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

Session 17 AMP Mode

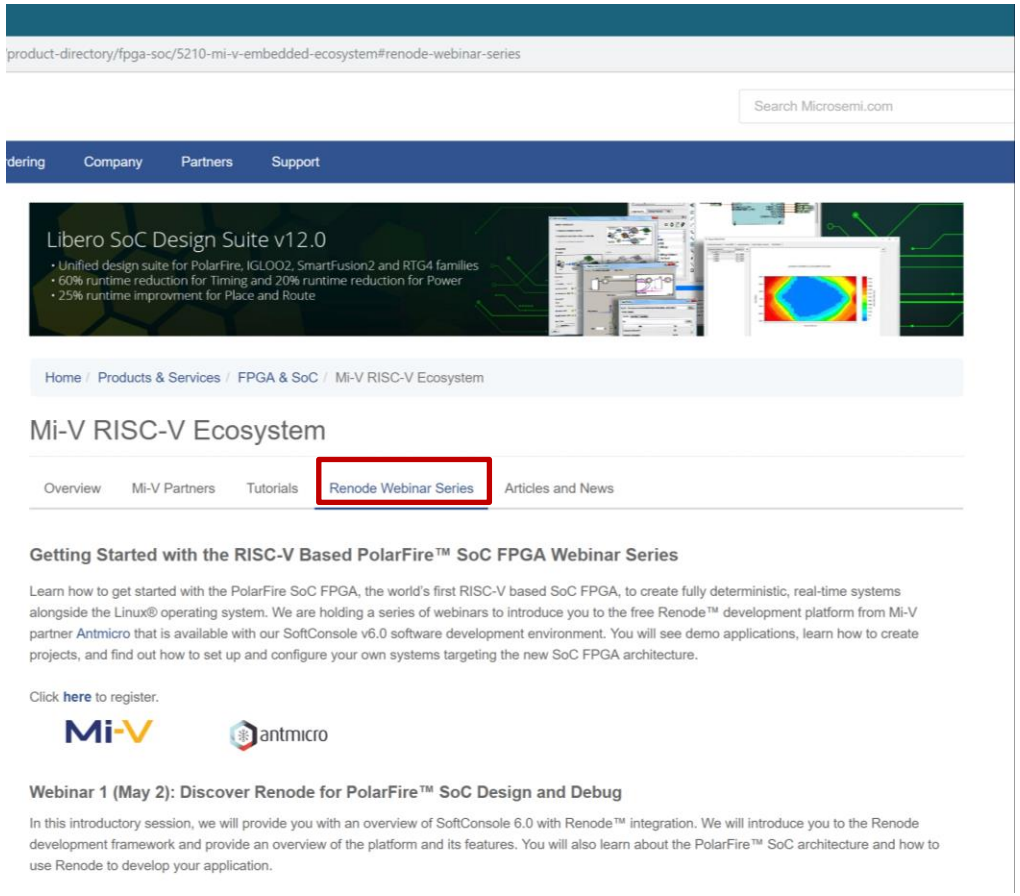
A Leading Provider of Smart, Connected and Secure Embedded Control Solutions

Hugh Breslin, Design Engineer

Thursday Sept. 10, 2020

Supporting Content

www.microsemi.com/Mi-V “Renode Webinar Series”



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

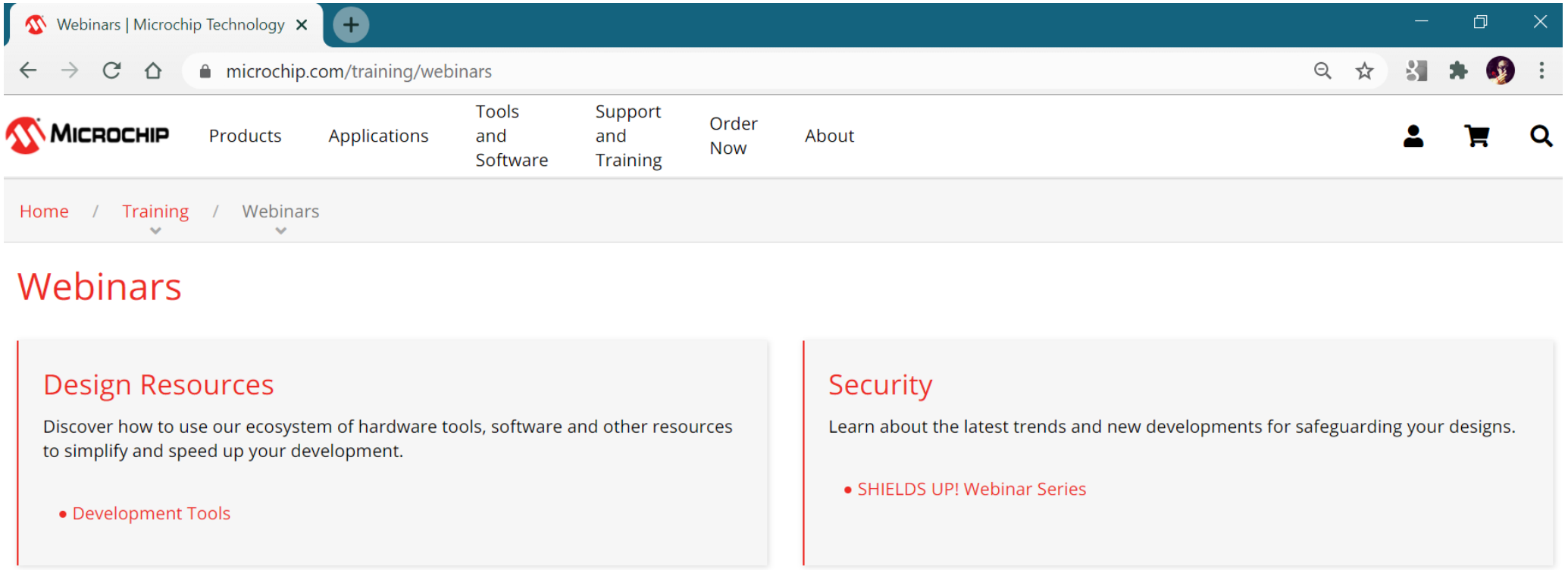
Webinar 14: The PolarFire SoC Icicle Kit Model on Renode

Webinar 15: Building and running Linux

Webinar 16: Linux on PolarFire SoC

Coming Soon! RISC-V® Innovation Unleashed

- **New PolarFire SoC webinar series**
- **Registrations go live from September 16th**
- **https://www.microchip.com/training/webinars**



MICROCHIP

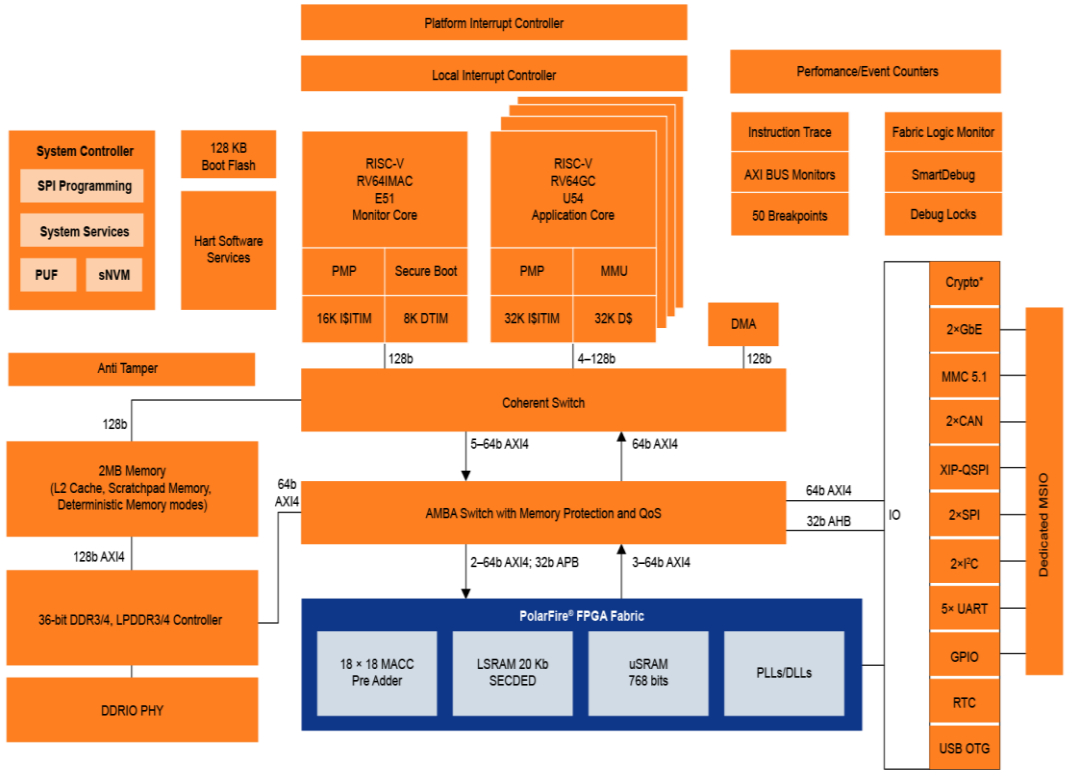
Agenda

- Multiprocessing
- Symmetric Multiprocessing
- Asymmetric Multiprocessing
- Configuring Asymmetric Multiprocessing

Multiprocessing

Multiprocessing

- Multiple CPUs in a single system
- Shared memory and peripherals
- Execute multiple tasks simultaneously
- This is not the same as parallel processing



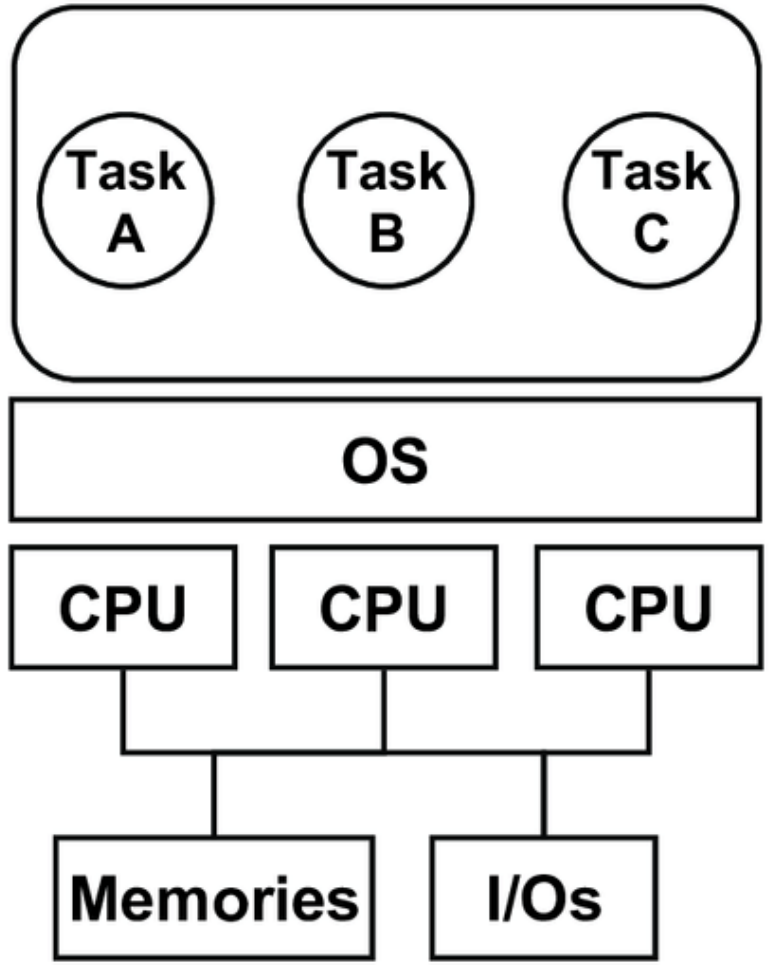
MICROCHIP

Multiprocessing

CPU Affinity

- This binds a process to a core / cores
- Run a process on a subset of cores
- Run a process on a specific core
- Achieved through a system call

✓Improve cache performance



Multiprocessing

CPU Affinity

- **Check affinity with**

taskset -cp [PID]

- **Returns**

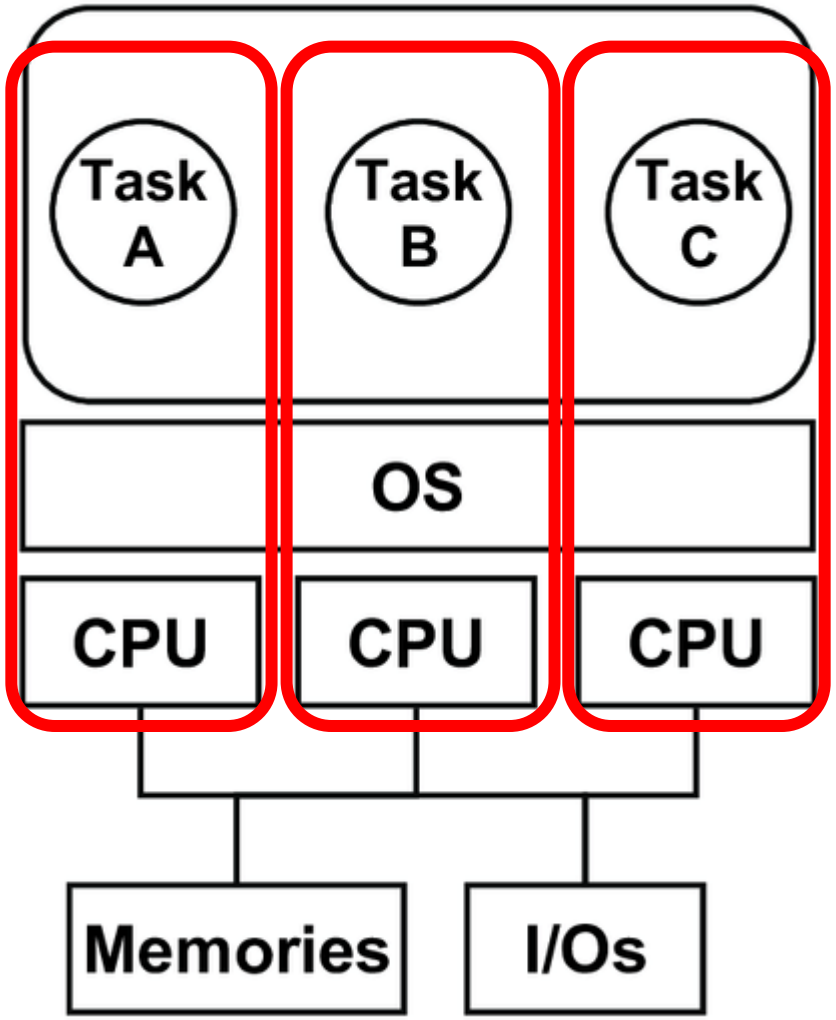
pid 9726's current affinity list: 0-3

- **Set affinity with**

taskset -cp 0,3 9726

- **Returns**

pid 9726's current affinity list: 0,1,2,3
pid 9726's new affinity list: 0,3



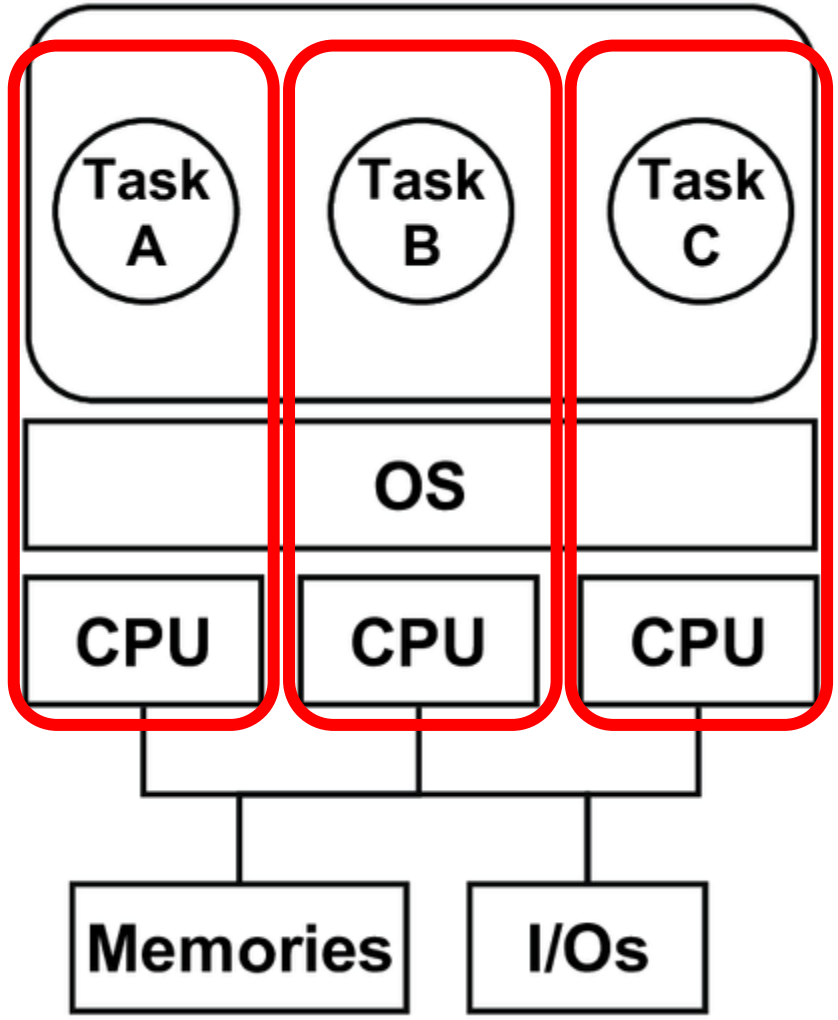
Multiprocessing

CPU Affinity

- **Launch a process with a certain affinity**

taskset 0xA webserver

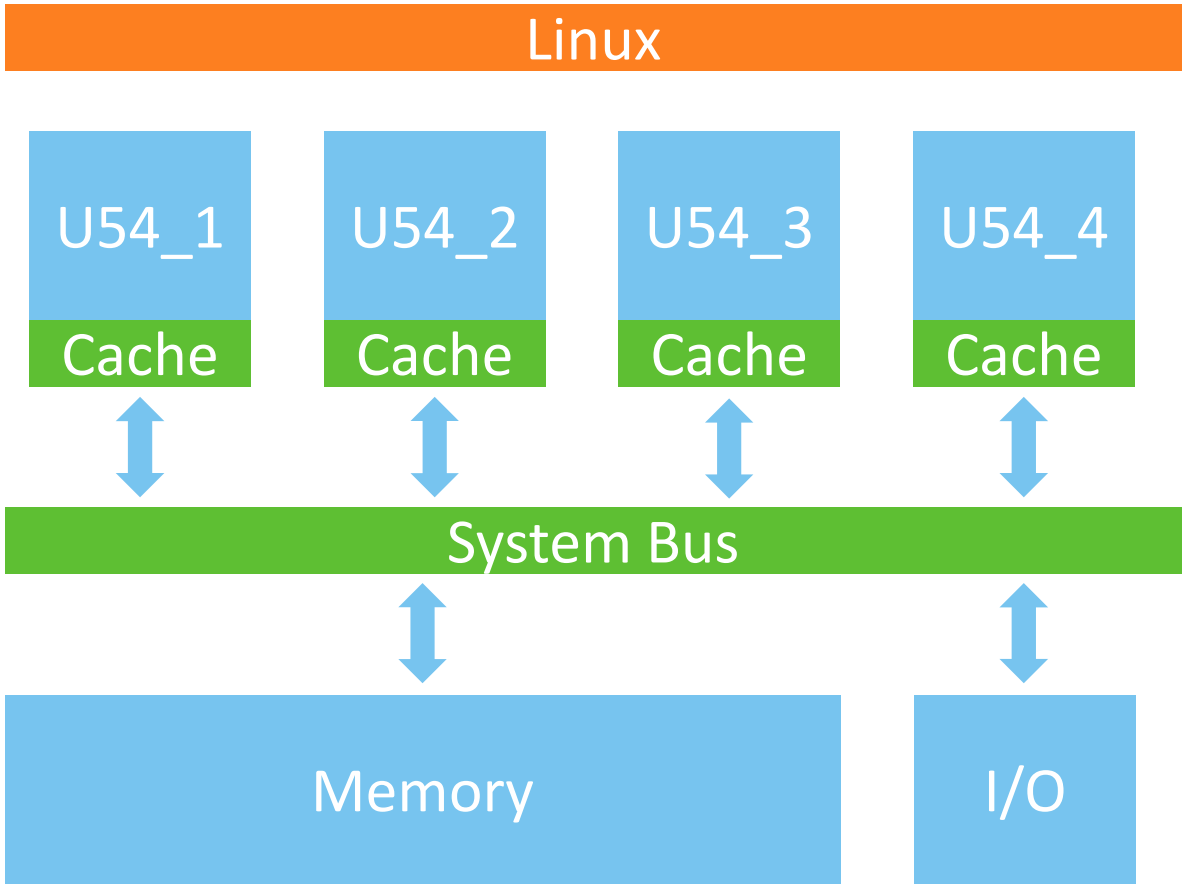
- ***0xA == 1010 – cores 1 and 3***



Symmetric Multiprocessing

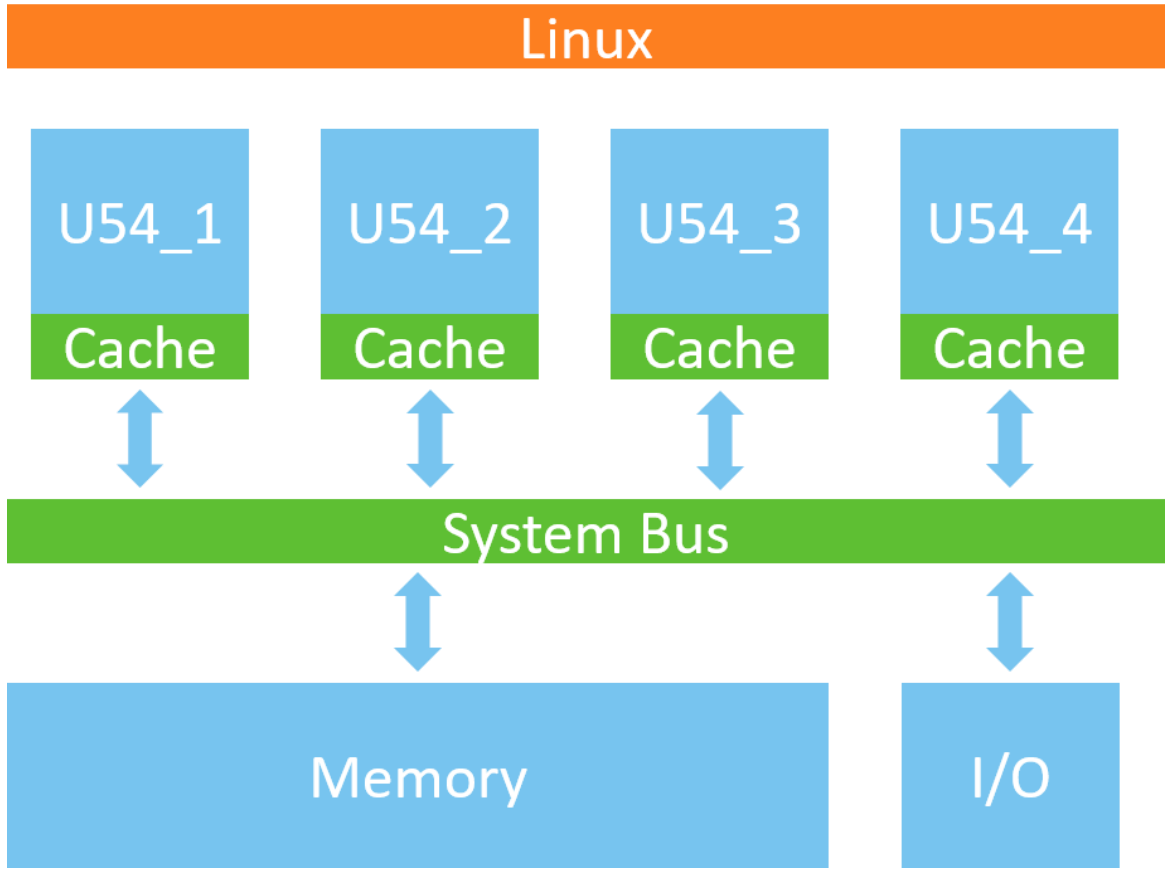
Symmetric Multiprocessing

- **A single OS with two or more of the same processor**



Symmetric Multiprocessing

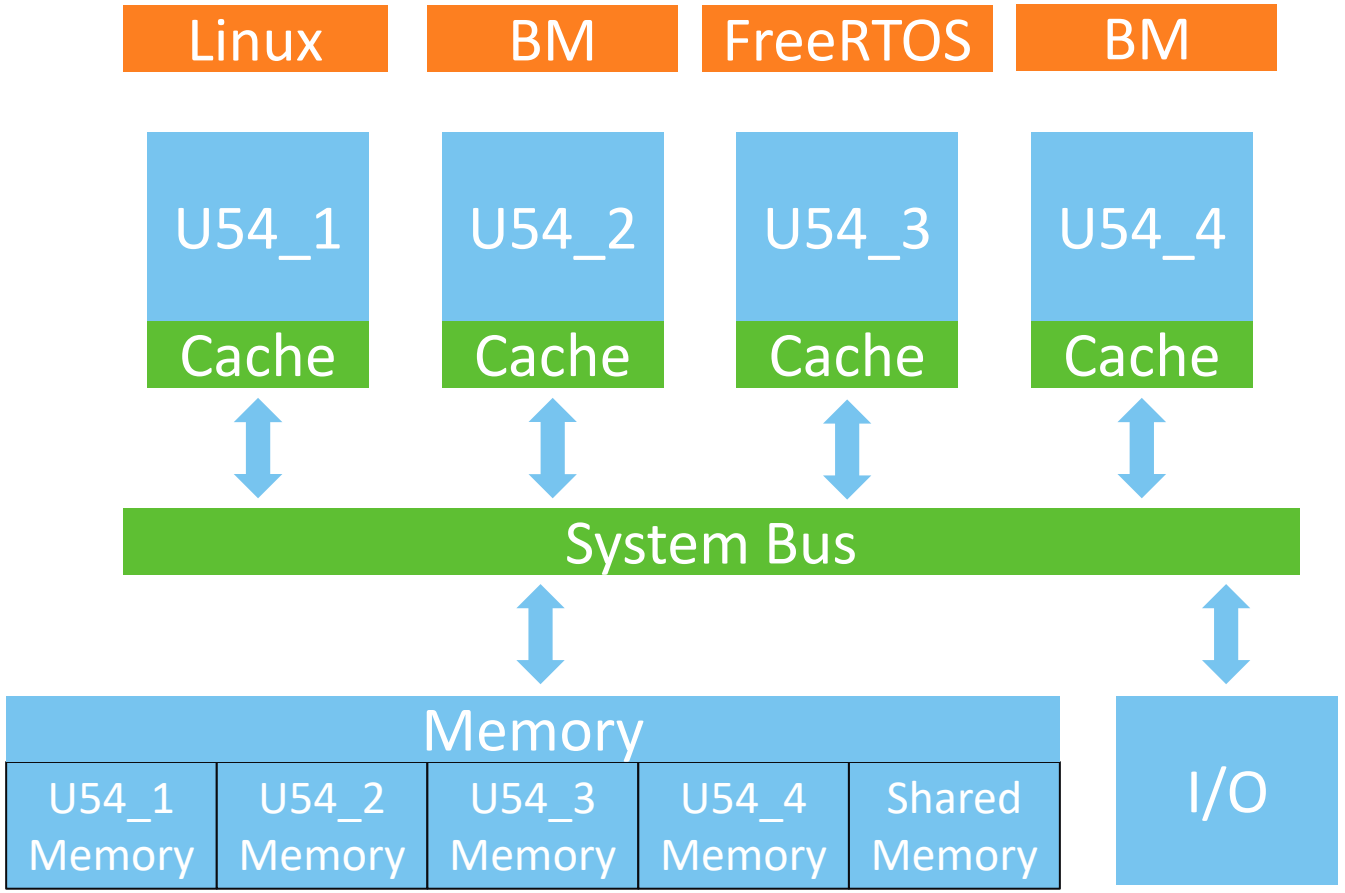
- All processors are treated equally
- Shared memory space
- Tasks can be run on any CPU
- The same task should not be run on 2 CPUs
- If a processor fails, the system stays up



Asymmetric Multiprocessing

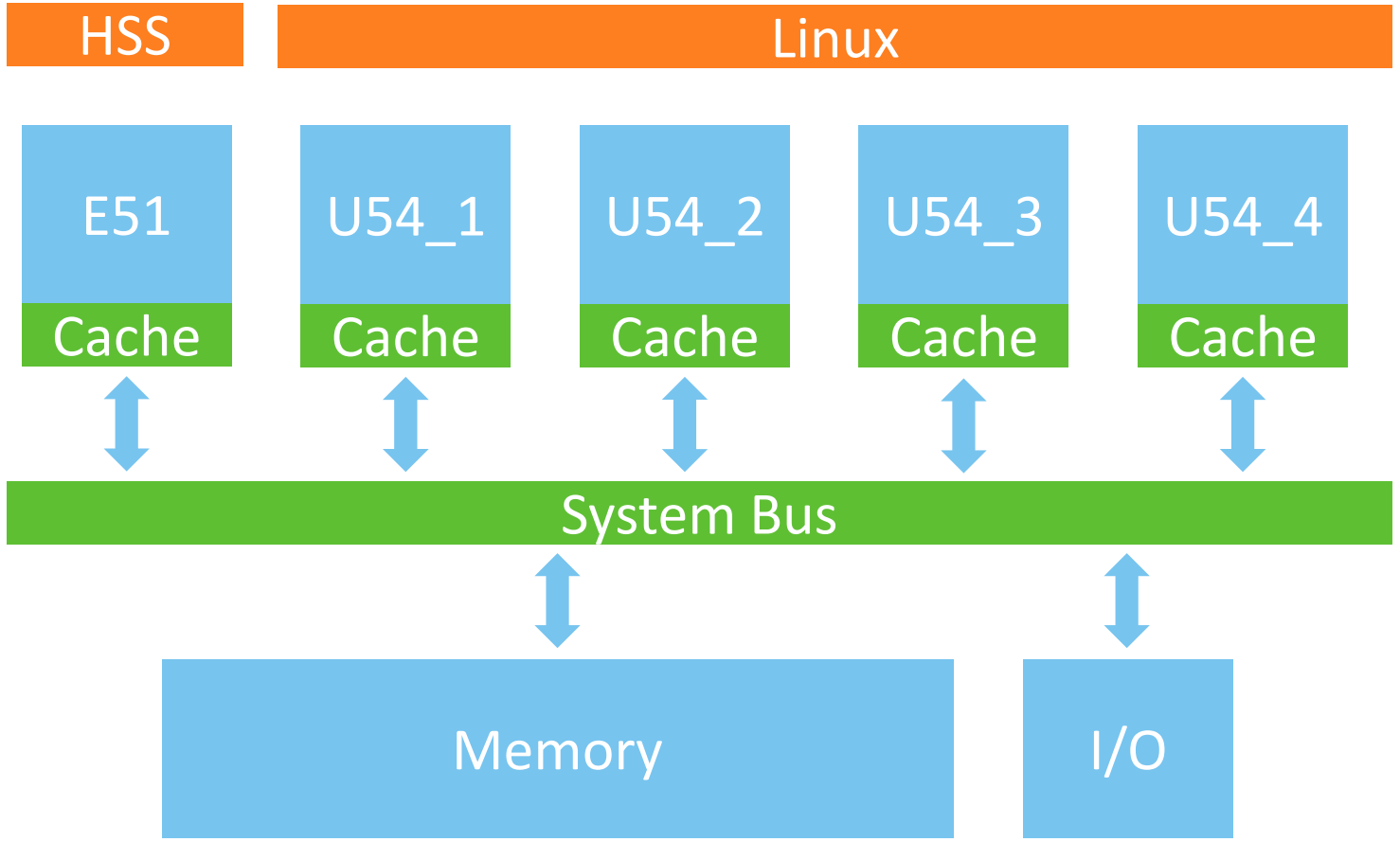
Asymmetric Multiprocessing

- **Multiple OS's / bare metal running on separate processors**



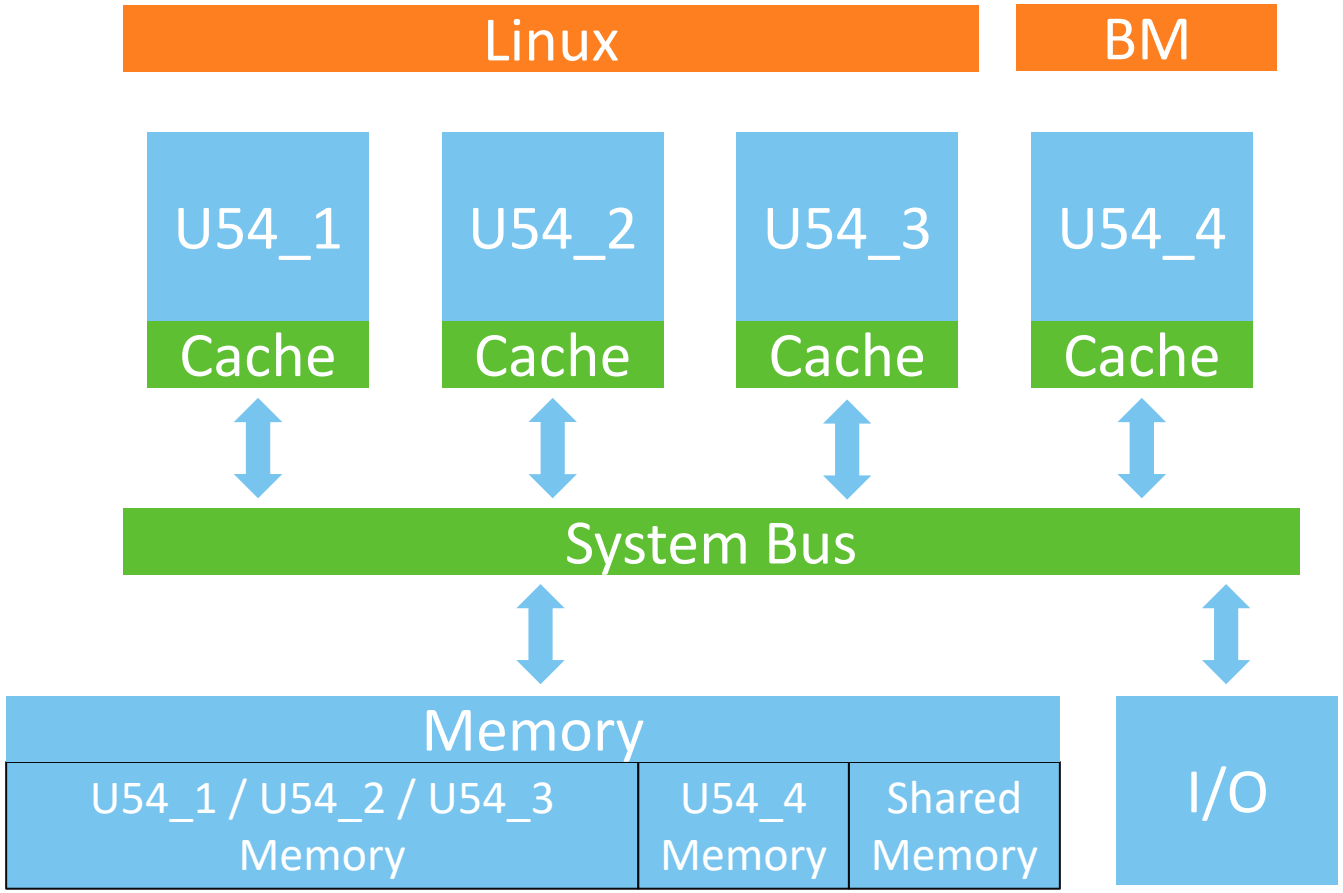
Asymmetric Multiprocessing

- **Multiple OS's / bare metal running on separate processors**



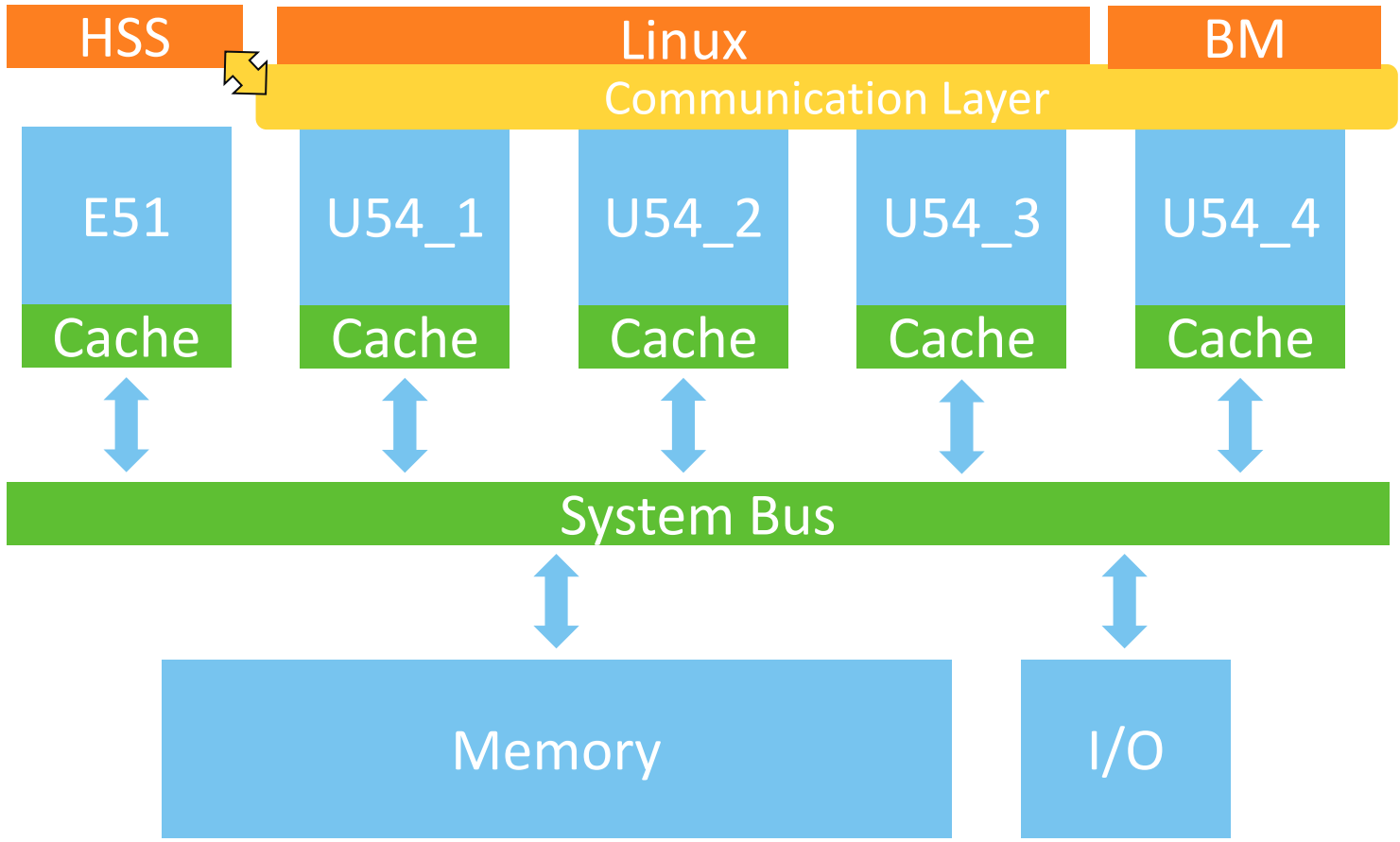
Asymmetric Multiprocessing

- **Blend SMP system into AMP**



Asymmetric Multiprocessing

- **Inter processor communication**



Configuring Asymmetric Multiprocessing

Asymmetric Multiprocessing

The HSS

- **A superloop monitor running on the E51 processor, which receives requests from the individual U54 application processors to perform certain services on their behalf.**

- **A Machine-Mode software interrupt trap handler, which allows the E51 to send messages to the U54s and request them to perform certain functions for it related to rebooting a U54.**
  - SSMB – “Secure Software Messaging Bus”

Asymmetric Multiprocessing

The HSS: Payloads

- **Payloads are created for the system configuration containing the binaries to be run**

- **See Webinar 13 “Two Bare-Metal Applications on PolarFire SoC”**

Asymmetric Multiprocessing

The HSS: Payloads

- **Payload 0: Linux for harts 1, 2 & 3**
- **Payload 1: Bare metal for hart 4**

Asymmetric Multiprocessing

The HSS: Linux payload

- **Linux for harts 1, 2 & 3**
  - Edit the device tree to remove hart 4 and build

master polarfire-soc-buildroot-sdk / conf / icicle-kit-es / **icicle-kit-es.dts**

ConchuOD dts: update dts from yocto

1 contributor

325 lines (321 sloc) | 10.9 KB | Raw | Blame

```
1   /dts-v1/;
2   / {
3           #address-cells = <2>;
4           #size-cells = <2>;
5           compatible = "SiFive,FU540G-dev", "fu540-dev", "sifive-dev";
6           model = "SiFive,FU540G";
7           L45: cpus {
8                   #address-cells = <1>;
9                   #size-cells = <0>;
10              timebase-frequency = <1000000>;
11                  L8: cpu@0 {
12                          clock-frequency = <0>;
```

master meta-polarfire-soc-yocto-bsp / recipes-kernel / linux / files / icicle-kit-es / **icicle-kit-es-a000-microchip.dts**

lhanlyu Clock Driver remove spaces in names. ... Latest commit 1df7cc

1 contributor

351 lines (346 sloc) | 11.7 KB | Raw | Blame | Copy File | 11.73 K

```
1   /dts-v1/;
2   / {
3           #address-cells = <2>;
4           #size-cells = <2>;
5           compatible = "SiFive,FU540G-dev", "fu540-dev", "sifive-dev";
6           model = "SiFive,FU540G";
7           L45: cpus {
8                   #address-cells = <1>;
9                   #size-cells = <0>;
10              timebase-frequency = <1000000>;
11                  L8: cpu@0 {
12                          clock-frequency = <0>;
```

Asymmetric Multiprocessing

The HSS: Bare metal payload

- Set the system “MPFS_HAL_FIRST_HART” to 4
- Target the desired memory in the linker script (e.g LIM / DDR)
- Build
- Watch webinar 11 “Handling Binaries”

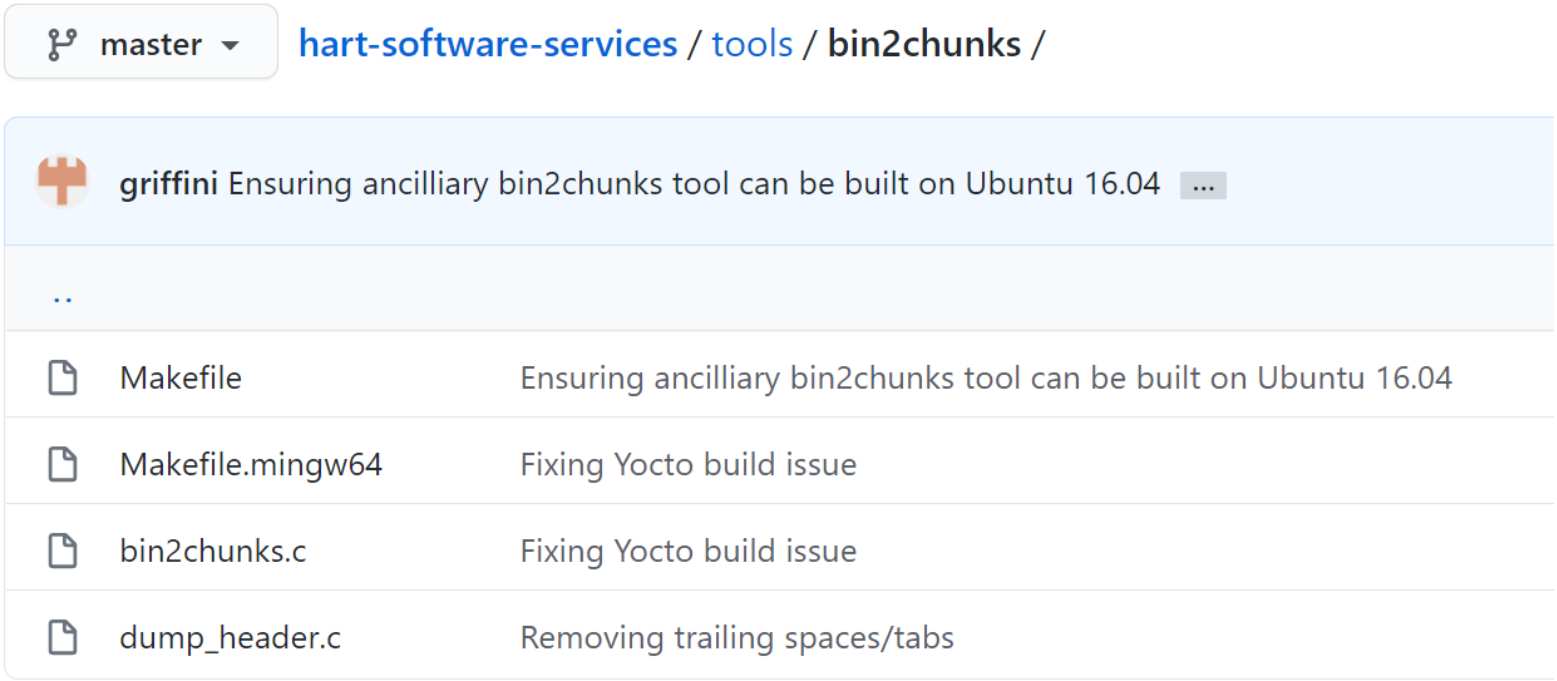
mss_sw_config.h

```
37   * to start and execute code on the E5
38   * Set MPFS_HAL_LAST_HART to a value s
39   * all U54 harts.
40   * Harts that are not started will rem
41   * through some other method
42   */
43  #ifndef MPFS_HAL_FIRST_HART
44  #define MPFS_HAL_FIRST_HART  4
45  #endif
46
47  #ifndef MPFS_HAL_LAST_HART
48  #define MPFS_HAL_LAST_HART   4
49  #endif
50
```

Asymmetric Multiprocessing

The HSS: Payloads

- **Use the payload generator in the HSS to generate a payload**
  - Currently bin2chunks – this will be updated



Asymmetric Multiprocessing

The HSS: Payloads

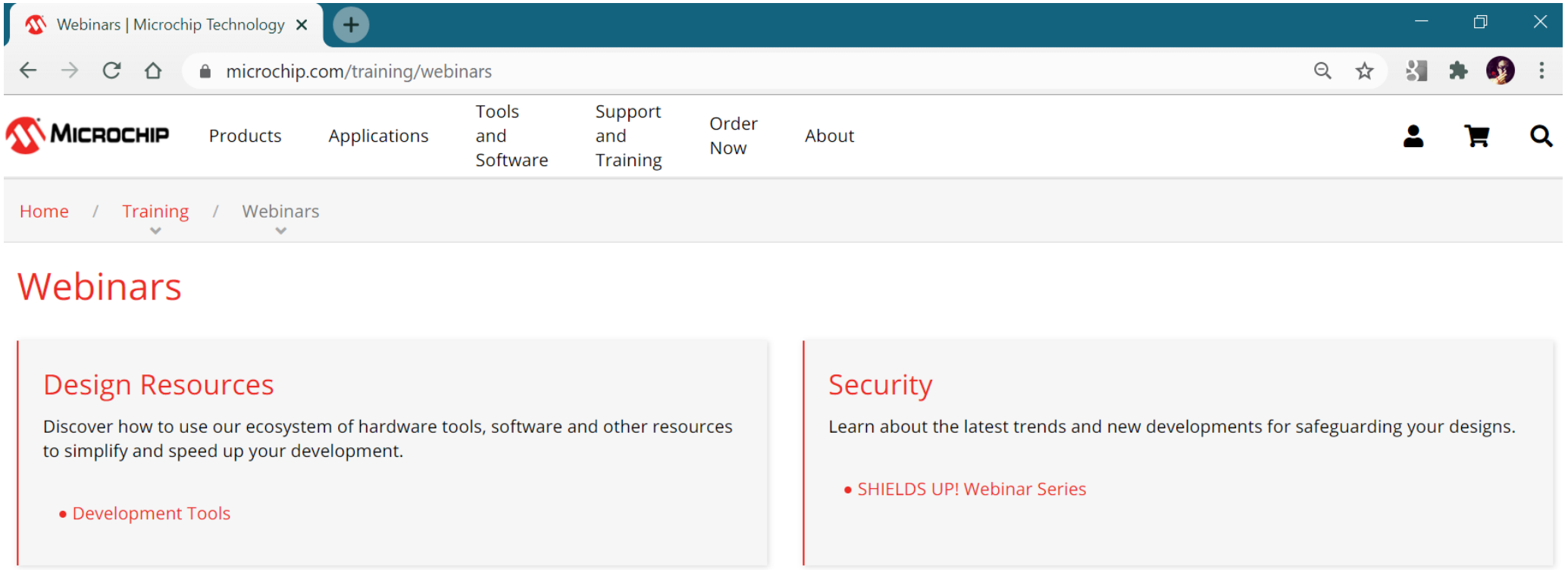
- Program the eNVM with the HSS (if not done already)
- Program the target (e.g eMMC / SD)
- Boot
- Voila!

Agenda

- Multiprocessing
- Symmetric Multiprocessing
- Asymmetric Multiprocessing
- Configuring Asymmetric Multiprocessing

Coming Soon! RISC-V® Innovation Unleashed

- **New PolarFire SoC webinar series**
- **Registrations go live from September 16th**
- **https://www.microchip.com/training/webinars**



MICROCHIP

Thank you!

Any questions?