Universiteit Antwerpen

ESP8266 Programming

De Borrekens Gauthier

Intro

After receiving the nodeMCU's we ordered, my goal was to understand how to program them, more specifically the WiFi aspect of the programming. I will go over the basic functions and methods used in WiFi programming.

Materials and Methods

I borrowed a breadboard with two cables and a 3.9 kOhm resistor from the university so I could connect the nodeMCU's that arrived from the company "antratek" in 'bootloader' mode (Connect d3 pin with the ground). For the programming of the ESP8266 I used the Arduino IDE Environment.

Results

I was able to connect the nodeMCU to the wifi as a client and setup the nodeMCU as a access point where my pc could connect

Information & conclusion

Setting up

My main source of information came from the "A Beginners guide to the ESP8266" written by Peter P. (P., 2018) where he explains the basic steps with elaborate examples.
First of all, I will go over the steps to install the Arduino IDE.

- Install the Arduino IDE from Windows Store
- Go to File -> Preferences
- Go to Additional Board Manager URL's and add the board URL of the esp8266 (http://arduino.esp8266.com/stable/package_esp8266com_index.json)
- Go to Tools -> Board Manager -> ESP8266 -> Install

Then we need to install the USB to Serial driver and we're done. Now we can start programming and uploading it to the ESP8266. (We need to make sure it is in bootloader mode (By connecting pin D3 with the ground, connected by a 3.9 kOhm resistor to limit the current to 12 mA, which is the maximum recommended current.

Programming

The basics can also be found in aforementioned site "A beginners guide to the ESP8266". The most important very basic one is the Serial.

- Serial: Establish a Serial in/out connection to be able to 'print' to a COM line. First a baudrate has to be set up by the Serial.Begin(baudrate) command. Once this is done, You can just print text to the serial output by Serial.print() or Serial.println();

We want to start programming with WiFi programming, which is included in the <ESP8266WiFi.h> package. Some functions:

- WiFi.begin(ssid, password) : Start a WiFi connection with a WiFi network as parameters. Auto IP assignment
- WiFi.config(ip, dns(optional), gateway(optional), subnet(optional)) configs static ip, dns, gateway and subnet
- WiFi.status() : Returns the status of the WiFi connection. Handy to check if WiFi.begin was successful Examples are WL_CONNECTED, WL_CONNECT_FAILED, WL_DISCONNECTED (Arduino, 2018)
- WiFi.localIP(), WiFi.subnetMask(), WiFi.gatewayIP() return values that are pretty self-explanatory.
- WiFi.scanNetworks() returns a byte with all the networks found.
- WiFi.softAP(ssid, password) : creates an access point with given ssid and password where clients can connect. No internet access is provided.
- WiFiClient : Datatype of a client which can connect to a specific ip address and port of that ip address. A variable of the type WiFiClient has the method “WiFiClient.connect(IPAddress, port);”. Can also .println() to a client.
- Udp.begin(port) : Begins a wifiUDP socket that listens on the given port. Returns a bool.
- Udp.parsePacket() : Checks presence of udp packet and returns the size. Must always be called before .read() !
- Udp.remorteIP() : gets the ip address of a remote connection. Must be called after ParsePacket().
- Udp.remotePort() : gets the port of a remote connection. Must be called after ParsePacket();
- Udp.read(Buffer, size) : reads ‘size’ characters of the incoming packet to the buffer. Arguments are optional
- Udp.beginPacket(Ip, Port) : start a packet on a specific ip address and port
- Udp.write(char data[]) : fill the packet with data
- Udp.endPacket() : stop the packet writing and send it.

Some extra useful thinkgs:

- IPAddress : The data type of an ip address, it is what the WiFi.localIP() returns.
- WiFiUDP : datatype of an UDP connection. “WiFiUDP Udp;” syntax (Arduino, 2018)

REFERENCE LIST

Arduino. (2018, Maart 21). *WiFi Library*. Opgehaald van Arduino.cc: https://www.arduino.cc/en/Reference/WiFi

Arduino. (2018, Maart 21). *WiFi.status*. Opgehaald van Arduino.cc: https://www.arduino.cc/en/Reference/WiFiStatus

P., P. (2018, Maart 21). *A beginners guide to the ESP8266*. Opgehaald van github: https://tttapa.github.io/ESP8266/Chap01%20-%20ESP8266.html