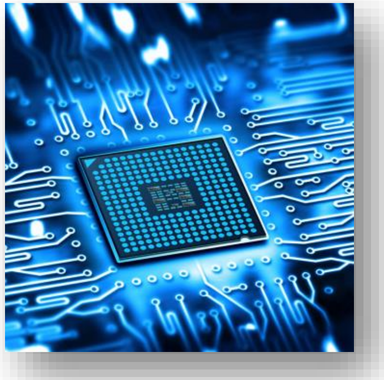


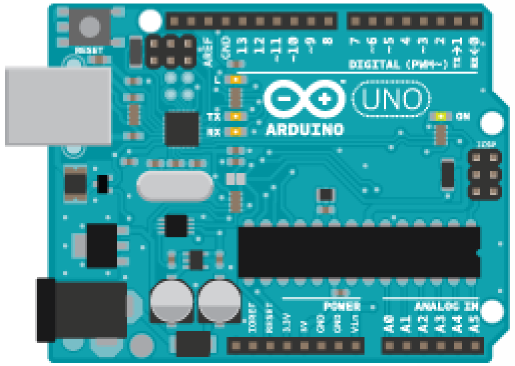
Hackathon Essentials

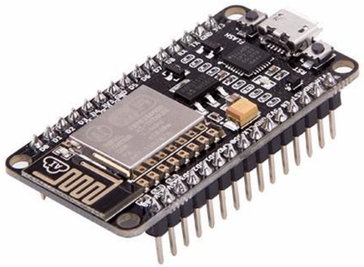
Electronics for Programmers

Microcontroller/ Microprocessor

- Microcontrollers
- Atmega/ESP8266…
- Arduino/NodeMCU
- CPU/RAM/ROM/Peripherals on same chip
- Slow CPU’s

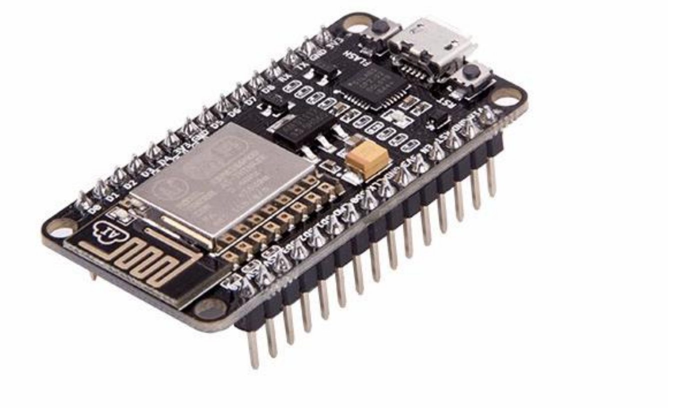
- Microprocessors
- Altera/ARM/Intel..
- Raspberry Pi/MK14 …
- Just a CPU, external RAM/ROM/Peripherals
- Waay faster

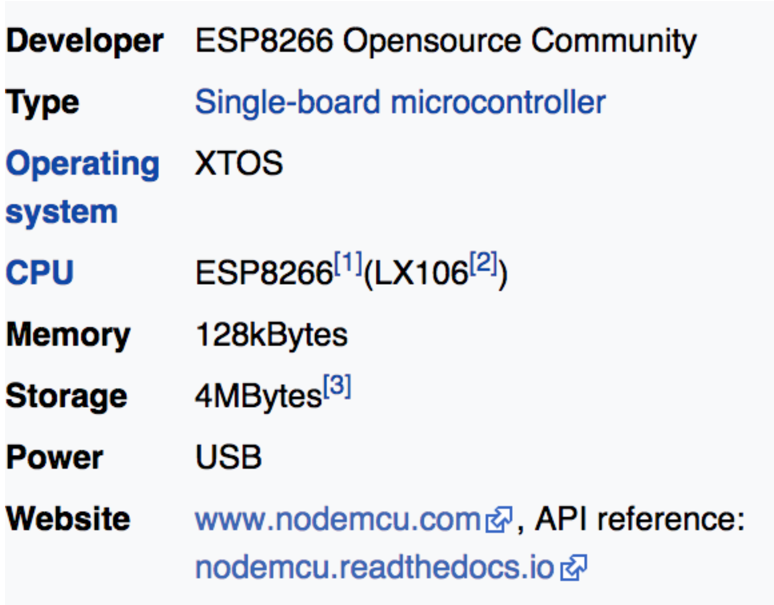




WiFi Enabled Microcontrollers : ESP8266

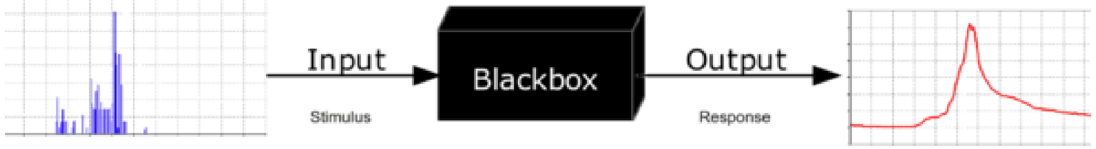
- Board Name: NodeMCU
- Open Source IoT Platform
- Extended on Arduino Platform, so a **delight for C++ developers**
- 802.11b/g/n compliant
- Create Access Points, Can also work as a router!
- Create HTTP servers/clients, Websocket server/clients



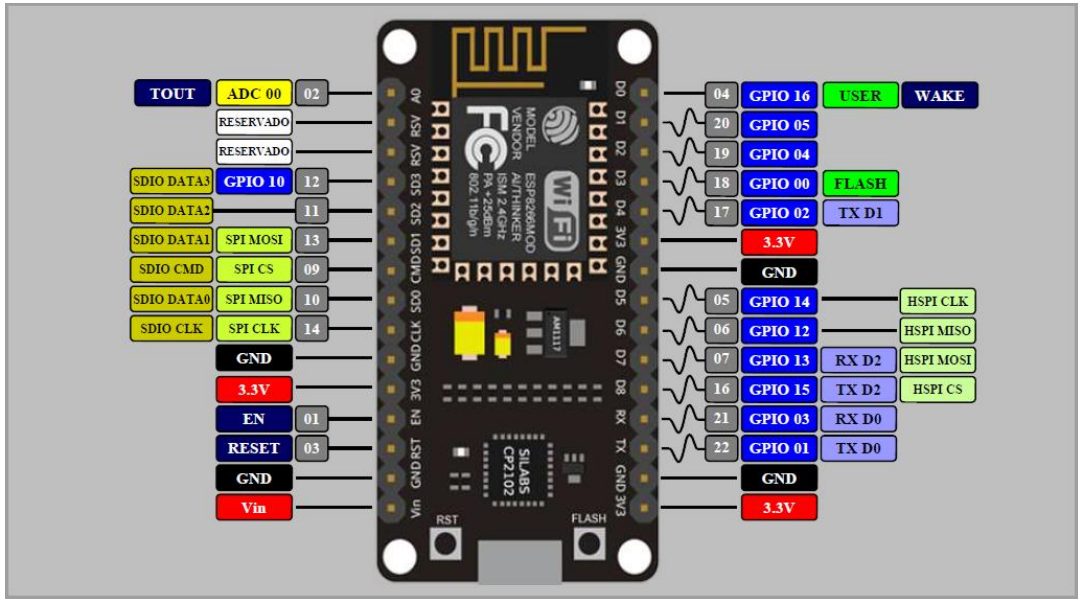


Developer	ESP8266 Opensource Community
Type	Single-board microcontroller
Operating system	XTOS
CPU	ESP8266[1](LX106[2])
Memory	128kBytes
Storage	4MBytes[3]
Power	USB
Website	www.nodemcu.com, API reference: nodemcu.readthedocs.io

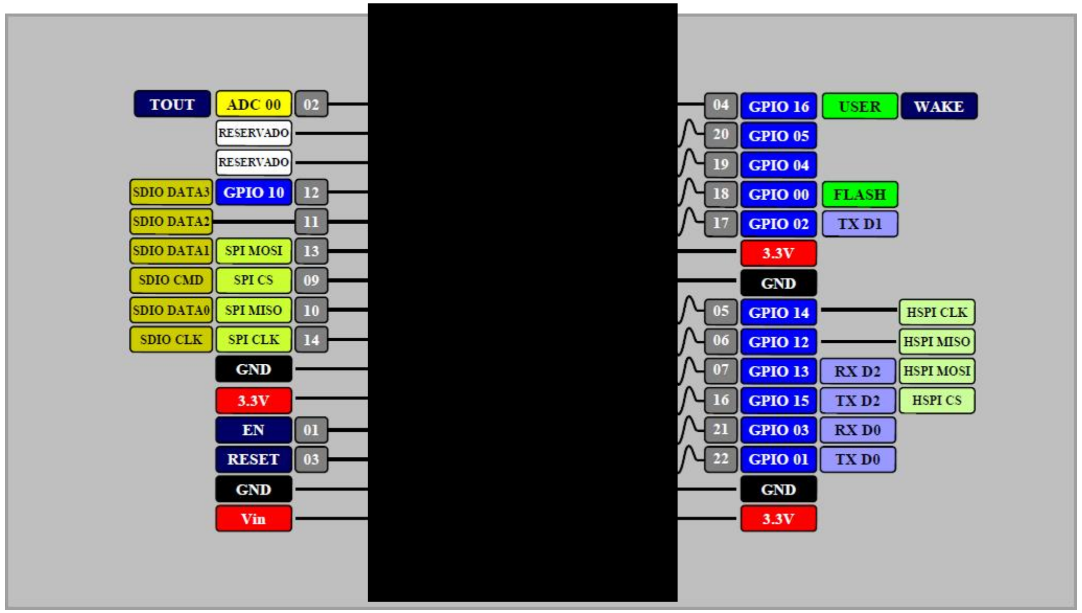
Black Box Theory!



- Hardware is a Black Box with pins to interact with the box!!
- We write code to play around and control pins and do some processing

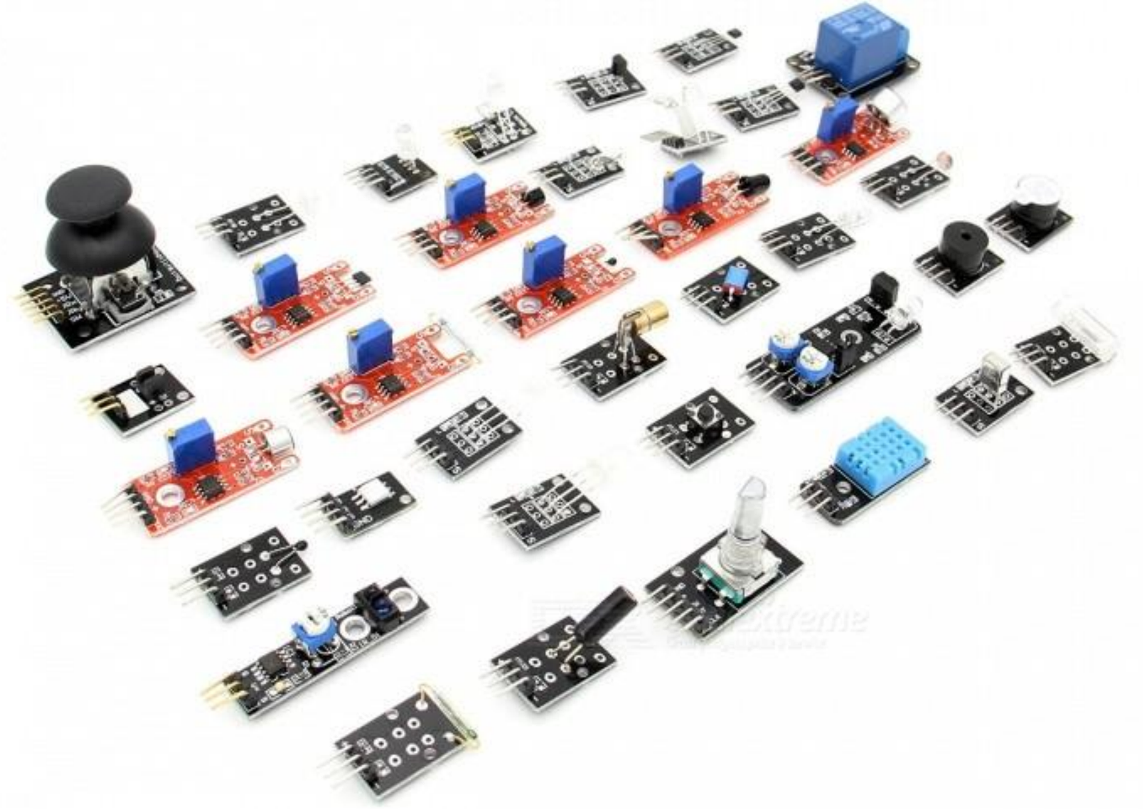


For Electronics Engineers



For Developers and Programmers

Sensors and Devices

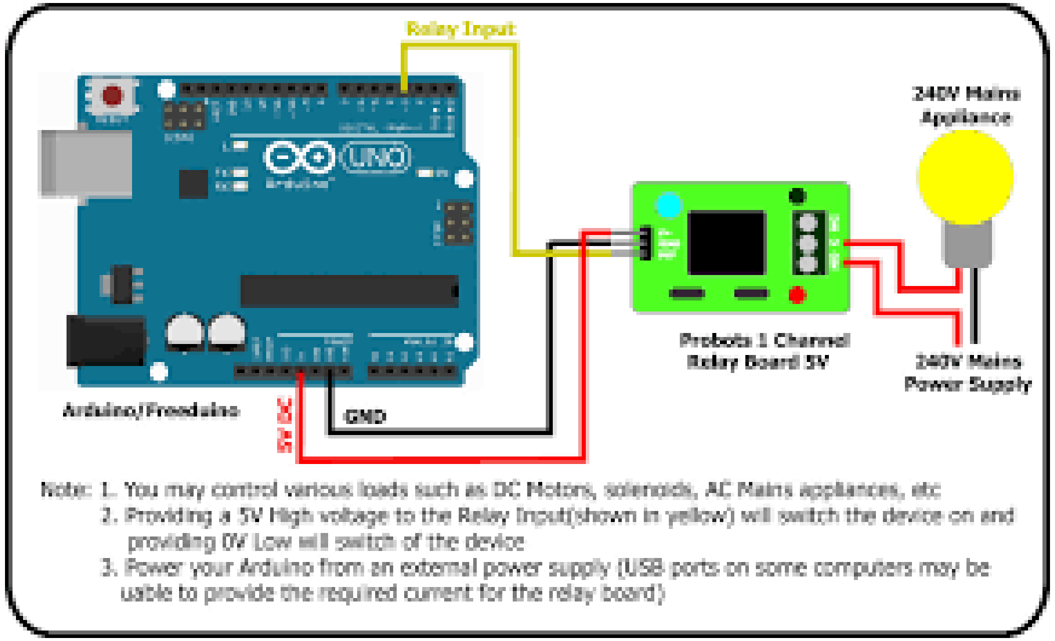


Sensors and Devices

- Real power of embedded platforms comes from the fact that they can be interfaced to **control sensors (Temperature, Humidity, Microphone etc.) and output devices(LED, Motors, Relays, Speakers etc..)**
- Make the connections (Power, Ground always present. Then depending on the sensor/device you will have to connect pins for data exchange and control).
- Several open source libraries available to perform various tasks with a sensor!
- Google the sensor/device you are working with and you'll find a working **library** very likely!

Today’s pick: Relays

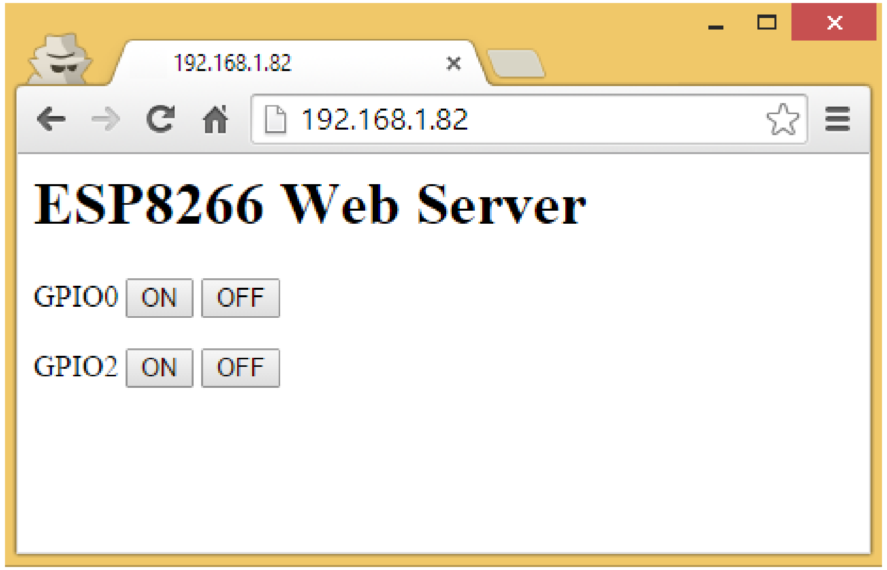
- Electrically operated switch
- Can turn an **electrical appliance ON/OFF** by receiving a signal from a pin from your microcontroller.
- You control the pin via the code you write.



Note: 1. You may control various loads such as DC Motors, solenoids, AC Mains appliances, etc
2. Providing a 5V High voltage to the Relay Input(shown in yellow) will switch the device on and providing 0V Low will switch of the device
3. Power your Arduino from an external power supply (USB ports on some computers may be uable to provide the required current for the relay board)

Today’s pick: ESP8266 WebSocket Server

- Cool Intro to Websockets: http://blog.teamtreehouse.com/an-introduction-to-websockets
- And the Code for ESP8266: https://github.com/maany/Hackathon_Essentials_ESP8266_Base



192.168.1.82

ESP8266 Web Server

GPIO0 ON OFF

GPIO2 ON OFF

Review for the developer

- Hardware is a Black Box with pins to interact with the box
- Pins connect to sensors/peripherals that I want to control with the microcontroller
- Google or talking with experts will help me to connect the pins between my microcontroller and the microcontroller
- Simple code for the microcontroller can control these pins and therefore control the connected devices or peripherals.

Questions