PRELIMINARY Basic NodeMCU-V2 Development Shield Getting Started Guide

Table of Contents

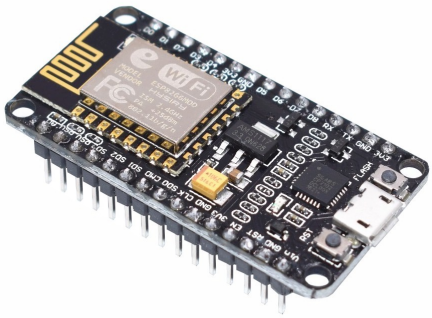
Intro

The purpose of this document is to be used as a "Getting Started" guide for the Basic NodeMCU-V2 Development Shield. This document has been written for a beginner (somebody who has little to no experience with Arduino's or electronics) to get them up and running with the shield and upload a basic example to it.

Prerequesites

Components

You should have the following 2 parts:

A NodeMCU V2 Development Board	Basic NodeMCU-V2 Development Shield
	Cell in column 2, row 2

If the NodeMCU isn't already plugged into the shield (plug it into the 2 pin headers on the board).

NOTE	This assumes that the shield is already soldered. If it isn't and you have a plain PCB and components, checkout the Soldering Guide.

Software

Arduino IDE

We will be using the Arduino IDE (Integrated Development Environment) to create programs and upload them to the NodeMCU. While it is the most common way of doing so, there are other tools and IDEs out there, like MicroPython for examples.

If you don't have the Arduino IDE installed on you machine, you should do that now. The program's download page can be found at https://www.arduino.cc/en/software. They have a Getting Started Guide if you're having trouble installing the program.

ESP8266 Board Package

When you first install the Arduino IDE, it doesn't come with the ability to upload to ESP8266 based modules (like the NodeMCU), so we must add that. Open up the IDE and go into File→Preferences. In the *Additional Board Managers URLs* section, enter the following URL:

```
https://arduino.esp8266.com/stable/package_esp8266com_index.json
```

This allows us to use the build-in Board Manager to install ESP8266 board. After that, exit out of the preferences menu (by clicking the *OK* button) and go into Tools→Board→Board Manager. Once the Board Manager menu pops us, type in `esp8266` in the filter entry and press Enter. You should see an `esp8266 by ESP8266 Community` board popup. Click on the Install button for that board. The download and installation process might take a hot minute, so grab a quick cup of coffee in the meanwhile (or tea which is my personal favorite). After the installation process is complete, close out of the Board Manager (with the Close button). You are now able to compile and upload programs to ESP8266 based modules including the NodeMCU which we are using.

Basic Dev. Shield Library

Lets Light This Place Up

Now let's make some LEDs blink on the Dev. Shield. Go to and select Files→Examples→PLACEHOLDER→Blink. A new sketch should pop up with some pre-written code.

New let's select the right board to compile and upload to. In the Tools menu, for the Board select `NodeMCU 1.0 (ESP-12E Module)` board. The rest of the settings should be fines, and it should look like the following:

IMAGE HERE

Now let's compile the Blink example we just loaded (turning the human-readable code into machine code). Click on the Compile button in the IDE. If all goes well, the IDE's status bar should say `Done Compiling`. Now let's plug in the NodeMCU Board into the computer. Use a MicroUSB cable to connect the NodeMCU module to the computer. Now go into the Tools menu, and in the Port submenu chose a COM port that isn't COM1 which should appear (or for Linux and Mac users the port which isn't `dev/ttyS0`). Now we are able to upload the program, and we do that by clicking on the Upload button in the IDE. When the status bar of the IDE says `Done Uploading`, the program has been uploaded to the board.

You should start noticing that the shield's blue LED (and the middle segment on the 7 segment display) should start blinking. If that's the case, CONGRATULATIONS!, you just uploaded a program on the NodeMCU board with the Basic Dev. Shield.

Now What?

Well really that's up to you. There is a vast possibility of what you can do with the NodeMCU board and the shield, and we've only done a tiny fraction of what's it's capable of (and we haven't started using the Wifi part of the NodeMCU). There are plenty of examples that you can try out, which the ESP8266 board package and the Basic NodeMCU Dev. Shield's library provides.

SOME HAPPY ENDING PLACEHOLDER