trgr / **node-gnublin**

Watch 1   ★ Star 0   Fork 0

<> Code   ⊙ Issues 0   Pull requests 0   Projects 0   Security   Insights

Node bindings for the gnublin API

⊙ **36** commits   ⑂ **1** branch   ⊡ **0** packages   ⬚ **0** releases   ⬚ **1** contributor   ⚖ **GPL-3.0**

Branch: master ⌄   New pull request   Find file   Clone or download ⌄

trgr Pumped version and updated readme   Latest commit e9f34ce on 7 Dec 2013

📁 src	Added more tests. Added WritePort method	6 years ago
📁 tests	Added more tests. Added WritePort method	6 years ago
📄 .gitignore	Added more tests. Added WritePort method	6 years ago
📄 LICENSE	Initial commit	6 years ago
📄 README.md	Pumped version and updated readme	6 years ago
📄 binding.gyp	Initial commit, workings on board, adding more methods	6 years ago
📄 package.json	Pumped version and updated readme	6 years ago

📖 **README.md**

node-gnublin

Bindings for the gnublin api http://en.gnublin.org/index.php/API

Supporting so far only the pca9555 subset. Compiles with `#define BOARD RASPBERRY_PI` but this does not affect pca9555 so it can be used with beaglebone, gnublin board, or raspberry pi + gnublin module

Install

Do the ususal `npm install gnublin`

Usage

```
var gnublin = require("gnublin");
var pca9555 = new gnublin.pca9555();

/* Then do stuff to your relay like:*/

pca9555.PinMode(0,gnublin.OUTPUT);
pca9555.DigitalWrite(0,gnublin.HIGH);

pca9555.PinMode(1,gnublin.INPUT);
pca9555.DigitalWrite(1,gnublin.LOW);

pca9555.SetAddress(0x21);

var byte = pca9555.ReadPort(0).toString(16);
```

So far the library supports:

`int PinMode(int pin,string mode)`

`int PortMode(int port, string mode)`

`int DigitalWrite(int pin, int value)`

`int DigitalRead(int pin)`

`int ReadState(int pin)`

`int ReadPort(int port)`

`int WritePort(int port,unsigned char value)`

`void SetDeviceFile(string file)`

`void SetAddress(int address)`

`bool Fail()`

`string GetErrorMessage()`