IoT.js
A JavaScript platform for the Internet of Things

Ziran Sun
ziran.sun@samsung.com

Samsung Open Source Group
Samsung Research, UK

FOSDEM 2018
Overview

• IoT.js Blinkt! light demo

• Demo walk-through
  ▪ IoT.js
  ▪ Blinkt! module

• Questions
IoT.js Blinkt! Light Demo
IoT.js Blinkt! Light Demo

Blinkt Light

GPIO

Raspberry Pi Zero W
(512k RAM, 8GB flash)

Blinkt Module
IoT.js
JerryScript
Linux

Weather data

https.request

OpenWeatherMapAPI
from
Openweathermap.org
IoT.js Blinkt! Light Demo

Demo implemented

**Fully in JavaScript**
IoT.js
IoT.js

- IoT.js aims to provide inter-operable service platform in the world of IoT.
- Based on web technology and run on top of JerryScript
- Designed to bring the success of Node.js to constrained IoT devices.
- Runs ECMAScript application which can access device resource (network, File system)
IoT.js architecture

- ECMAScript Application
- IoT.js
  - file system
  - network
  - device
  - event
  - etc ...
- IoT.js ECMAScript API
- IoT.js C core
  - ECMAScript binding
  - I/O Event binding
  - IoT binding
- JerryScript
- libtuv
- IoTivity
- OS (RTOS, Linux)
JerryScript

• An ultra lightweight JavaScript engine
• Originally developed from scratch by Samsung
• Heavily optimized for low memory footprint
• Self-contained and extremely portable
JerryScript

- Transferred to JS Foundation in 2016
- Lightweight, modular, ECMAScript
- Features added:
  - Introduced a debugger
  - ES6 related features: promise, TypedArray
  - More hardware support: e.g. ARM m-bed
IoT.js

• Released 1.0 (July 2017)
  – Asynchronous, event driven I/O (libtuv)
  – Module system: native (C) and JavaScript modules
  – Node.js friendly architecture
  – Supports a subset of core Node.js modules, e.g. fs, events, http, net and stream etc
  – Provide a list of I/O control hardware modules
Blinkt! Light Module
Blinkt! Light Module

• A JavaScript library to Interact with Blinkt! LEDs on RPi

• Use GPIO interfaces in IoT.js

```javascript
var gpio = require('gpio');
Blinkt.prototype.setup = function setup() {
  gpio_dat = gpio.open({
    pin: DAT,
    direction: gpio.DIRECTION.OUT
  }, function(err) {
  
  });
}

function writeByte(byte) {
  var bit;
  ...
  gpio_dat.write(bit);
  ...
}
```
Integrate Blinkt! Module in IoT.js

• Add module in `iotjs/src/modules.json`

```
"blinkt": {
  "js_file": "js/blinkt.js",
  "require": ["gpio"]
}
```

• Enable module in Build

```
-- ENABLE_MODULE_BLINKT
```
Thank you.