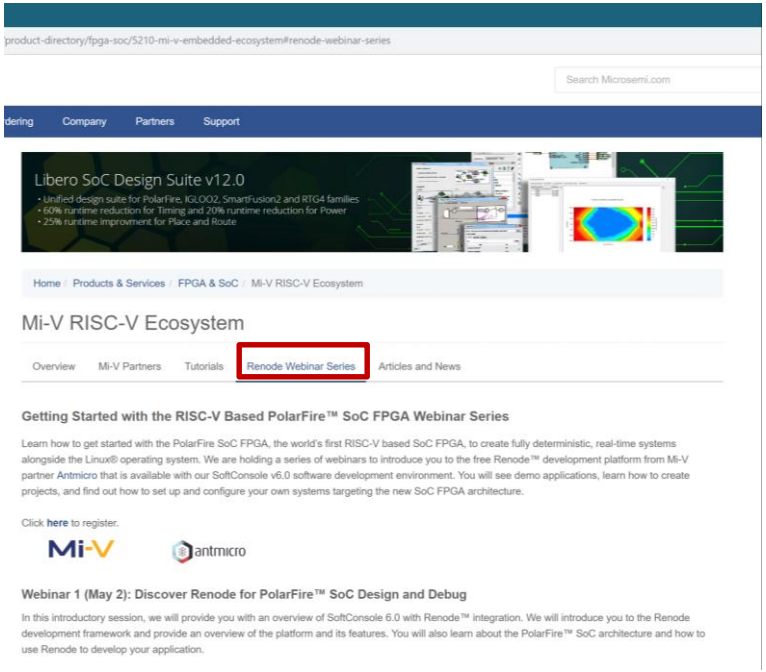
First / Second Thursdays

Dec. 5	-	Webinar 8: What's New in SoftConsole v6.2
Jan. 9	-	Webinar 9: Getting Started with PolarFire® SoC
Feb. 13	-	Webinar 10: Introduction to the PolarFire SoC Baremetal Library
Mar. 12	-	Webinar 11: Handling Binaries
April 9	-	Webinar 12: Two Baremetal Applications on PolarFire SoC
May 14	-	Webinar 13: Linux on Renode
June 11	-	Webinar 14: Building Applications for Linux on PolarFire SoC
July 9	-	Webinar 15: Real-Time (AMP Mode) on PolarFire SoC

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

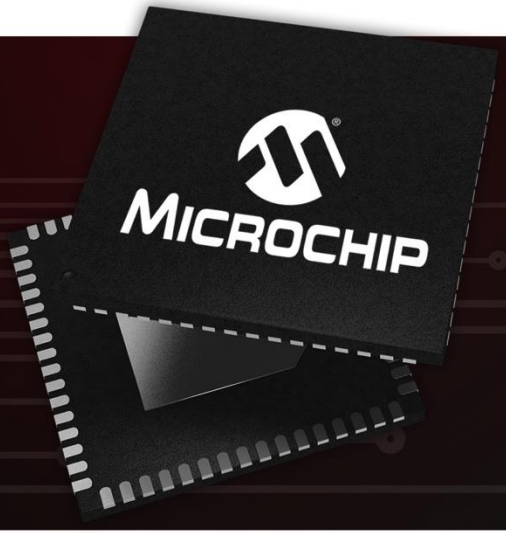
Webinar 4: Tips and Tricks for Even Easier PolarFire SoC Debug with Renode

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

MICROCHIP



A Leading Provider of Microcontroller, Security, Mixed-Signal, Analog & Flash-IP Solutions




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

Session 8: What's New in SoftConsole 6.2

Hugh Breslin, Embedded Linux Engineer

Thursday Dec. 5, 2019

Agenda

- **Updated Renode**
  - antmicro Training
  - Renode Scripts Moved
- **GDB Multi-Core Debugging**
- **Updated to Example Projects**
  - Changes to PSE Blinky
  - New Sample Project: mpfs-mustein-julia
  - New Sample Project: mpfs-freertos-lwip

Updated Renode

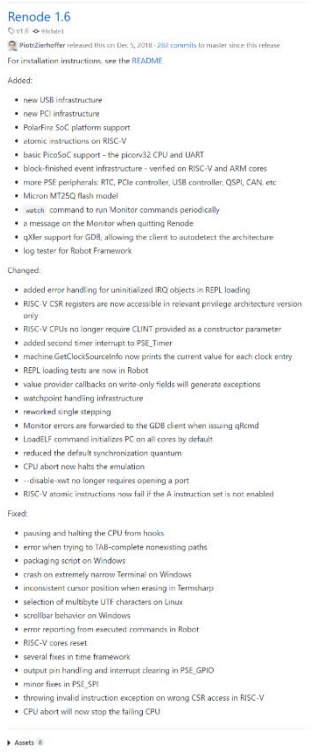
Updated Renode

Release	Bundled Renode Version
SoftConsole v6.0	1.6.0
SoftConsole v6.1	1.7.1
SoftConsole v6.2	1.8.2

https://github.com/renode/renode/releases

Updated Renode

https://github.com/renode/renode/releases



Renode 1.6

v1.6 · f9cbde1

PiotrZierhoffer released this on Dec 5, 2018 · 282 commits to master since this release

For installation instructions, see the README.

Added:

- new USB infrastructure
- new PCI infrastructure
- PolarFire SoC platform support
- atomic instructions on RISC-V
- basic PicoSoC support - the picorv32 CPU and UART
- block-finished event infrastructure - verified on RISC-V and ARM cores
- more PSE peripherals: RTC, PCIe controller, USB controller, QSPI, CAN, etc
- Micron MT25Q flash model
- `watch` command to run Monitor commands periodically
- a message on the Monitor when quitting Renode
- qXfer support for GDB, allowing the client to autodetect the architecture
- log tester for Robot Framework

Changed:

- added error handling for uninitialized IRQ objects in REPL loading
- RISC-V CSR registers are now accessible in relevant privilege architecture version only
- RISC-V CPUs no longer require CLINT provided as a constructor parameter
- added second timer interrupt to PSE_Timer
- machine.GetClockSourceInfo now prints the current value for each clock entry
- REPL loading tests are now in Robot
- value provider callbacks on write-only fields will generate exceptions
- watchpoint handling infrastructure
- reworked single stepping
- Monitor errors are forwarded to the GDB client when issuing qRcmd
- LoadELF command initializes PC on all cores by default
- reduced the default synchronization quantum
- CPU abort now halts the emulation
- --disable-xwt no longer requires opening a port
- RISC-V atomic instructions now fail if the A instruction set is not enabled

Fixed:

- pausing and halting the CPU from hooks
- error when trying to TAB-complete nonexisting paths
- packaging script on Windows
- crash on extremely narrow Terminal on Windows
- inconsistent cursor position when erasing in Termsharp
- selection of multibyte UTF characters on Linux
- scrollbar behavior on Windows
- error reporting from executed commands in Robot
- RISC-V cores reset
- several fixes in time framework
- output pin handling and interrupt clearing in PSE_GPIO
- minor fixes in PSE_SPI
- throwing invalid instruction exception on wrong CSR access in RISC-V
- CPU abort will now stop the failing CPU

▸ Assets 6

Renode 1.6.1

v1.6.1 · 6bf7286

PiotrZierhoffer released this on Jan 2 · 286 commits to master since this release

For installation instructions, see the README.

Added:

- CC2538 Flash Controller
- ECB mode for CC2538 Cryptoprocessor

Changed:

- unhandled read/write logs are now decorated with the CPU name instead of the number
- message acknowledge logic on PolarFire CAN controller

Fixed:

- race condition in PromptTerminal used by the Robot Framework
- Monitor socket not opening in certain situations
- unaligned accesses in RISC-V not setting the proper badaddr value
- handling of data exceeding the maximum packet size of USB endpoint
- memory map and CPU definition for SiFive FE310
- out of bounds access when using Ctrl+R with wrapped lines in the Monitor

Renode 1.6.2

v1.6.2 · 7c1897a

PiotrZierhoffer released this on Jan 10 · 248 commits to master since this release

For installation instructions, see the README.

Added:

- instructions on running in Docker
- --pid-file option to save Renode's process ID to a file

Changed:

- RISC-V X0 register is now protected from being written from the Monitor
- Renode will now close when it receives a signal from the environment (e.g. Ctrl+C from the console window)
- invalid instructions in RISC-V will no longer lead to CPU abort - an exception will be issued instead, to be handled by the guest software
- Robot tests will now log more

Fixed:

- formatting of symbol logging
- error reporting in Robot tests using the `Requires` keyword
- Microsemi's M-V CPU description

Renode 1.7.1

v1.7.1 · bc47e8b

PiotrZierhoffer released this on May 15 · 131 commits to master since this release

For installation instructions, see the README.

Added:

- integration layer for Verilator
- base infrastructure for verilated peripherals
- base class for verilated UARTs, with analyzer support
- Linux on LiteX with VexRiscv demo

Changed:

- RISC-V CPUs now don't need CLINT in their constructor, but will accept any abstract time provider
- updated LiteX with PicoRV32 and LiteX with VexRiscv platform`

Fixed:

- sharing violation when trying to run downloaded files

Renode 1.8.1

v1.8.1 · dcbe1a0

PiotrZierhoffer released this on Oct 9 · 10 commits to master since this release

For installation instructions, see the README.

Added:

- LiteX with VexRiscv configuration running Zephyr
- USB/IP Server for attaching Renode peripherals as USB devices to host
- optional NMI support in RISC-V
- flash controller for EFR32
- I2C controller for LiteX
- SPI controller for PicoRV
- framebuffer controller for LiteX
- USB keyboard model

Changed:

- `-e` parameter for commands executed at startup can be provided multiple times
- `polarfire` platform is now renamed to `polarfire-soc`
- style of Robot Framework result files
- M25Q flash backend has changed from file to memory, allowing software to execute directly from it
- improved LiteX on Fomu platform
- terminals based on sockets now accept reconnections from clients

Fixed:

- `Bad IL` exceptions when running on Mono 6.4

Renode 1.7

v1.7 · b0f033b

PiotrZierhoffer released this on May 7 · 156 commits to master since this release

For installation instructions, see the README.

Added:

- PicoRV32 CPU
- LiteX platform with PicoRV32
- LiteX timer and ethernet (LiteEth) model
- Marax SoC with UART, timer and GPIO controller models
- Fomu target support with LiteX and VexRiscv
- SAM E70 Xplained platform with USART, TRNG and ethernet controller models
- STM32F4 Random Number Generator model
- PSE watchdog model
- PTP support in Cadence GEM ethernet model, along with several fixes
- option to execute CPUs in serial instead of parallel
- support for custom instructions in RISC-V
- `empty` keyword in REPL
- graphical display analyzer support on Windows
- multi-target GPIO support, along with the new REPL syntax
- local interrupts in PolarFire SoC platform
- option to pass variables to Robot tests via test.sh
- some SiFive FU540 tests
- network interface tester for Robot tests
- tests for PTP implementation in Zephyr

Changed:

- Micron MT25Q is now able to use file as a backend and does not need to have a separate memory provided in REPL
- Micron MT25Q now has selectable endianess
- `logFile` command will now create a copy of the previous log before overwriting it
- `sysbus LogPeripheralAccess` will now add the active CPU name and current PC to log messages
- single stepping of a CPU is now easier, it requires only a single call to `cpu Step` on a paused CPU
- NVIC reload value is now 24-bit
- reimplemented the STM32 UART model
- updated the PolarFire SoC memory map
- updated the SiFive FU540 memory map
- `GetClockSourceInfo` will now display the name of the timer
- Termsharp will no longer print the NULL character
- RISC-V cores will now abort when trying to run a disabled F/D instruction

Fixed:

- handling of divider in CompareingTimer
- reporting of download progress on some Mono versions
- running Robot tests on Windows
- generation of TAP helper on newest Mono releases
- Renode crashing after opening a socket on the same port twice
- serialization of data storage structures
- architecture name reported on GDB connection for Cortex-M CPUs
- highlighting of wrapped lines in the terminal on Windows
- TAB completion in the Monitor on Windows
- RNG determinism and serialization for multicore/multi-node systems
- SiFive FE310 interrupt connection
- instruction counting in RISC-V on MMU faults
- time progress in multicore systems
- fixes in M-V GPIO controller model
- several fixes and improvements in file backend storage layer
- several fixes in testing scripts
- several fixes in various LiteX peripherals
- several fixes in PSE QSPI and Micron MT25Q model

Renode 1.8

v1.8 · 1d4a200

PiotrZierhoffer released this on Sep 2 · 41 commits to master since this release

For installation instructions, see the README.

Added:

- support for RISCY core and the VEGA board
- UART and timer models for RISCY
- support for Minerva, a 32-bit RISC-V soft CPU
- LiteX with Minerva platform
- LiteX with VexRiscv on Arty platform
- SPI, Control and Status, SPI Flash and GPIO port peripheral models for LiteX
- PSE_PDMA peripheral model for the PolarFire SoC platform
- basic slave mode support in PSE I2C
- EtherBone bridge model to connect Renode with FPGA via EtherBone
- EtherBone bridge demo on Fomu
- RTCC and GPCRC peripheral models for EFR32
- support for deep sleep on Cortex-M cores
- option of bundling Renode as an ELF executable on Linux

Changed:

- GDB server is now started from the `machine` level instead of `cpu` and is able to handle multiple cores at once
- renamed `SetLossRangeWirelessFunction` to `SetRangeLossWirelessFunction`
- LiteX Ethernet now supports the MDIO interface
- updated memory map for several EFR32 platforms
- changed the interrupt handling of EFR32_USART
- several changes in Ethernet PHY
- switch is now started immediately after creation
- the Monitor (and other mechanisms) now uses caching, increasing its performance
- Robot tests are now part of packages
- Robot tests no longer cause the Monitor telnet server to start automatically
- REPL files now accept multiline strings delimited with triple apostrophe
- UART analyzers are writing to the Renode log when running from Robot
- simplified command line switches for running Robot tests
- some Robot keywords (e.g. `LogToFile`) are not saved between related tests

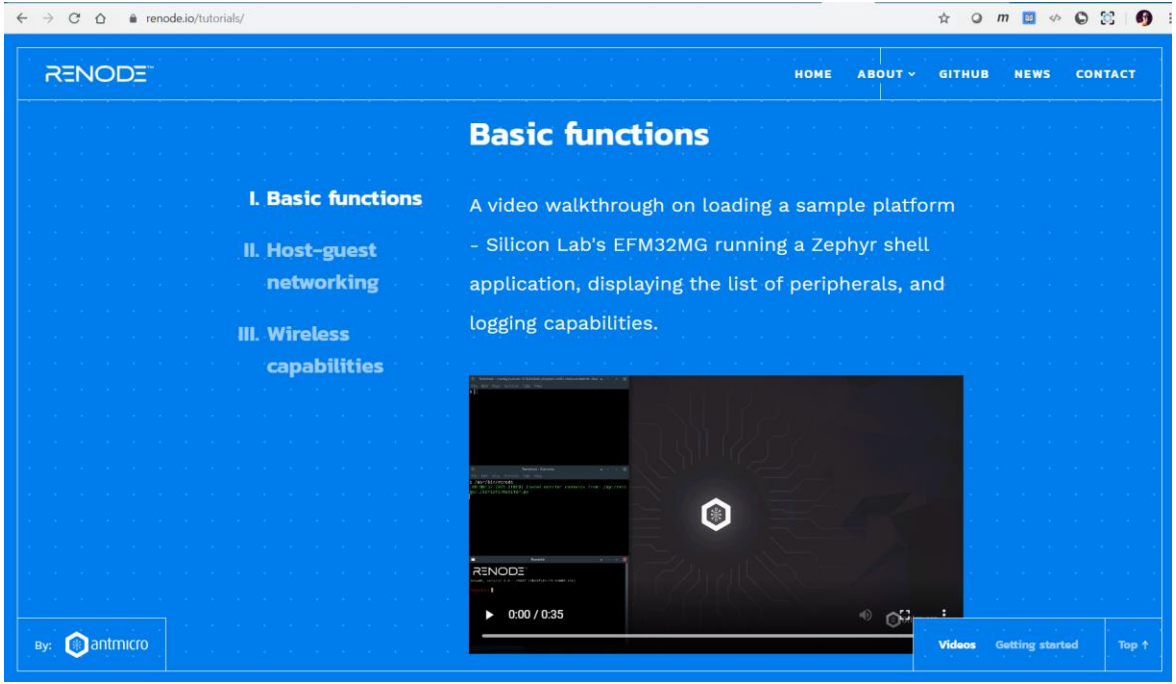
Fixed:

- compilation of verilated peripheral classes in Windows (backported to 1.7.1 package)
- determinism of SAM E70 tests
- crash when using `LogLevel` command with `--hide-log` switch
- ad-hoc compiler behavior in Windows
- crash on too short Ethernet packets
- byte read behavior in NS16550
- auto update behavior of PSE_Timer
- connection mode when running the Monitor via telnet
- deserialization of `SerializableStreamView`
- crash when completing interrupts in PLIC when no interrupt is pending
- Renode startup position on Windows with desktop scaling enabled
- fence.* operation decoding in RISC-V
- invalid size reported by SD card
- crash when trying to set the same log file twice
- compilation issues on GCC 9

Updated Renode: antmicro Training

Updated Renode: antmicro Training

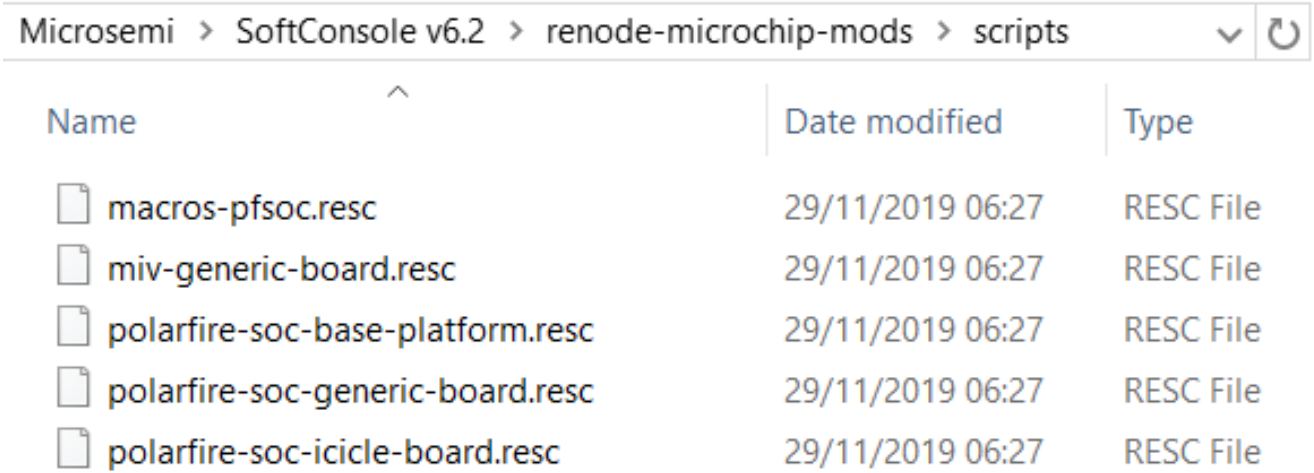
https://renode.io/tutorials/



Updated Renode:
Microchip Renode Scripts Moved

Updated Renode: Microchip Renode Scripts Moved

SoftConsole Install Directory -> renode-microchip-mods -> scripts



All launch scripts can now be found here
Board files are **still** in renode\boards
SoftConsole v6.2 Install Directory -> Renode

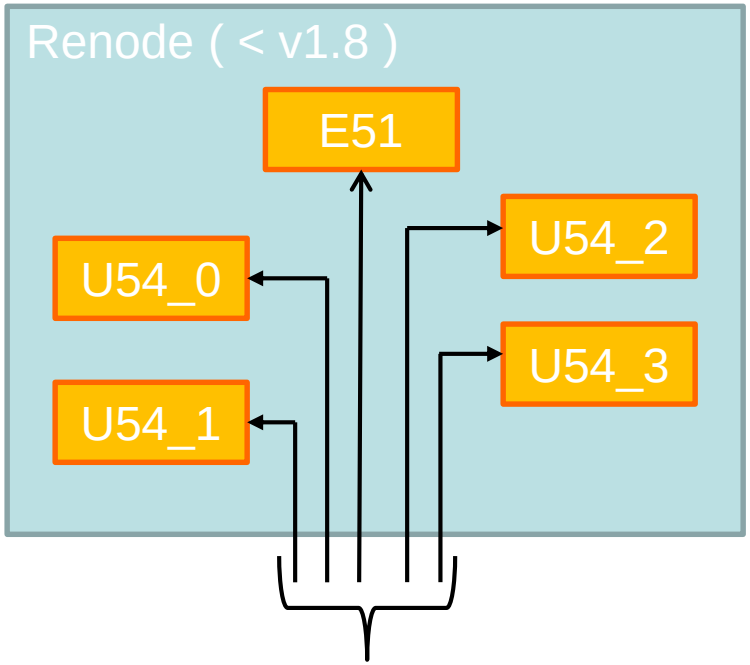
GDB Multi-Core Debugging

GDB Multi-Core Debugging

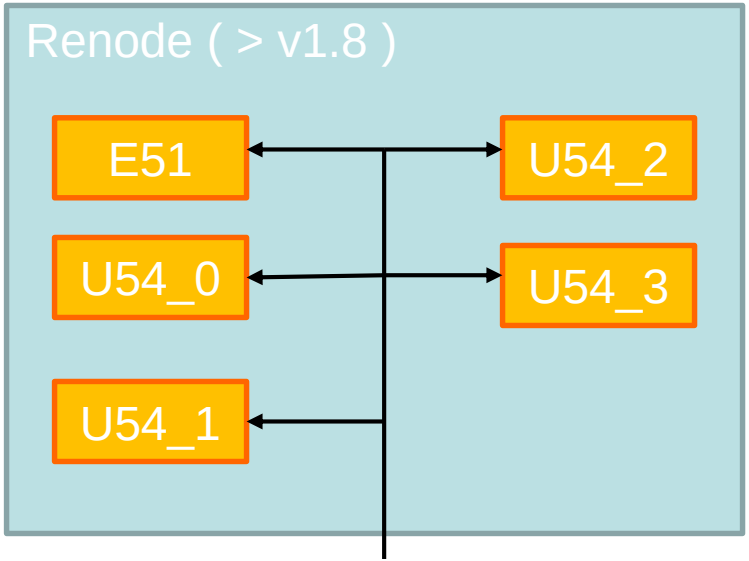
- **Previously each CPU needed its own GDB server**
  - “cpu0 StartGDBServer 3333 true”
  - “cpu1 StartGDBServer 3334 true”
  - Etc …

- **Now the machine has a single server for each thread / core**
  - “machine StartGDBServer $GDB_SERVER_PORT true”
  - The GDB server port can be passed as an argument

GDB Multi-Core Debugging

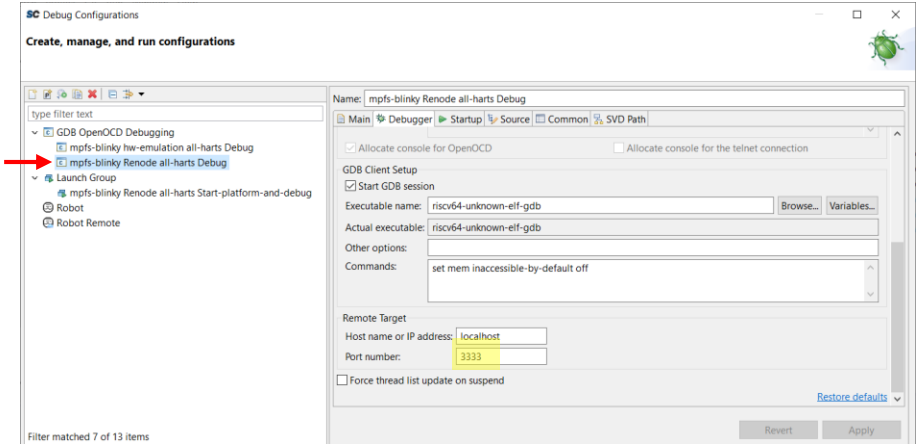


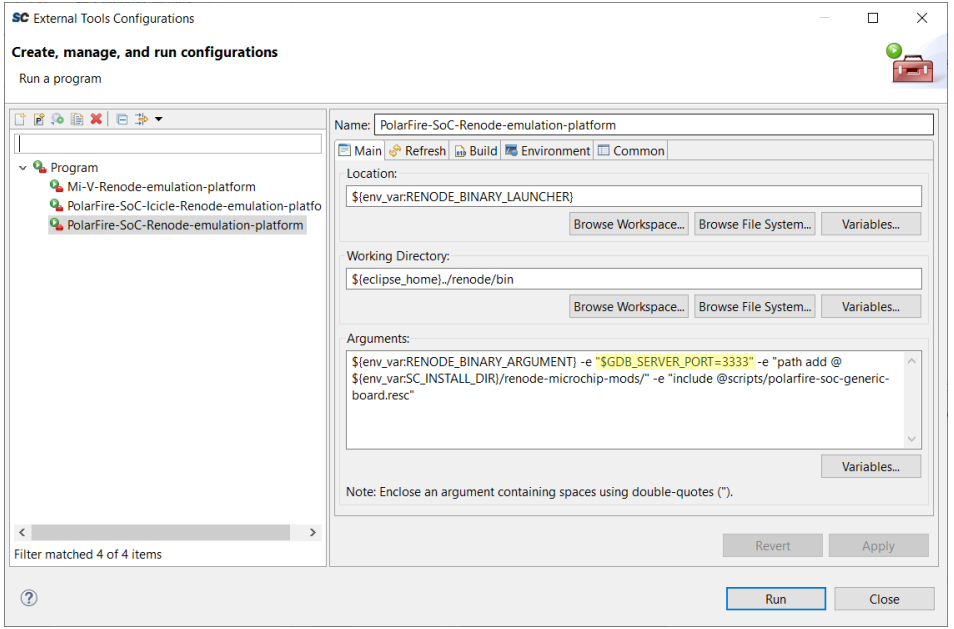
Individual GDB sessions



Single GDB session

GDB Multi-Core Debugging





workspace.examples - C:\Users\hbreslin\Downloads\scWindows-6.2.0.249-20191129-134636 (1)\scWindows-6.2.0.249-20191129-134636\extras\workspace.examples\mpfs-blinky\src\application\hart0\e51.c - Microsemi SoftConsole v6.2.0.249-generic Renode ...

File Edit Source Refactor Navigate Search Project Run Window Help

Quick Access

Project Explorer

- fpga-cortex-m1-blinky
- m1fpga-cortex-m1-blinky
- miv-rv32im-interrupt-blinky
- miv-rv32im-systick-blinky
- miv-rv32imaf-mandelbrot-uart
- miv-rv32imaf-raytracer-uart-cpp
- mpfs-blinky
  - Binaries
  - Includes
  - src
    - application
      - hart0
        - e51.c
      - hart1
      - inc
    - modules
    - platform
  - Debug
  - mpfs-blinky hw-emulation all-harts Debug.launch
  - mpfs-blinky Renode all-harts Debug.launch
  - mpfs-blinky Renode all-harts Start-platform-and-debug.launc
  - README.md
- mpfs-freertos-lwip
- mpfs-mustein-julia
- smartfusion-cortex-m3-blinky
- smartfusion2-cortex-m3-blinky

e51.c

```
168         for (uint64_t i = 0; i< delay_loop_max; i++) {
169             delay_loop_sum = delay_loop_sum + i;
170         }
171
172         safe_MSS_UART0_polled_tx_string("Setting outputs 0, 1 and 2 to low\r\n");
173         MSS_GPIO_set_output(GPIO1_LO, MSS_GPIO_0, 0);
174         MSS_GPIO_set_output(GPIO1_LO, MSS_GPIO_1, 0);
175         MSS_GPIO_set_output(GPIO1_LO, MSS_GPIO_2, 0);
176     }
177 }
178
179
180 /* Main function for the HART0(E51 processor).
181  * Application code running on HART0 is placed here.
182  * UART0 PLIC interrupt is enabled on hart0.*/
183 void e51(void)
184 {
185     e51_setup();
186     e51_application();
187
188     /* Shouldn't never reach this point */
189     while (1)
190     {
191        volatile static uint64_t counter = 0U;
192
193        /* Added some code as gdb hangs when stepping through an empty infinite loop */
194        counter = counter + 1;
195     }
196 }
197
198
199
```

Outline

- mpfs_hal/mss_hal.h
- drivers/mss_gpio/mss_gpio.h
- drivers/mss_uart/mss_uart.h
- inc/common.h
- uart_lock : uint64_t
- gpio0_bit0_or_gpio2_bit13_plic_0_IRQHandler
- gpio0_bit1_or_gpio2_bit13_plic_1_IRQHandler
- gpio0_bit2_or_gpio2_bit13_plic_2_IRQHandler
- gpio0_non_direct_plic_IRQHandler(void) : uint
- gpio1_non_direct_plic_IRQHandler(void) : uint
- gpio2_non_direct_plic_IRQHandler(void) : uint
- e51_setup(void) : void
- e51_application(void) : void
- e51(void) : void

Problems Tasks Console Properties Debug

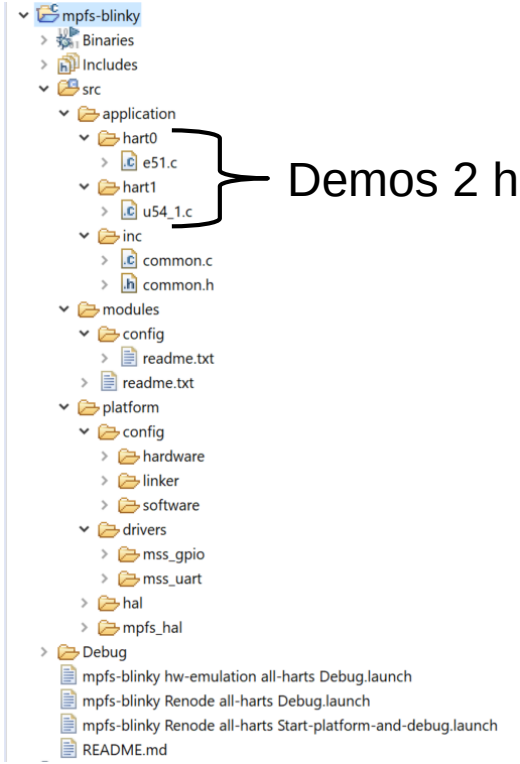
Writable Smart Insert 188 : 43

Updates to Example Projects

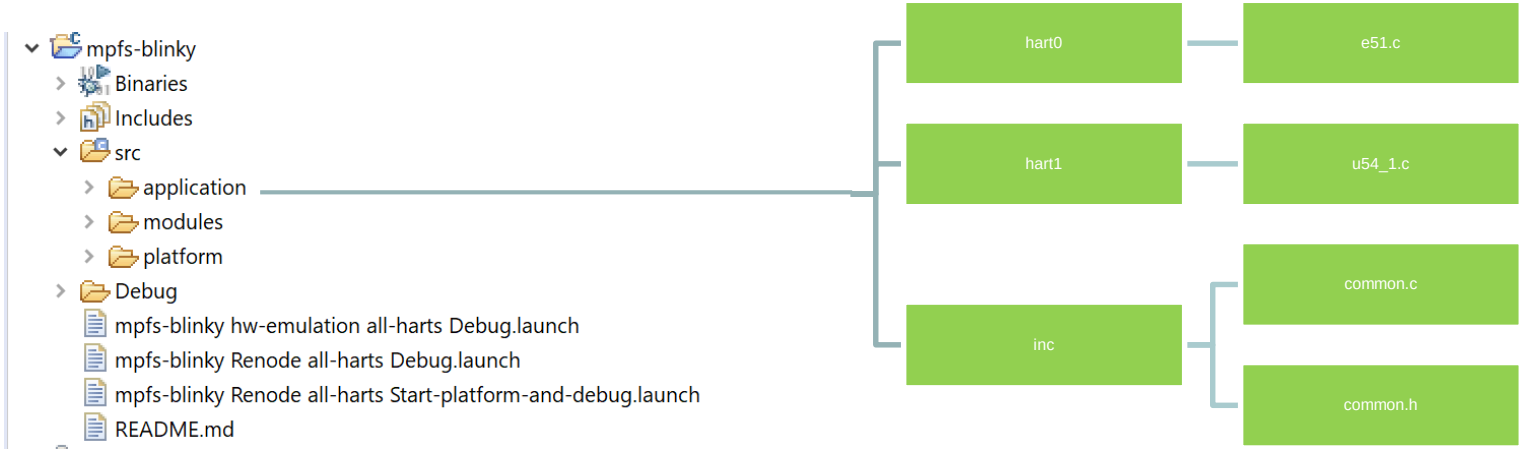
Updates to pse-blinky

PSE Blinky is now mpfs-blinky

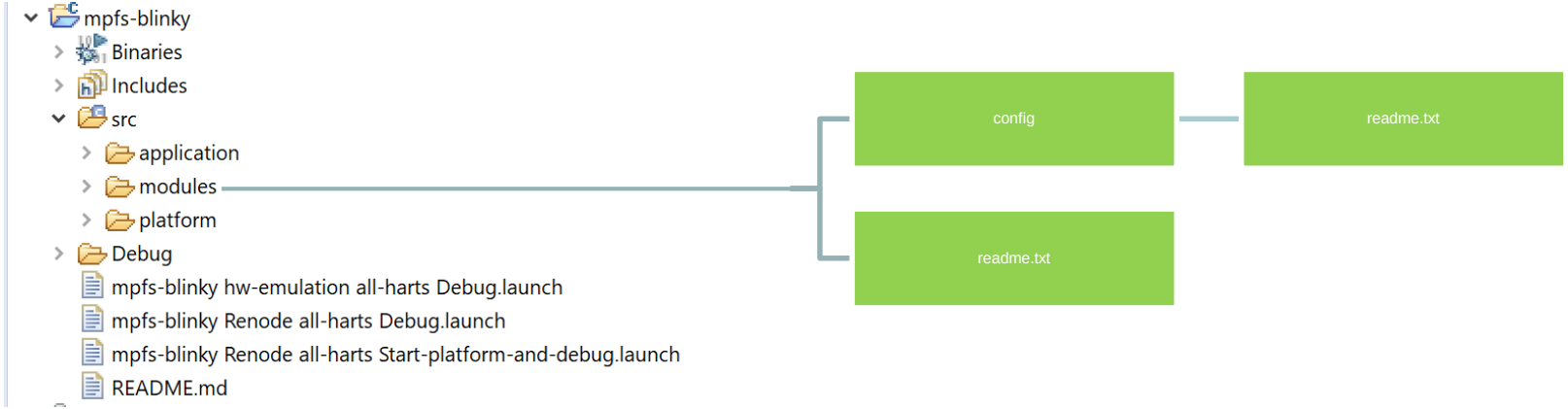
Microchip **P**olar**F**ire **S**oC



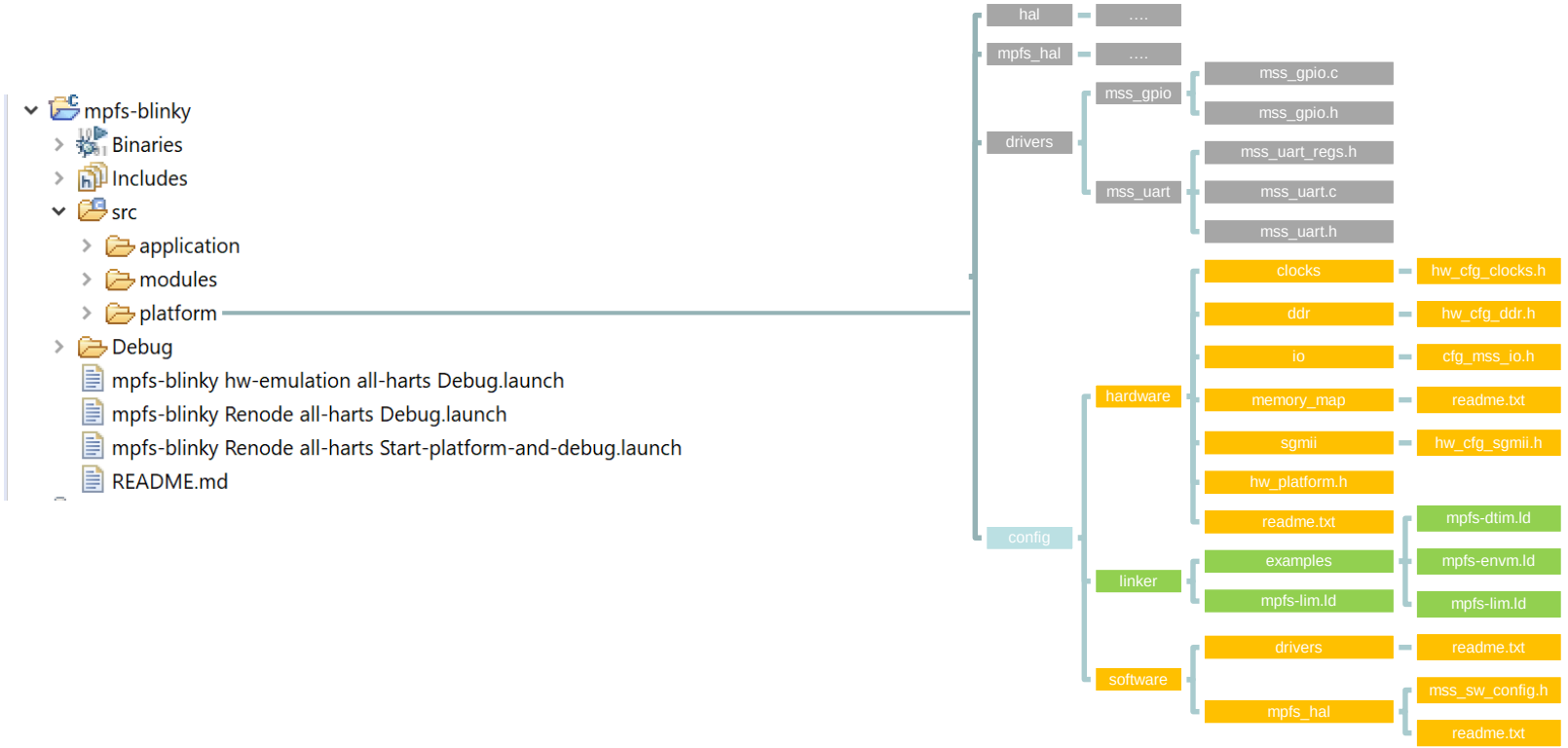
Updates to pse-blinky



Updates to pse-blinky



Updates to pse-blinky



Updates to pse-blinky

mpfs-blinky only uses one UART

u54_1.c

```
36  MSS_UART_DATA_8_BITS | MSS_UART_NO_PARITY);
37  MSS_UART_polled_tx_string(&g_mss_uart1_lo,
38          "Hello World from u54 core 0 - hart1.\r\n");
39
40  while (1) {
41      mcycle_start = readmcycle();
42
43      for (i = 0; i < num_loops; i++) {
44          dummy_h1 = i;
45      }
46      MSS_UART_polled_tx_string(&g_mss_uart1_lo, "for loop finished\r\n");
47
48      hartid = read_csr(mhartid);
```

e51.c

```
130     // Stay in the infinite loop, never return from main
131     volatile uint64_t delay_loop_sum = 0;
132     uint64_t mcycle_start = readmcycle();
133
134     for (uint64_t i = 0; i< 1000; i++) {
135         delay_loop_sum = delay_loop_sum + i;
136     }
137
138     MSS_UART_polled_tx_string(&g_mss_uart2_lo,
139             "Setting outputs 0, 1 and 2 to high\r\n");
140     MSS_GPIO_set_output(GPIO1_LO, MSS_GPIO_0, 1);
141     MSS_GPIO_set_output(GPIO1_LO, MSS_GPIO_1, 1);
142     MSS_GPIO_set_output(GPIO1_LO, MSS_GPIO_2, 1);
```

pse-blinky

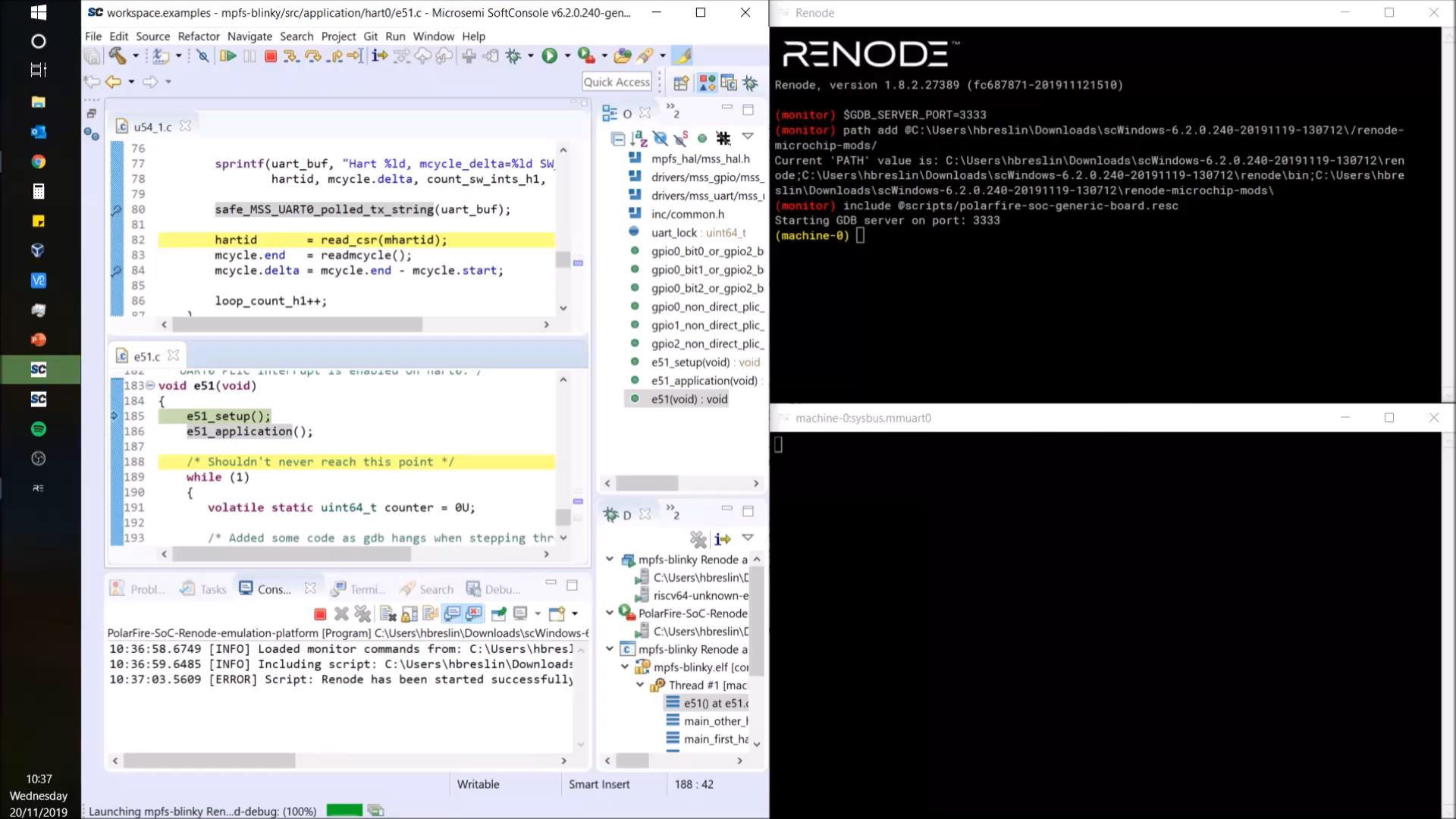
u54_1.c

```
73      for (uint64_t i = 0; i < num_loops; i++) {
74          dummy_h1 = i;
75      }
76
77      sprintf(uart_buf, "Hart %ld, mcycle_delta=%ld SW_IRQs=%ld mcycle=%ld\r\n",
78              hartid, mcycle.delta, count_sw_ints_h1, readmcycle());
79
80      safe_MSS_UART0_polled_tx_string(uart_buf);
81
82      hartid       = read_csr(mhartid);
83      mcycle.end   = readmcycle();
84      mcycle.delta = mcycle.end - mcycle.start;
```

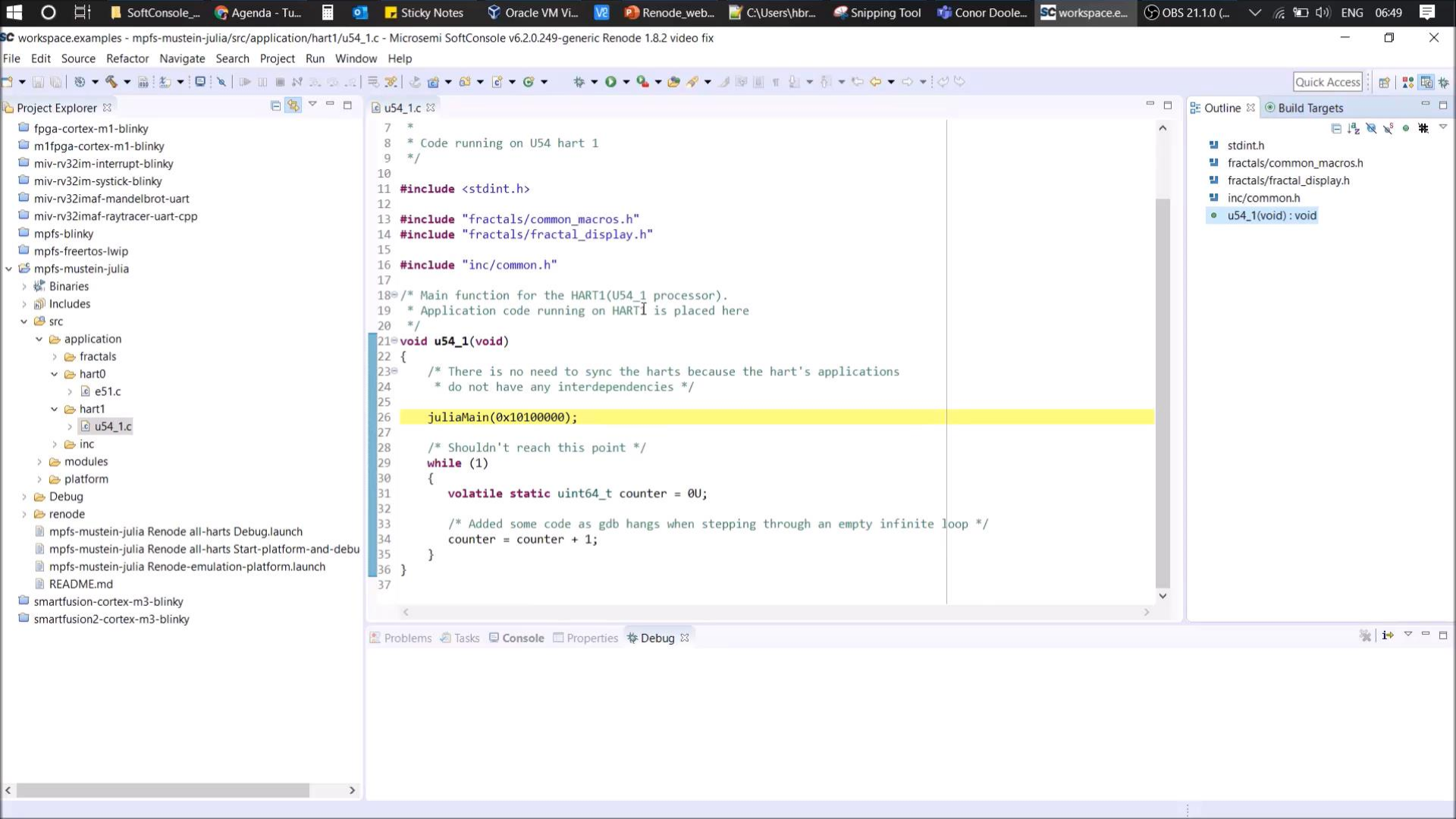
e51.c

```
157
158     for (uint64_t i = 0; i< delay_loop_max; i++) {
159         delay_loop_sum = delay_loop_sum + i;
160     }
161
162     safe_MSS_UART0_polled_tx_string("Setting outputs 0, 1 and 2 to high\r\n");
163
164     MSS_GPIO_set_output(GPIO1_LO, MSS_GPIO_0, 1);
165     MSS_GPIO_set_output(GPIO1_LO, MSS_GPIO_1, 1);
166     MSS_GPIO_set_output(GPIO1_LO, MSS_GPIO_2, 1);
167
168     for (uint64_t i = 0; i< delay_loop_max; i++) {
```

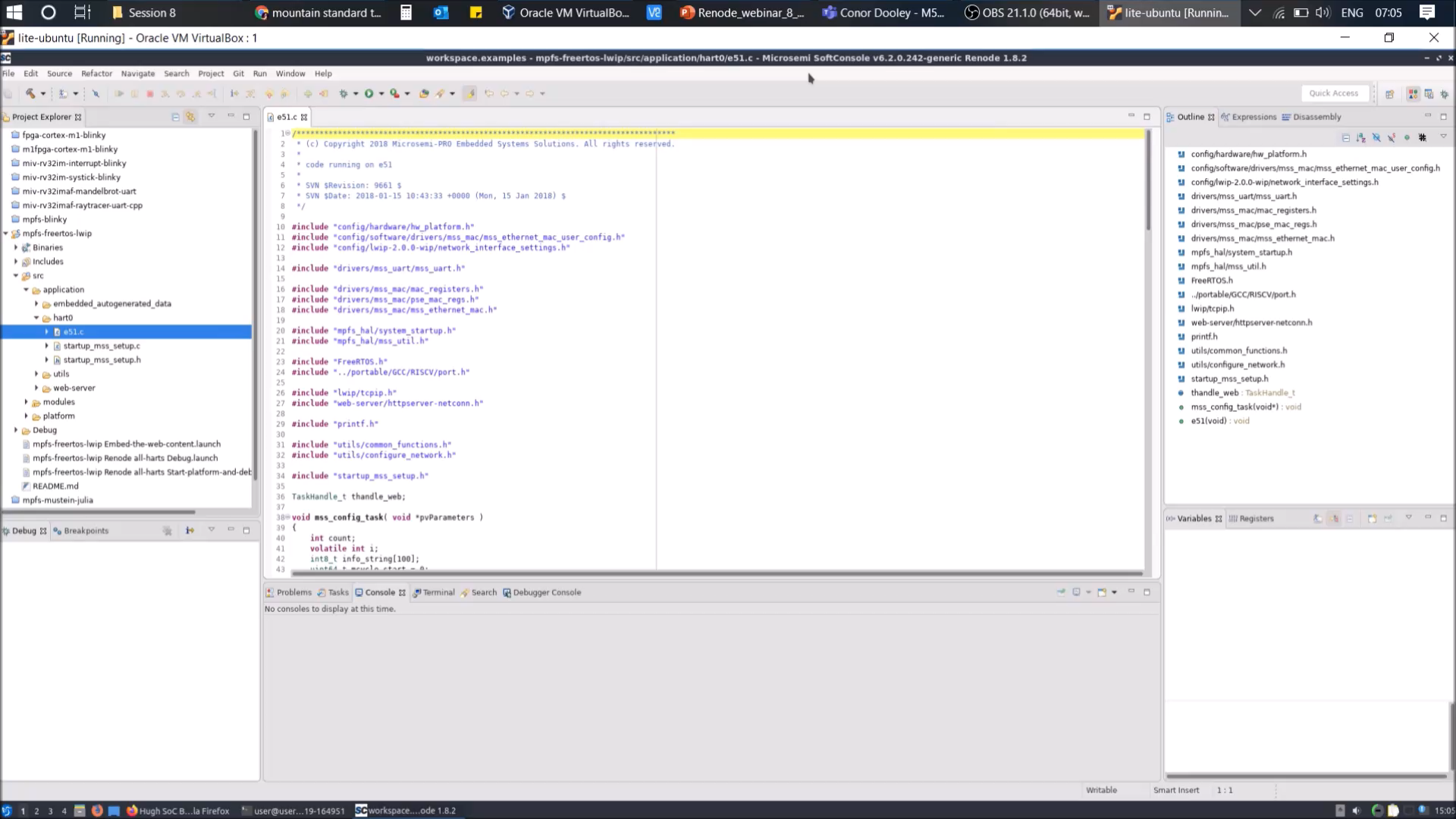
mpfs-blinky

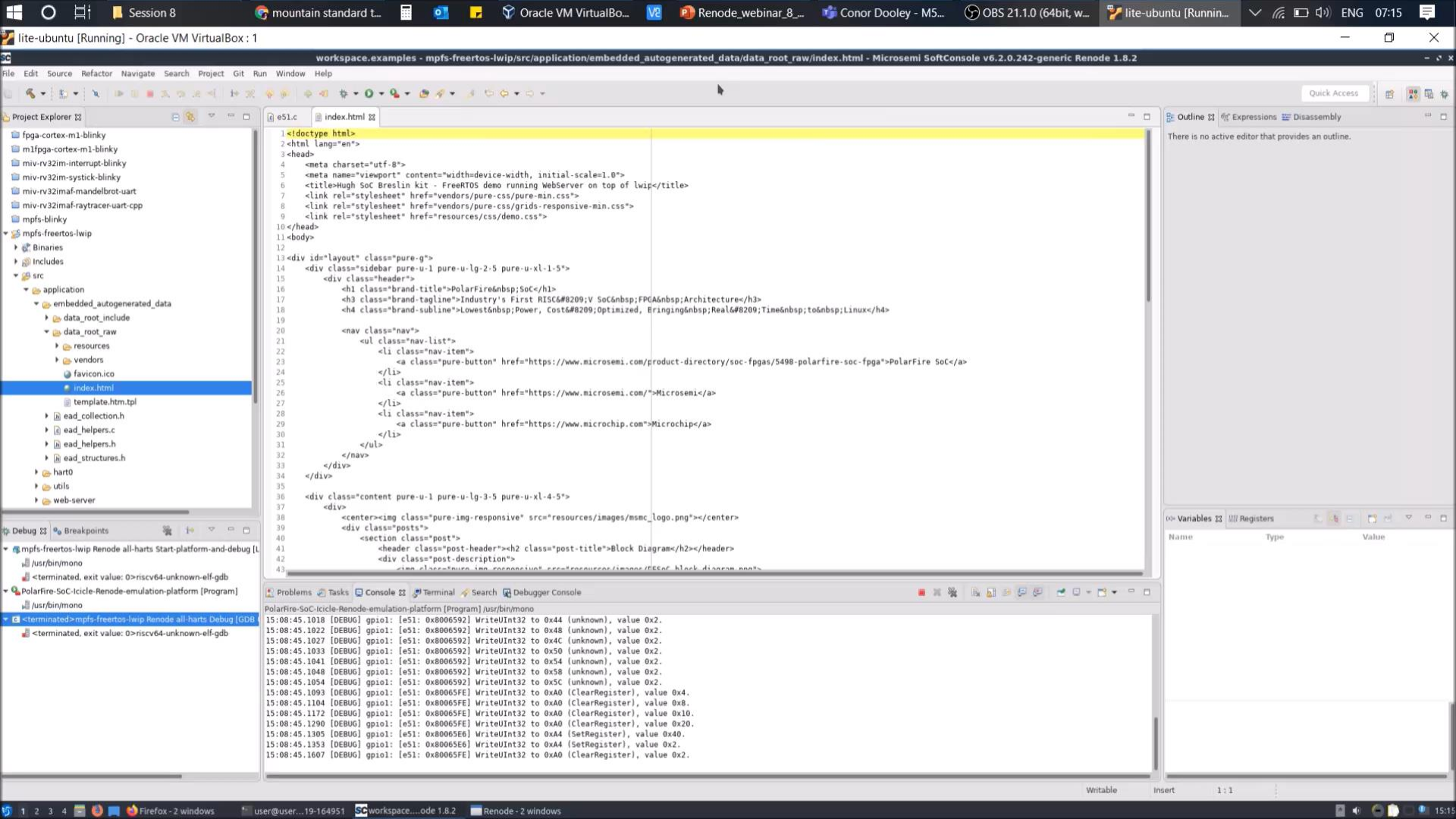

workspace.examples - mpfs-blinky/src/application/hart0/e51.c - Microsemi SoftConsole v6.2.0.240-gen...
File Edit Source Refactor Navigate Search Project Git Run Window Help
Quick Access
u54_1.c
sprintf(uart_buf, "Hart %ld, mcycle_delta=%ld SW
hartid, mcycle.delta, count_sw_ints_h1,
safe_MSS_UART0_polled_tx_string(uart_buf);
hartid = read_csr(mhartid);
mcycle.end = readmcycle();
mcycle.delta = mcycle.end - mcycle.start;
loop_count_h1++;
e51.c
void e51(void)
{
e51_setup();
e51_application();
/* Shouldn't never reach this point */
while (1)
{
volatile static uint64_t counter = 0U;
/* Added some code as gdb hangs when stepping thr
mpfs_hal/mss_hal.h
drivers/mss_gpio/mss_
drivers/mss_uart/mss_
inc/common.h
uart_lock : uint64_t
gpio0_bit0_or_gpio2_b
gpio0_bit1_or_gpio2_b
gpio0_bit2_or_gpio2_b
gpio0_non_direct_plic_
gpio1_non_direct_plic_
gpio2_non_direct_plic_
e51_setup(void) : void
e51_application(void) :
e51(void) : void
Probl... Tasks Cons... Termi... Search Debu...
PolarFire-SoC-Renode-emulation-platform [Program] C:\Users\hbreslin\Downloads\scWindows-
10:36:58.6749 [INFO] Loaded monitor commands from: C:\Users\hbres
10:36:59.6485 [INFO] Including script: C:\Users\hbreslin\Downloads
10:37:03.5609 [ERROR] Script: Renode has been started successfully
mpfs-blinky Renode a
C:\Users\hbreslin\D
riscv64-unknown-e
PolarFire-SoC-Renode
C:\Users\hbreslin\D
mpfs-blinky Renode a
mpfs-blinky.elf [co
Thread #1 [mac
e51() at e51.c
main_other_h
main_first_ha
Writable
Smart Insert
188 : 42
Launching mpfs-blinky Ren...d-debug: (100%)
Renode
RENODE
Renode, version 1.8.2.27389 (fc687871-201911121510)
(monitor) $GDB_SERVER_PORT=3333
(monitor) path add @C:\Users\hbreslin\Downloads\scWindows-6.2.0.240-20191119-130712\/renode-microchip-mods/
Current 'PATH' value is: C:\Users\hbreslin\Downloads\scWindows-6.2.0.240-20191119-130712\renode;C:\Users\hbreslin\Downloads\scWindows-6.2.0.240-20191119-130712\renode\bin;C:\Users\hbreslin\Downloads\scWindows-6.2.0.240-20191119-130712\renode-microchip-mods\
(monitor) include @scripts/polarfire-soc-generic-board.resc
Starting GDB server on port: 3333
(machine-0)
machine-0:sysbus.mmuart0
10:37
Wednesday
20/11/2019

New Sample Project: mpfs-mustein-julia


workspace.examples - mpfs-mustein-julia/src/application/hart1/u54_1.c - Microsemi SoftConsole v6.2.0.249-generic Renode 1.8.2 video fix
File Edit Source Refactor Navigate Search Project Run Window Help
Project Explorer
fpga-cortex-m1-blinky
m1fpga-cortex-m1-blinky
miv-rv32im-interrupt-blinky
miv-rv32im-systick-blinky
miv-rv32imaf-mandelbrot-uart
miv-rv32imaf-raytracer-uart-cpp
mpfs-blinky
mpfs-freertos-lwip
mpfs-mustein-julia
Binaries
Includes
src
application
fractals
hart0
e51.c
hart1
u54_1.c
inc
modules
platform
Debug
renode
mpfs-mustein-julia Renode all-harts Debug.launch
mpfs-mustein-julia Renode all-harts Start-platform-and-debu
mpfs-mustein-julia Renode-emulation-platform.launch
README.md
smartfusion-cortex-m3-blinky
smartfusion2-cortex-m3-blinky
u54_1.c
7 *
8 * Code running on U54 hart 1
9 */
10
11 #include <stdint.h>
12
13 #include "fractals/common_macros.h"
14 #include "fractals/fractal_display.h"
15
16 #include "inc/common.h"
17
18 /* Main function for the HART1(U54_1 processor).
19 * Application code running on HART1 is placed here
20 */
21 void u54_1(void)
22 {
23 /* There is no need to sync the harts because the hart's applications
24 * do not have any interdependencies */
25
26 juliaMain(0x10100000);
27
28 /* Shouldn't reach this point */
29 while (1)
30 {
31 volatile static uint64_t counter = 0U;
32
33 /* Added some code as gdb hangs when stepping through an empty infinite loop */
34 counter = counter + 1;
35 }
36 }
37
Outline
Build Targets
stdint.h
fractals/common_macros.h
fractals/fractal_display.h
inc/common.h
u54_1(void) : void
Problems Tasks Console Properties Debug

New Sample Project: mpfs-freertos-lwip


lite-ubuntu [Running] - Oracle VM VirtualBox : 1
workspace.examples - mpfs-freertos-lwip/src/application/hart0/e51.c - Microsemi SoftConsole v6.2.0.242-generic Renode 1.8.2
File Edit Source Refactor Navigate Search Project Git Run Window Help
Project Explorer
fpga-cortex-m1-blinky
m1fpga-cortex-m1-blinky
miv-rv32im-interrupt-blinky
miv-rv32im-systick-blinky
miv-rv32imaf-mandelbrot-uart
miv-rv32imaf-raytracer-uart-cpp
mpfs-blinky
mpfs-freertos-lwip
Binaries
Includes
src
application
embedded_autogenerated_data
hart0
e51.c
startup_mss_setup.c
startup_mss_setup.h
utils
web-server
modules
platform
Debug
mpfs-freertos-lwip Embed-the-web-content.launch
mpfs-freertos-lwip Renode all-harts Debug.launch
mpfs-freertos-lwip Renode all-harts Start-platform-and-deb
README.md
mpfs-mustein-julia
Debug
Breakpoints
e51.c
/***
* (c) Copyright 2018 Microsemi-PRO Embedded Systems Solutions. All rights reserved.
*
* code running on e51
*
* SVN $Revision: 9661 $
* SVN $Date: 2018-01-15 10:43:33 +0000 (Mon, 15 Jan 2018) $
*/
#include "config/hardware/hw_platform.h"
#include "config/software/drivers/mss_mac/mss_ethernet_mac_user_config.h"
#include "config/lwip-2.0.0-wip/network_interface_settings.h"
#include "drivers/mss_uart/mss_uart.h"
#include "drivers/mss_mac/mac_registers.h"
#include "drivers/mss_mac/pse_mac_regs.h"
#include "drivers/mss_mac/mss_ethernet_mac.h"
#include "mpfs_hal/system_startup.h"
#include "mpfs_hal/mss_util.h"
#include "FreeRTOS.h"
#include "../portable/GCC/RISCV/port.h"
#include "lwip/tcpip.h"
#include "web-server/httpserver-netconn.h"
#include "printf.h"
#include "utils/common_functions.h"
#include "utils/configure_network.h"
#include "startup_mss_setup.h"
TaskHandle_t thandle_web;
void mss_config_task(void *pvParameters)
{
int count;
volatile int i;
int8_t info_string[100];
Outline
Expressions
Disassembly
config/hardware/hw_platform.h
config/software/drivers/mss_mac/mss_ethernet_mac_user_config.h
config/lwip-2.0.0-wip/network_interface_settings.h
drivers/mss_uart/mss_uart.h
drivers/mss_mac/mac_registers.h
drivers/mss_mac/pse_mac_regs.h
drivers/mss_mac/mss_ethernet_mac.h
mpfs_hal/system_startup.h
mpfs_hal/mss_util.h
FreeRTOS.h
../portable/GCC/RISCV/port.h
lwip/tcpip.h
web-server/httpserver-netconn.h
printf.h
utils/common_functions.h
utils/configure_network.h
startup_mss_setup.h
thandle_web : TaskHandle_t
mss_config_task(void*) : void
e51(void) : void
Variables
Registers
Problems
Tasks
Console
Terminal
Search
Debugger Console
No consoles to display at this time.
Writable
Smart Insert
1 : 1


lite-ubuntu [Running] - Oracle VM VirtualBox : 1
workspace.examples - mpfs-freertos-lwip/src/application/embedded_autogenerated_data/data_root_raw/index.html - Microsemi SoftConsole v6.2.0.242-generic Renode 1.8.2
File Edit Source Refactor Navigate Search Project Git Run Window Help
Project Explorer
fpga-cortex-m1-blinky
m1fpga-cortex-m1-blinky
miv-rv32im-interrupt-blinky
miv-rv32im-systick-blinky
miv-rv32imaf-mandelbrot-uart
miv-rv32imaf-raytracer-uart-cpp
mpfs-blinky
mpfs-freertos-lwip
Binaries
Includes
src
application
embedded_autogenerated_data
data_root_include
data_root_raw
resources
vendors
favicon.ico
index.html
template.htm.tpl
ead_collection.h
ead_helpers.c
ead_helpers.h
ead_structures.h
hart0
utils
web-server
e51.c
index.html
1 <!doctype html>
2 <html lang="en">
3 <head>
4 <meta charset="utf-8">
5 <meta name="viewport" content="width=device-width, initial-scale=1.0">
6 <title>Hugh SoC Breslin kit - FreeRTOS demo running WebServer on top of lwip</title>
7 <link rel="stylesheet" href="vendors/pure-css/pure-min.css">
8 <link rel="stylesheet" href="vendors/pure-css/grids-responsive-min.css">
9 <link rel="stylesheet" href="resources/css/demo.css">
10 </head>
11 <body>
12
13
14
15
16 <h1 class="brand-title">PolarFire SoC</h1>
17 <h3 class="brand-tagline">Industry's First RISC‑V SoC FPGA Architecture</h3>
18 <h4 class="brand-subline">Lowest Power, Cost‑Optimized, Bringing Real‑Time to Linux</h4>
19
20 <nav class="nav">
21 <ul class="nav-list">
22 <li class="nav-item">
23 PolarFire SoC
24
25 <li class="nav-item">
26 Microsemi
27
28 <li class="nav-item">
29 Microchip
30
31
32 </nav>
33
34
35
36
37
38 <img class="pure-img-responsive" src="resources/images/msmc_logo.png">
39
40 <section class="post">
41 <header class="post-header"><h2 class="post-title">Block Diagram</h2></header>
42
43
Outline
Expressions
Disassembly
There is no active editor that provides an outline.
Variables
Registers
Name
Type
Value
Debug
Breakpoints
mpfs-freertos-lwip Renode all-harts Start-platform-and-debug [L
/usr/bin/mono
<terminated, exit value: 0>riscv64-unknown-elf-gdb
PolarFire-SoC-Icicle-Renode-emulation-platform [Program]
/usr/bin/mono
<terminated>mpfs-freertos-lwip Renode all-harts Debug [GDB
<terminated, exit value: 0>riscv64-unknown-elf-gdb
Problems
Tasks
Console
Terminal
Search
Debugger Console
PolarFire-SoC-Icicle-Renode-emulation-platform [Program] /usr/bin/mono
15:08:45.1018 [DEBUG] gpio1: [e51: 0x8006592] WriteUInt32 to 0x44 (unknown), value 0x2.
15:08:45.1022 [DEBUG] gpio1: [e51: 0x8006592] WriteUInt32 to 0x48 (unknown), value 0x2.
15:08:45.1027 [DEBUG] gpio1: [e51: 0x8006592] WriteUInt32 to 0x4C (unknown), value 0x2.
15:08:45.1033 [DEBUG] gpio1: [e51: 0x8006592] WriteUInt32 to 0x50 (unknown), value 0x2.
15:08:45.1041 [DEBUG] gpio1: [e51: 0x8006592] WriteUInt32 to 0x54 (unknown), value 0x2.
15:08:45.1048 [DEBUG] gpio1: [e51: 0x8006592] WriteUInt32 to 0x58 (unknown), value 0x2.
15:08:45.1054 [DEBUG] gpio1: [e51: 0x8006592] WriteUInt32 to 0x5C (unknown), value 0x2.
15:08:45.1093 [DEBUG] gpio1: [e51: 0x80065FE] WriteUInt32 to 0xA0 (ClearRegister), value 0x4.
15:08:45.1104 [DEBUG] gpio1: [e51: 0x80065FE] WriteUInt32 to 0xA0 (ClearRegister), value 0x8.
15:08:45.1172 [DEBUG] gpio1: [e51: 0x80065FE] WriteUInt32 to 0xA0 (ClearRegister), value 0x10.
15:08:45.1290 [DEBUG] gpio1: [e51: 0x80065FE] WriteUInt32 to 0xA0 (ClearRegister), value 0x20.
15:08:45.1305 [DEBUG] gpio1: [e51: 0x80065E6] WriteUInt32 to 0xA4 (SetRegister), value 0x40.
15:08:45.1353 [DEBUG] gpio1: [e51: 0x80065E6] WriteUInt32 to 0xA4 (SetRegister), value 0x2.
15:08:45.1607 [DEBUG] gpio1: [e51: 0x80065FE] WriteUInt32 to 0xA0 (ClearRegister), value 0x2.
Writable
Insert
1 : 1
Firefox - 2 windows
user@user...19-164951
workspace....ode 1.8.2
Renode - 2 windows

Agenda

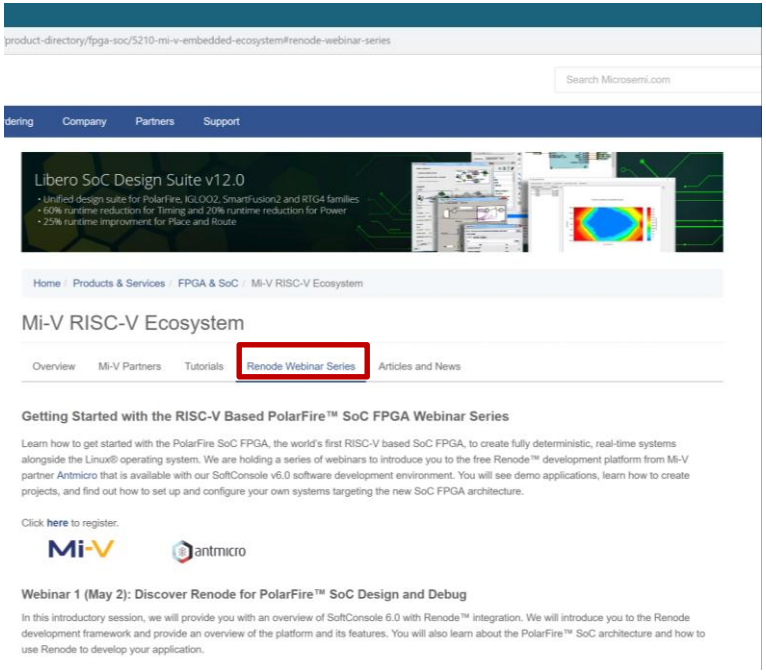
- **Updated Renode**
  - antmicro Training
  - Renode Scripts Moved
- **GDB Multi-Core Debugging**
- **Updated to Example Projects**
  - Changes to PSE Blinky
  - New Sample Project: mpfs-mustein-julia
  - New Sample Project: mpfs-freertos-lwip

First / Second Thursdays

Dec. 5	-	Webinar 8: What's New in SoftConsole v6.2
Jan. 9	-	Webinar 9: Getting Started With PolarFire SoC
Feb. 13	-	Webinar 10: Introduction to the PolarFire SoC Baremetal Library
Mar. 12	-	Webinar 11: Handling Binaries
April 9	-	Webinar 12: Two Baremetal Applications on PolarFire SoC
May 14	-	Webinar 13: Linux on Renode
June 11	-	Webinar 14: Building Applications for Linux on PolarFire SoC
July 9	-	Webinar 15: Real-Time (AMP Mode) on PolarFire SoC

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 and Pre-Existing model in PolarFire SoC

Webinar 7: How to Write Custom Models

Thank You

Any Questions?