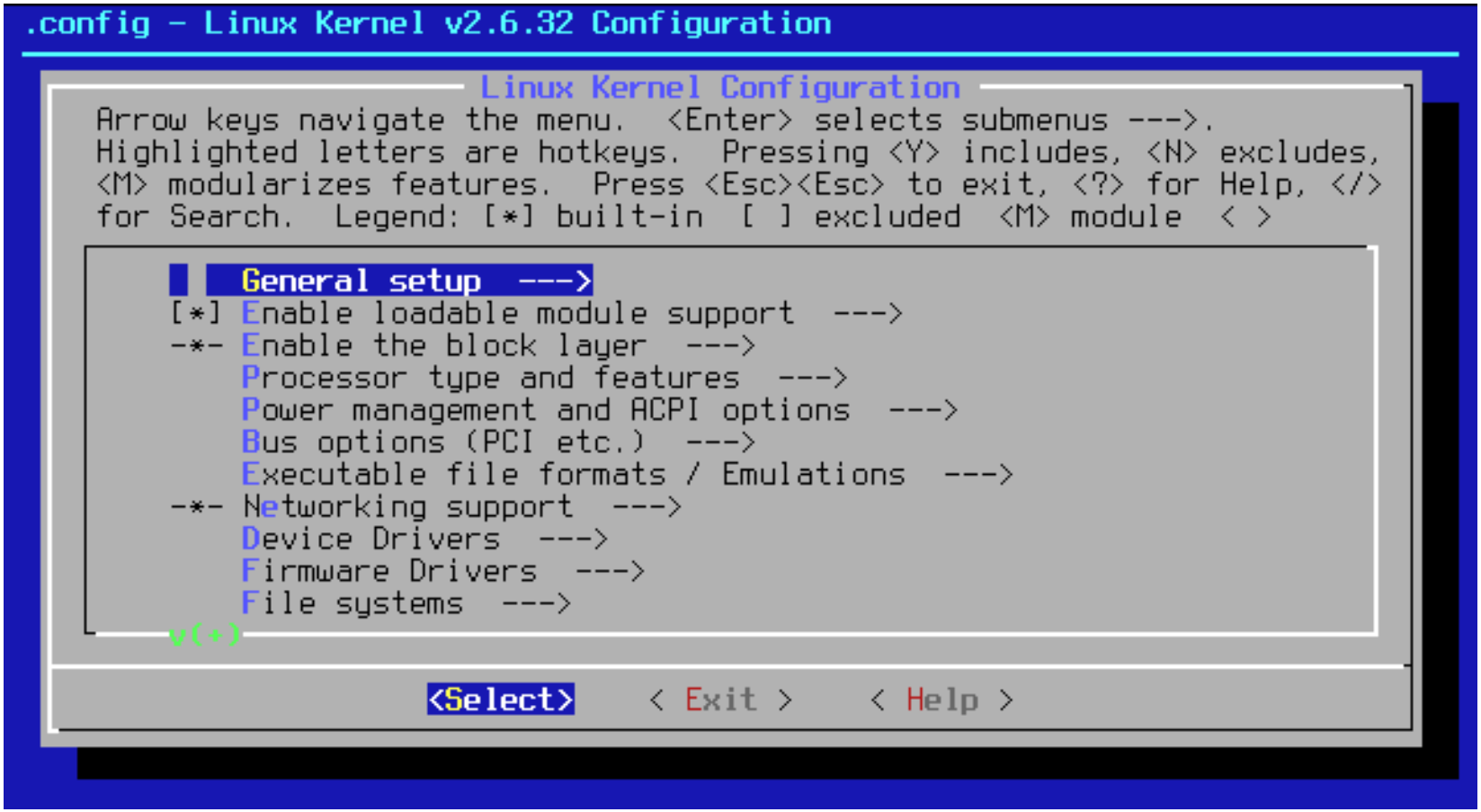
03-3 Graphical Front End via node.js

How to add a pretty face via a web browser

Ways to Add Graphics

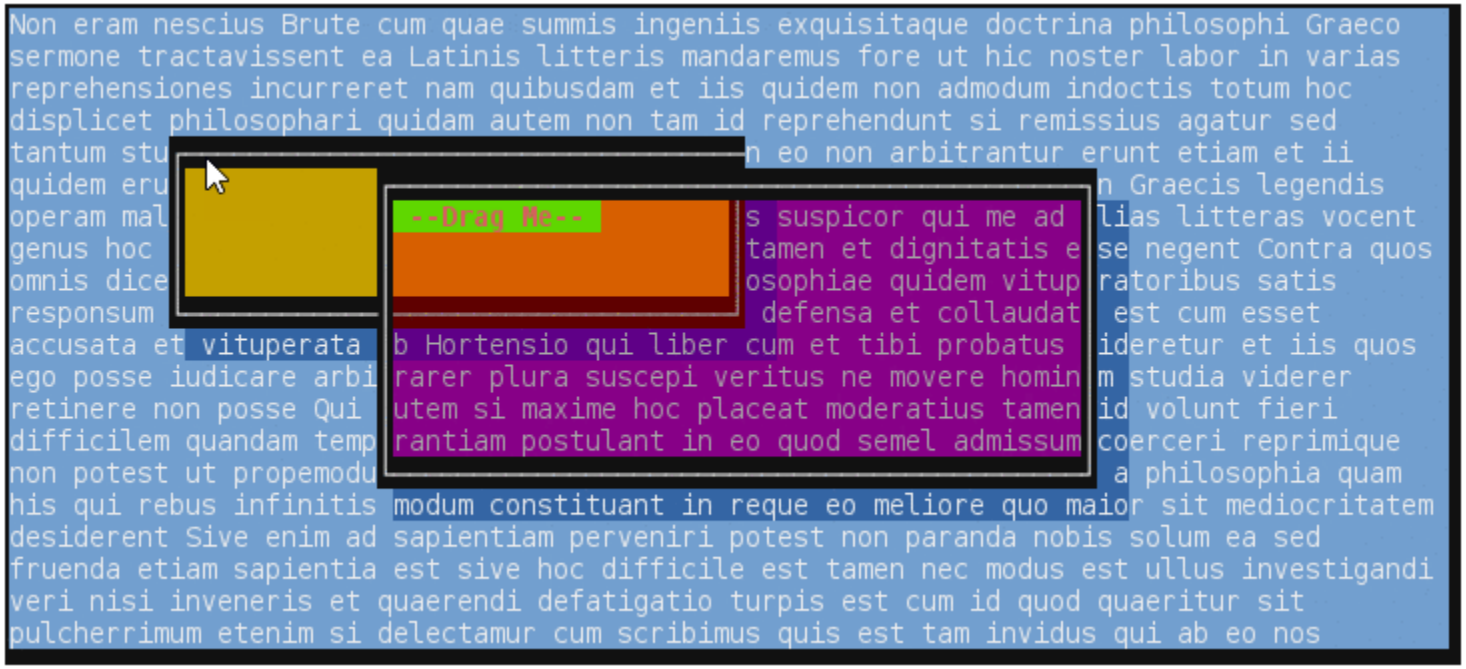
- ncurses (http://www.gnu.org/software/ncurses/)
- Blessed (https://www.npmjs.com/package/blessed)
- X Window System (http://www.x.org/wiki/)
- Qt (http://qt.digia.com/) both X-based and embedded
- Web server
  - node.js (http://nodejs.org/)

ncurses

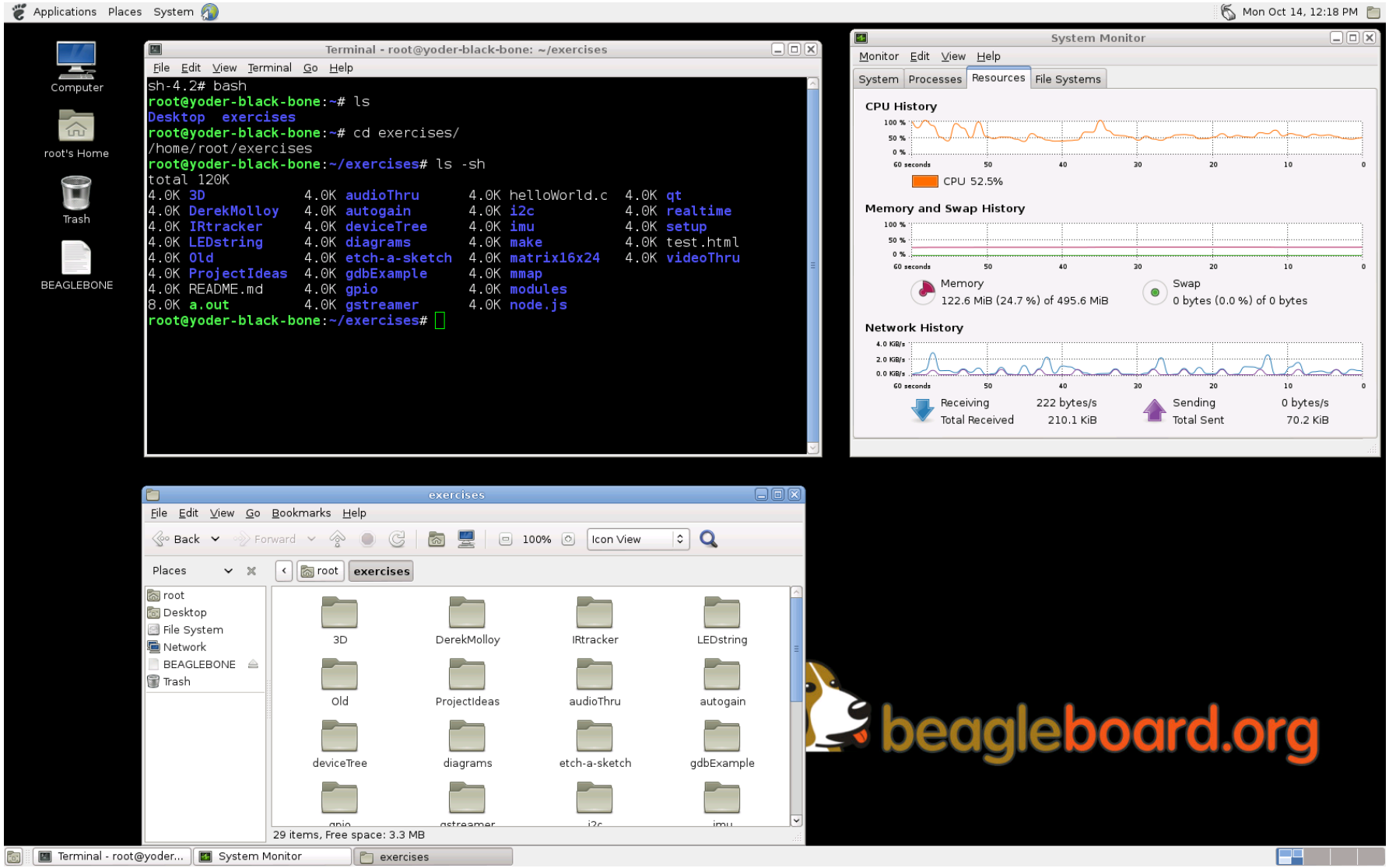


http://en.wikipedia.org/wiki/File:Linux-menuconfig.png

blessed



X Windows

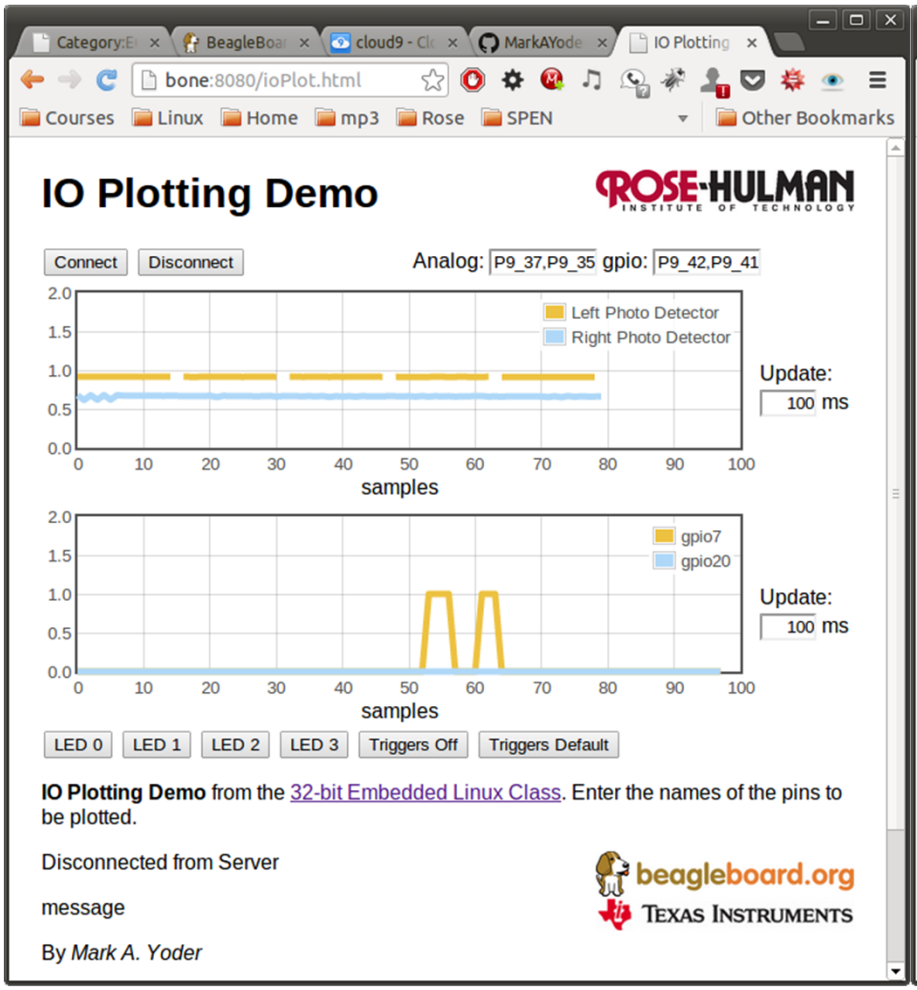

Applications Places System
Mon Oct 14, 12:18 PM
Computer
root's Home
Trash
BEAGLEBONE
Terminal - root@yoder-black-bone: ~/exercises
File Edit View Terminal Go Help
sh-4.2# bash
root@yoder-black-bone:~# ls
Desktop exercises
root@yoder-black-bone:~# cd exercises/
/home/root/exercises
root@yoder-black-bone:~/exercises# ls -sh
total 120K
4.0K 3D 4.0K audioThru 4.0K helloWorld.c 4.0K qt
4.0K DerekMolloy 4.0K autogain 4.0K i2c 4.0K realtime
4.0K IRtracker 4.0K deviceTree 4.0K imu 4.0K setup
4.0K LEDstring 4.0K diagrams 4.0K make 4.0K test.html
4.0K Old 4.0K etch-a-sketch 4.0K matrix16x24 4.0K videoThru
4.0K ProjectIdeas 4.0K gdbExample 4.0K mmap
4.0K README.md 4.0K gpio 4.0K modules
8.0K a.out 4.0K gstreamer 4.0K node.js
root@yoder-black-bone:~/exercises#
System Monitor
Monitor Edit View Help
System Processes Resources File Systems
CPU History
CPU 52.5%
Memory and Swap History
Memory
122.6 MiB (24.7 %) of 495.6 MiB
Swap
0 bytes (0.0 %) of 0 bytes
Network History
Receiving 222 bytes/s
Total Received 210.1 KiB
Sending 0 bytes/s
Total Sent 70.2 KiB
exercises
File Edit View Go Bookmarks Help
Back Forward 100% Icon View
Places root exercises
root
Desktop
File System
Network
BEAGLEBONE
Trash
3D DerekMolloy IRtracker LEDstring
Old ProjectIdeas audioThru autogain
deviceTree diagrams etch-a-sketch gdbExample
29 items, Free space: 3.3 MB
Terminal - root@yoder...
System Monitor
exercises
beagleboard.org

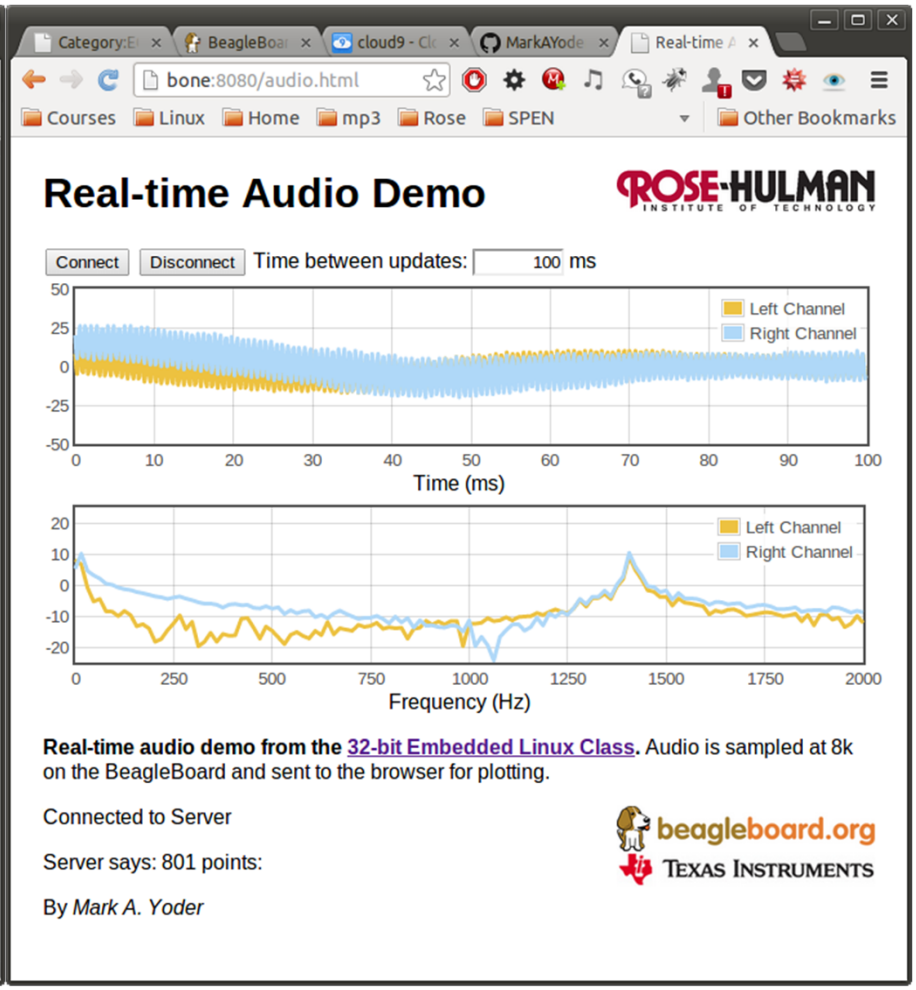
Qt – Both X and embedded

```
$ sudo apt install qt4-demos
$ qtdemo
```

http://elinux.org/ECE497_Notes_on_Qt

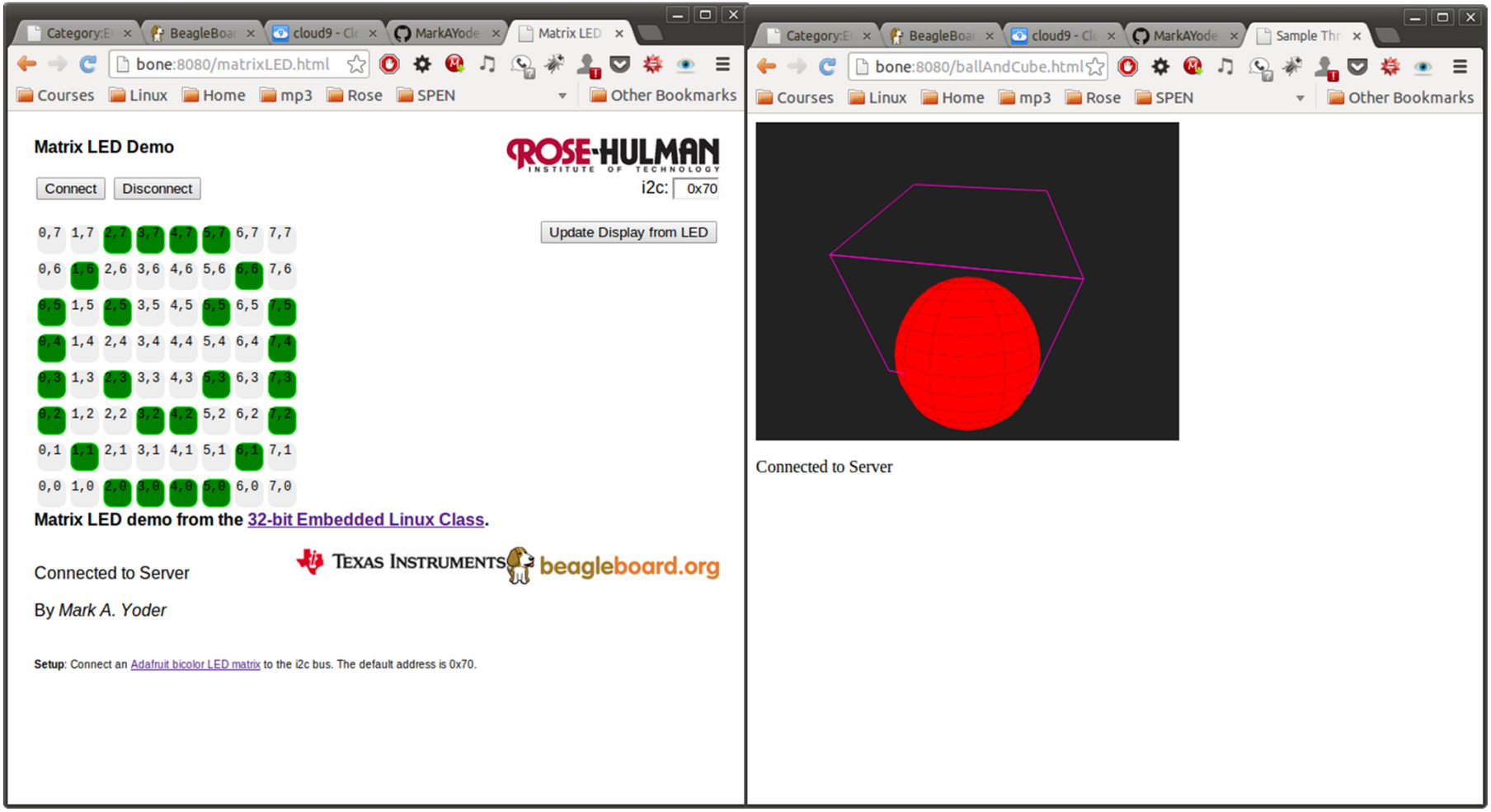
Via the Web via node.js






```
bone$ cd ~/exercises/realtime
bone$ ./boneServer.js
```

Via the Web via node.js




```
bone$ cd ~/exercises/realtime
bone$ ./boneServer.js
```

node.js

- Platform built on Chrome's JavaScript runtime for easily building fast, scalable network applications.
- Uses an event-driven, non-blocking I/O model that makes it lightweight and efficient, perfect for data-intensive real-time applications that run across distributed devices.
- Programmed in JavaScript on both server and client.

http://nodejs.org/

node.js example: Webserver

- This simple web server written in Node responds with "Hello World" for every request.

```
var http = require('http');
http.createServer(function (req, res) {
  res.writeHead(200, {'Content-Type': 'text/plain'});
  res.end('Hello World\n');
}).listen(1337);
console.log('Server running on port 1337');
```

- To run the server, put the code into a file example.js and execute it with the node program:

```
$ node example.js
Server running on port 1337
```

Things to know

- JavaScript
  - socket.io
  - jQuery
  - DOM
- html
- CSS

- Where to you start?

Javascript – C-like

```
#include <stdio.h>
main() {
    int i;
    for(i=0; i<5; i++) {
       printf("i=%d\n", i);
    }
}
```

```
var i;
for(i=0; i<5; i++) {
    console.log("i=%d", i);
}
```

Things to know

- JavaScript
  - socket.io
  - jQuery
  - DOM
- html
- CSS

socket.io

- http://socket.io/
- **Socket.IO** aims to make realtime apps possible in every browser and mobile device, blurring the differences between the different transport mechanisms.
- It's care-free realtime 100% in JavaScript.

socket.io

- Server

```
var io = require('socket.io').listen(80);
io.sockets.on('connection', function (socket) {
  socket.emit('news', { hello: 'world' });
  socket.on('my other event', function (data) {
    console.log(data);
  });
});
```

- Client - Browser

```
<script>
var socket = io.connect('http://localhost');
socket.on('news', function (data) {
  console.log(data);
  socket.emit('my other event', { my: 'data' });
});
</script>
```

socket.io

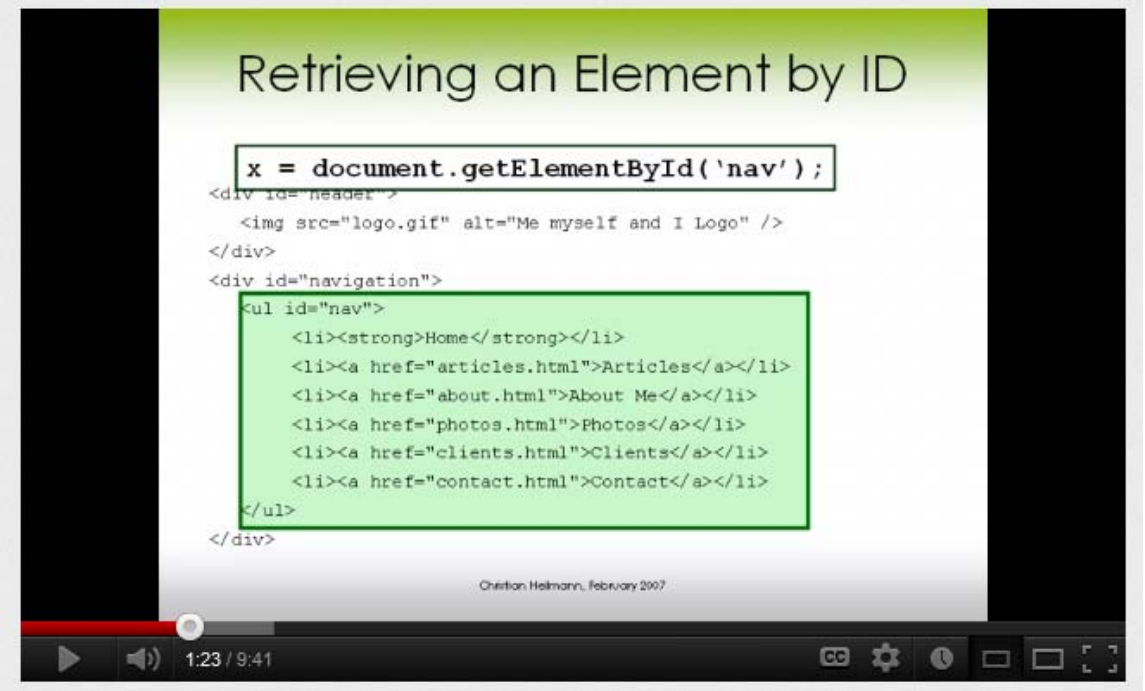
- See: **Getting Started With node.js and socket.io (v0.7+) – Part 2**
- http://www.codehenge.net/2011/12/getting-started-with-node-js-and-socket-io-v0-7-part-2/
- My code is based on this

Things to know

- JavaScript
  - socket.io
  - DOM
  - jQuery
- html
- CSS

DOM

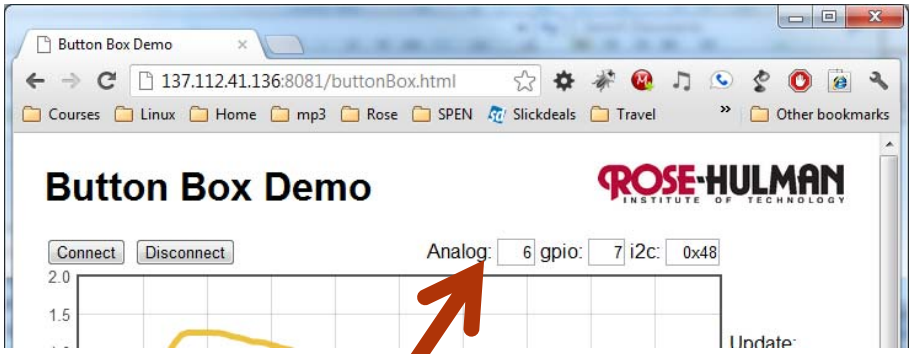
- **Essentials of the DOM and JavaScript in 10 Minutes**
- **http://www.youtube.com/watch?v=URF2sVQWuxU**
- **10 minute YouTube video**
- **However we'll use jQuery, it's much more compact**



Things to know

- JavaScript
  - socket.io
  - DOM
  - jQuery
- html
- CSS

jQuery



- http://jquery.org/
- jQuery is a fast and concise JavaScript Library that simplifies HTML document traversing, event handling, animating, and Ajax interactions for rapid web development.
- jQuery is designed to change the way that you write JavaScript
- Looks like **$()** in html

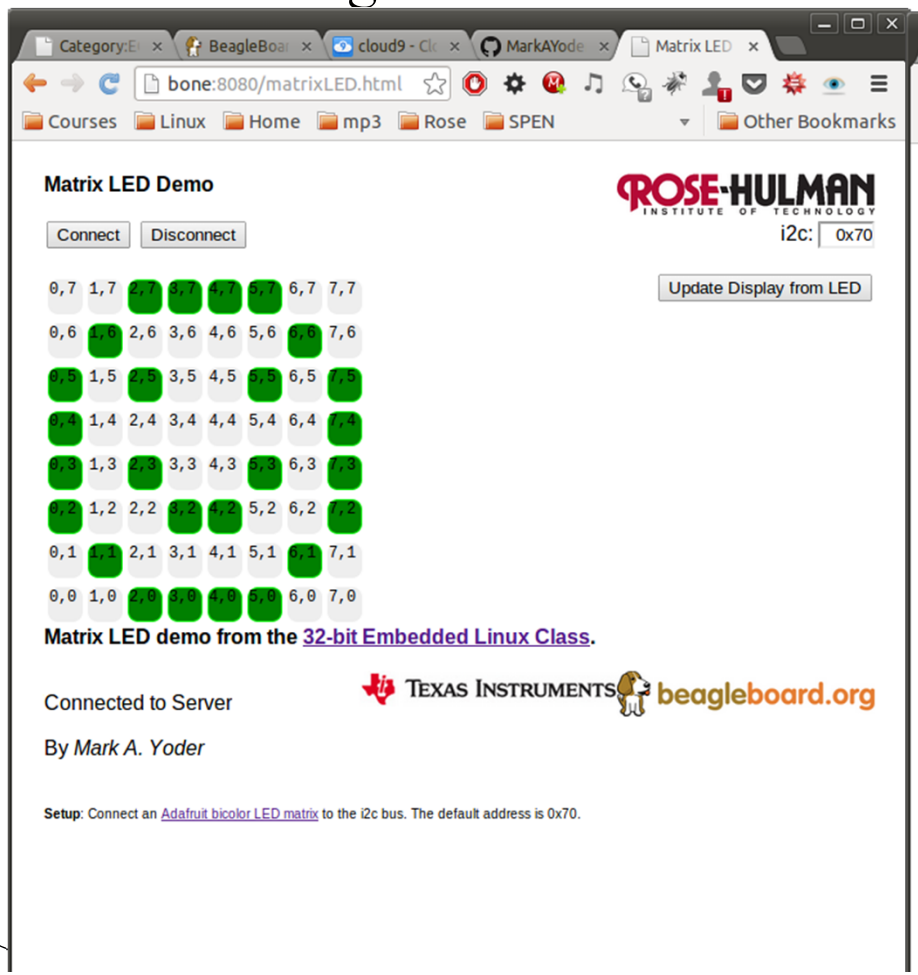
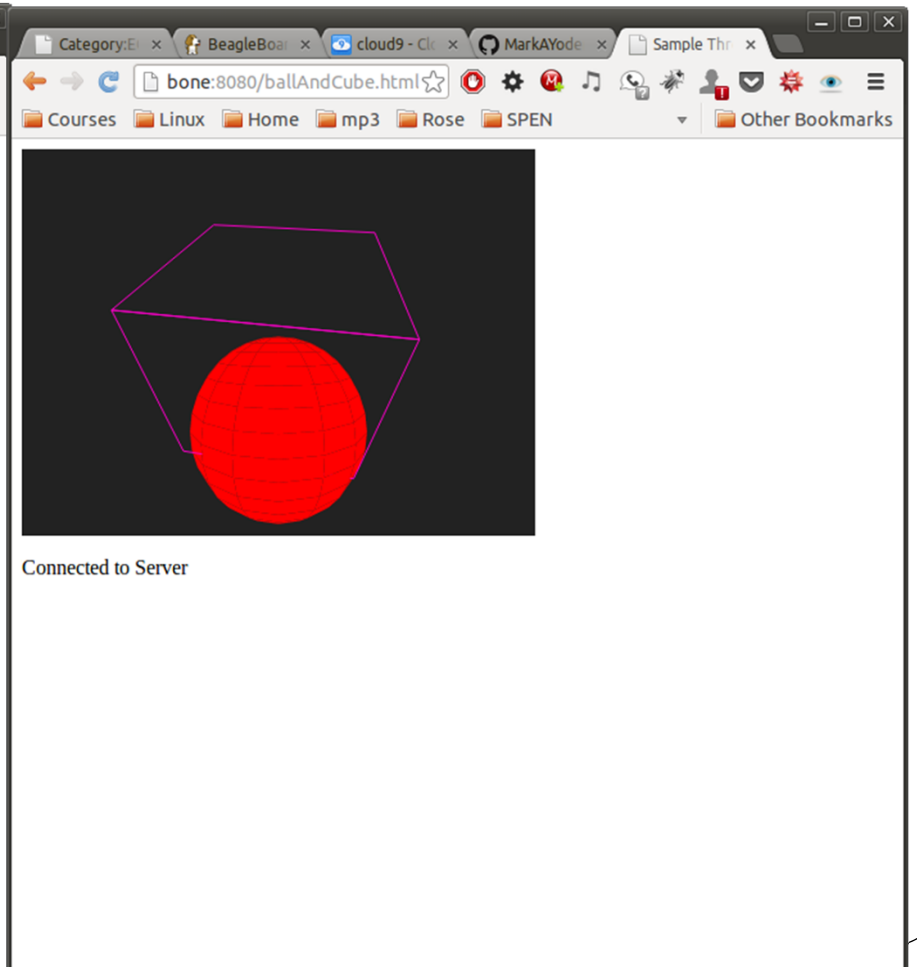
```
Analog: <input id="ainNum" type="text" value=""
style="text-align: right; width:2em">
```

- **In JavaScript**

```
$("#ainNum").val(ainNum).change(function () {
        ainNum = $(this).val();
    });
```

Things to know

- JavaScript
  - socket.io
  - DOM
  - jQuery
- html
- CSS
- FLOT
- http://www.flotcharts.org/
- Flot is a pure JavaScript plotting library for jQuery, with a focus on simple usage, attractive looks and interactive features.

To Do

- Look at **`~/exercises/realtime`** and see what you can figure out.