Impact Factor (SJIF): 3.632

International Journal of Advance Research in Engineering, Science & Technology

e-ISSN: 2393-9877, p-ISSN: 2394-2444

Volume 3, Issue 11, November-2016

ESP 128266 exploration

Vidhi B patel[1], Chandresh Parekh[2], Malav Patel[3]

[1]*Cyber Security Dept., Raksha-Shakti University*
[2]*Cyber Security Dept., Raksha-Shakti University*
[3]*Embedded Firmware Developer,Teksun Microsys Pvt Ltd*

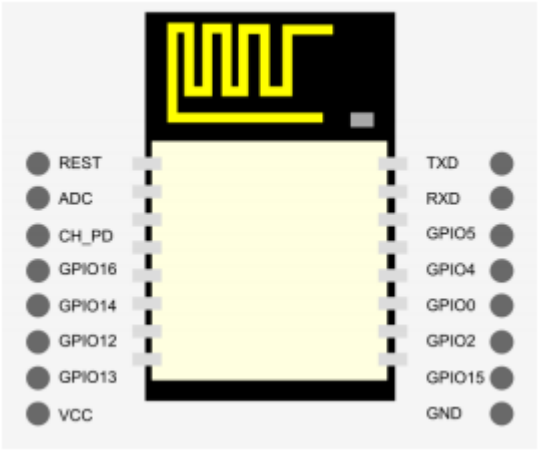
Abstract *— The ESP8266 is the name of a micro controller designed by Espressif Systems. Espressif is a Chinese company based out of Shanghai. The ESP8266 advertises itself as a self-contained WiFi networking solution offering itself as a bridge from existing micro controller to WiFi and is also capable of running self contained applications. the number of GPIO pins exposed, the amount of flash memory provided, the style of connector pins and various other considerations related to construction. From a programming perspective, they are all the same. A micro controller is an integrated circuit that is capable of running programs. There are many instances of those on the market today from a variety of manufacturers. The prices of these micro controllers keeps falling. These micro controller and architectures are great and there will always be a place for them. However, there is a catch … and that is networking. These devices have an amazing set of capabilities including direct electrical inputs and outputs (GPIOs) and support for a variety of protocols including SPI, I2C, UART and more, however, none of them so far come with wireless networking included.*

Keywords- ESP8266; **Wi-Fi; AT-Commands; Access point**

I INTRODUCTION

ESP8266 Modules the ESP8266 integrated circuit comes in a small package, maybe five millimeters square. The ESP8266 is designed to be used with a partner memory module and this is most commonly flash memory. There are a variety of board styles available. The two that I am going to focus on have been given the names ESP-1 and ESP-12. It is important to note that there is only one ESP8266 processor and it is this processor that is found on ALL breakout boards. ESP-12 is the current most popular and flexible configuration available today. It exposes the most GPIO pins for use. The basic ESP-12 module really needs its own expander module to make it breadboard and 0.1" strip board friendly.

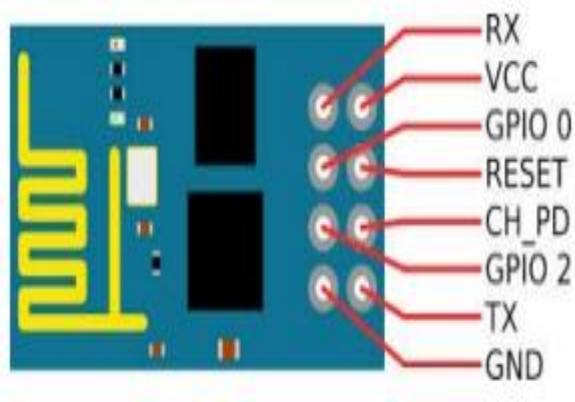
Figure 1 ESP8266 module



Name	Description
VCC	3.3V.
GPIO 13	Also used for SPI MOSI.
GPIO 12	Also used for SPI MISO.
GPIO 14	Also used for SPI Clock.
GPIO 16	
CH_PD	Chip enable. Should be high for normal operation. • 0 – Disabled • 1 – Enabled
ADC	
REST	External reset. • 0 – Reset • 1 – Normal
TXD	UART 0 transmit.
RXD	UART 0 Receive.
GPIO 4	Regular GPIO.
GPIO 5	Regular GPIO.
GPIO 0	Should be **high** on boot, low for flash update.
GPIO 2	Should be **high** on boot.
GND	Ground.

Figure 2 ESP12 module pin discription

ESP-1 board is an ESP8266 on an 8 pin board. It is not at all breadboard friendly but fortunately we can make adapters for it extremely easily.



Function	Color	Description
TX		Transmit
RX		Receive. **Always** used a level converter for incoming data. This device is **not** 5V tolerant.
CH_PD		Chip enable. Should be high for normal operation. • 0 – Disabled • 1 – Enabled
RST		External reset. • 0 – Reset • 1 – Normal
GPIO 0		Should be **high** on boot, low for flash update.
GPIO 2		Should be **high** on boot.
VCC		3.3V
GND		Ground

Figure 3 ESP1 module pin description

II Connecting to the ESP8266

The ESP8266 is a Wi-Fi device and hence we will eventually connect to it using Wi-Fi protocols but some bootstrapping is required first. The device doesn't know what network to connect to, which password to use and other necessary parameters. This of course assumes we are connecting as a station, if we wish the device to be an access point or we wish to load our own applications into it, the story gets deeper. This implies that there is a some way to interact with the device other than Wi-Fi. The ESP8266 has a dedicated UART interface with pins labeled TX and RX. The TX pin is the ESP8266 transmission (outbound from ESP8266) and the RX pin is used to receive data (inbound into the ESP8266). These pins can be connected to a UART (serial) partner. This is used extensively when working with the AT commands. A second purpose of the UART is to receive binary data used to "flash" the flash memory of the device to record new applications for execution. When the ESP8266 is performing the role of an access point, it is likely that you will want it to also behave as a DHCP server so that connecting stations will be able to be automatically assigned IP addresses and learn their subnet masks and gateways. TCP/IP is the network protocol that is used on the Internet. It is the protocol that the ESP8266 natively understands and uses with WiFi as the transport.

III Wi-Fi Theory

When working with a WiFi oriented device, it is important that we have at least some understanding of the concepts related to WiFi. At a high level, WiFi is the ability to participate in TCP/IP connections over a wireless communication link. WiFi is specifically the set of protocols described in the IEEE 802.11 Wireless LAN architecture. A device called a Wireless Access Point (access point or AP) acts as the hub of all communications. Typically it is connected to as TCP/IP router to the rest of the TCP/IP network. WiFi connections are then formed to the access point and TCP/IP traffic flows through the access point to the Internet.

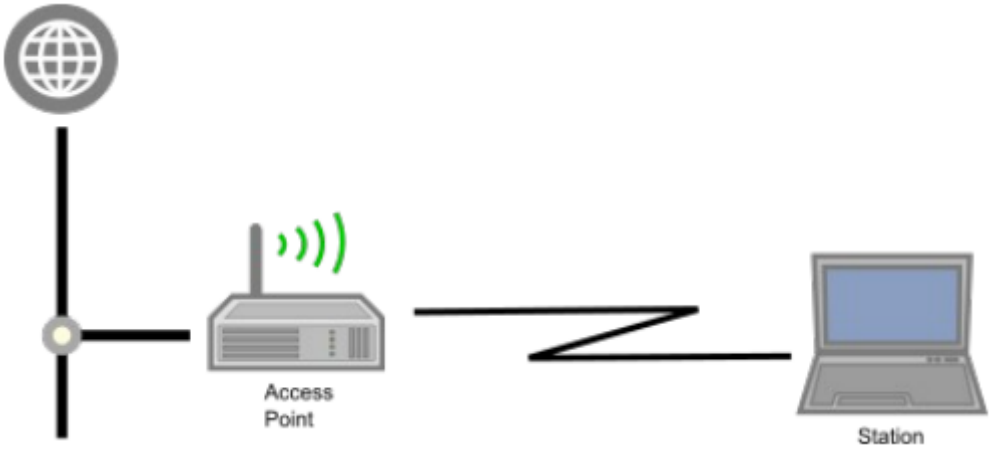


Figure 4 the devices that connect to the access points

An ESP8266 device can play the role of an Access Point, a Station or both at the same time. The access point also has a network connection to the Internet and acts as a bridge between the wireless network and the broader TCP/IP network that is the Internet. A collection of stations that wish to communicate with each other is termed a Basic Service Set (BSS). The common configuration is what is known as an Infrastructure BSS. In this mode, all communications inbound and outbound from an individual station are routed through the access point. A station may only be associated with a single access point at any one time. The network has a unique identifier called the MAC address. This is a 48bit value. When we have multiple access points within wireless range, the station needs to know with which one to connect. Each access point has a network identifier called the BSSID (or more commonly just SSID). SSID is service set identifier. It is a 32 character value that represents the target of packets of information sent over the network.

IV AT Command Programming

The quickest and easiest way to get started with an ESP8266 is to access it via the AT command interface. When we think about an ESP8266 device we find that it has a built in UART (Serial) connection. This means that it can both send and receive data using the UART protocol. We also know that the device can communicate with WiFi. This would then allow us to use the ESP8266 without ever having to know the programming languages that are native to the device. This is exactly what a program that has so far been found to be pre-installed on the ESP8266 does for us. The program is called the "AT command processor" named after the format of the commands sent through the serial link. All "AT" instructions end with the "\r\n" pair.

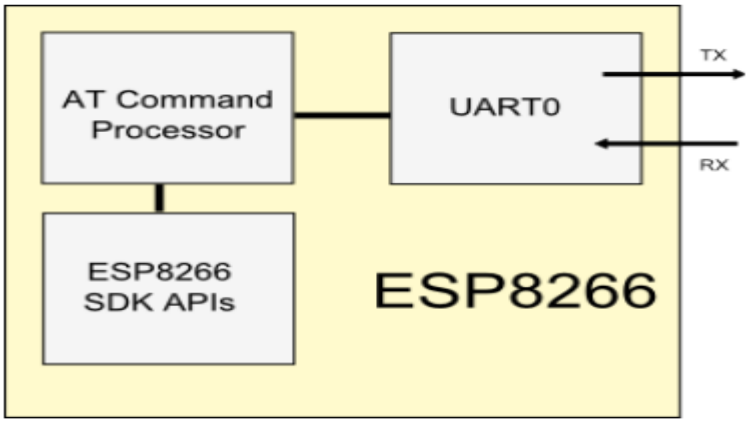


Figure 5 At command process

The services of the ESP8266 as a client and the ESP8266 as a server capable of servicing those commands as a server, then the client sends strings of characters through the UART connection to the server and server responds with the outcome.

The following command mostly use for Wi-Fi connection:
1 AT
2 AT+WD
3 AT+NSET = ipadress
4 AT+CWMODE_CUR=<mode>
5 AT+RST

V RESULT AND CONCLUSTION

I learn the different paper and understand by ESP8266 modules working for Wi-Fi technology.ESP128266 Wi-Fi module using AT command for connection. This module use for different protocol like UART, DHCP and TCP/TP. Also understanding Wi-Fi technology and its working with different protocol.

ACKNOWLEDGMENT

We take this opportunity to express our deepest gratitude and appreciation to all those who have helped us directly or indirectly towards the successful completion of this paper

REFERENCES

[1] NEIL KOLBAN, "book of ESP8266", USA, pp. 19-40, September 2015
[2] http://bbs.espressif.com/viewforum.php?f=5
[3] http://randomnerdtutorials.com/getting-started-with-esp8266-wifi-transceiver-review/