Open Source RTOS Ports on RISC-V

Nitin Deshpande
Sr. Staff Engineer – Firmware
Open Source RTOS Ports on RISC-V

Motivation
- Various application domains – Industrial, IoT, Military, Space
- Bare metal to large scale applications including networking stacks
- Resource constrained environment
- Create infrastructure to support variety of applications on RISC-V processors

Infrastructure
- Multiple open source RTOS support
- Software development and debugging tools
- Firmware and example applications
Open Source RTOS Ports on RISC-V

- **FreeRTOS:**
  - Modified GNU General Public License

- **Mynewt:**
  - [https://mynewt.apache.org/](https://mynewt.apache.org/)
  - Apache License, version 2.0

- **Huawei LiteOS:**
  - [https://github.com/LITEOS](https://github.com/LITEOS)
  - Open Source License
FreeRTOS

• **Features**
  – Popular cross-platform RTOS
  – Prioritized preemptive scheduling with time-slicing
  – Tick-less mode of operation
  – Idle task hook for applications
  – SafeRTOS – Certified version of FreeRTOS for safety critical applications
  – No BSP/HAL
  – Requires third party network protocols

• **Port**
  – 32bit: Running on RISC-V Soft Processor
  – 64bit: Running on Spike emulator

• **Next steps**
  – Publish as contributed port
  – Work towards distributing it as official port
• **Features**
  - Wireless stacks NimBLE, Bluetooth mesh, WiFi and more
  - IoT protocols e.g. CoAP
  - Secure bootloader to verify firmware integrity and authenticity
  - Power management
  - Cross-platform, well defined HAL and BSP
  - Package management and build using Newt tool
  - [https://mynewt.apache.org/](https://mynewt.apache.org/)
  - Need to get familiar with Newt configuration and build tool

• **Port**
  - BSP support
  - MCU/HAL support
    - UART, GPIO

• **Next steps**
  - Create application with networking stack
  - Upstream the MCU and BSP port
Huawei LiteOS

• **Features**
  – Lightweight RTOS for IoT
  – Multiple toolchain support
  – Well defined project structure and HAL
  – Test suite for kernel features
  – Wireless networks, IoT gateway, home gateway
  – Ocean Connect IoT Platform(PaaS)
  – [www.huawei.com/liteos](http://www.huawei.com/liteos)
  – Less Documentation/porting guide for new architectures
  – Need for run-time statistics and task trace/debug support.

• **Port**
  – LiteOS Kernel port to RISC-V
  – BSP and HAL support
    – UART, GPIO, WiFi (esp8266)

• **Next Steps**
  – Port already pulled into LiteOS GitHub
  – Collaborate With Huawei
LiteOS Port

• **Kernel Port for a new ISA**
  – Context frame - TSK_CONTEXT_S
  – Context switching - los_dispatch_gcc.S
  – Interrupt handling functions
    – Interrupt lock, unlock, enable, disable functions
  – Memory model
  – Startup code

• **BSP and HAL port**
  – Board specific code e.g. memory, clock
  – Port device drivers in the HAL framework

```
Software_IRQHandler:
  li    t0, 0x08
  csrrc zero, mstatus, t0
  la    t0, g_pfnTaskSwitchHook
  lw    t1, 0x0(t0)
  beqz t1, TaskSwitch

TaskSwitch:
  la    t0, g_stLosTask
  lw    t1, 0(t0)
  sw    a2, 0(t1)

//Clear the task running bit of pstRunTask.
  la    t0, g_stLosTask
  lw    t1, (t0)
  lb    t2, 0x4(t1)
  andi t2, t2, OS_TASK_STATUS_NOT_RUNNING
  sb    t2, 0x4(t1)

//copy pstNewTask into pstRunTask
  la    t0, g_stLosTask
  lw    t1, 0x4(t0)
  sw    t1, 0x0(t0)

//set the task running bit=1
  lh    t2, 0x4(t1)
  ori  t2, t2, OS_TASK_STATUS_RUNNING
  sh    t2, 0x4(t1)

//retrive stack pointer
  lw    sp, (t1)
```
RISC-V experience

- **RISC-V ISA**
  - Easy for assembly programming
    - Small ISA specification document.
    - Fewer addressing modes
    - zero register (x0)
  - RV32I context frame – 31 registers
    - RV32E extension (15 registers) is desired for deeply embedded applications
  - Absence of multiple register load/store instruction
  - Software interrupt for implementing context switch

- **Privilege spec implementation**
  - Soft reset – may be required in some cases such as reset after remote firmware upgrade.
Mi-V Ecosystem

- A comprehensive offering to ease the adoption of RISC-V

Mi-V is the Microsemi RISC-V Ecosystem
Mi-V Soft Processors on Microsemi FPGA

- **Mi-V Soft Processors**
  - MiV_RV32IMA_L1_AHB
  - MiV_RV32IMAF_L1_AHB

- **Features**
  - Based on Rocket Chip
  - Provides a single hardware thread (hart)
  - Machine-mode privileged architecture
  - RV32I Base ISA with ‘M’, ‘A’ and ‘F’ extensions
  - Integrated 8Kbytes instructions cache and 8Kbytes data cache
  - Two external AHB interfaces for IO and memory
  - Support up to 31 programmable interrupts
  - Debug unit with a JTAG interface
  - Best suited for low-mid range microcontroller applications
SoftConsole

- Eclipse based IDE for software development
  - Supported platforms
    - Windows
    - Linux
  - Supported CPUs
    - Mi-V
    - ARM Cortex-M (M1, M3)
  - Supported JTAG programmers
    - FlashPro5
    - Olimex
    - Any programmer supported by openocd
  - Free Download
    - https://www.microsemi.com/products/fpga-soc/design-resources/design-software/softconsole
Mi-V RTOS Support and Solutions

- Ready to run example projects

- SoftConsole example projects should run on any (ideally) RV32I platform.
  - Verified on Mi-V and HiFive1

- Plans to enhance and maintain the RTOS ports

- FreeRTOS
  - [https://github.com/RISCV-on-Microsemi-FPGA/FreeRTOS](https://github.com/RISCV-on-Microsemi-FPGA/FreeRTOS)

- Mynewt
  - [https://github.com/RISCV-on-Microsemi-FPGA/MyNewt](https://github.com/RISCV-on-Microsemi-FPGA/MyNewt)

- LiteOS
  - [https://github.com/RISCV-on-Microsemi-FPGA/LiteOS](https://github.com/RISCV-on-Microsemi-FPGA/LiteOS)
Microsemi Corporation (Nasdaq: MSCC) offers a comprehensive portfolio of semiconductor and system solutions for aerospace & defense, communications, data center and industrial markets. Products include high-performance and radiation-hardened analog mixed-signal integrated circuits, FPGAs, SoCs and ASICs; power management products; timing and synchronization devices and precise time solutions, setting the world’s standard for time; voice processing devices; RF solutions; discrete components; enterprise storage and communication solutions, security technologies and scalable anti-tamper products, Ethernet solutions; Power-over-Ethernet ICs and midspans; as well as custom design capabilities and services. Microsemi is headquartered in Aliso Viejo, Calif., and has approximately 4,800 employees globally. Learn more at www.microsemi.com

Microsemi makes no warranty, representation, or guarantee regarding the information contained herein or the suitability of its products and services for any particular purpose, nor does Microsemi assume any liability whatsoever arising out of the application or use of any product or circuit. The products sold hereunder and any other products sold by Microsemi have been subject to limited testing and should not be used in conjunction with mission-critical equipment or applications. Any performance specifications are believed to be reliable but are not verified, and Buyer must conduct and complete all performance and other testing of the products, alone and together with, or installed in, any end-products. Buyer shall not rely on any data and performance specifications or parameters provided by Microsemi. It is the Buyer’s responsibility to independently determine suitability of any products and to test and verify the same. The information provided by Microsemi hereunder is provided “as is, where is” and with all faults, and the entire risk associated with such information is entirely with the Buyer. Microsemi does not grant, explicitly or implicitly, to any party any patent rights, licenses, or any other IP rights, whether with regard to such information itself or anything described by such information. Information provided in this document is proprietary to Microsemi, and Microsemi reserves the right to make any changes to the information in this document or to any products and services at any time without notice.