/>)**

**Careers (<https://www.embecosm.com/about/careers/>)**

**Contact (<https://www.embecosm.com/contact/>)**

## **Resources (<https://www.embecosm.com/resources/>)**

**Application Notes (<https://www.embecosm.com/resources/appnotes/>)**

**Software Packages (<https://www.embecosm.com/resources/software/>)**

**Articles (<https://www.embecosm.com/resources/articles/>)**

**Hardware (<https://www.embecosm.com/resources/hardware/>)**

## **Our commitment to accessibility (<https://www.embecosm.com/our-commitment-to-accessibility/>)**





[info@embecosm.com](mailto:info@embecosm.com) (<mailto:info@embecosm.com>).

[+44 1590 610184](tel:+441590610184) ([tel:+44 1590 610184](tel:+441590610184)).



[\\_ \(http://www.github.com/embecosm\)](http://www.github.com/embecosm)



[\\_ \(http://www.twitter.com/embecosm\)](http://www.twitter.com/embecosm)



[\\_ \(https://www.youtube.com/user/embecosm\)](https://www.youtube.com/user/embecosm)

© 2019 Embecosm Ltd. All rights reserved.

Company No. 6577021 (England & Wales). VAT No. GB 131 5223 59

[Terms and conditions \(https://www.embecosm.com/terms-and-conditions/\)](https://www.embecosm.com/terms-and-conditions/)

[Privacy policy \(https://www.embecosm.com/privacy-policy/\)](https://www.embecosm.com/privacy-policy/)

This site uses cookies: [Find out more.](#)

[\(https://www.embecosm.com/cookie-policy/\)](https://www.embecosm.com/cookie-policy/)

Accept Cookies