FreeRTOS

...What’s new in the FreeRTOS project

Richard Barry
Founder, FreeRTOS Project
Principal Engineer, AWS IoT
Agenda

The FreeRTOS Kernel
Amazon FreeRTOS
New Ecosystem Projects
New Architecture Ports
FreeRTOS—Open source real time kernel

is everywhere...
FreeRTOS downloads per month over 15 years
New Ecosystem Products
SEGGER Driven Remote Development

Development Board

SWD/JTAG

J-Link Remote Server (Tunnel Mode)

Amazon Workspaces Client

AWS Cloud

EC2 instance

SEGGER Tunnel Server

SEGGER Embedded Studio

SEGGER J-Link Library
New Architecture Ports
FreeRTOS Kernel V10.2.0
ARMv8-M (Cortex-M33)

Instructions to Build and Run:
- The Keil multi-project workspace FreeRTOSDemo.uvmpw contains projects for both the secure project, and non secure project.
- Set the FreeRTOSDemo_s project as Active - Right click on "Project: FreeRTOSDemo_s" and select "Set as Active Project".
- Build the FreeRTOSDemo_s project using "Project --> Build" or by pressing "F7".
- Set the FreeRTOSDemo_ns project as Active - Right click on "Project: FreeRTOSDemo_ns" and select "Set as Active Project".
- Build the FreeRTOSDemo_ns project using "Project --> Build" or by pressing "F7".
- Start Debug Session using "Debug --> Start/Stop Debug Session" or by pressing "Ctrl+F5".

int main( void )

// Boot the non-secure code. */
BootNonSecure( mainNONSECURE_APP_START_ADDRESS );

/* Non-secure software does not return. This code is not executed. */
RISC-V
RISC-V Ecosystem Growth
RISC-V Summit Dec 2018

- over 1000 registered attendees
- ~ 2X the attendance of the Milpitas Workshop in Dec 2017
- ~ 250 abstracts submitted
- 59 sessions including keynotes, tutorials and 3 tracks
- 29 exhibitors
RISC-V Foundation: 200+ Members
So what’s all the fuss about?

It's just an ISA right?

How did we get here?
RISC-V Background

- Started as a research project at UC Berkeley in 2010 and resulted in the release of the base user spec in May, 2014
  - many tapeouts and several research publications along the way
- The name RISC-V (pronounced risk-five), was chosen to represent the fifth major RISC ISA design effort at UC Berkeley
  - RISC-I, RISC-II, SOAR, and SPUR were the first four projects with the original RISC-I publications dating back to 1981
- In August 2015, articles of incorporation were filed to create a non-profit RISC-V Foundation to govern the ISA
Most CPU chips are SoCs with many ISAs

- Applications processor
- Graphics processors
- Image processors
- Radio DSPs
- Audio DSPs
- Security processors
- Power-management processor
- ....

- Apps processor ISA too large for base accelerator ISA
- IP bought from different places, each proprietary ISA
- Home-grown ISA cores
- Over a dozen ISAs on some SoCs – each with unique software stack
Why so Many ISAs?

Do we need all these different ISAs?

Must they be proprietary?

*What if there was one free and open ISA everyone could use for everything?*
What’s Different about RISC-V?

- **Simple**
  - Far smaller than other commercial ISAs
  
- **Clean-slate design**
  - Clear separation between user and privileged ISA
  - Avoids µarchitecture or technology-dependent features

- **A modular ISA**
  - Small standard base ISA
  - Multiple standard extensions

- **Designed for extensibility/specialization**
  - Variable-length instruction encoding
  - Vast opcode space available for instruction-set extensions

- **Stable**
  - Base and standard extensions are frozen
  - Additions via optional extensions, not new versions
Summary

- The free and open RISC-V ISA is enabling a new innovation frontier across all computing devices
- Strong Industry Support
  - ~200+ members; Broad commercial and academic interest
- RISC-V Twitter [http://twitter.com/risc_v @risc_v](http://twitter.com/risc_v)
- RISC-V LinkedIn Page
  - [http://www.linkedin.com/company/risc-v-foundation](http://www.linkedin.com/company/risc-v-foundation)
- RISC-V mail lists / groups
  - [https://riscv.org/mailing-lists/](https://riscv.org/mailing-lists/)
FreeRTOS on RISC-V
RISC-V examples in FreeRTOS V10.2.0

Pre-configured examples:
• QEMU sifive_e model using Freedom Studio
• Microsemi/Future Electronics M2GL025-Creative-Board using SoftConsole
• VEGAboard using vanilla Eclipse (https://www.open-isa.org)

FreeRTOS RISC-V specifics:
• Including the source files
• Setting the assembler’s include file
• Set configCLINT_BASEADDRESS
• #define portasmHANDLE_INTERRUPT
• Install the FreeRTOS trap handler
Thank You!

www.FreeRTOS.org
aws.amazon.com/freertos

Download, share and support
opensource.amazon.com/enterprise-oss-book
@AWSOpen | opensource.amazon.com | aws.github.io