EE333 Introduction to Microcontrollers

Lab#8 Wireless Communication

Allan A. Douglas, Associate Professor

**Lab Overview**

The objective of this lab is to interface the microcontroller with an ESP8266 Serial-to-WiFi device and create a small web server.

**Required Materials**

|  |  |
| --- | --- |
| Breadboard | Capacitors (2x 10uF) |
| Arduino UNO (or Spark Fun RedBoard equivalent) | Arduino ISE Software |
| ATmega328P Microcontroller Datasheet | WiFi Module - ESP8266 |
| Level Shifter (3.3V to 5.0V) | Sparkfun FTDI Basic |
| 3.3V regulator (Use LD33V from “Build your Own Arduino” kit |  |

**Required Work**

The areas of **bold** text below are items required in your Lab Report.

Part1: Soldering

The ESP8266 has 3.3V inputs and outputs, whereas our Atmega328P microcontroller has 5V inputs and outputs. We need add additional circuitry to convert the logic levels so that we do no damage the ESP8266 chip. One way to accomplish this is to use a SparkFun Bi-Directional Logic Level Converter. You will need to solder pins to the Converter so that it can be used in your breadboard.



In order to make the ESP8266 breadboard-friendly, you can either use female to male jumper wires (easier) or solder wires to the pins of the ESP8266.



**Include pictures of the soldered Logic Level Converter and the ESP8266 (if you choose to solder wires to it) in your Lab Report.**

**In your own words, explain why the Logic Level Converter is required and what might occur if the part is not included in the lab.**

**Explain any challenges you had with the soldering tasks in your Lab Report.**

Part2: ESP8266 Firmware Update

Next, you will need to upgrade the firmware on the ESP8266 to v0.9.2.2. See lecture slides for detailed circuit and steps needed.

**Include a screenshot of the nodemcu software while performing the firmware update in your Lab Report.**

**Include a screenshot of the TeraTerm output when you verify the new firmware version with the “AT+GMR” command in your Lab Report.**

Part3: Arduino Sketch for WiFi Server

Remove the FTDI board and connect to Arduino. The following figure shows the complete circuit you need to build for this part of the lab.

TX\_33V



RX\_33V

For this part of the lab you, use the WiFi.ino sketch included in the Canvas folder.

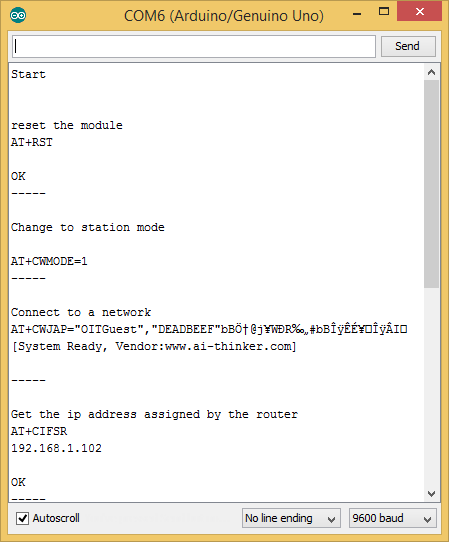
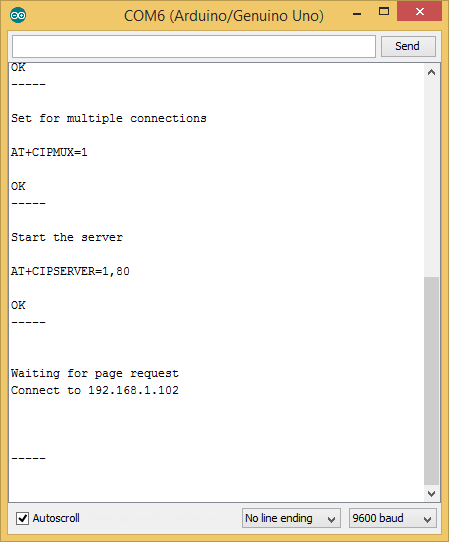
Note, you will need to change the code on line 49 to your wireless network SSID and Password.

Also, this sketch uses microcontroller memory very inefficiently. You will notice a message stating “Low memory available, stability problems may occur.” This is normal and has not caused problems in any testing so far.

**Include a complete description of the sketch in your Lab Report that describes the functionality of each part of the code. Your goal should be to demonstrate a complete understanding of the operation of the code and the AT commands sent to the WiFi module. (Link to command reference in lecture slides)**

**Include an explanation of why the SoftwareSerial.h file is required for this lab in your Lab Report.**

Open the Serial Monitor and set the baud rate for 9600. When you have everything connected properly, you should see the following output:

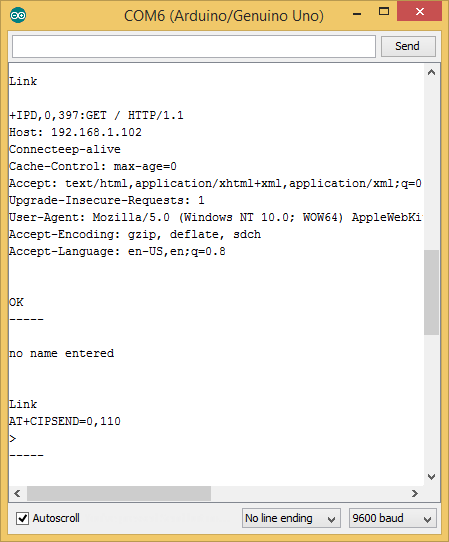
 

The wireless network must be running DHCP in order to assign the WiFi module an IP address. In this example, the IP address that was assigned is 192.168.1.102. This example uses a wireless router with SSID=OITGuest and Password=DEADBEEF.

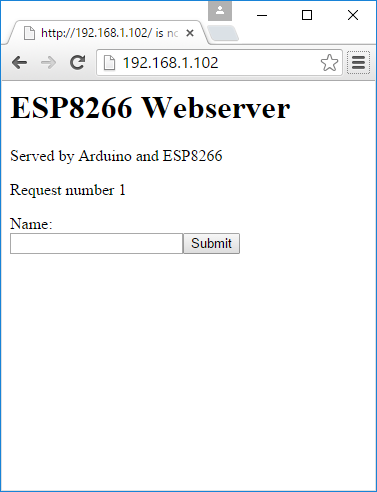
**Include screenshots of this output in your Lab Report.**

At this point, the microcontroller is waiting for a client to connect and request a webpage.

Use a PC that can connect to your wireless network and open a browser to IP address 192.168.1.102. You should see the following output on the Serial Monitor:



The webpage displayed in the browser should look like:



Enter your name in the text entry box and click submit. The microcontroller should generate a new webpage, display your name (text you entered), and the number of times the webpage has been requested. The output will look similar to the following figure.



**Include screenshots of your webpage outputs as shown above in your Lab Report.**

**Explain the HTML commands in the sketch and how the webpages are generated. Explain the strcpy() and strcat() functions. How are the command and html character strings constructed? Explain the difference between sending the espSerial.print(command) and sending the espSerial.print(html).**

**Describe the getReply() function. Explain why this is necessary. Make sure you have fully described what this code does and demonstrated your understanding.**

Here is a link that may be useful during this lab.

<http://www.instructables.com/id/Using-the-ESP8266-module/>