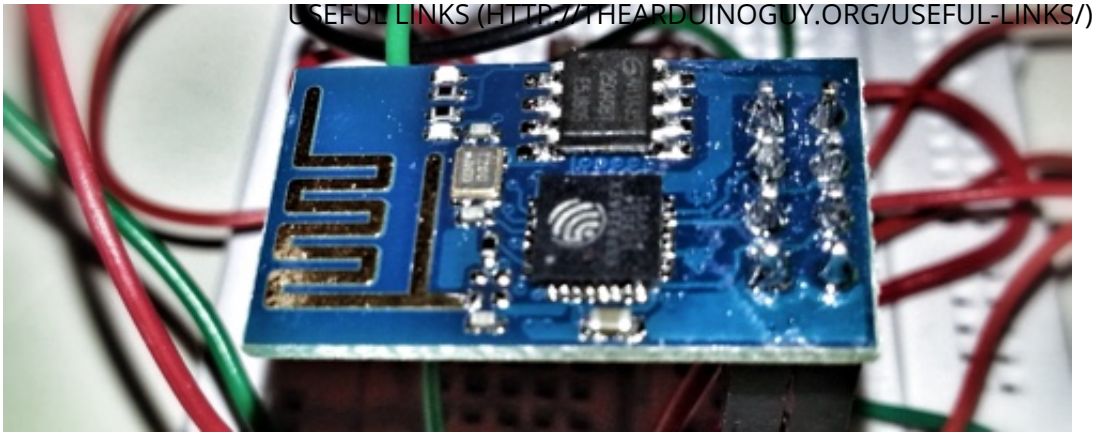
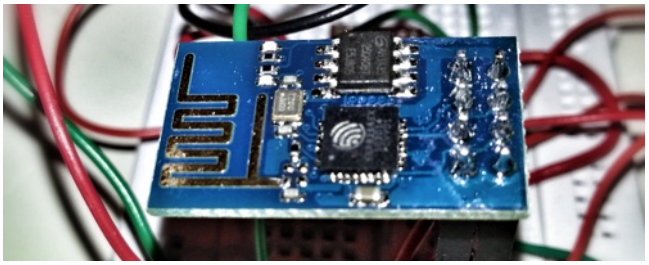


Using an ESP8266 as a time source (Part 1)

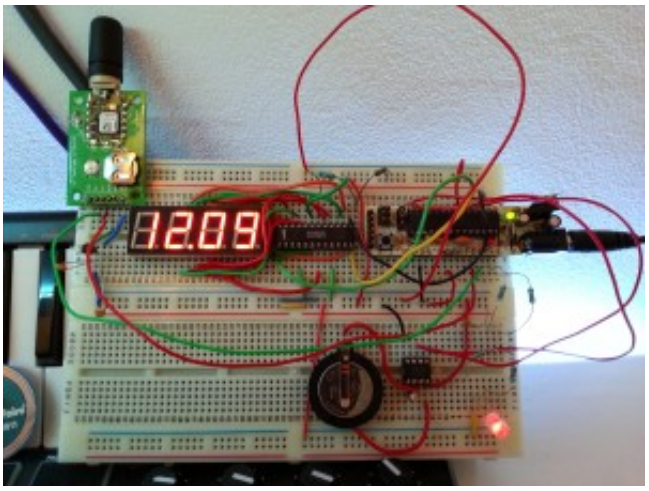
February 25, 2015 (http://thearduinoguy.org/using-an-esp8266-as-a-time-source/) Mike McRoberts
(http://thearduinoguy.org/author/mikethearduinoguy-org/)

So i've obtained some ESP8266 WiFi modules lately and have been having a play with them. If you
heard of the ESP8266 they are tiny serial controlled WiFi modules that are about £5 each and ena
easy and cheap method of connecting an electronics project to your home network or the intern
also have an onboard chip (SoC) that can be accessed and programmed although i've not gone th
and have simply used the serial interface so far.



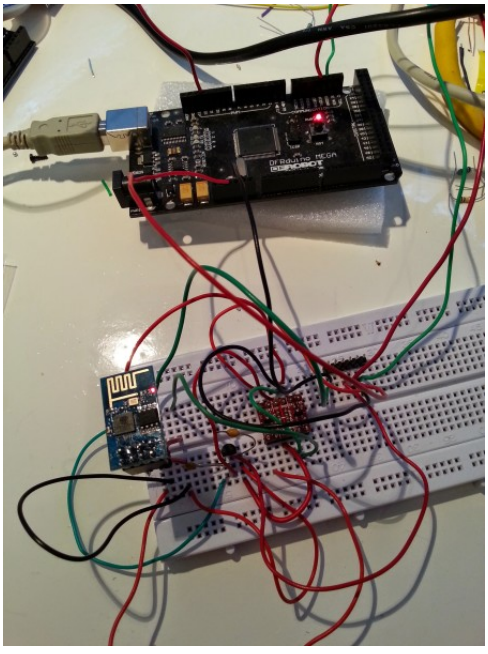
(http://thearduinoguy.org/wp-content/uploads/2015/02/ESP8266.jpg)

The reason I got one was to try and obtain an a time signal from the internet using NTP or som method. I had built a digital clock circuit with a segment display and was trying to find ways of ensuring the time was always accurate. I tried out DCF and MSF time receiver units and found that on their own they were able to successfully rece time signal, once added to the rest of the circuit they were hyper-sensitive to the tiniest bit of ele noise and failed to sync.



Next I tried a GPS module. GPS modules return accurate time and date in the NMEA string and are a great way of obtaining accurate time. How they need a view of the sky to receive the satelli signals and so indoors usually only work when window. This is not much good for my clock pro don't want the clock near a window all the time

So the next idea was to connect to the internet using NTP or some other method obtain time th have several ethernet shields for my Arduino's have some Arduino Yun's but this was complete overkill for a clock. I then heard about the ESP82 modules and decided to purchase one. AT £4.95 inc. delivery from eBay it was worth the risk.

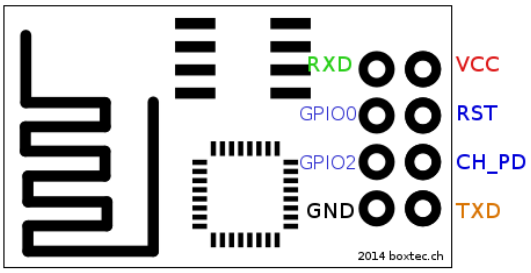


(http://thearduinoguy.org/wp-content/uploads/2015/02/IMG_20150225_115737.jpg)Since been playing around with the module and learning how to first had to update the firmware on them which was a pain ass and at one point I thought i'd bricked the unit. In the en able to update both the firmware and the AT command set am now running at version 0020000903_NTP.

The module hooks up to the Arduino via one of the serial p used a Mega 1280 as it has 4 serial ports and enables me t for comms with the ESP8266 and the other to send debug messages to the serial monitor window of the Arduino IDE. need to make sure that your baud rate matches that of you (Mine was set to 115,200 baud) and you are then able to se commands to the device to control it.

The pins for the device are as in the diagram below.

The GND pin goes to ground, VCC goes to 3.3v (NOT 5v!) and the RX and TX pins go to the opposi TX pins on the appropriate serial port of your Arduino. The CH_PD pin need sto be held high so c to VCC also. The other pins are only used for special functions such as upgrading the firmware w not go into here.



(http://thearduinoguy.org/wp-content/uploads/2015/02/esp8266_pinout.png)

Thanks to Boxtec.com

Note that the ESP8266 draws a considerable mo current and it is recommended that it is powere separate power supply. This makes it an interes choice for battery powered projects, though it c a sleep mode to conserve power. It has been kr draw up to 700mA. Your FTDI cable or Arduino be able to provide this much and if the device k resetting this is probably the cause. Where poss power it with a dedicated PSU.

To send a command to the device you simply us `Serial.println` and send your command, e.g

```
Serial1.println("AT+RST");
```

To reset the device you send

```
AT+RST
```

and the device will reset and send back a reset message. With the latest firmware the reset mess my opinion, needlessly large and is

```
 ets Jan  8 2013,rst cause:4, boot mode:(3,7)

wdt reset
load 0x40100000, len 25816, room 16
tail 8
chksum 0x0d
load 0x3ffe8000, len 3476, room 0
tail 4
chksum 0x8b
load 0x3ffe8da0, len 7376, room 4
tail 12
chksum 0x33
csum 0x33
rl
ready
```

Essentially what you are looking for is the 'ready' message which shows the device has been rese now ready for other commands.

You can set the device in one of 3 modes using the AT+CWMODE command, as a client as an acce or as both. This means you can actually connect other devices directly to it when in AP mode, suc phone. For my use I am using mode 1 so the device is a client.

```
Serial1.println("AT+CWMODE=1");
```

Then you need to connect the device to your WiFi router withthe CWJAP command by feeding it y and Password thus

```
#define SSID "YourSSIDHere"
#define PASS "YourPasswordHere"
```

```
#define ESP8266 Serial1   // Use Serial1 to talk to ESP8266 (Mega)

String cmd;   // AT command string

cmd = "AT+CWJAP=\""; // Join the Access Point
cmd += SSID;
cmd += "\",\"";
cmd += PASS;
cmd += "\"";
ESP8266.println(cmd); //send command to device
```

You can find out the IP and MAC address of the device using

```
ESP8266.println("AT+CIFSR");
```

There will be two sets, one for STA (Station mode) and one for AP (Access point) in Mode 3 and or Modes 1 and 2.

```
AT+CIFSR
+CIFSR:APIP,"192.168.4.1"
+CIFSR:APMAC,"1a:ef:34:f8:46:6c"
+CIFSR:STAIP,"192.168.0.8"
+CIFSR:STAMAC,"18:da:13:9f:12:7b"
```

If you want to know your firmware version you can use the GMR command

```
ESP8266.println("AT+GMR");
```

Which will return something like

```
AT+GMR
0020000903_NTP
```

Now to obtain the time there are two methods. The first is by using the NTP command built into AT command firmware.

```
AT+CIPNTP=0
```

The format is AT+CIPNTP=<offset from GMT>. Then to query the time from the NTP server simply

```
AT+CIPNTP?
```

This works sometimes and will return the current date, time and GMT offset like this

```
AT+CIPNTP?
Time: 22:22:42 12/02/2014 GMT+02
```

However, that is only if it works. Most of the time NTP severs are notoriously overloaded and bus you will be unable to get the time. On those occasions you will receive this message

```
Sntp initializing TZ: GMT00...
Sntp initializing...
pool.ntp.orgsntp_request: Waiting for server address to be resolved.
sntp_dns_found: Server address resolved, sending request
sntp_send_request: Sending request to server
sntp_process: 1424870150, ussntp_recv: Scheduled next time request: 3600000 ms
```

So, I had to find another way of getting the time. Thanks to some posts on Pete Scargill's blog I wa use a method similar to one he had used with a HTTP GET command. You simply upload a text fil server and then request the file using GET commands. In return (when it works) you get back the

time in the HTTP header.

This time we need to make a TCP connection to the relevant IP address of the server and send a request with the location of the file.

```
#define DST_IP "173.254.30.60" //my web site, replace with yours
cmd = "AT+CIPSTART=\"TCP\",\"";
cmd += DST_IP;
cmd += "\",80";
ESP8266.println(cmd); //send command to device
cmd = "GET /Test.txt HTTP/1.1\r\n\r\n"; //construct http GET request
cmd += "Host: thearduinoguy.org\r\n\r\n"; //test file on my web
ESP8266.print("AT+CIPSEND=");
ESP8266.println(cmd.length()); //esp8266 needs to know message length of incoming messc
ESP8266.println(cmd);
ESP8266.println("AT+CIPCLOSE"); // Close the TCP connection
```

The relevant commands are the CIPSTART, CIPSEND and CIPCLOSE commands.

If you are lucky you will get back a response such as this

```
Date: Wed, 25 Feb 2015 13:34:09 GMT
```

A lot of the time, for reasons I have yet to fathom, you get a busy s... message by return and othe messages. As far as I can make out this is a bug in the firmware so will hopefully be ironed out ev

```
AT+CIPSTART="TCP","173.254.30.60",80
CONNECT

OK

AT+CIPSEND=53
>

GET /Test.txt HTTP/1.0 Host: thearduinoguy.org busy s...
AT+CIPCLOSE busy s...

SEND OK
```

However, the date and time string is returned often enough successfully that it can be parsed an sync the time on an RTC chip for your clock project. Obviously, don't use my web page, use your though your welcome to use it for testing purposes only.

The whole test code for this is below. Note that this is not complete and is mainly code for sendir commands and seeing the response. The relevant timeout is allowed for a response after each c (some take longer than others).

Once the code is complete I can add it to the code for the clock project. I need to parse the returr date/time header and then inject that latest time into the RTC chip on the clock circuit.

Message me if you need any help with any of the above.

Mike

```
1  #define BUFFER 1024 // size of the buffer in bytes
2
3  #define SSID   "YourSSIDHere"
4  #define PASS   "YourPasswordHere"
```

```
#define DST_IP "173.254.30.60" //my web site, replace with yours

char buffer[BUFFER];  // buffer for returned messages
String cmd;   // AT command string

#define DEBUG Serial      // Send debug messages to serial monitor
#define ESP8266 Serial1   // Use Serial1 to talk to ESP8266 (Mega)

// By default we expect OK\r\n back
char OK[] = "OK\r\n";

//====================================================================
void setup() {
  delay(1000);
  ESP8266.begin(115200); // Start ESP8266 comms
  DEBUG.begin(115200); // Start seriam monitor comms for debug messages
  DEBUG.println("Initialising...");

  initESP8266(); // Initialise the ESP8266 module
}

//====================================================================
byte waitForResponse(int timeout, char* term=OK) {
  unsigned long t=millis();
  bool found=false;
  int i=0;
  int len=strlen(term);
  // wait for at most timeout milliseconds
  // or if OK\r\n is found
  while(millis()<(t+timeout)) {
    if(ESP8266.available()) {
      buffer[i++]=ESP8266.read();
      if(i>=len) {
        if(strncmp(buffer+i-len, term, len)==0) {
          found=true;
          break;
        }
      }
    }
  }
  buffer[i]=0;
  DEBUG.println(buffer);
  delay(100);
  return found;
}

//====================================================================
void initESP8266() {
  ESP8266.println("AT+RST"); // Reset the ESP8266
  waitForResponse(10000);

  ESP8266.println("AT+CWMODE=1"); // Set Mode 1 (STA = Client)
  waitForResponse(5000);

  cmd = "AT+CWJAP=\""; // Join the Access Point
  cmd += SSID;
  cmd += "\",\"";
  cmd += PASS;
  cmd += "\"";
  ESP8266.println(cmd);  //send command to device
  waitForResponse(10000);

  ESP8266.println("AT+CIPMUX=0"); // Single connection
  waitForResponse(5000);

```

```
70    ESP8266.println("AT+CIPMODE=0"); // Normal mode
71    waitForResponse(5000);
72
73    ESP8266.println("AT+CIFSR"); // Print the IP address of module
74    waitForResponse(5000);
75
76    ESP8266.println("AT+GMR"); // Print the Firmware version
77    waitForResponse(5000);
78
79    DEBUG.println("============================");
80  }
81
82  //=========================================================================
83  void getTheTime() {
84    cmd = "AT+CIPSTART=\"TCP\",\"";
85    cmd += DST_IP;
86    cmd += "\",80";
87    ESP8266.println(cmd);  //send command to device
88    waitForResponse(10000);
89
90    cmd =  "GET /Test.txt HTTP/1.0\r\n\r\n";  //construct http GET request
91    cmd += "Host: thearduinoguy.org\r\n\r\n";        //test file on my web
92    ESP8266.print("AT+CIPSEND=");
93    ESP8266.println(cmd.length());  //esp8266 needs to know message length of incomi
94    waitForResponse(10000);
95
96    ESP8266.println(cmd);
97    waitForResponse(60000);
98
99    ESP8266.println("AT+CIPCLOSE"); // Close the TCP connection
100   waitForResponse(5000);
101
102   DEBUG.println("----------------------------");
103 }
104
105 //=========================================================================
106 void loop() {
107   getTheTime();
108   delay(30000); // Check for time once a minute
109 }
```

Like this:

★ Like

Be the first to like this.

(/#facebook) (/#twitter) (/#google_plus) (https://www.addtoany.com/share#url=a-time-source%2F&title=Using%20an%20ESP826

ESP8266 (http://thearduinoguy.org/category/esp8266/) permalink (http://thearduinoguy.org/using esp8266-as-a-time-source/)

← ATMEGA328P POWER SAVING TECHNIQUES (HTTP://THEARDUINOGUY.ORG/ATMEGA328P-POWER-SAVING-TECHNIQUES/)

USING AN ESP8266 AS A TIME SOURCE → (HTTP://THEARDUINOGUY.ORG/USI ESP8266-AS-A-TIME-SOURCE-PART-2/)

13 thoughts on "Using an ESP8266 as a time source (Part 1)"

chicthomson (http://chicthomson.wordpress.com) says:
February 25, 2015 at 2:40 pm (http://thearduinoguy.org/using-an-esp8266-as-a-time-source/#comme

MIke
Worth adding that the ESP8266 will draw significant current at 3V3, (mine takes >700mA).
This can cause your 'duino, or your USB port to bork if you don't power it seperately.

Chic

Reply (http://thearduinoguy.org/using-an-esp8266-as-a-time-source/?replytocom=

mike@thearduinoguy.org says:
February 25, 2015 at 3:10 pm (http://thearduinoguy.org/using-an-esp8266-as-a-time-source/#com

Yeah good point. I'll add that in. Thanks.

Reply (http://thearduinoguy.org/using-an-esp8266-as-a-time-source/?replytocom=

Hardik says:
May 30, 2015 at 11:22 am (http://thearduinoguy.org/using-an-esp8266-as-a-time-source/#comment-1

Hello.....if i test your script that is http://thearduinoguy.org/Test.txt (http://thearduinoguy.org/Test.t bowser then i gets response like :This is a test

so if i send same request using ESP8266 using GET method then should i get same kind of response terminal or no response becuase when i tested it with ESP8266 i could not get any response...

where is my mistake?

Reply (http://thearduinoguy.org/using-an-esp8266-as-a-time-source/?replytocom=

Mike McRoberts says:
June 2, 2015 at 5:15 pm (http://thearduinoguy.org/using-an-esp8266-as-a-time-source/#comment

I would need to take a look at your code to comment.

Reply (http://thearduinoguy.org/using-an-esp8266-as-a-time-source/?replytocom=

Les_Phil says:
January 12, 2016 at 7:56 pm (http://thearduinoguy.org/using-an-esp8266-as-a-time-source/#commen

can you link the firmware of the esp8266 please?

Reply (http://thearduinoguy.org/using-an-esp8266-as-a-time-source/?replytocom=

Les_Phil says:
January 12, 2016 at 8:05 pm (http://thearduinoguy.org/using-an-esp8266-as-a-time-source/#comn

Nevermind

Reply (http://thearduinoguy.org/using-an-esp8266-as-a-time-source/?replytocom=

Phumlani (http://facebook) says:
February 18, 2016 at 9:36 pm (http://thearduinoguy.org/using-an-esp8266-as-a-time-source/#comme

can you please mike show me the schematic diagram of the clock

Reply (http://thearduinoguy.org/using-an-esp8266-as-a-time-source/?replytocom=

L.K. says:
April 11, 2016 at 3:34 pm (http://thearduinoguy.org/using-an-esp8266-as-a-time-source/#comment-9

173.254.30.60 is a personal web server you run?

I have a hosted website, but I don't have an IP address for it.
I only have the URL.
Can I still use the HTTP GET method to get the NTP time?

Reply (http://thearduinoguy.org/using-an-esp8266-as-a-time-source/?replytocom=

Mike McRoberts says:
April 13, 2016 at 2:28 pm (http://thearduinoguy.org/using-an-esp8266-as-a-time-source/#commer

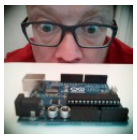
Hi, yes it is a personal web server with the time file included. You could just as easily get the time header of any small website. HTTP time and NTP time are two seperate things.

Reply (http://thearduinoguy.org/using-an-esp8266-as-a-time-source/?replytocom=

L.K. says:
April 14, 2016 at 12:38 am (http://thearduinoguy.org/using-an-esp8266-as-a-time-source/#con

Do you find that HTTP time is adequate for your clock?

Reply (http://thearduinoguy.org/using-an-esp8266-as-a-time-source/?replytocom=



Mike McRoberts says:
April 22, 2016 at 12:12 pm (http://thearduinoguy.org/using-an-esp8266-as-a-time-source/#100)

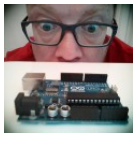
Yes it works pretty well. You only have to update the time about once a week or even once anyway so it's no big deal.

Reply (http://thearduinoguy.org/using-an-esp8266-as-a-time-source/?replytocom=1

L.K> says:
April 19, 2016 at 11:43 pm (http://thearduinoguy.org/using-an-esp8266-as-a-time-source/#con

How does your text.txt file get updated with the correct time?

Reply (http://thearduinoguy.org/using-an-esp8266-as-a-time-source/?replytocom=



Mike McRoberts says:
April 22, 2016 at 12:13 pm (http://thearduinoguy.org/using-an-esp8266-as-a-time-source/#101)

It doesn't. When a HTML page is requested over HTTP the header of the page includes the time it was requested. With a tiny text file this will be done very quicky and hence is quick do a time sync.

Reply (http://thearduinoguy.org/using-an-esp8266-as-a-time-source/?replytocom=1

Leave a Reply

Enter your comment here...

Search

Search ...

Recent Posts

- Radio Controlled LED Sequencer (http://thearduinoguy.org/radio-controlled-led-sequencer/)
- UAV Project (http://thearduinoguy.org/uavproject/)
- FLiR Lepton Thermal Camera Module (http://thearduinoguy.org/flir-lepton-thermal-camera-module/)
- Retro Ramblings (http://thearduinoguy.org/retro-ramblings/)
- The Arduino Academy – Lesson 2 – Basic Outputs (http://thearduinoguy.org/the-arduino-academy-lesson-2-basic-outputs/)
- The Arduino Acadamy – Lesson 1 – An Introduction to the Arduino (http://thearduinoguy.org/the-arduino-acadamy-lesson-1-an-introduction-to-the-arduino/)

Pages

- My Social Links (http://thearduinoguy.org/my-social-links/)
- Tutorials (http://thearduinoguy.org/tutorials/)
  - The Complete Beginners Guide to the Arduino (http://thearduinoguy.org/tutorials/arduinoguide/)
    - Tutorial 1 (http://thearduinoguy.org/tutorials/tutorial-1/)
- Useful Links (http://thearduinoguy.org/useful-links/)

Follow me on Twitter

Tweets by @TheArduinoGuy

Mike McRoberts
@TheArduinoGuy

Looks like ideal conditions for todays maiden flight of the X-Uav Skua. T-6 hours. #Arduino #RaspberryPi

4h

Mike McRoberts
@TheArduinoGuy

UAV all set and ready for its maiden flight tomorrow (Weather permitting). #Arduino #RaspberryPi

19h

Mike McRoberts
@TheArduinoGuy

Assembling the UAV for some tests ready for tomorrows maiden flight. #Arduino #RaspberryPi

07 May

Mike McRoberts
@TheArduinoGuy

Aeroponic V3 — Controlled By Arduino - DZone IoT - klou.tt/o4fc8z4an4tm #Arduino #RaspberryPi

Aeroponic V3 — Controlled By Arduin…
Assembling a new aeroponic control syst…
dzone.com

06 May

Mike McRoberts
@TheArduinoGuy

Adafruit is Now Selling Pi-Top, a Laptop Creation Kit for Raspberry Pi Users - klou.tt/m4qc369sx0dg #RaspberryPi

Adafruit is Now Selling Pi-Top, a Laptop Creation K…
Raspberry Pi is not included in the kit, but it comes with…
techtimes.com

06 May

Embed View on Twitter

Subscribe to Blog via Email

Enter your email address to subscribe to this blog and receive notifications of new posts by email.

Email Address

Subscribe