Introduction to IoT.js

Tilmann Scheller
Senior LLVM Compiler Engineer
t.scheller@samsung.com

Samsung Open Source Group
Samsung Research UK

FOSDEM 2016
Brussels, Belgium, January 30 – 31, 2016

Overview

- Introduction

- Demo

- Questions

Introduction

What is IoT.js?

- A lightweight version of Node.js

- Attempt to bring the success of Node.js to the embedded world

- Retains backwards compatibility with Node.js as much as possible

- Mostly written in JavaScript

- Runs on top of JerryScript

What is JerryScript?

- An extremely lightweight JavaScript engine

- Written from scratch by Samsung

- Designed to run on heavily resource-constrained microcontrollers

- Has a base RAM footprint of 10KB

- Not just a JavaScript subset: Implements the full ECMAScript 5.1 standard

- Written in C

- Binary size is around 200KB (compiled for ARM Thumb-2)
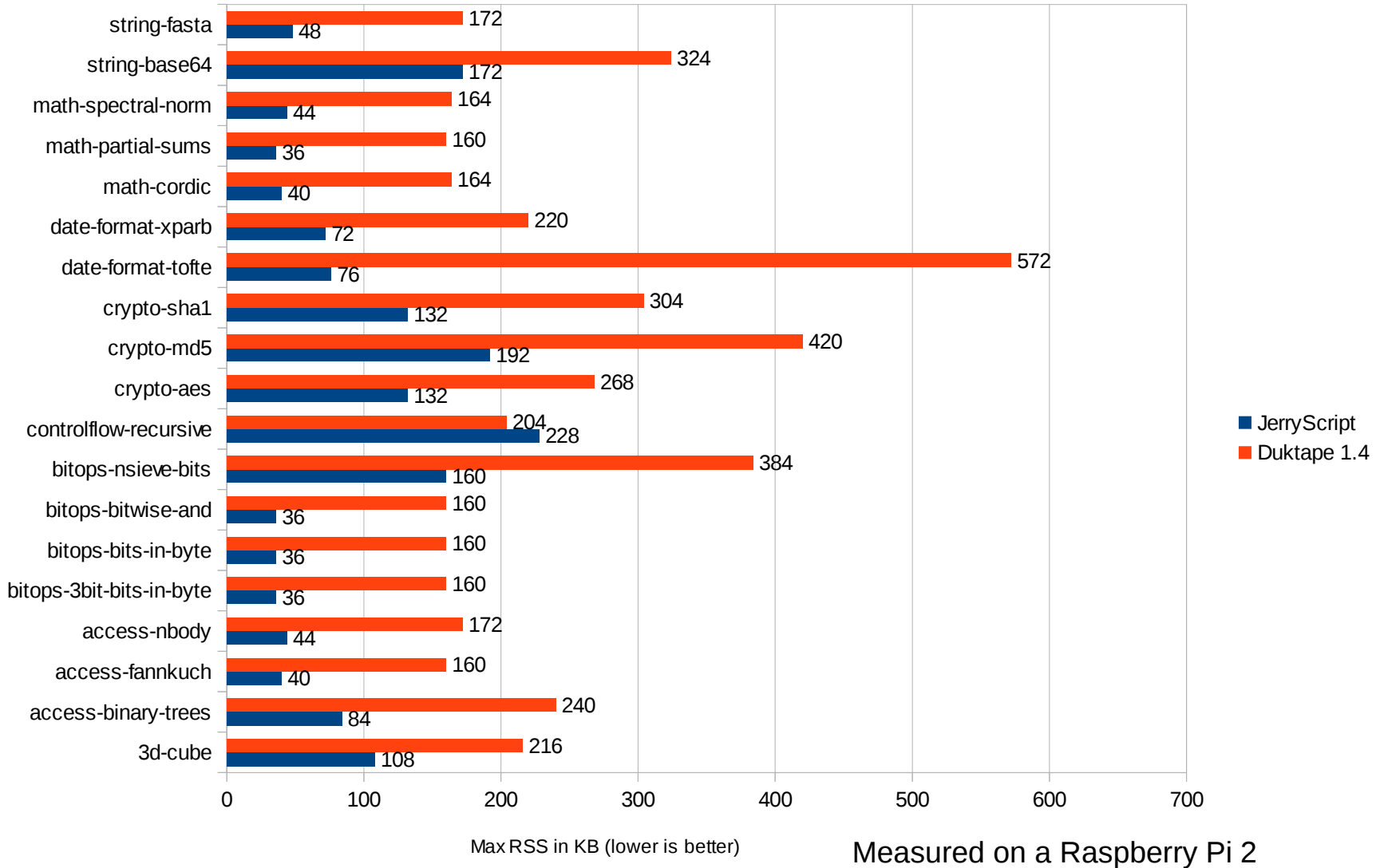
Why JavaScript on microcontrollers?

- 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

Open source

- Actively developed on GitHub

- JerryScript and IoT.js are both open source released under the Apache 2.0 license

- Feature-complete, supports the full ECMAScript 5.1 standard

- Looking for bug reports and feedback

- More information on http://www.jerryscript.net and http://www.iotjs.net

SunSpider 1.0.2 - Memory consumption

Max RSS in KB (lower is better)
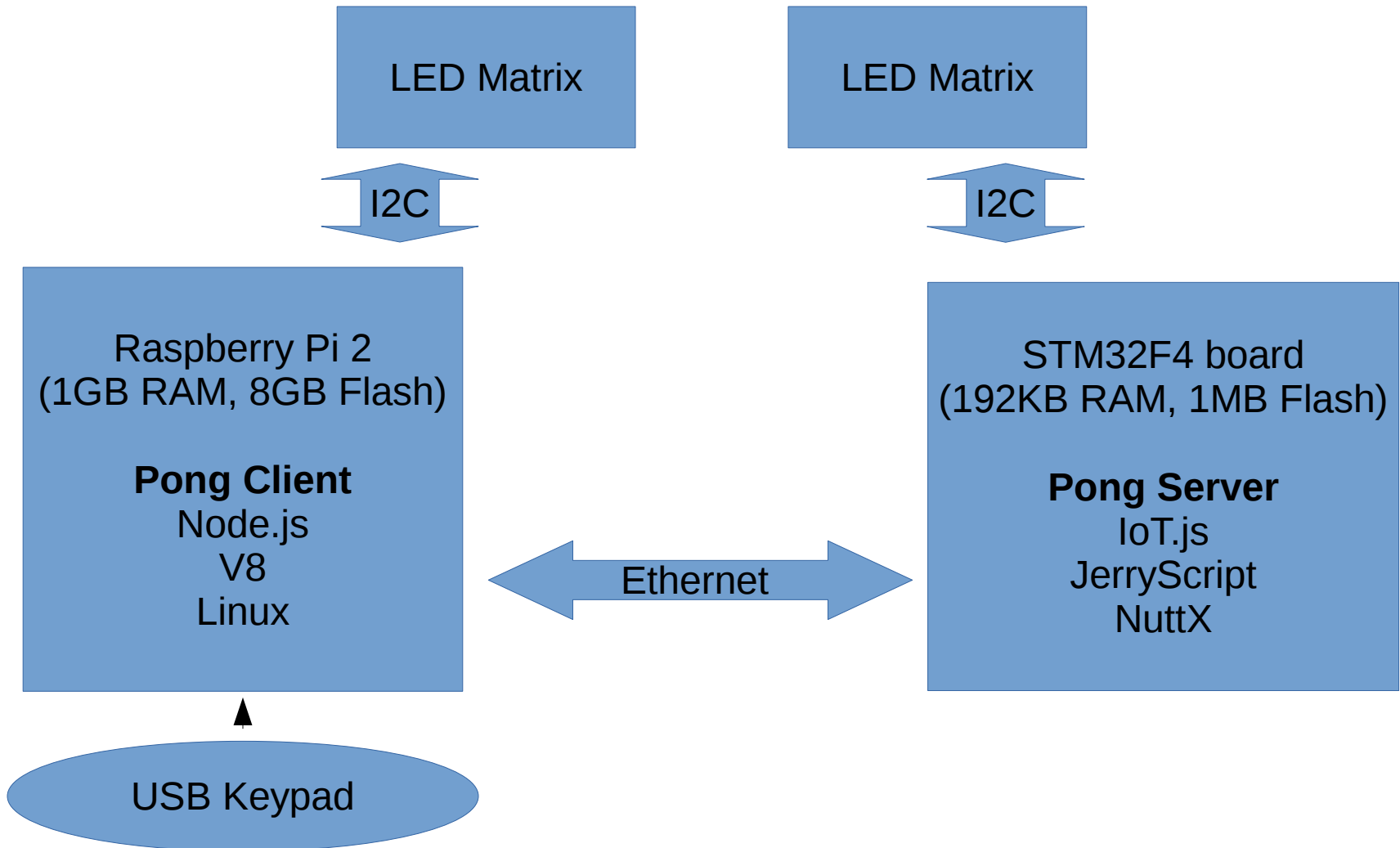
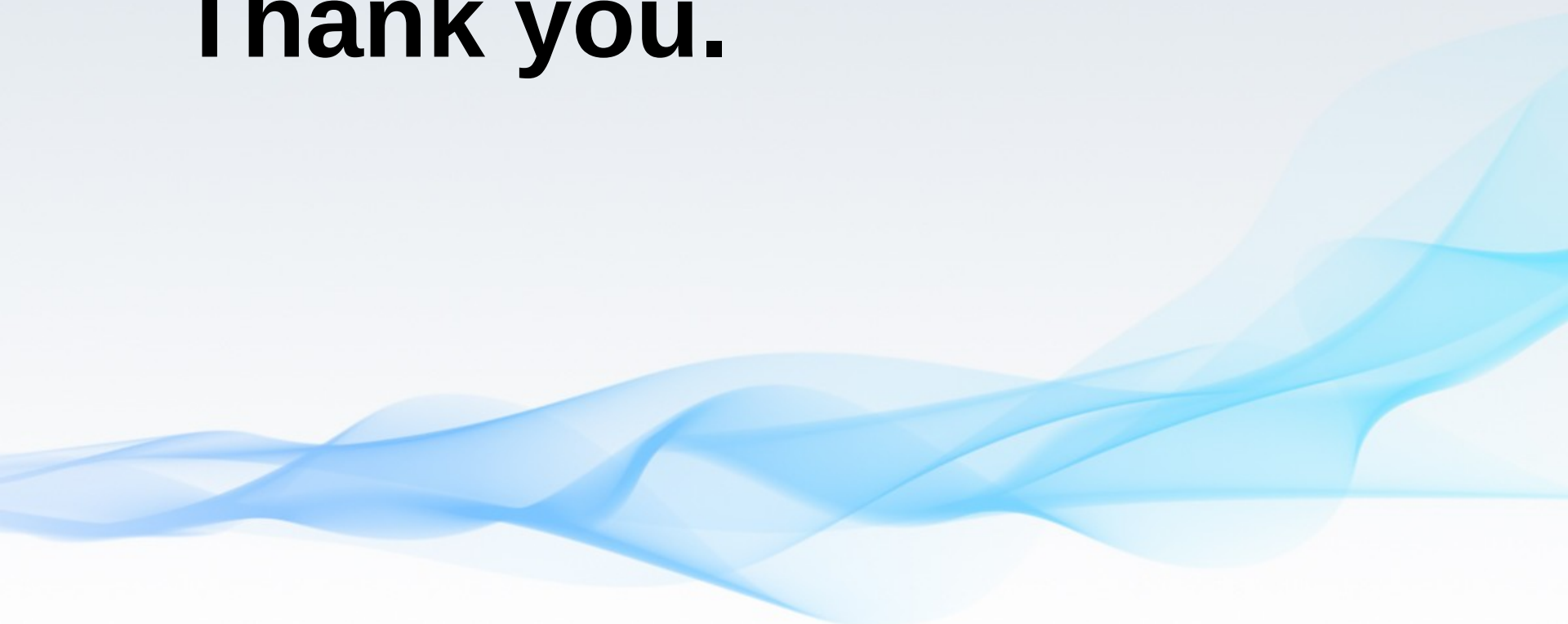Measured on a Raspberry Pi 2

Demo

- Implementation of the classic Pong game

- Display shared across two devices

- Each device drives one LED matrix

- Implemented as a Node.js module

- "AI" oppenent running on the microcontroller

Demo

Thank you.

Contact Information:

Tilmann Scheller
t.scheller@samsung.com

Samsung Open Source Group
Samsung Research UK