

WiFi Control of 8x8 LED Matrix: A Presentation in the Senior Design Project Course

Cristinel Ababei

Dept. of Electrical and Computer Engineering, Marquette University

Spring 2016

1. Objective

The objective of this exercise is to use an ESP8266 WiFi module connected to the Arduino board to control an 8x8 LED matrix. The controlling commands are sent from an Android app running on an Android device.

Notes:

- *This example is one of several examples that are part of a lab in the COEN-4720 Embedded Systems course. More information, including how one can upgrade the firmware of the ESP8266 module, can be found here: <http://dejazz.com/coen4720/index.html> (see Labs section)*

2. Introduction to WiFi

TODO. See presentation slides.

3. ESP8266 Arduino + 8x8 LED matrix control from Android app

In this example, we control what lighting effect to use to drive an 8x8 LED matrix connected to the Arduino board. We have implemented just two lighting effects to display a heart beating or fading on the 8x8 LED matrix. The controls are sent from a custom Android app to the Arduino board, which has an ESP8266 wifi module attached to it.

Note that this example has the advantage of not requiring any Arduino library. You can implement it in two slightly different ways: one, where your local network name and password are needed (for better security), and two, without any local network name and password (but needs the Android device to connect directly to the IP address assigned to the ESP8266; please read comments inside the Arduino sketch for more details on this).

Main components needed:

- I used the ESP8266 module that I upgraded the firmware to v1.5.0.
- Separate power adapter + 5V to 3.3V breadboard power converter.
- Arduino Uno board.
- 8x8 LED matrix (<http://tronixstuff.com/2013/10/11/tutorial-arduino-max7219-led-display-driver-ic/>)
- Samsung Galaxy S5 phone

Create the experimental setup described by the block diagram in the figure below. Note that we use digital pins 10,11 as RX,TX for the software serial connected to ESP8266 module.

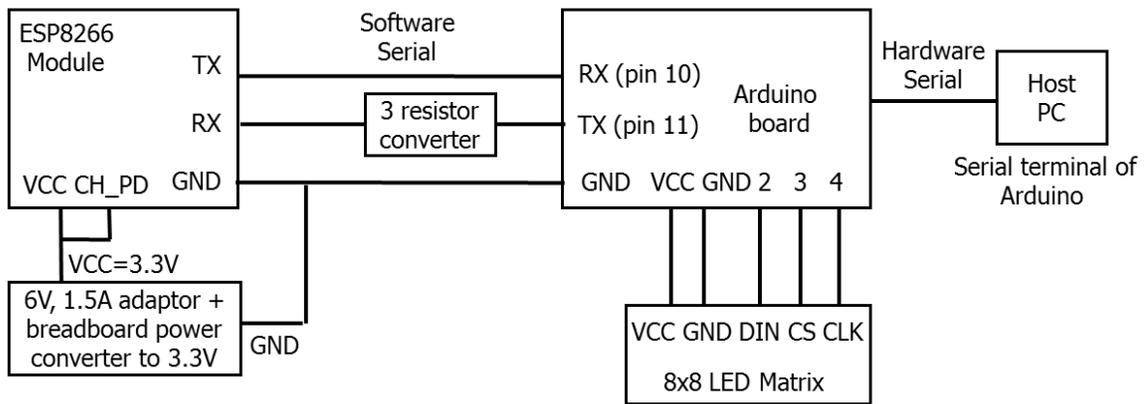


Figure 1 Block diagram of Arduino example.

Once you created all connections, you should have something like in the figure below.

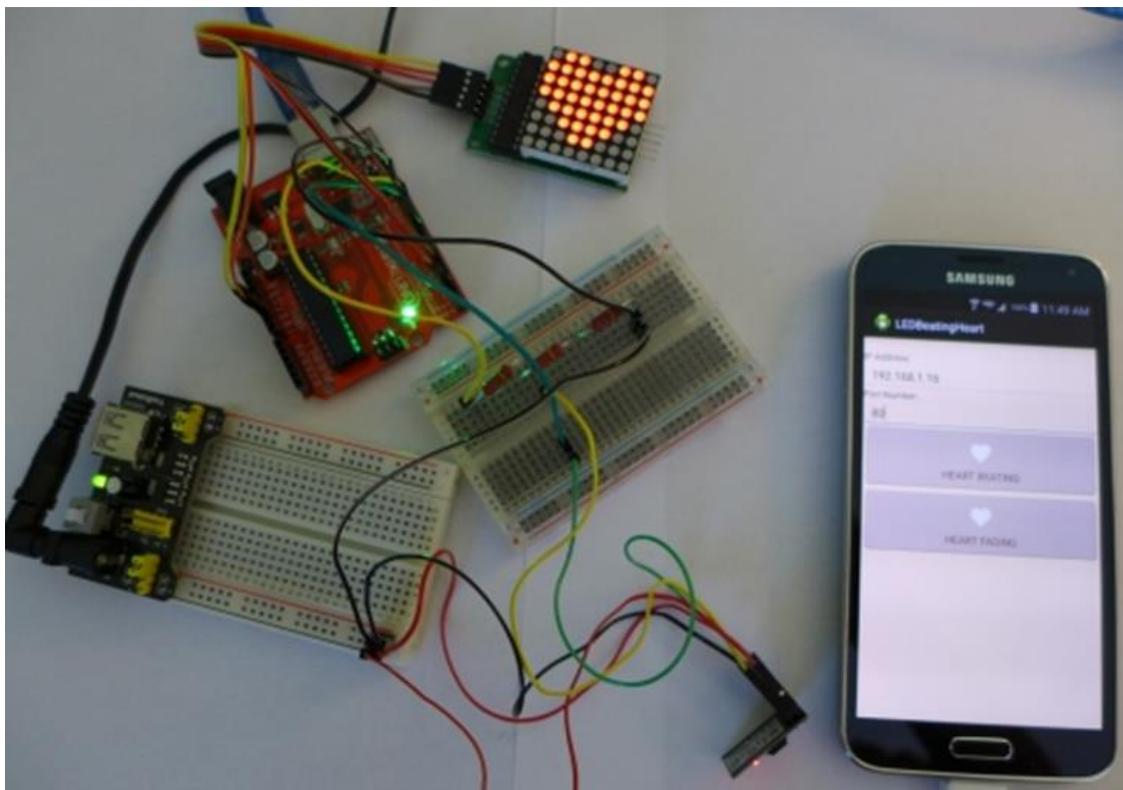


Figure 2 Complete experimental setup.

Use the provided sketch to program the Arduino board. Also, use the provided custom Android app to program your Android device. Here, I assume that you know the basics of how to create a simple Android app and program your device. The UI of the custom app looks like in the figure below.

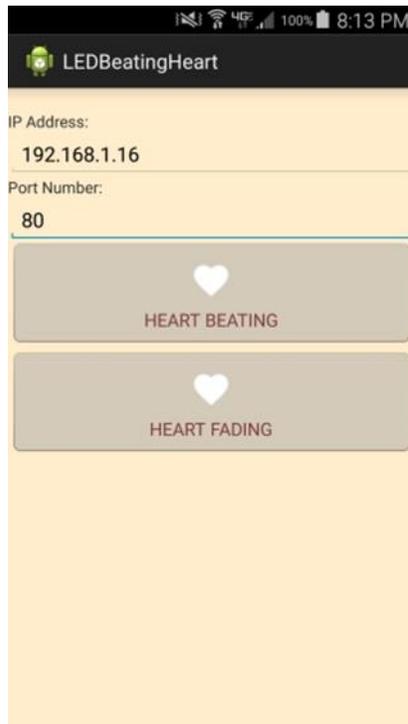


Figure 3 UI of the custom Android app.

Power up, open the Serial terminal of the Arduino IDE, launch the app, and play! Observe operation and comment.

References and Credits

[1] This project was created by adapting the ideas and source code from the projects described in following sources:

<http://yaab-arduino.blogspot.com/2015/05/beating-heart-with-arduino-and-max7219.html>

<http://allaboutee.com/2015/01/20/esp8266-android-application-for-arduino-pin-control/>