JerryScript on RIOT
An ultra-lightweight JavaScript engine for the Internet of Things

Tilmann Scheller
Principal Compiler Engineer
t.scheller@samsung.com
Samsung Open Source Group
Samsung Research UK
RIOT Summit 2016
Berlin, Germany, July 15 – 16, 2016
Overview

- Introduction
- JerryScript
- Memory consumption/Performance
- JerryScript on RIOT
- Demo
- Future work
- Summary
Introduction
What is JerryScript?

- A really lightweight JavaScript engine
- Has a base footprint of ~3KB of RAM
- Optimized for microcontrollers
- Developed from scratch by Samsung
- Actively developed on GitHub
- JerryScript is an open source project released under the Apache License 2.0
Why JavaScript on microcontrollers?

- There's a huge pool of JavaScript developers
- Opens up the possibility for web developers to easily write software for embedded devices
- Performance overhead of JavaScript less of an issue for control tasks
- Increased productivity, shorter time to market
- Ability to load code dynamically over the network
- Security: Executing JavaScript code is safer than executing arbitrary native code
JerryScript
JerryScript History

- Development started in June 2014
- Released as open source in June 2015
- JerryScript passed 100% of the test262 conformance test suite in August 2015
- Rewritten compact byte code implementation landed in January 2016
- Current focus on performance optimization
JerryScript

- Heavily optimized for a low memory footprint
- Interpreter-only
- Compact object representation
- Compressed pointers
- No AST, directly creating byte code
- Compact byte code heavily optimized for low memory consumption
JerryScript Portability

- Extremely portable
- Self-contained
- Small C library
- Can run bare-metal
- Supports the STM32F4, ESP8266, Arduino 101, FRDM-K64F boards
- OS support: NuttX, Zephyr, mbed, RIOT
- Runs on Linux/macOS as well
Target hardware

- STM32F4 developer board
- Cortex-M4F clocked at 168 MHz
- 192KB of RAM
- 1MB of flash memory
JerryScript

- Written in C99
- About 82KLOC
- Code size 156KB when compiled with GCC in LTO mode for ARM Thumb-2
- Implements the entire ECMAScript 5.1 standard, passes 100% of the test262 conformance test suite
- C API for embedding JerryScript
- Byte code snapshot feature
Memory consumption/Performance
SunSpider 1.0.2 - Memory consumption

Measured on a Raspberry Pi 2
SunSpider 1.0.2 - Performance

Measured on a Raspberry Pi 2 (-Os -flto)
JerryScript on RIOT
JerryScript on RIOT

- Ported JerryScript over to RIOT
- Without any prior knowledge of RIOT the port took only a couple of days to complete
- Running RIOT on the STM32F4 board
- Ran into issue #4488 which causes crashes when writing to the serial console
- RIOT now officially supported as a target OS in upstream JerryScript
Demo
Tetris Demo

• Implementation of the classic Tetris game
• An LED matrix is used as a display
• Written in JavaScript
Tetris Demo
Future work
Future work

• Further performance and memory optimizations
• 1.0 release of JerryScript in the coming weeks
• Debugging support
• Memory profiling
• Support for more boards
Summary
Summary

- Significantly lowers barrier of entry for JavaScript development targeting heavily constrained embedded devices
- Speeds up development
- Active community
- More information on http://samsung.github.io/jerryscript
- Looking for bug reports and feedback
Thank you.
Contact Information:

Tilmann Scheller

t.scheller@samsung.com

Samsung Open Source Group
Samsung Research UK