



Node-js (for Starters) and Electronics/Robotics

Presented by
Onuralp SEZER
Fedora Ambassador

Event Topics

1. Node-js ?

a.How to install ?

b.How to use ? (simple and quick way)

2. And Electronics,

3. And Robotics

4.IOT

- Security
- How to stay safe

5. Other tools (python,android,c,c++ etc.)

Event Topics

1. Node-js ?
 - a.How to install ?
 - b.How to use ? (simple and quick way)
2. And Electronics,
3. And Robotics

Node-JS

What is it ?

- It's a java-script run-time built on Chrome's V8 JavaScript Engine.
- It's lightweight plus efficient
- It has own package ecosystem call **npm** !
- "npm" depend on node-js
 - Without node It cannot work
- Install > `#su -c 'dnf install nodejs npm -y'`
- It's simple to use ! **I ❤️** DOCS! And run "user" mode If possible in "dev" mode.

Electronics/Robotics

even IOT

Relationship with NodeJS



- Connection Methods
 - SPI,I2C and Serial-port
 - Web Server and socket
 - Industrial Ways (Modbus,Bacnet(RTU,IP))
- Circuit and board types (Examples)
 - Arduino with Firmata or manual codes
 - Embedded Cards(Rs-pi,Beagleboard etc.)
 - Install node-js and communicate with them.
 - Microchip (PIC series with manual code)
 - OpenWRT devices

I2C and SPI

- Both connection depend on addresses list.
- Can be connect multiple device on same line
- Can be use as multiplexer(digital or analog)
- Each addresses need to be call and determine on libraries
- There is no general libraries for each I2C or SPI

Example device/code

- PCA9685 / I2C chip
- 16-Channel 12-bit
- PWM driver
- Possible to multiple on top of it.
 - Need soldering work!
 - Npm I use ;
 - <https://www.npmjs.com/package/pca9685>
 - Only works for that chip.

Some basic code...

```
var i2cBus = require("i2c-bus");
var Pca9685Driver = require("pca9685").Pca9685Driver;
var options = {
  i2c: i2cBus.openSync(1),
  address: 0x40,
  frequency: 50,
  debug: false};
pwm = new Pca9685Driver(options, function() {
  console.log("Initialization done");
});
// Set channel 0 to turn on on step 42 and off on step 255
pwm.setPulseRange(0, 42, 255);
// Set the pulse length to 1500 microseconds for channel 2
pwm.setPulseLength(2, 1500);
// Set the duty cycle to 25% for channel 8
pwm.setDutyCycle(8, 0.25);
```

Serialport

- It's general way to communicate devices
- All we need is know what's the addresses of device then we connect
- `$ npm install --save-dev serialport` OR
- `$ sudo npm install -g serialport` (install global)
- Also we can use command line for quick check.

Magical codes..



```
var serialport = require("serialport");
```

```
var SerialPort = serialport;
```

```
var serialPort = new  
SerialPort("/dev/ttySX", { baudrate:  
9600,
```

```
  parser: serialport.parsers.readline("\n")  
});
```

- PS: Make sure you have permissions ! :)
- Do not rely on "sudo" cuz USERS!

Web Socket/Server

- We can connect multiple device who can talk on Ethernet
- It's a general way we can use on website/device
- Internet is the way for cloud and automation infrastructures / IOT
- E.g : built-in "http" or external library like a express..
-

WebServer (http)



```
const http = require('http');
const hostname = '127.0.0.1';
const port = 2016;
const server = http.createServer((req, res) => {
  res.statusCode = 200;
  res.setHeader('Content-Type', 'text/plain');
  res.end('Hello World\n');
});
server.listen(port, hostname, () => {
  console.log('Server running at http://${hostname}:${port}/');
  console.log('Server need to run on Flock 2016 hopefully')
});
```

Web Server(express)

```
var express = require('express')
var app = express()

app.get('/', function (req, res) {
  res.send('Hello World')
})

app.listen(3000)
```

- Short
- Fast
- Well doc
- External

- Express still active project and has many use too.

•Industrial Ways...

- Modbus and Bacnet using on industrial automations(buildings,factories etc.)
- Bacnet and Modbus not new protocols but Bacnet is most recent and fast/easy to use it.
- Modbus is old but still using on more basics devices like;
 - Sensors,counters,converters etc.
- Bacnet protocol need more space(on small board need code effective)

Robotics (Depend on HW)

- Nodejs also meant to be on any robotics too.
- Node-Red (research needed quickly)
 - CylonJS >> Meant to be robotics
 - `npm install cylon` | `npm install -g cylon`
 - Johnny five > Arduino focus/Robotics/
 - GORT > CLI based control
 - Scan devices, Install firmware and use with others
 - Compatible with cylon and johnny-five, Gobot, Artoo etc.

Cylon-jS

- It based on Arduino/Rs-pi and other embedded boards.
- You can control over 43+ devices
- You can make your own control-algorithm
- It's simple to install and well documented.
- And we have small example to show up.

Cylon-Js addons

- Cylon-firmata (Arduino+Firmata needed)
 - Cylon-beaglebone (I2C and SPI)
 - Cylon-ble (BLE communication)
 - Cylon-opencv
 - Cylon-raspi
 - Cylon-keyboard
 - Cylon-joystick
- And more...

Resources

- <https://cylonjs.com/>
- <http://gort.io/>
- <http://johnny-five.io/>
- <http://socket.io/>
- <https://expressjs.com/>

Demos

Hardware List(Demos)

- EMG controller
- Leap motion controller (camera and IR)
- Rs-pi/PC
- PWM controller
- External Power Adapter
- 12 Servo Motor (0-180 high torque)
 - (Always get extra back-up If you broke the servos)

Questions?



Contact:
thunderbirdtr@fedoraproject.org