

**Getting Started with the RISC-V Based PolarFire® SoC  
FPGA Webinar Series**  
Session 15 Building and running Linux



---

A Leading Provider of Smart, Connected and Secure Embedded Control Solutions



SMART | CONNECTED | SECURE

***Hugh Breslin, Design Engineer***

*Thursday July. 9, 2020*

# Supporting Content

[www.microsemi.com/Mi-V](http://www.microsemi.com/Mi-V) “Renode Webinar Series”

product-directory/fpga-soc/5210-mi-v-embedded-ecosystem#renode-webinar-series

Search Microsemi.com

Ordering Company Partners Support

Libero SoC Design Suite v12.0

- Unified design suite for PolarFire, IGL002, SmartFusion2 and RTG4 families
- 60% runtime reduction for Timing and 20% runtime reduction for Power
- 25% runtime improvement for Place and Route

Home / Products & Services / FPGA & SoC / Mi-V RISC-V Ecosystem

## Mi-V RISC-V Ecosystem

Overview Mi-V Partners Tutorials **Renode Webinar Series** Articles and News

### Getting Started with the RISC-V Based PolarFire™ SoC FPGA Webinar Series

Learn how to get started with the PolarFire SoC FPGA, the world's first RISC-V based SoC FPGA, to create fully deterministic, real-time systems alongside the Linux® operating system. We are holding a series of webinars to introduce you to the free Renode™ development platform from Mi-V partner Antmicro that is available with our SoftConsole v6.0 software development environment. You will see demo applications, learn how to create projects, and find out how to set up and configure your own systems targeting the new SoC FPGA architecture.

Click [here](#) to register.

#### Webinar 1 (May 2): Discover Renode for PolarFire™ SoC Design and Debug

In this introductory session, we will provide you with an overview of SoftConsole 6.0 with Renode™ integration. We will introduce you to the Renode development framework and provide an overview of the platform and its features. You will also learn about the PolarFire™ SoC architecture and how to use Renode to develop your application.

Webinar 1: Discover Renode for PolarFire® SoC Design and Debug

Webinar 2: How to Get Started with Renode for PolarFire SoC

Webinar 3: Learn to Debug a Bare-Metal PolarFire SoC Application with Renode

Webinar 4: Tips and Tricks for Even Easier PolarFire SoC Debug with Renode

Webinar 5: Add and Debug PolarFire SoC Models with Renode

Webinar 6: Add and Debug Pre-Existing Model in PolarFire SoC

Webinar 7: How to Write Custom Models

Webinar 8: What's New in SoftConsole v6.2

Webinar 9: Getting Started with PolarFire SoC

Webinar 10: Introduction to the PolarFire SoC Bare-Metal Library

Webinar 11: Handling Binaries

Webinar 12: Simple Peripheral as Software Stimulus

Webinar 13: Two Baremetal Applications on PolarFire SoC

# Agenda

- **Where to find the Linux sources**
- **Building Linux using Yocto**
- **Building Linux using BuildRoot**
- **Building the HSS**
- **Booting Linux**

# Where to find the Linux sources

# Where to find the Linux sources

- <https://github.com/polarfire-soc>
- **Yocto build:** <https://github.com/polarfire-soc/meta-polarfire-soc-yocto-bsp>
- **BuildRoot build:** <https://github.com/polarfire-soc/polarfire-soc-buildroot-sdk>
- **HSS:** <https://github.com/polarfire-soc/hart-software-services>

# Where to find the Linux sources

- **Yocto or BuildRoot is up to the user**
  - Yocto Icicle Kit support is up streamed
  - BuildRoot Icicle Kit support coming in a few days
- **More info available**
  - <https://www.yoctoproject.org/>
  - <https://buildroot.org/>

# Building Linux with Yocto

---

# Building Linux with Yocto

- **There are several first run steps to set up your system:**
  - Install required Yocto packages for the host
  - Install the repo command from Google
  - Install kconfiglib
- **To run a build:**
  - Clone the PF SoC Yocto repository
  - Run the configuration script
  - Run the bitbake command specifying your machine target and image
- **Steps available in the readme for the repository**

 **PolarFire-SoC**  
PolarFire SoC Embedded Software  
<http://www.microsemi.com/polarfiresoc>

Repositories 5 Packages People Projects

**Grow your team on GitHub**  
GitHub is home to over 50 million developers working together. Join them to grow your own development teams, manage permissions, and collaborate on projects.  
[Sign up](#) Dismiss

Pinned repositories

- polarfire-soc-bare-metal-library**  
Bare metal embedded software drivers and examples for PolarFire SoC  
C 4 stars 2 forks
- meta-polarfire-soc-yocto-bsp**  
PolarFire SoC yocto Board Support Package  
Python 1 star 2 forks
- polarfire-soc-buildroot-sdk**  
PolarFire SoC Buildroot Software Development Kit  
Makefile 1 star 2 forks

Find a repository... Type: All Language: All

**polarfire-soc-documentation**  
PolarFire SoC Documentation  
0 forks 0 stars 0 issues 0 pull requests Updated 2 days ago

**hart-software-services**  
PolarFire SoC hart software services

Top languages  
C Python Makefile

People  
This organization has no public members.

# Building Linux with Yocto

- **Yocto will build the HSS and Linux image and they will all be contained in the same output folder**
- **Instructions on programming eNVM or SD cards are found in the readme**

Recent

Starred

Home

Desktop

Documents

Downloads

Music

Pictures

Videos

Trash

+ Other Locations



build

meta-  
openembe  
ddedmeta-  
polarfire-  
soc-yocto...

meta-riscv

openembe  
dded-core

## Copy the created Disk Image to flash device (USB mmc flash/SD/uSD)

Be very careful while picking /dev/sdX device! Look at dmesg, lsblk, GNOME Disks, etc. before and after plugging in your usb flash device/uSD/SD to find a proper device. Double check it to avoid overwriting any of system disks/partitions!

```
cd yocto-dev/build
zcat tmp-glibc/deploy/images/icicle-kit-es/mpfs-dev-cli-icicle-kit-es.wic.gz | sudo dd of=/dev/sdX bs
```

## Yocto Setup and BSP

Yocto Release Activity: Dunfell (Revision 3.1) (Released April 2020)

## Required packages for the Build Host

This document assumes you are running on a modern Linux system. The process documented here was tested using

# Building Linux with Yocto

- **The HSS is used to load the Linux image and needs to be configured for the PF SoC MSS configuration used**
- **The Libero generated XML can be used for this and simply replacing 1 file will update your configuration**
- **This only needs to be done if the MSS has been reconfigured**

```
MACHINE=<MACHINE> bitbake <recipe> -c <command>
```

BSP recipes available:

- hss (Microchip HSS)
- u540-c000-bootloader (Sifive FSBL)
- u-boot
- mpfs-linux (kernel our BSP)

Available commands: clean / configure / compile / install

## HSS Hardware Configuration from Libero Design

(Support for the Icycle-kit only)

The HSS recipe generates embedded software header files from information supplied by Libero from the Libero design. Libero supplies the information in the form of an xml file. This can be found in the Libero component subdirectory e.g: /component/work/PFSOC\_MSS\_C0/PFSOC\_MSS\_C0\_0

Update the following folder with the updated XML file (use the same name) :

```
meta-polarfire-soc-yocto-bsp/recipes-bsp/hss/files/${MACHINE}/ICICLE_MSS_0.xml
```

## Yocto Image and Binaries directory

```
build/tmp-glibc/deploy/images/{MACHINE}
```

For Example the following is the path for the Icycle-kit-es

```
build/tmp-glibc/deploy/images/icicle-kit-es
```

## Running wic.gz image on hardware

Compressed Disk images files use <image>-<machine>.wic.gz format, for example,

```
mpfs-dev-cli-<machine>.wic.gz . We are interested in .wic.gz disk images for writing to emmc/uSD/SD card.
```

# Building Linux with BuildRoot

---

# Building Linux with BuildRoot

- **There are several packages that need to be installed on the host system and these are listed in the readme**
- **To run a build:**
  - Clone the PF SoC BuildRoot repository
  - Run the make file
- **Steps available in the readme for the repository**
- **Note that on the first build BuildRoot will also build the RISC-V Toolchain**



PolarFire-SoC  
PolarFire SoC Embedded Software  
<http://www.microsemi.com/polarfiresoc>

Repositories 5 Packages People Projects

**Grow your team on GitHub**

GitHub is home to over 50 million developers working together. Join them to grow your own development teams, manage permissions, and collaborate on projects.

[Sign up](#)

[Dismiss](#)

Pinned repositories

**polarfire-soc-bare-metal-library**

Bare metal embedded software drivers and examples for PolarFire SoC

C 4 2

**meta-polarfire-soc-yocto-bsp**

PolarFire SoC yocto Board Support Package

Python 1 2

**polarfire-soc-buildroot-sdk**

PolarFire SoC Buildroot Software Development Kit

Makefile 1 2

Find a repository...

Type: All

Language: All

**polarfire-soc-documentation**

PolarFire SoC Documentation  
0 0 0 0 Updated 3 days ago

**hart-software-services**

PolarFire SoC hart software services

Top languages

C Python Makefile

People

This organization has no public members.

# Building the HSS

---

# Building the HSS

- **Yocto will build a copy of the HSS based on example XML from Libero**
- **Buildroot will not build the HSS**
- **If you have a different set up to the base used in Yocto you can manually build a standalone HSS, if you use BuildRoot you need to build the HSS**



PolarFire-SoC  
PolarFire SoC Embedded Software  
http://www.microsemi.com/polarfiresoc

Repositories 5 Packages People Projects

**Grow your team on GitHub**

GitHub is home to over 50 million developers working together. Join them to grow your own development teams, manage permissions, and collaborate on projects.

[Sign up](#)

[Dismiss](#)

Pinned repositories

- polarfire-soc-bare-metal-library**  
Bare metal embedded software drivers and examples for PolarFire SoC  
C 4 stars 2 forks
- meta-polarfire-soc-yocto-bsp**  
PolarFire SoC yocto Board Support Package  
Python 1 star 2 forks
- polarfire-soc-buildroot-sdk**  
PolarFire SoC Buildroot Software Development Kit  
Makefile 1 star 2 forks

Find a repository... Type: All Language: All

**polarfire-soc-documentation**

PolarFire SoC Documentation  
0 forks 0 stars 0 issues 0 pull requests Updated 3 days ago

**hart-software-services**

PolarFire SoC hart software services

**Top languages**

- C
- Python
- Makefile

**People**

This organization has no public members.

# Booting Linux

---

# Booting Linux

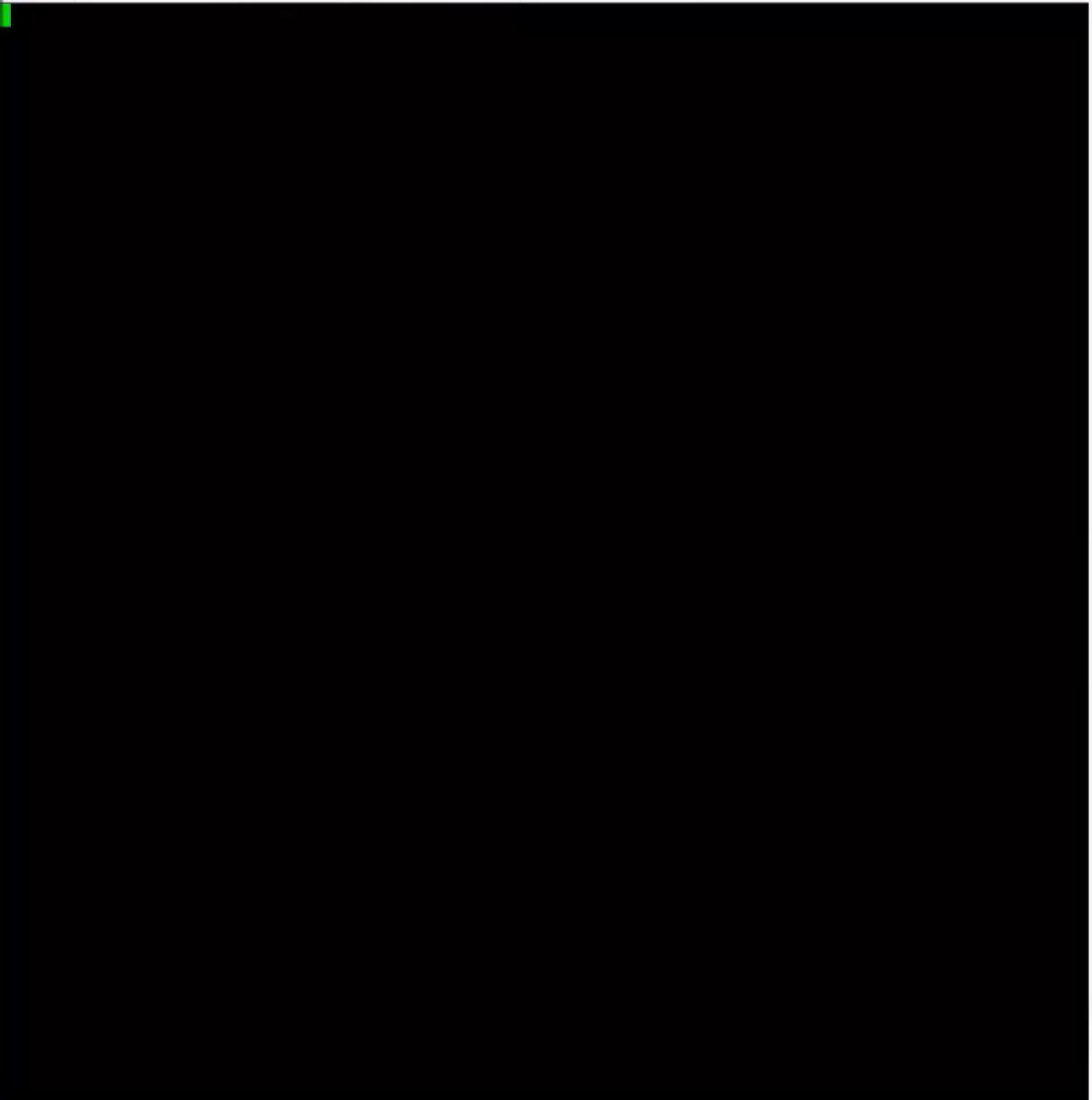
- **Depending on your configuration there are several ways to supply the Linux image to PF SoC**
  - eMMC storage
  - SD card
  - TFTP
- **eMMC can be programmed using the HSS**
- **SD cards can be programmed directly from a host**
- **Ethernet can be used to download an image via TFTP**
- **The HSS will be stored in eNVM**

# Booting Linux

1. HSS launches
2. Option to halt boot and run several commands (e.g program eMMC)
3. When booting the Linux image will be copied to DDR
4. uboot starts and Linux boots

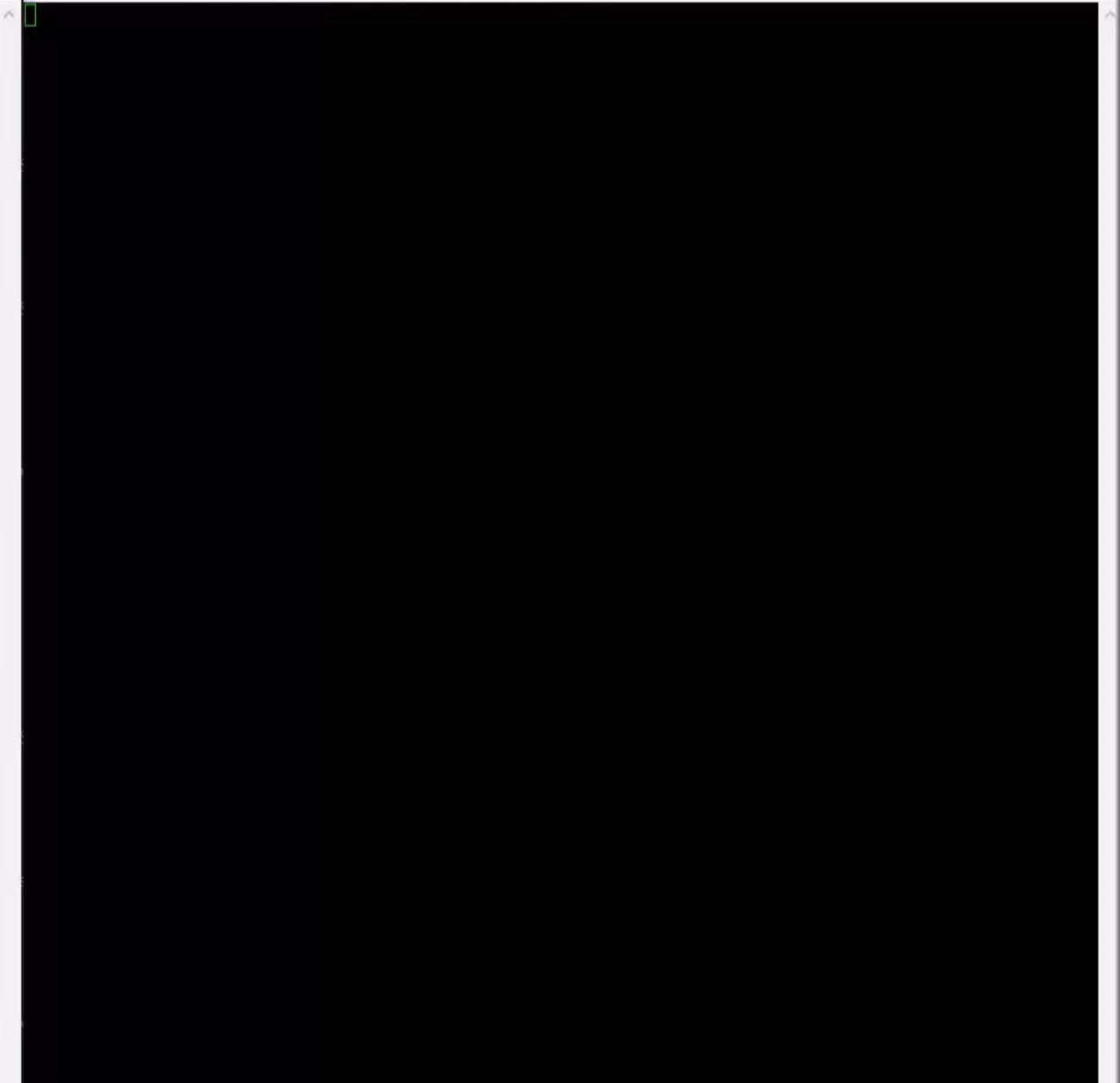
COM22 - PuTTY

Session Special Command Window Logging Files Transfer Hangup ?



COM21 - PuTTY

Session Special Command Window Logging Files Transfer Hangup ?



# Agenda

- **Where to find the Linux Source**
- **Building Linux using Yocto**
- **Building Linux using BuildRoot**
- **Building the HSS**
- **Booting Linux**

# Thank you!

---

Any questions?

# Second Thursdays

**Aug. 13 - Webinar 16: Building Applications for Linux on PolarFire SoC**

**Sep. 10 - Webinar 17: Real-Time (AMP Mode) on PolarFire SoC**