

RISE OF THE NODEBOTS

Presented by Joel Lord
iJS '17 Munich, DE
October 26th 2017

ABOUT ME, EH?

SPIRIA



AGENDA

Demo

Hardware

Software

Setting up the Arduino board

Setting up the project

Building a Hello World!

Building the Night light

I LIKE ROBOTS



I LIKE ROBOTS

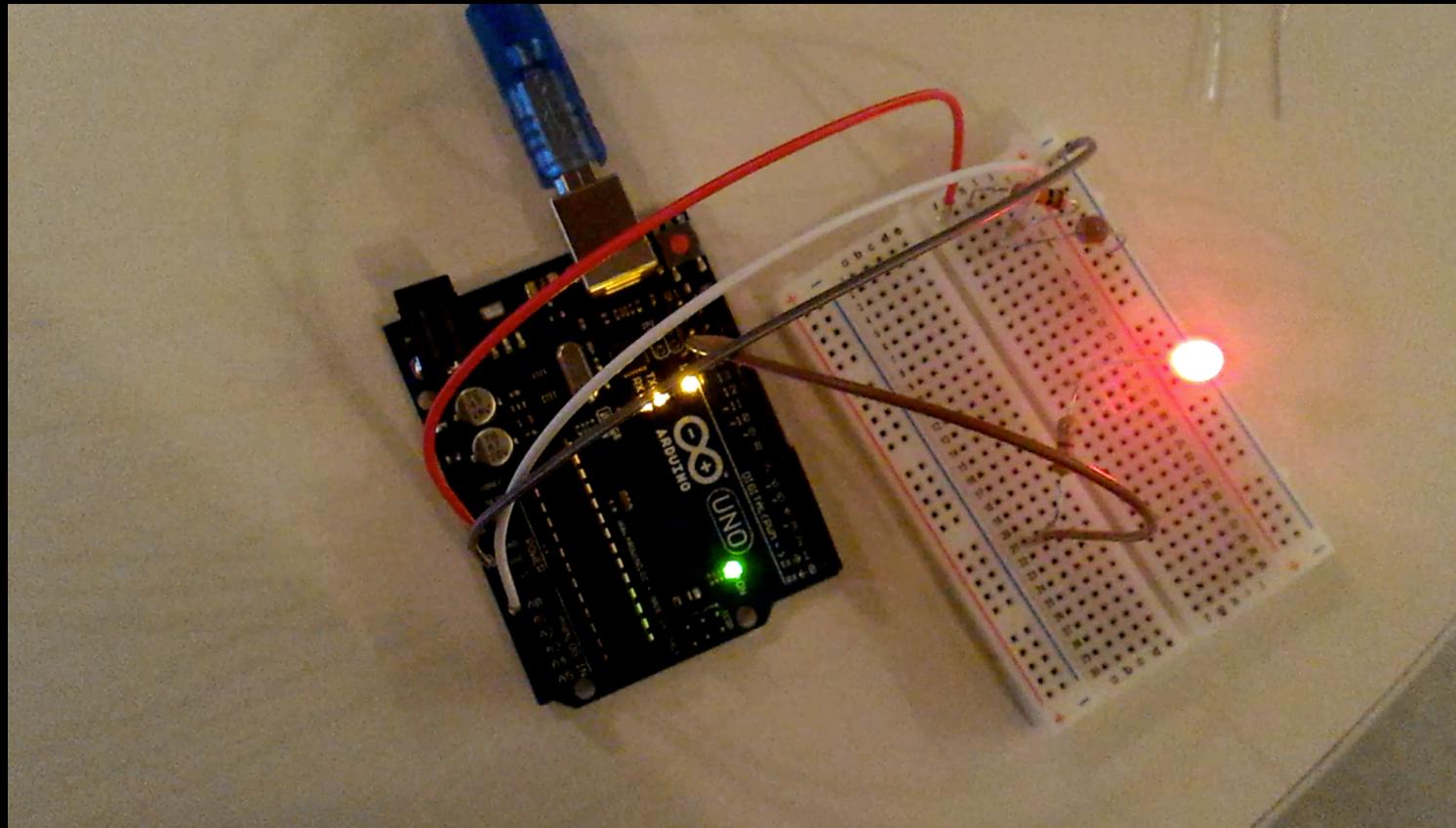
Travel Tip !



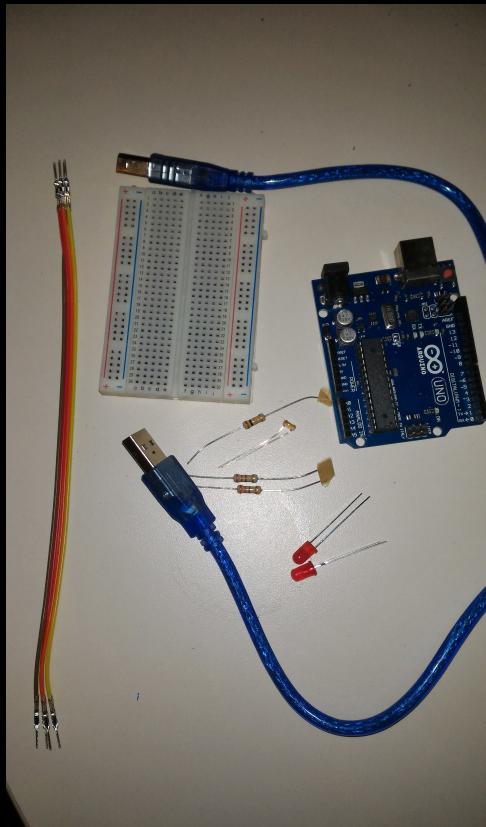
I LIKE ROBOTS



OUR FINAL PRODUCT



YOUR KIT



1 Arduino board

1 USB Cable

1 Breadboard

4 Jumper Cables

2 Red LEDs

2 220 ohms Resistors

1 Photoresistor

1 1k ohms Resistor

HARDWARE - ARDUINO



Open Source hardware and software

Uses an ATmega programmable chip

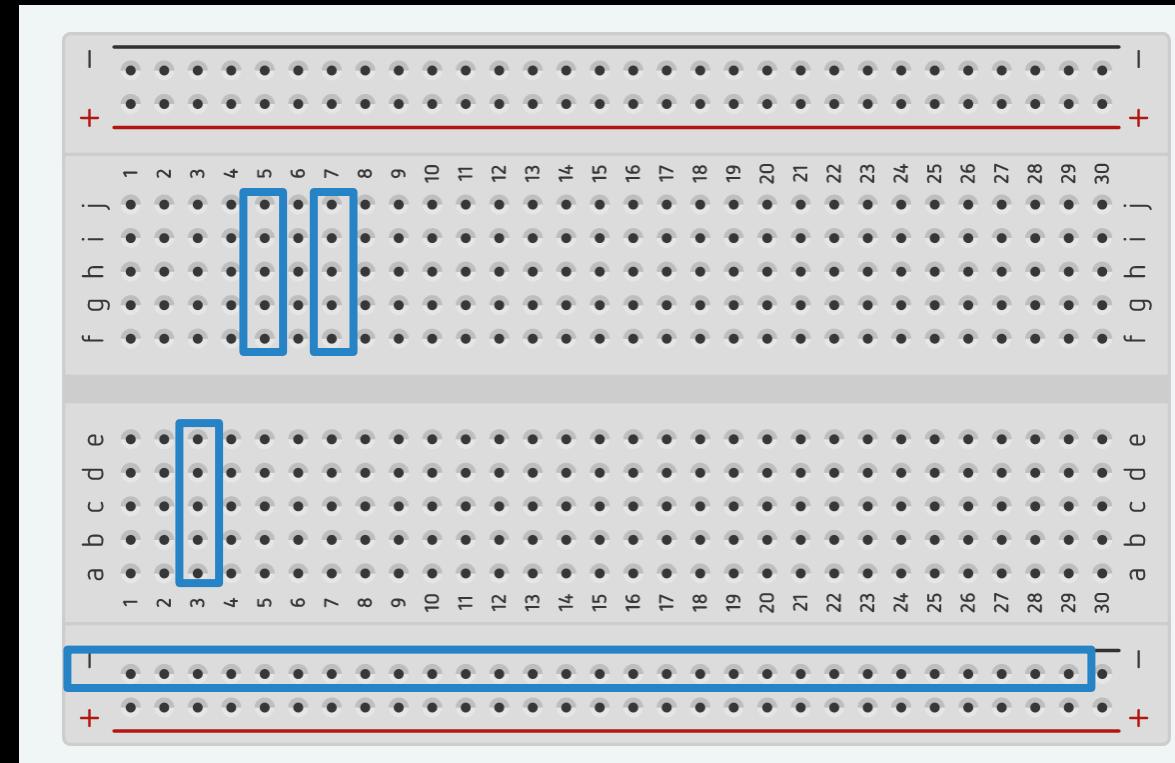
Programmable in C

We can use the serial port to send
instructions using the Standard
Firmata

(<https://github.com/firmata/arduino>)

HARDWARE

HARDWARE - BREADBOARDS



HARDWARE - RESISTORS

$$R = \frac{V}{I}$$

$$R = \frac{5v}{0.02A}$$

$$R = 250 \frac{v}{A}$$

$$R = 250\Omega$$

SOFTWARE

SOFTWARE - NODE



JavaScript runtime built on Chrome's V8 JavaScript engine.

Open Source and available freely

Available at <https://nodejs.org/>

SOFTWARE - NPM

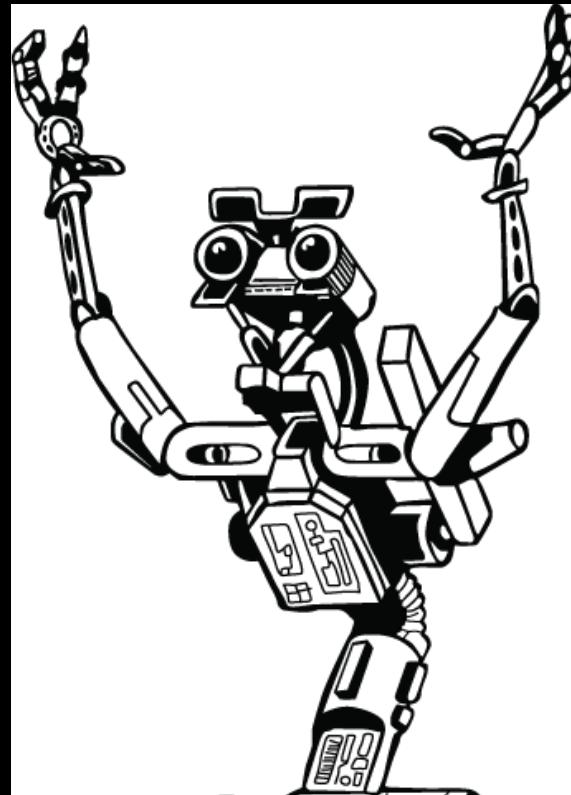


Node Package Manager

Comes out of the box with NodeJs

Available at <https://www.npmjs.com/>

SOFTWARE – JOHNNY FIVE



Available through npm

Documentation at <http://johnny-five.io/>

PROJECT SETUP

SETTING UP THE ARDUINO

To enable communication with the serial port, you need the Standard Firmata Plus installed

<https://github.com/firmata/arduino>

All the boards today come pre-configured

Still, here is how to install it

SETUP YOUR PROJECT

Create a new project using

- `npm init`

Install Johnny-Five using

- `npm install --save johnny-five`

SETUP YOUR PROJECT – NPM INIT

```
> npm init

name: (test)
version: (1.0.0)
description:
entry point: (index.js)
test command:
git repository:
keywords:
author:
license: (ISC)
About to write to /Users/jlord/Documents/Projects/test/package.json:
```

```
{
  "name": "test",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \\\"Error: no test specified\\\" && exit 1"
  },
  "author": "",
  "license": "ISC"
}
```

Is this ok? (yes)

SETUP YOUR PROJECT – INSTALL JOHNNY-FIVE

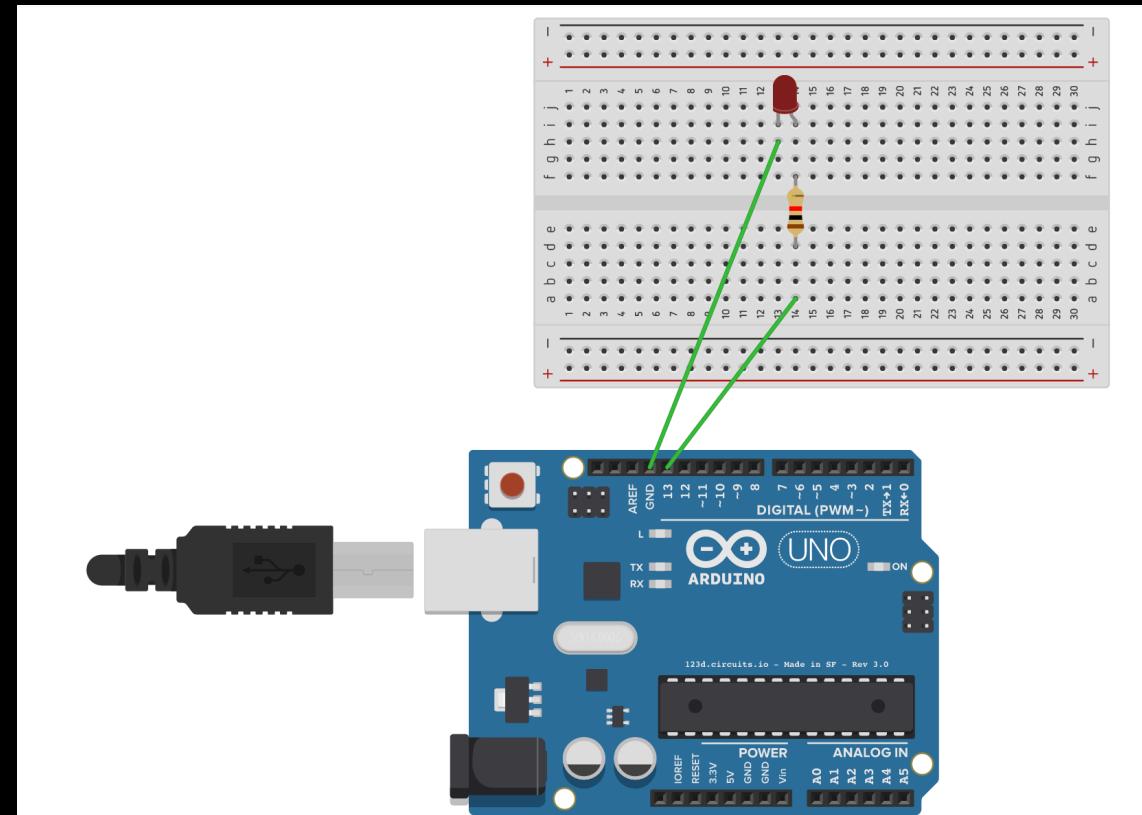
```
> npm install --save johnny-five

npm WARN package.json test@1.0.0 No description
npm WARN package.json test@1.0.0 No repository field.
npm WARN package.json test@1.0.0 No README data
 \
> serialport@4.0.7 install /Users/jlord/Documents/Projects/test/node_modules/johnny-five/node_modules/serialport
> node-pre-gyp install --fallback-to-build

[serialport] Success: "/Users/jlord/Documents/Projects/test/node_modules/johnny-
five/node_modules/serialport/build/Release/serialport.node" is installed via remote
johnny-five@0.10.6 node_modules/johnny-five
|   ├── lodash.debounce@4.0.8
|   ├── lodash.clonedeep@4.5.0
|   ├── ease-component@1.0.0
|   ├── color-convert@1.2.2
|   ├── browser-serialport@2.1.0
|   ├── temporal@0.5.0
|   ├── nanotimer@0.3.10
|   ├── firmata@0.15.0
|   ├── chalk@1.1.3 (escape-string-regexp@1.0.5, ansi-styles@2.2.1, supports-color@2.0.0, has-ansi@2.0.0, strip-ansi@3.0.1)
|   ├── es6-shim@0.35.2
└── serialport@4.0.7 (bindings@1.2.1, commander@2.9.0, lie@3.1.0, debug@2.6.0, nan@2.5.0, object.assign@4.0.4)
```

HELLO WORLD

WIRING UP A LED



CODING OUR LED

```
var j5 = require("johnny-five");

var board = new j5.Board();

board.on("ready", function() {
  var led = new j5.Led(13);
  led.blink(500);
}) ;
```

CODING OUR LED

```
var j5 = require("johnny-five");

var board = new j5.Board();

board.on("ready", function() {
  var led = new j5.Led(13);
  led.blink(500);
}) ;
```

CODING OUR LED

```
var j5 = require("johnny-five");

var board = new j5.Board();

board.on("ready", function() {
  var led = new j5.Led(13);
  led.blink(500);
}) ;
```

CODING OUR LED

```
var j5 = require("johnny-five");

var board = new j5.Board();

board.on("ready", function() {
  var led = new j5.Led(13);
  led.blink(500);
});
```

CODING OUR LED

```
var j5 = require("johnny-five");

var board = new j5.Board();

board.on("ready", function() {
  var led = new j5.Led(13);
  led.blink(500);
});
```

CODING OUR LED

```
var j5 = require("johnny-five");

var board = new j5.Board();

board.on("ready", function() {
  var led = new j5.Led(13);
  led.blink(500);
}) ;
```

Issues on your Mac ? http://www.wch.cn/download/CH341SER_MAC_ZIP.html

SHOW ME SOME BLINK

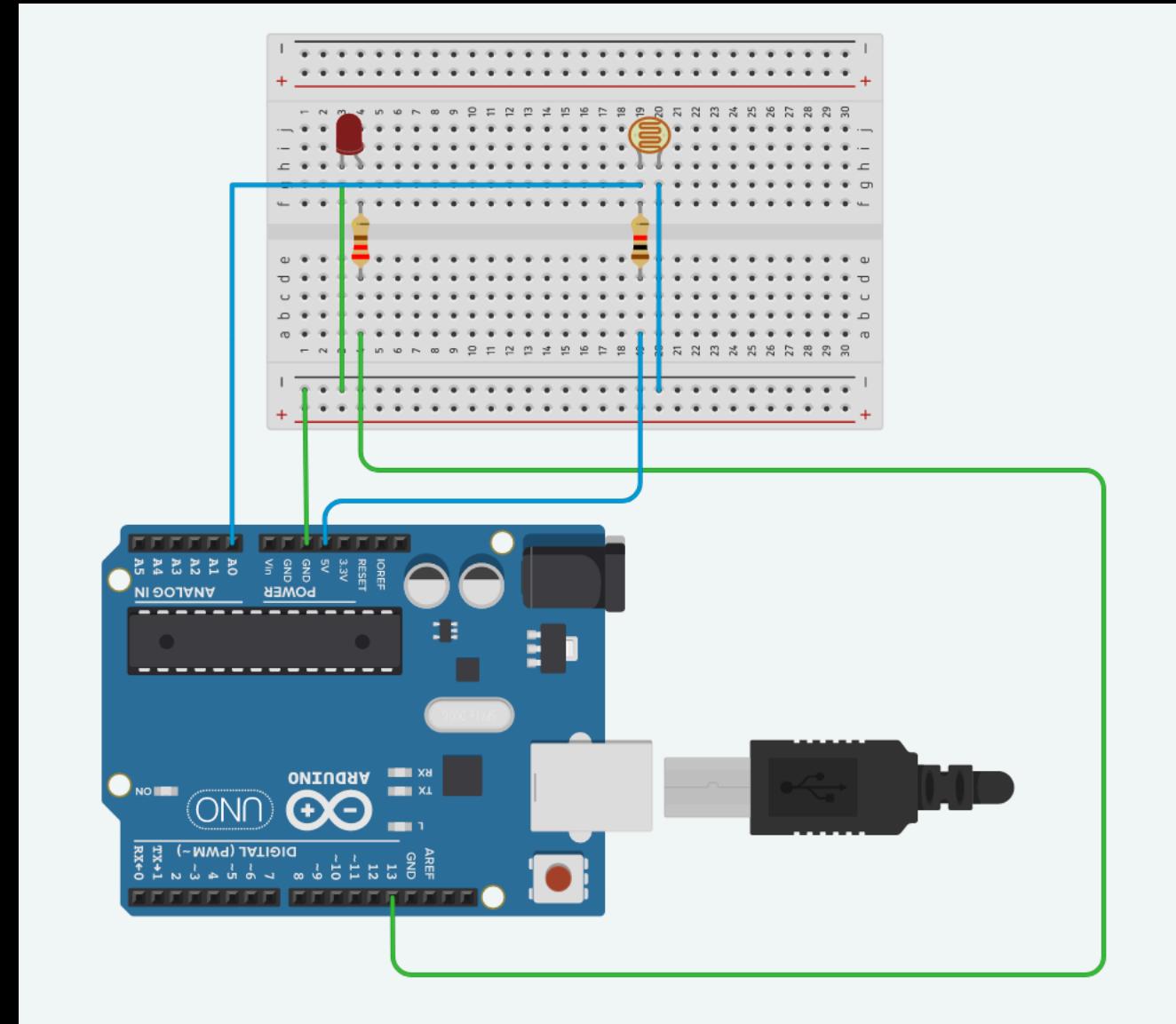
```
➤ node hello.js
```

```
1484955651472 Device(s) /dev/cu.usbmodem1421
1484955651484 Connected /dev/cu.usbmodem1421
1484955653114 Repl Initialized
```

```
➤
```

BUILD A NIGHT LIGHT

WIRING



CODING

```
var j5 = require("johnny-five");
var board = new j5.Board();
board.on("ready", function() {
  var led = new j5.Led(13);
  var sensor = new j5.Sensor("A0");
  sensor.on("change", function() {
    if (sensor.value > 750) {
      led.on();
    } else {
      led.off();
    }
  });
});
```

CODING

```
var j5 = require("johnny-five");
var board = new j5.Board();
board.on("ready", function() {
  var led = new j5.Led(13);
  var sensor = new j5.Sensor("A0");
  sensor.on("change", function() {
    if (sensor.value > 750) {
      led.on();
    } else {
      led.off();
    }
  });
});
```

CODING

```
var j5 = require("johnny-five");
var board = new j5.Board();
board.on("ready", function() {
  var led = new j5.Led(13);
  var sensor = new j5.Sensor("A0");
  sensor.on("change", function() {
    if (sensor.value > 750) {
      led.on();
    } else {
      led.off();
    }
  });
});
```

CODING

```
var j5 = require("johnny-five");
var board = new j5.Board();
board.on("ready", function() {
  var led = new j5.Led(13);
  var sensor = new j5.Sensor("A0");
  sensor.on("change", function() {
    if (sensor.value > 750) {
      led.on();
    } else {
      led.off();
    }
  });
});
```

CODING

```
var j5 = require("johnny-five");
var board = new j5.Board();
board.on("ready", function() {
  var led = new j5.Led(13);
  var sensor = new j5.Sensor("A0");
  sensor.on("change", function() {
    if (sensor.value > 750) {
      led.on();
    } else {
      led.off();
    }
  });
});
```

CODING

```
var j5 = require("johnny-five");
var board = new j5.Board();
board.on("ready", function() {
  var led = new j5.Led(13);
  var sensor = new j5.Sensor("A0");
  sensor.on("change", function() {
    if (sensor.value > 750) {
      led.on();
    } else {
      led.off();
    }
  });
});
```

CODING

```
var j5 = require("johnny-five");
var board = new j5.Board();
board.on("ready", function() {
  var led = new j5.Led(13);
  var sensor = new j5.Sensor("A0");
  sensor.on("change", function() {
    if (sensor.value > 750) {
      led.on();
    } else {
      led.off();
    }
  });
});
```

CODING

```
var j5 = require("johnny-five");
var board = new j5.Board();
board.on("ready", function() {
  var led = new j5.Led(13);
  var sensor = new j5.Sensor("A0");
  sensor.on("change", function() {
    if (sensor.value > 750) {
      led.on();
    } else {
      led.off();
    }
  });
});
});
```

CODING

```
var j5 = require("johnny-five");
var board = new j5.Board();
board.on("ready", function() {
  var led = new j5.Led(13);
  var sensor = new j5.Sensor("A0");
  sensor.on("change", function() {
    if (sensor.value > 750) {
      led.on();
    } else {
      led.off();
    }
  });
});
```

CODING

```
➤ node index.js
```

```
1484955651472 Device(s) /dev/cu.usbmodem1421
1484955651484 Connected /dev/cu.usbmodem1421
1484955653114 Repl Initialized
```

```
➤
```

THE END – THANK YOU !

SPIRIA

Presented by

Joel Lord

October 25th 2017

iJS'17 Munich, DE

Twitter: @joel__lord

Github: joellord