

TA0297
WEMOS D1 R2 WIFI
ARDUINO
DEVELOPMENT BOARD
ESP8266



Contents

1. Overview – TA0297	3
2. Getting started:.....	3
2.1. What is WeMos D1 R2 Wifi Arduino Development Board?	3
2.2. What is IDUINO UNO?.....	3
2.3. What are the differences between Arduino development boards and WeMos-D1R2?.....	3
3. Software installation	4
3.1. Arduino Software/IDE	4
3.2. Play with your first “Hello World” LED example.....	5
4. Before we getting started	6
4.1. Connect your WeMos- D1R2 to your computer.....	6
4.2. Driver installation	7
4.3. Connecting correctly with your PC?	8
5. Have fun.....	8
5.1. Telling Arduino IDE where to find Wemos-D1R2 Library.....	8
5.2. Trying the Blink Example.....	12

1. Overview – TA0297

In this instruction, We will introduce you through one simple example of WeMos D1 R2 Wifi Arduino Development Board ESP8266 using the Arduino IDE. Get your WeMos Wifi board. Let's get it started!

2. Getting started:

2.1. What is WeMos D1 R2 Wifi Arduino Development Board?

WeMos-D1R2 is an ESP8266-12 based WiFi enabled microprocessor unit on an Arduino-UNO footprint. That means the board looks like an UNO. Apparently several shields, sensors and output devices that are manufactured for the Arduino platform will work on the UNO with the added advantage of built-in WiFi.

2.2. What is IDUINO UNO?

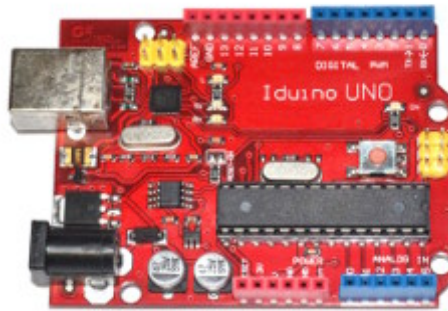


Figure 1 UNO

The Iduino Uno is based on the ATmega328. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header, and a reset button.

It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. You can tinker with your UNO without worrying too much about doing something wrong, worst case scenario you can replace the chip for a few dollars and start over again.

For more information: <https://store.arduino.cc/usa/arduino-uno-rev3>

2.3. What are the differences between Arduino development boards and WeMos-D1R2?

Though the Arduino UNO and the WeMos-D1R2 are similar, there are a few differences in their pin assignment. The above illustration provides the mapping of the pins.

In some situations programs written for the UNO will need to be modified a little to the proper pin assignments of the WeMos-D1R2. More on this when we write our first sketch.

Table 1 Comparison Table

Arduino-UNO			WeMos-D1R2	
SCL	I2C: SCL	→	GPIO05	I2C: SCL
SDA	I2C: SDA	→	GPIO04	I2C: SDA
AREF		→		
GND		→	GND	
GPIO13	SPI: SCK	→	GPIO14	SCK
GPIO12	SPI: MISO	→	GPIO12	MISO
GPIO11	SPI: MOSI	→	GPIO13	MOSI
GPIO10	SPI: SS	→	GPIO15	SS
GPIO9		→	GPIO13	

Arduino-UNO			WeMos-D1R2	
GPIO8		→	GPIO12	
GPIO7		→	GPIO14	
GPIO6		→	GPIO2	
GPIO5		→	GPIO0	
GPIO4		→	GPIO04	
GPIO3		→	GPIO05	
GPIO2		→	GPIO16	
GPIO1	TX	→	GPIO01	TX0
GPIO0	RX	→	GPIO03	RX0

Difference in pin assignments between Arduino UNO and Wemos-D1R2

3. Software installation

In this section, We will introduce you the development platform where you translate creative mind into codes and let it fly.

3.1. Arduino Software/IDE

Downloads from [here](#). Open Windows-based app by double clicking it and follow the instruction to complete(Remember to install everything driver for Arduino). Easy!

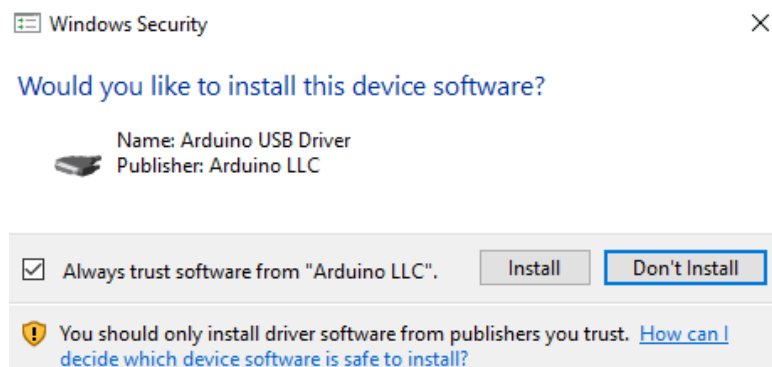


Figure 2 Installation of drivers

Connect your WeMos board with your computer

Connect WeMos and your PC by a blue USB cable, and if connect correctly you will see the green power LED light on and another orange LED is blinking.

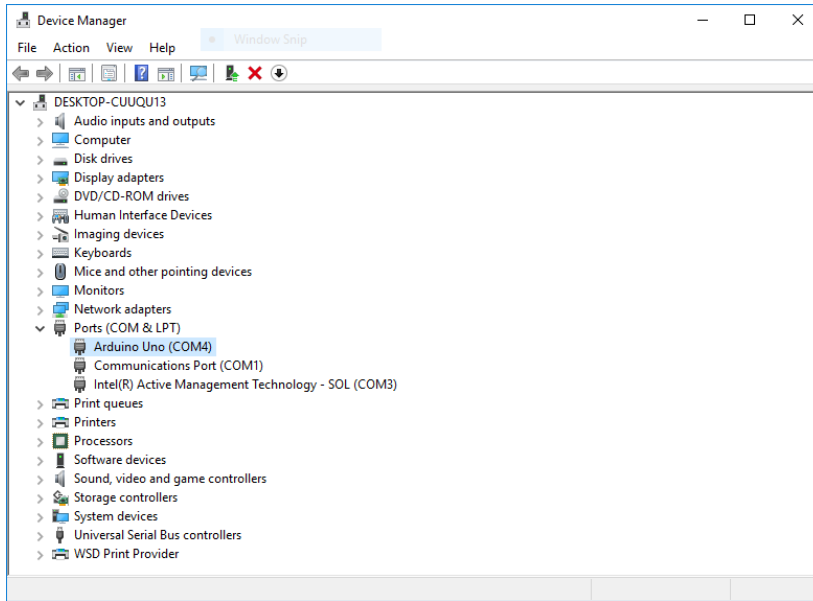


Figure 3 Check Your special COM and note it down the number

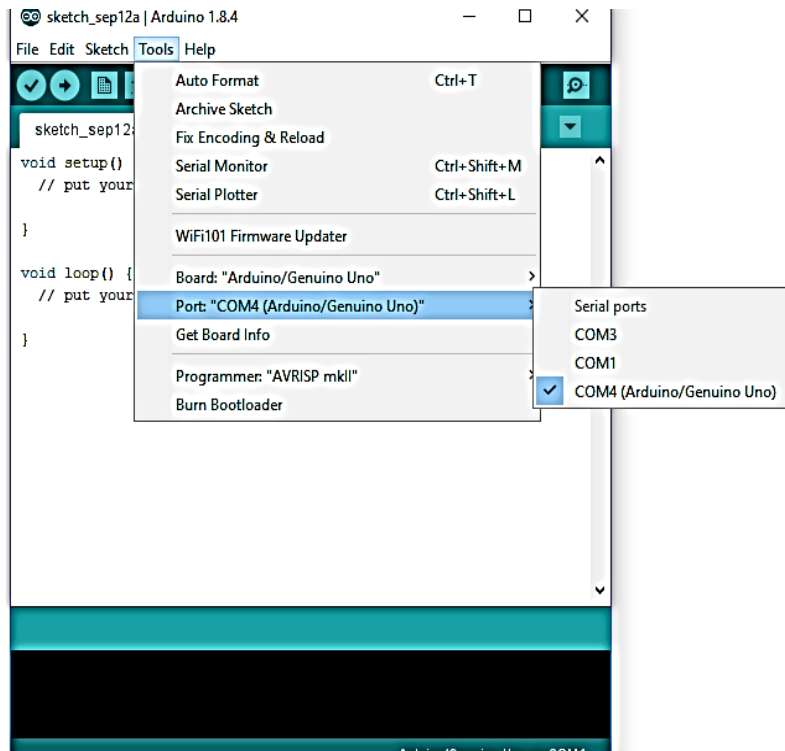
Find your Serial COM number and note it down.

Then we need to figure out which channel COM is currently communicating between PC and UNO. Following the path: Control panel | Hardware and Sound | Devices and Printers | Device Manager | Ports (COM & LPT) | Arduino UNO (COMx)

Note down the COM number as this will be required later. For this case, we are using the COM 4

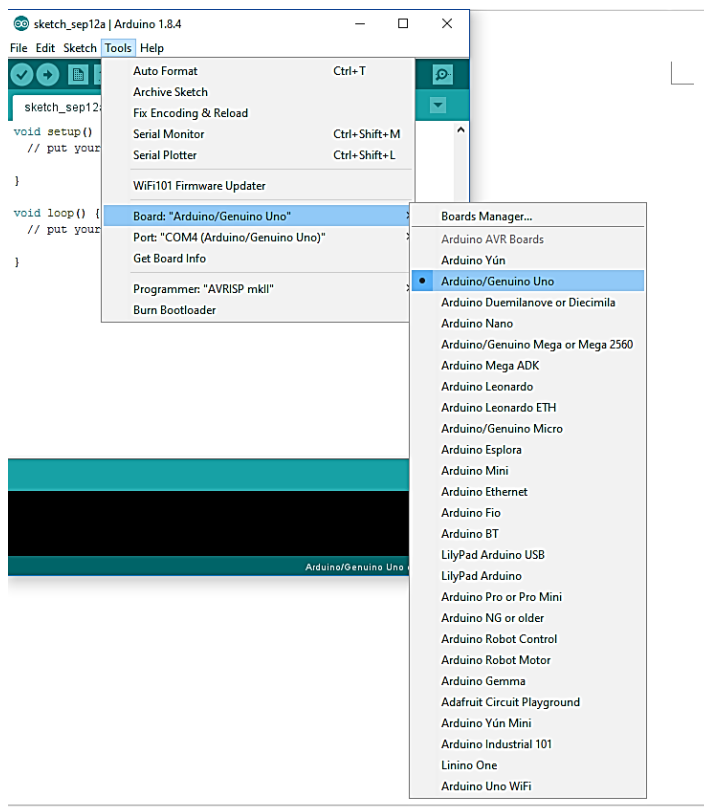
3.2. Play with your first “Hello World” LED example

Firstly let’s tell IDE where to find our Arduino port and which board you are currently using: The following instruction (Figure 3 and 4) shows the details:



Configuration of Ports

Figure 4 Port changes



Configuration of the Board

Figure 5 Board Changes

Then it's time to play with your first simple example. Following the path by File | Examples | 01. Basics | Blink and a new code window would pop out and just single press the upload. You will notice the orange LED is blinking almost every second.

4. Before getting started

4.1. Connect your WeMos- D1R2 to your computer

The WeMos-D1R2 board comes without a USB cable. Make sure you have a data cable and not an ordinary charging cable.



Figure 6 Data Cable

A data cable causes your Windows OS to react. It announces the attachment of a new hardware and tries to install the USB drivers. It either finds the driver on your system or it does not. If it does not then you will need to download the USB drivers. The USB interface chip is the CH340G. You will need to install the drivers for this chip.



4.2. Driver installation

A Google search for the "CH340G drivers" finds several sites. You can download the Windows 7 drivers from [here](#) or the Windows 8 from [here](#). Other operating systems can be downloaded from [here](#). Windows 10 operating system come with the drivers.

- a) Download the Windows CH340 Driver
- b) Unzip the file
- c) Run the installer which you unzipped
- d) In the Arduino IDE when the CH340 is connected you will see a COM Port in the Tools > Serial Port menu, the COM number for your device may vary depending on your system.

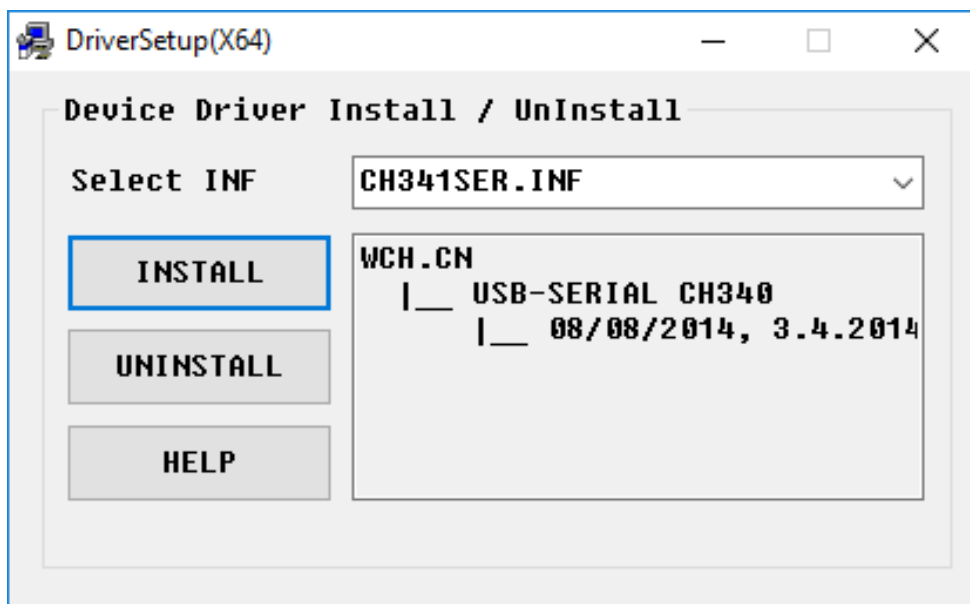


Figure 7 Driver installation

4.3. Connecting correctly with your PC?

Lets find out if the WeMos-D1R2 and your computer are communicating with each other. Getting this step right is important for any further action. Open the Device Manager via Control Panel| Hardware and Sound.

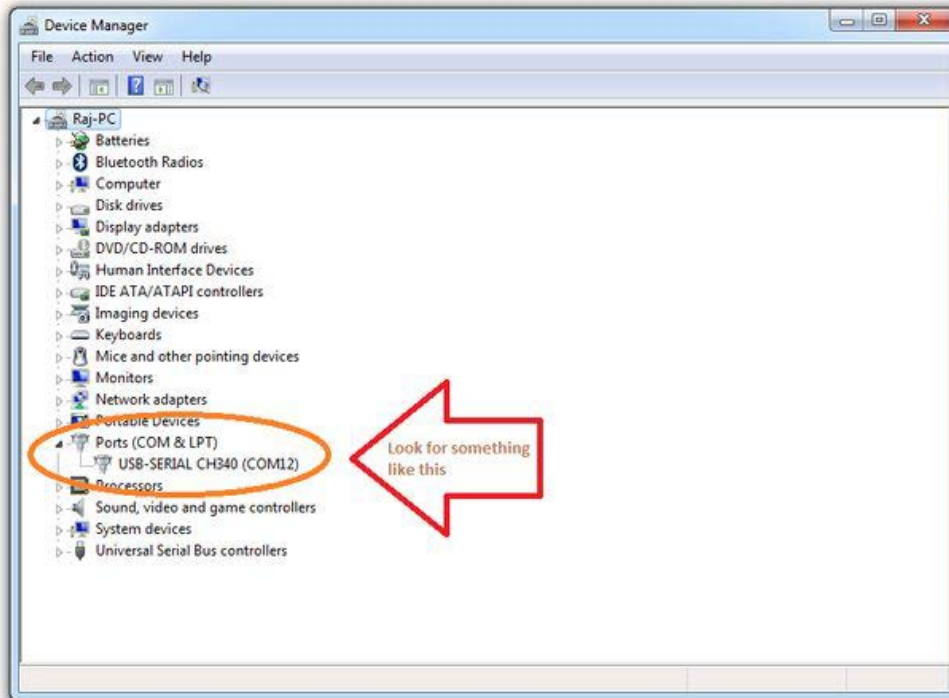
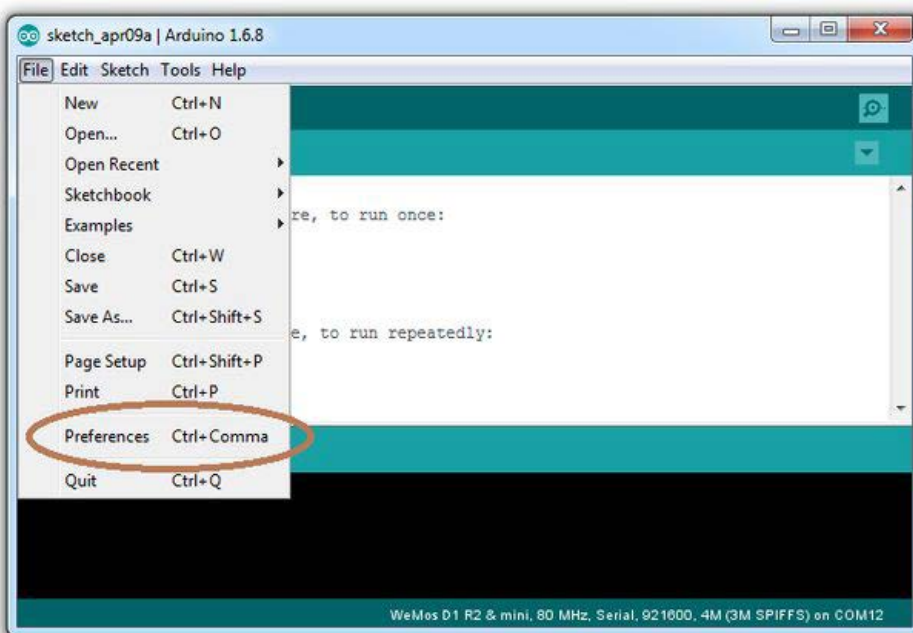


Figure 8 Device Manager

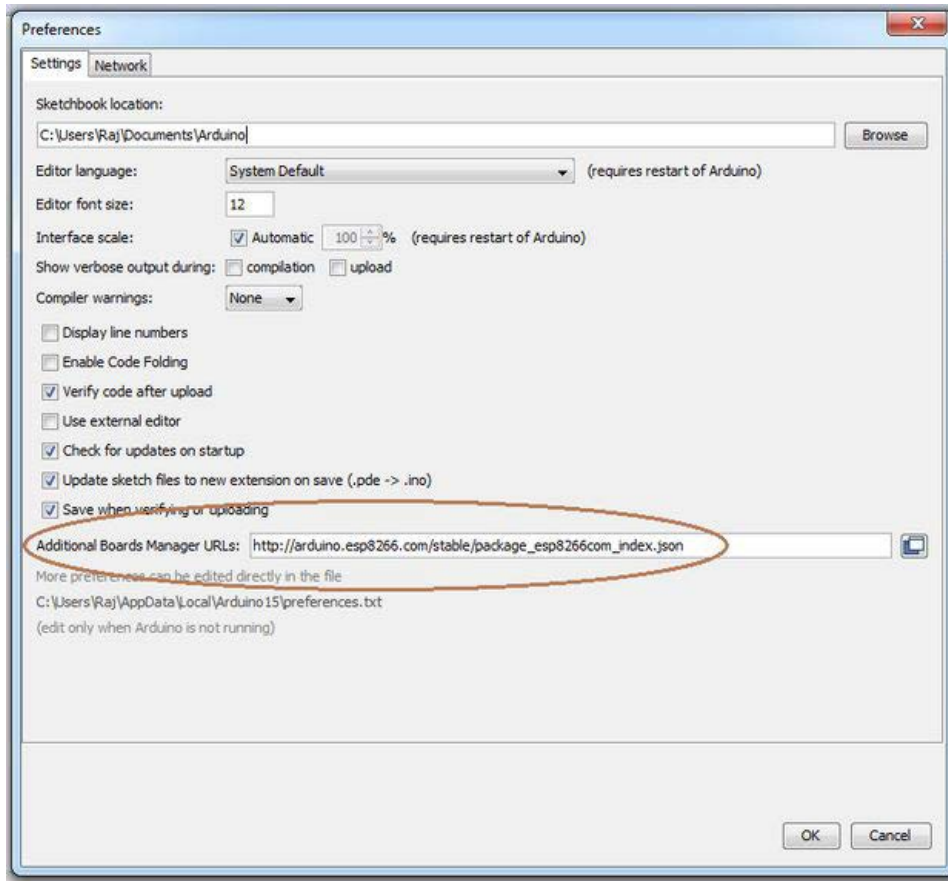
5. Have fun

5.1. Telling Arduino IDE where to find Wemos-D1R2 Library

The following steps:



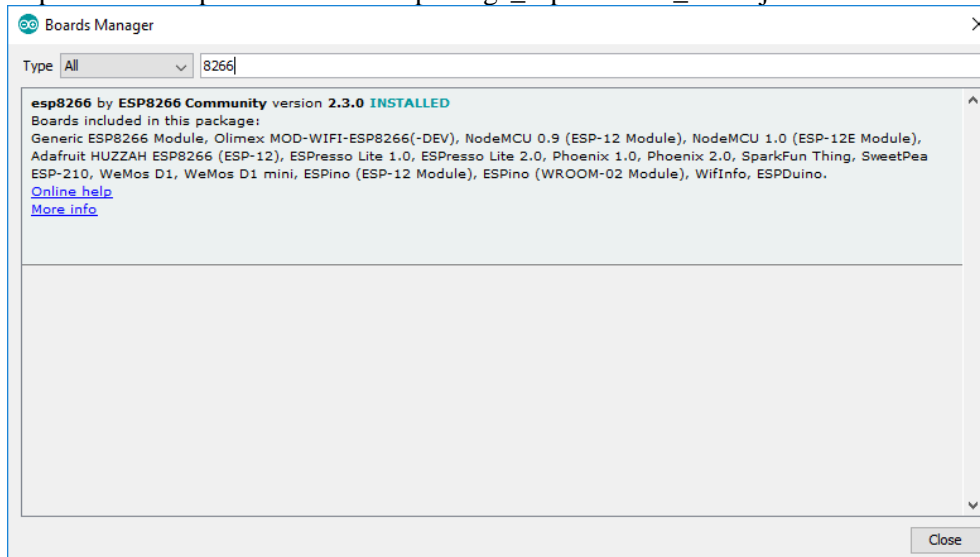
Click the Preferences



Preferences

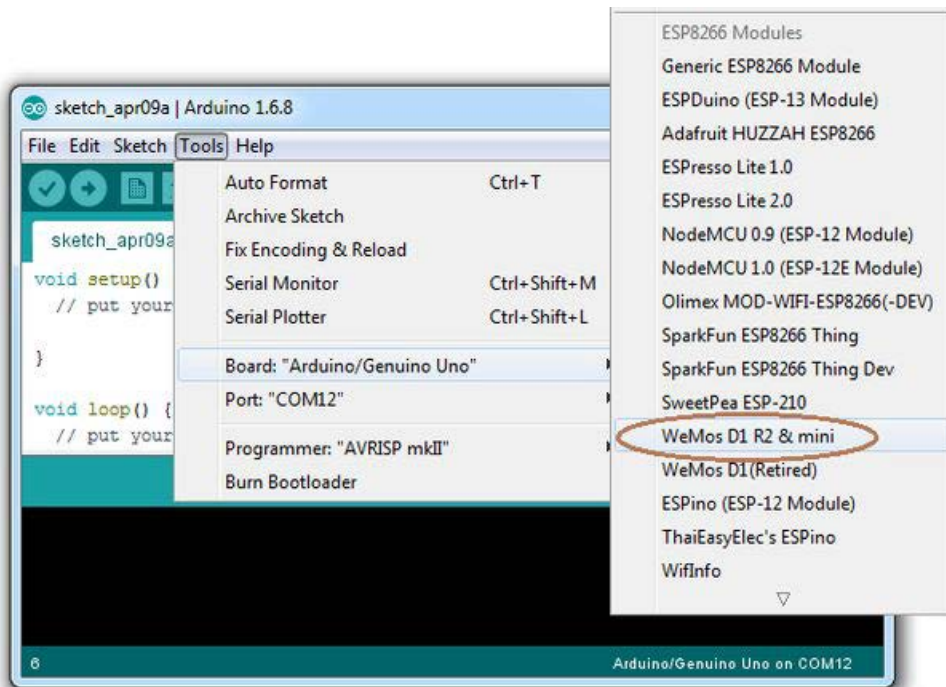
Fill in the Additional Board manager:

http://arduino.esp8266.com/stable/package_esp8266com_index.json

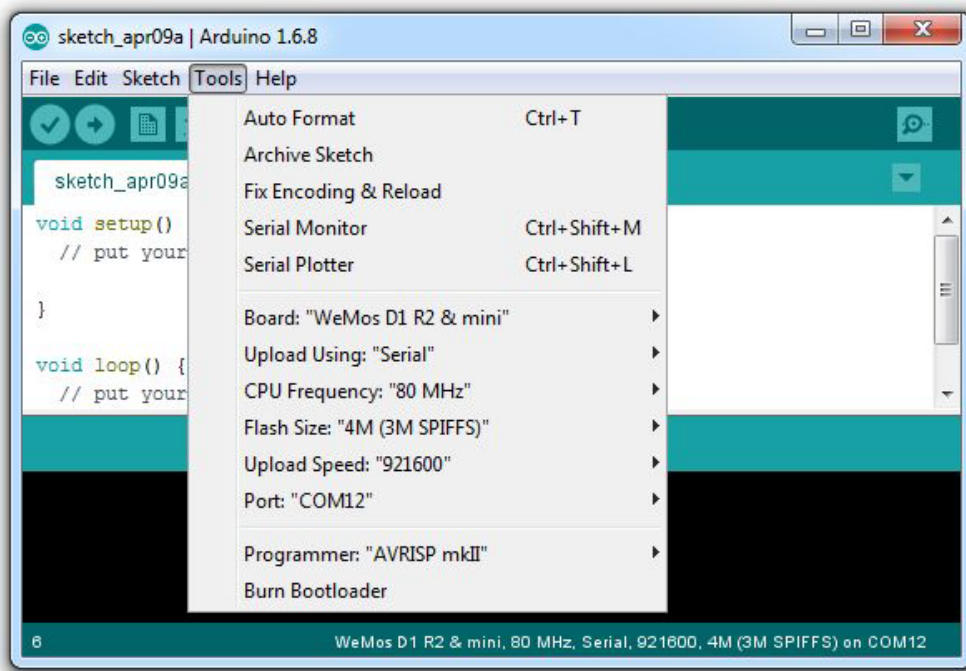


Board Manager

Open Tools | Board:|Boards Manager dialog box. Somewhere in there you will see the entry “Esp8266 of ESP8266 Community version 2.3.0” select that. The Install button will appear, click the Install button. Wait for a while... This process will take some time to download and complete. After the install it is a good idea to shut the Arduino program and restarted it.

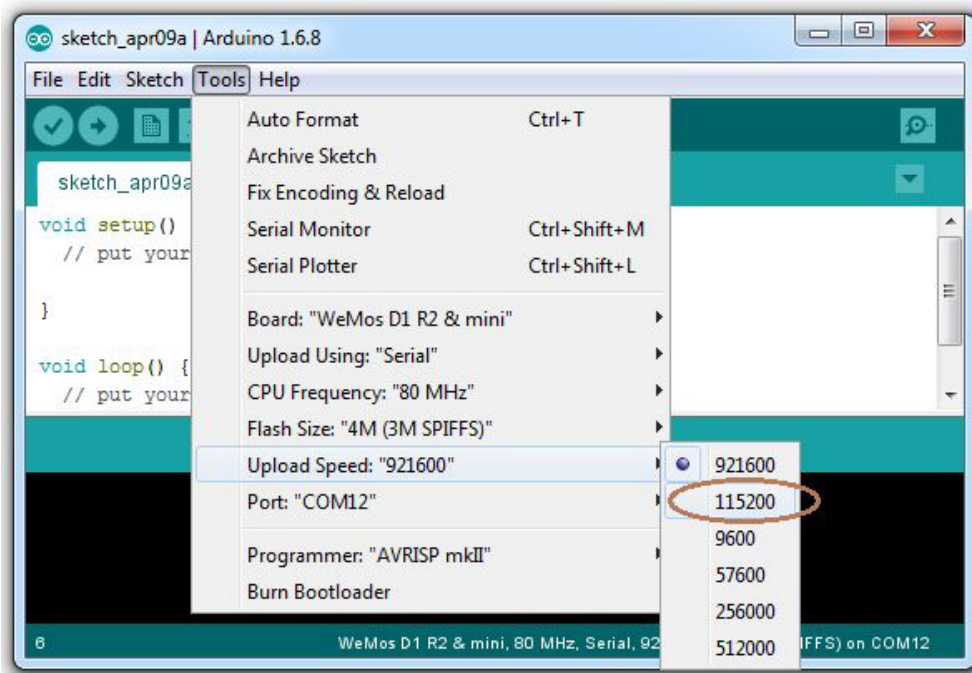


Go to Tools | Board:... scroll way down and select WeMos D1 R2 & mini. The Tools menu changes to the second image shown above.

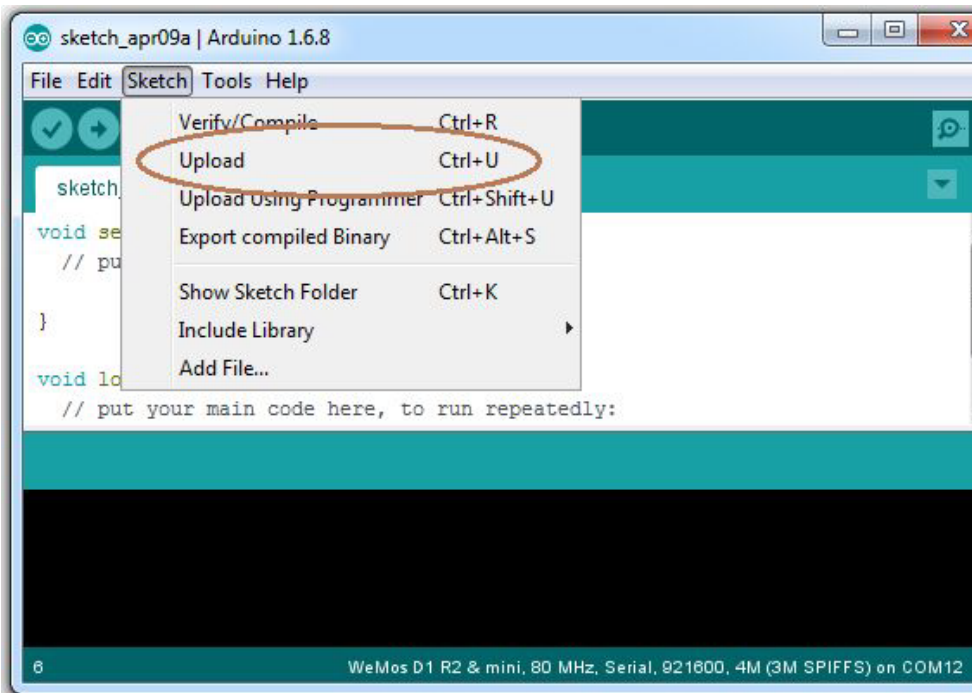


Tools

Determined the COM port WeMos-D1R2 appears on. To select the COM port, go to Tools | Port: and select the COM port, in my case it was COM 12



To select the speed of communication between the WeMos-D1R2 and the computer. Go to Tools | Upload Speed: and select 115200. We are ready to upload our first sketch.



Sketch

Test drive the complete setup using the boiler plate sketch.

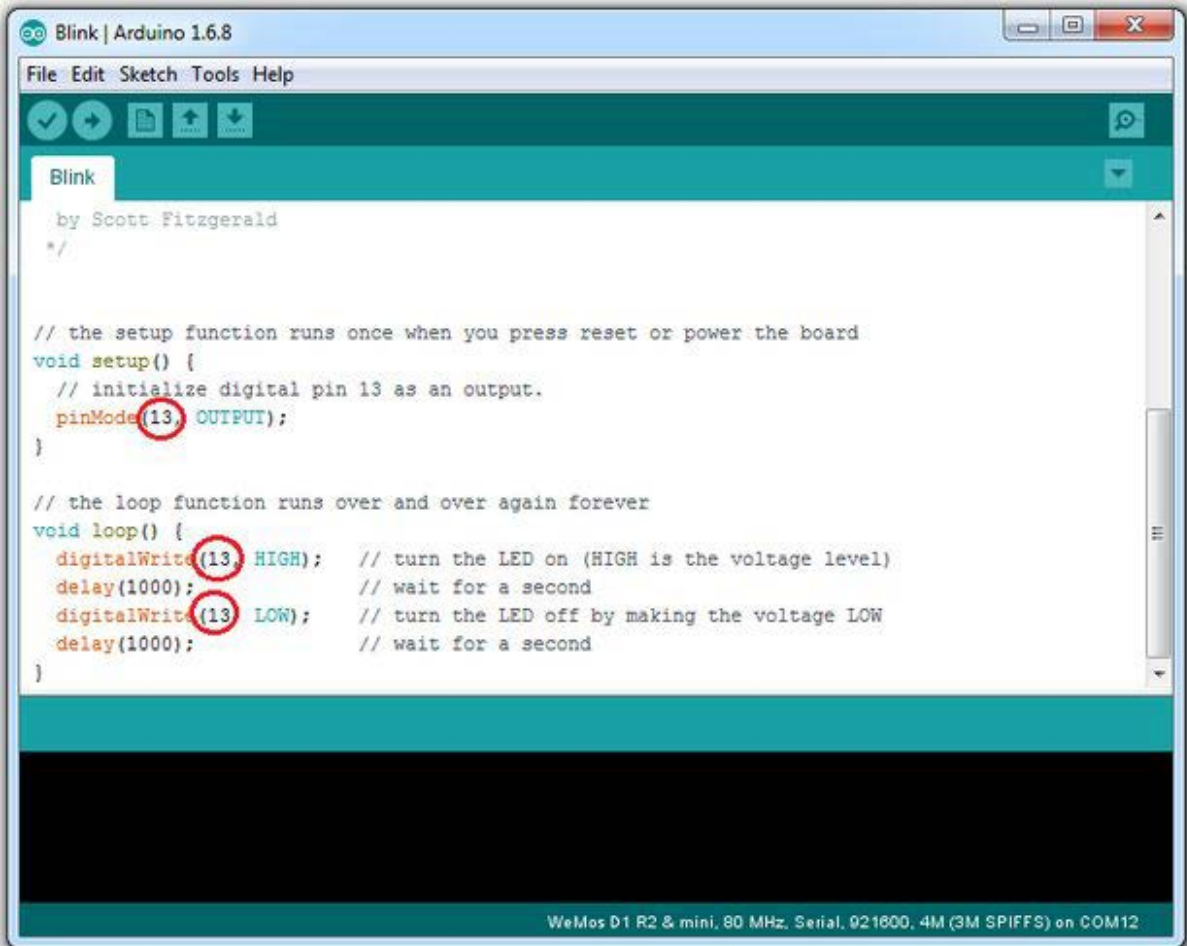
To upload the boiler plate sketch go to Sketch | Upload or use the shortcut key Ctrl+U or click on the right arrow button next to the check mark. A successful compile and upload means all systems are working and we are ready to get to the real stuff.

5.2. Trying the Blink Example

Load the packaged example Blink. To find it go to File | Examples | 01.Basics | Blink. Compile and upload as instructed in the previous step. Nothing happens. :(

The reason nothing happens is because the on-board LED on the WeMos is connected to pin 13. Refer to the table in Step 2. On the WeMos-D1R2 the on-board LED is connected to pin 14.

Replace all instances of pin 13 with pin 14 and save the sketch as WeMosBlink. Viola the blue LED starts blinking.



The screenshot shows the Arduino IDE interface with the 'Blink' sketch loaded. The code is as follows:

```
Arduino 1.6.8
File Edit Sketch Tools Help
Blink
by Scott Fitzgerald
*/

// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin 13 as an output.
  pinMode(13, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(13, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000); // wait for a second
  digitalWrite(13, LOW); // turn the LED off by making the voltage LOW
  delay(1000); // wait for a second
}

WeMos D1 R2 & mini, 80 MHz, Serial, 921600, 4M (3M SPIFFS) on COM12
```

In the original image, the number '13' in the following lines is circled in red:

- `pinMode(13, OUTPUT);`
- `digitalWrite(13, HIGH);`
- `digitalWrite(13, LOW);`

```
WemosBlink.ino | Arduino 1.6.8
File Edit Sketch Tools Help
WemosBlink.ino
// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin 14 as an output.
  pinMode(14, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(14, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000);           // wait for a second
  digitalWrite(14, LOW); // turn the LED off by making the voltage LOW
  delay(1000);           // wait for a second
}
Done Saving.
28 WeMos D1 R2 & mini, 80 MHz, Serial, 921600, 4M (3M SPIFFS) on COM12
```

Disconnect the WeMos-D1R2 from your computer and connect an external 5V power source. I connected it to the power bank that I use to charge my phone. The LEDs will start blinking.

Happy IoTing.