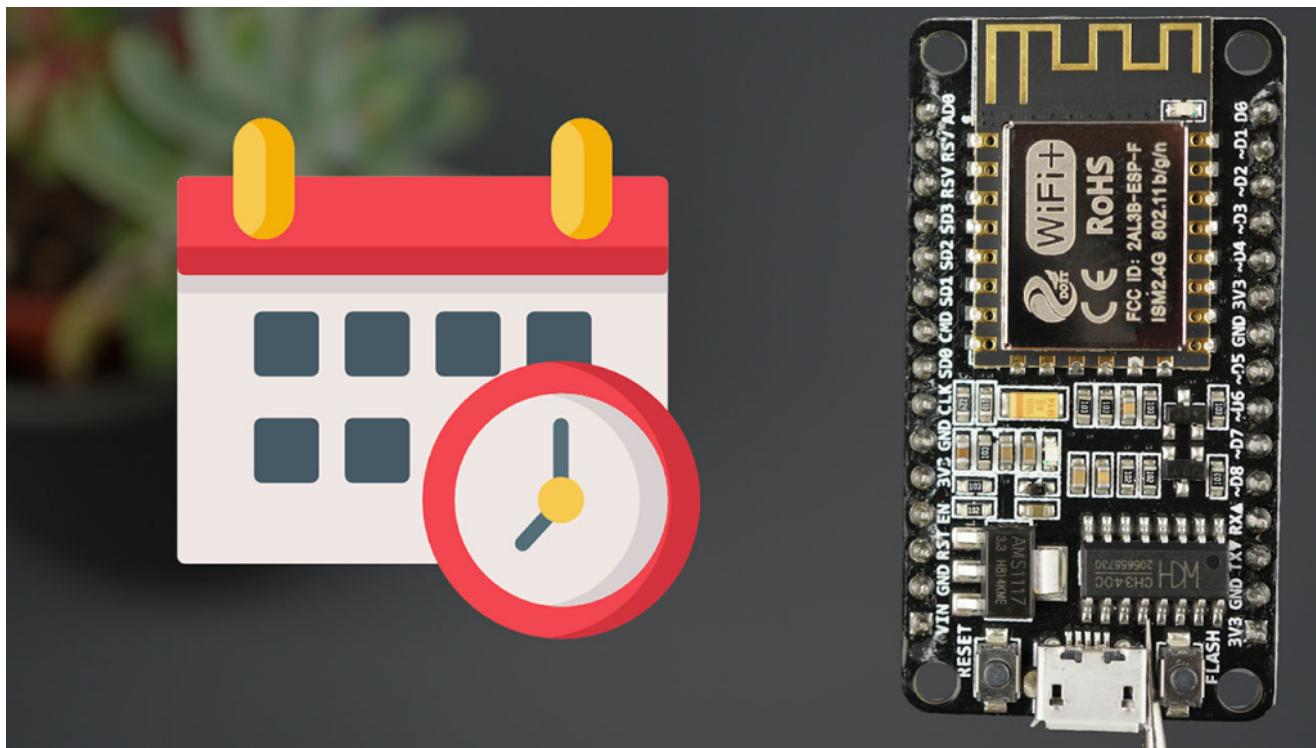


ESP8266 NodeMCU NTP Client-Server: Get Date and Time (Arduino IDE)

In this tutorial you'll learn how to get date and time from an NTP server using the ESP8266 NodeMCU with Arduino IDE. Getting date and time is useful in data logging projects to timestamp readings. To get time from an NTP Server, the [ESP8266](#) needs to have an Internet connection and you don't need additional hardware (like an RTC clock).



Before proceeding make sure you have the ESP8266 board installed in Arduino IDE:

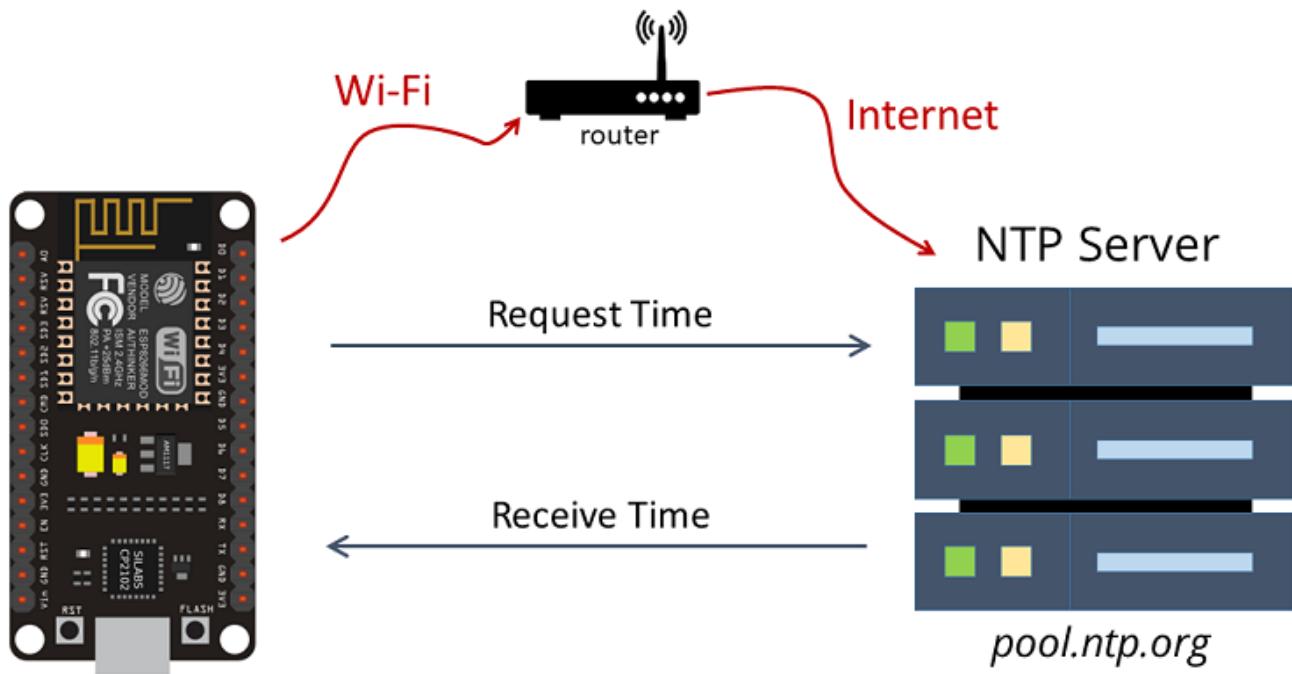
- [Installing ESP8266 Board in Arduino IDE \(Windows, Mac OS X, Linux\)](#)

Recommended: [Get Date and Time with ESP32 NTP Client-Server](#)

NTP (Network Time Protocol)

NTP stands for Network Time Protocol and it is a networking protocol for clock synchronization between computer systems. In other words, it is used to synchronize computer clock times in a network.

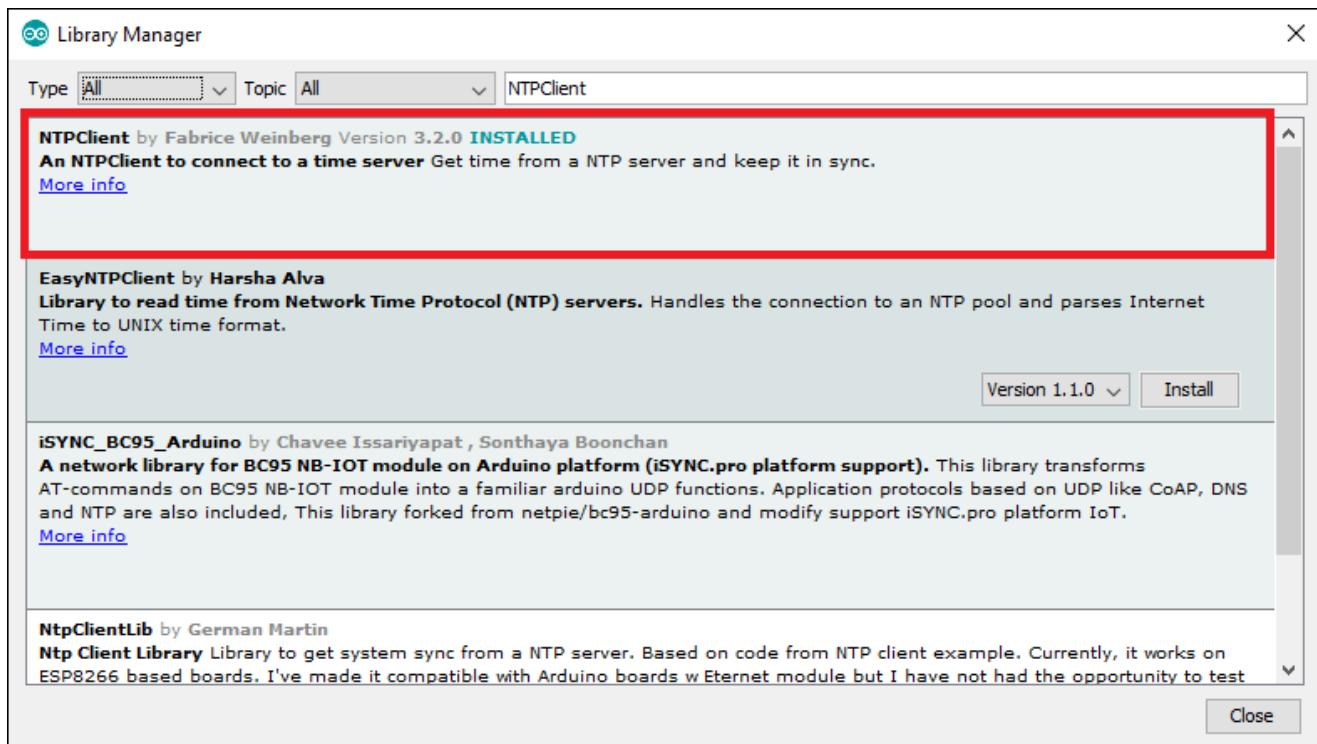
There are NTP servers like *pool.ntp.org* that anyone can use to request time as a client. In this case, the ESP8266 is an NTP Client that requests time from an NTP Server (*pool.ntp.org*).



Installing the NTPClient Library

We'll use the **NTPClient** library to get time. In your Arduino IDE, go to **Sketch > Library > Manage Libraries**. The Library Manager should open.

Search for **NTPClient** and install the library by Fabrice Weinber as shown in the following image.



NTPClient Library Time Functions

The NTPClient Library comes with the following functions to return time:

`getDay()` – returns an int number that corresponds to the the week day (0 to 6) starting on Sunday;

`getHours()` – returns an int number with the current hour (0 to 23) in 24 hour format;

`getMinutes()` – returns an int number with the current minutes (0 to 59);

`getSeconds()` – returns an int number with the current second;

`getEpochTime()` – returns an unsigned long with the epoch time (number of seconds that have elapsed since January 1, 1970 (midnight GMT));

`getFormattedTime()` – returns a String with the time formatted like HH:MM:SS;

This library doesn't come with functions to return the date, but we'll show you in the code how to get the date (day, month and year).

ESP8266 NodeMCU Code

The following code connects the ESP8266 to an NTP Server (pool.ntp.org) to request date and time. It displays the current date and time in several formats in the Serial Monitor.

```
/*
Rui Santos
Complete project details at https://RandomNerdTutorials.com/

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files.

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

*/
#include <ESP8266WiFi.h>
#include <NTPClient.h>
#include <WiFiUdp.h>

// Replace with your network credentials
const char *ssid      = "REPLACE_WITH_YOUR_SSID";
const char *password  = "REPLACE_WITH_YOUR_PASSWORD";

// Define NTP Client to get time
WiFiUDP ntpUDP;
NTPClient timeClient(ntpUDP, "pool.ntp.org");

//Week Days
String weekDays[7]={ "Sunday", "Monday", "Tuesday", "Wednesday",
                     "Thursday", "Friday", "Saturday" };

//Month names
```

[View raw code](#)

How the Code Works

First, include the necessary libraries.

```
#include <ESP8266WiFi.h>
#include <NTPClient.h>
#include <WiFiUdp.h>
```

Insert your network credentials in the following variables so that the ESP8266 can connect to your router to have access to the internet to request date and time from the NTP server.

```
const char *ssid      = "REPLACE_WITH_YOUR_SSID";
const char *password = "REPLACE_WITH_YOUR_PASSWORD";
```

Define an NTP client to get date and time.

```
WiFiUDP ntpUDP;
NTPClient timeClient(ntpUDP, "pool.ntp.org");
```

We'll request the time from *pool.ntp.org*, which is a cluster of times servers that anyone can use to request the time.

Next, we create two arrays to hold the days of the week and the month names.

```
//Week Days
String weekDays[7]={"Sunday", "Monday", "Tuesday", "Wednesday", ' '
//Month names
String months[12]={"January", "February", "March", "April", "May"}
```

setup()

In the `setup()`, initialize the Serial Monitor to display the information.

```
Serial.begin(115200);
```

Next, connect the ESP8266 to the internet.

```
// Connect to Wi-Fi
Serial.print("Connecting to ");
Serial.println(ssid);
WiFi.begin(ssid, password);
while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
}
```

Initialize the NTPClient.

```
timeClient.begin();
```

Set Timezone

You can use the `setTimeOffset()` method to adjust the time for your timezone in seconds.

```
timeClient.setTimeOffset(3600);
```

Here are some examples for different timezones:

- GMT +1 = 3600
- GMT +8 = 28800
- GMT -1 = -3600
- GMT 0 = 0

We live in Portugal, so we don't need to adjust the time.

```
timeClient.setTimeOffset(0);
```

loop()

In the `loop()`, call the `update()` function to get the current date and time from the NTP server.

```
timeClient.update();
```

Get Time

Then, we can use the functions provided by the library to get time. For example, to get the epoch time:

```
unsigned long epochTime = timeClient.getEpochTime();
Serial.print("Epoch Time: ");
Serial.println(epochTime);
```

The `getFormattedTime()` function returns the time in HH:MM:SS format.

```
String formattedTime = timeClient.getFormattedTime();
Serial.print("Formatted Time: ");
Serial.println(formattedTime);
```

You can get the hours, minutes or seconds separately using the `getHours()`, `getMinutes()` and `getSeconds()` functions as follows:

```
int currentHour = timeClient.getHours();
Serial.print("Hour: ");
Serial.println(currentHour);

int currentMinute = timeClient.getMinutes();
Serial.print("Minutes: ");
Serial.println(currentMinute);

int currentSecond = timeClient.getSeconds();
```

```
Serial.print("Seconds: ");
Serial.println(currentSecond);
```

Get Date

The `getDay()` function returns a number from 0 to 6, in which 0 corresponds to Sunday and 6 to Saturday. So, we can access the week day name from the array we've created previously as follows

```
String weekDay = weekDays[timeClient.getDay()];
Serial.print("Week Day: ");
Serial.println(weekDay);
```

The NTP Client doesn't come with functions to get the date. So, we need to create a time structure (`struct tm`) and then, access its elements to get information about the date.

```
struct tm *ptm = gmtime ((time_t *)&epochTime);
```

The time structure contains the following elements:

- `tm_sec` : seconds after the minute;
- `tm_min` : minutes after the hour;
- `tm_hour` : hours since midnight;
- `tm_mday` : day of the month;
- `tm_year` : years since 1900;
- `tm_wday` : days since Sunday;
- `tm_yday` : days since January 1;
- `tm_isdst` : Daylight Saving Time flag;
- [tm structure documentation](#).

The following lines get the day of the month as follows:

```
int monthDay = ptm->tm_mday;
Serial.print("Month day: ");
Serial.println(monthDay);
```

To get the other elements, you use a similar approach. For example, for the month:

```
int currentMonth = ptm->tm_mon+1;
Serial.print("Month: ");
Serial.println(currentMonth);
```

Because the `tm_mday` starts at 0, we add 1 to the month so that January corresponds to 1, February to 2, and so on.

Then, we can get the name of the month using the `months` array we've created previously. The arrays numbering starts at 0, that's why we subtract 1.

```
String currentMonthName = months[currentMonth-1];
Serial.print("Month name: ");
Serial.println(currentMonthName);
```

To get the year, we need to add 1900 because the `tm_year` saves the number of years after 1900.

```
int currentYear = ptm->tm_year+1900;
Serial.print("Year: ");
Serial.println(currentYear);
```

Finally, we create a String called `currentDate` that holds the current date in the YYYY-MM-DD format.

```
String currentDate = String(currentYear) + "-" + String(currentM
Serial.print("Current date: ");
```

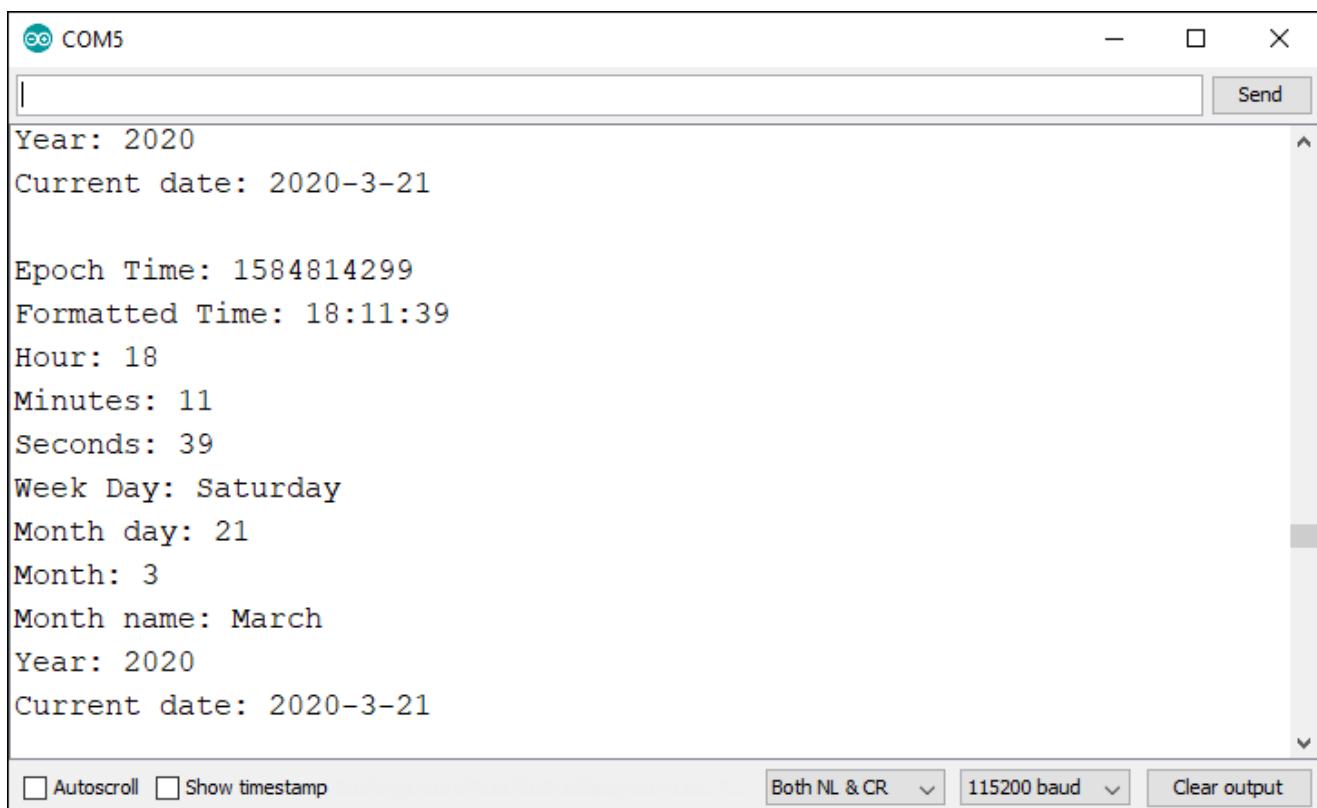
```
Serial.println(currentDate);
```

Demonstration

After inserting your network credentials and modifying the variables to adjust the time to your timezone, test the example.

Upload the code to your ESP8266 board. Make sure you have the right board and COM port selected.

Open the Serial Monitor at a baud rate of 115200. The date and time should be displayed in several formats as shown below.



Wrapping Up

In this tutorial you've learned how to get date and time from an NTP server using the ESP8266. This is specially useful for data logging projects that have access to the internet.

If you don't have access to the internet, you can use an [RTC module like the DS1307](#).

If you want to learn more about the ESP8266, check our resources:

- Home Automation using ESP8266
 - MicroPython Programming with ESP32 and ESP8266
 - More ESP8266 projects...

Thanks for reading.

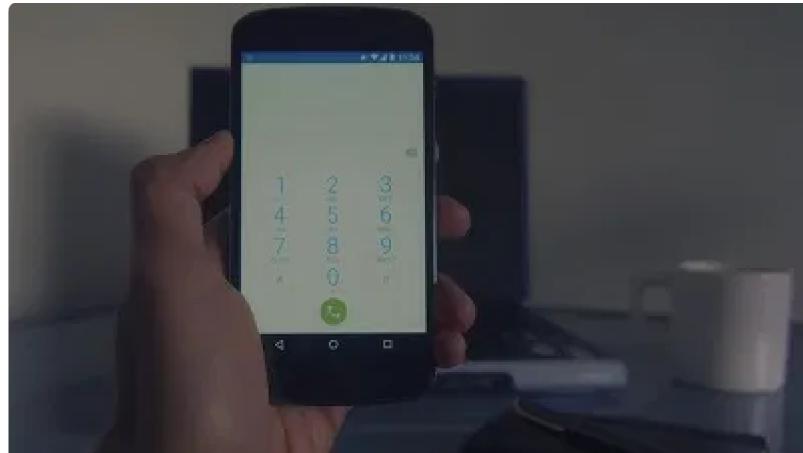
An advertisement for PCBWay featuring a green and yellow design. The top left has the PCBWay logo and 'PCB Fabrication & Assembly' text. The top right shows the website address 'wwwpcbway.com'. The center features a large orange 'ONLY \$5 for 10 PCBs' text. Below it is a list of benefits with checkmarks and color swatches. A large image of a complex PCB is shown in a circular inset on the right.



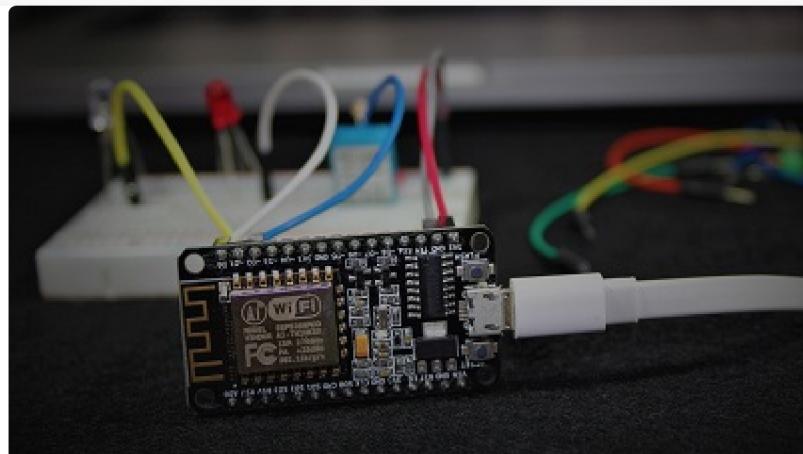
[eBook] Build Web Servers with ESP32 and ESP8266 (2nd Edition)

Build Web Server projects with the ESP32 and ESP8266 boards to control outputs and monitor sensors remotely. Learn HTML, CSS, JavaScript and client-server communication protocols [DOWNLOAD »](#)

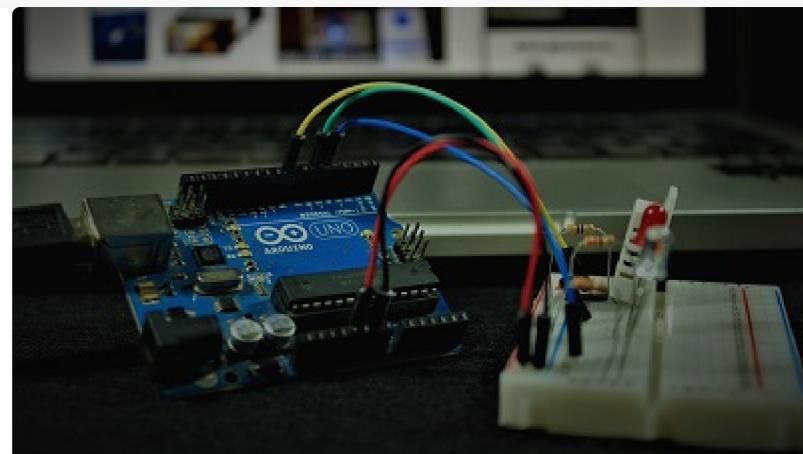
Recommended Resources



[Build a Home Automation System from Scratch »](#) With Raspberry Pi, ESP8266, Arduino, and Node-RED.



[Home Automation using ESP8266 eBook and video course »](#) Build IoT and home automation projects.



[Arduino Step-by-Step Projects »](#) Build 25 Arduino projects with our course, even with no prior experience!

What to Read Next...

[ESP32 Capacitive Touch Sensor Pins with Arduino IDE](#)

[ESP32 Save Data Permanently using Preferences Library](#)

Enjoyed this project? Stay updated by subscribing our newsletter!

SUBSCRIBE

66 thoughts on “ESP8266 NodeMCU NTP Client-Server: Get Date and Time (Arduino IDE)”



Jan

March 25, 2020 at 4:11 pm

Thanks for your project. Because of the summer time next week, I was experimenting and searching for this kind of solution. Do you have an suggestion for switch between normal and summertime?

Thanks in advantage.

PS. Timezone Amsterdam

[Reply](#)



Rui Santos

April 3, 2020 at 10:12 am

Hello Jan, unfortunately this example doesn't cover timezone adjustment. We've looked into that subject, but it wasn't implemented in the libraries.

[Reply](#)



Jan

April 3, 2020 at 5:12 pm

Hi Rui,

Thanks for your response and sorry I was not clear enough. By the way, your books and examples have helped me a lot with getting started with my smart home.

I believe I have done all what you wrote.

Downloaded and installed the lib's.

But I get this message:

exit status 1

freertos/FreeRTOS.h: No such file or directory

See:

De volumenaam van station C is OS_W7

Het volumenummer is FA0D-EFD7

Map van C:\Users\Jan\Mijn documenten\Arduino\libraries

01-04-2020 19:37 .

01-04-2020 19:37 ..

19-01-2020 15:30 Adafruit_ADXL343

01-04-2020 17:45 Adafruit_BME280_Library

19-01-2020 15:30 Adafruit_Circuit_Playground

03-02-2020 19:43 Adafruit_Unified_Sensor

18-11-2019 22:01 Arduino_SigFox_for_MKRFox1200

01-04-2020 19:35 AsyncTCP

01-04-2020 19:22 async_mqtt_client
22-12-2019 18:34 DHT_sensor_library
13-03-2020 09:45 ESP32_Mail_Client
22-12-2019 15:46 ESPAsyncTCP
22-12-2019 15:44 ESPAsyncWebServer
22-03-2020 18:46 EspMQTTClient
22-03-2020 18:47 MFUthings
21-03-2020 22:35 NTPClient
21-03-2020 22:28 NTPClient-master
22-12-2019 15:50 NTPClient-oud
17-01-2018 22:52 NTPClient-Patched
22-03-2020 18:46 PubSubClient
18-08-2019 22:16 108 readme.txt
18-11-2019 22:01 SD
22-12-2019 15:50 Servo
22-12-2019 15:50 SpacebrewYun
1 bestand(en) 108 bytes
23 map(pen) 2.921.226.240 bytes beschikbaar

[Reply](#)



Jeroen

May 2, 2022 at 2:18 pm

Can someone help me out.
I do i change serialprintln to something like.

digitalWrite(getseconds, HIGH);

Tnx in advanced

[Reply](#)

**Jan**

April 3, 2020 at 5:18 pm

Hi Sara and Rui,

Sorry, reaction before was a reaction on ESP32 with BME280.

Correct reaction is; I fixed Summertime with testing for weeknumber , sunday and the time.

[Reply](#)**Enrique**

December 12, 2022 at 9:33 am

Hello Rui and Jan.

Greetings from Andalusia!

I must first say that I am a bit new to these issues. Forgive me if I say something nonsense.

I tried your code on a NodeMCU and it was great, it worked first time. Thank you very much for this and also for how well its operation is explained.

I also wanted it to automatically switch to daylight saving time and I did the following which seems to work:

1.

In the void loop I added “int summertime = ptm->tm_isdst;” to get the data from the structure and store it in a variable.

I then printed it out the serial port and saw that it is currently (ending fall) 0.

According to the documentation at your link..."The daylight saving time

flag (tm_isdst) is greater than zero if daylight saving time is in effect, zero if daylight saving time is not in effect, and less than zero if the information is not available .”

I continued to modify the original code and add a few things.

2.

I created a variable “int secondsOffset;”

I took out the offset setting “timeClient.setTimeOffset(secondsOffset);” from the void setup and included it at the beginning of the void loop.

Under the line of point 1 I added

```
“if (summerTime > 0) {  
secondsOffset = 7200;  
}  
else {  
secondsOffset = 3600;  
}”
```

In this way, in each loop, if I receive the data that we are in summer time, the offset will be adjusted to 2 hours, and if it is not, it will be 1 hour.

(Or so I hope).

Thanks for everything.

[Reply](#)



Geert

March 27, 2020 at 7:03 pm

Hello Rui,

For the ntp client, does it also support: sunset and sunrise to pilot a chicken coop door?

Regards

[Reply](#)**Sara Santos**

March 28, 2020 at 6:20 pm

Hi Geert.

No, it doesn't.

For that, it is better to search for an API that returns that kind of information. For example, <https://openweathermap.org/current>

Regards,

Sara

[Reply](#)**WME**

March 28, 2020 at 6:04 pm

Hello,

nice tutorial.

I have changed this program a little bit.

I use an ESP32S NodeMCU and added a OLED and a temperature sensor.

Regards

Wolfgang

[Reply](#)**Sara Santos**

March 28, 2020 at 6:15 pm

Great!

[Reply](#)



Giano

October 12, 2020 at 1:22 pm

Could you share the sketch?

[Reply](#)



Alnoor Ratansi

May 12, 2020 at 4:33 pm

Hi , Great Tutorial. is there a way incorporate the timezone /dst/est into the code ?.

my programing skills are very good.

[Reply](#)



Clive

May 16, 2020 at 8:05 am

I'm sure I have read somewhere that you shouldn't continually call an ntp, and that you should only update a clock at interval?

Have you a tutorial on using an esp 8266 as a clock running on its own, maybe looking to sync on the hour or every couple of hours?

I know of RTC chips and milli but is there a Clock library you could point me at that you consider to be a good one for a newbie

Cheers Clive

[Reply](#)



Luc Berger

June 30, 2020 at 2:35 pm

Local clock can be probably done with millis as offset to rt clock.

If you get the RT, the millis() is an offset to add to the UTC time obtained at the powerup of the module.

Then update every x seconds.

But as the wifi is connected, it is far more easy to get the RT every x seconds.

[Reply](#)



Laurent

May 23, 2020 at 4:28 pm

You saved me, once again!

Great job.

[Reply](#)

**Luc Berger**

June 30, 2020 at 12:07 pm

Thanks very much !

Correction:

“Set Timezone: You can use the setTimeOffset() method to adjust the time for your timezone in milliseconds.

”

It is ‘seconds’.

[Reply](#)**Techno**

September 3, 2020 at 11:09 am

Nicely explained !

Can you please write tutorial about creating own local NTP server using ESP8266 and RTC module ?

[Reply](#)**Richie**

September 20, 2020 at 7:50 am

Hello Sara, Rui,

I love your website, it is so informative!

I am a beginner with Arduino/ESP, so maybe my question is lame, but when I tried to verify this ntp code, it said: 'gmtime' was not declared in this scope. I tried to check, if some library is missing, but everything looks good. Any advice?

Thank you,

Richie

[Reply](#)



Sara Santos

September 20, 2020 at 11:32 pm

Hi.

Make sure you have an ESP8266 board selected in Tools > Board before compiling.

Regards,

Sara

[Reply](#)



Clive

November 15, 2020 at 9:44 pm

Thanks for doing these tutorials but here is a suggestion that I am thinking about but may not be a simple as it sounds?

How can you link NTP to ESPNow?

It's going to take me a while to work this one out as I am working on some dht sensors that will feed back via espnow to a esp that is linked to

my home router to display and store the readings.

I have had a little success due to your tutorials but what I am thinking is this;

If the remote ends send data to the centre in a many to one configuration how can I pass back or check the remote ends to sync a timestamp with an ntp?

One of the remote end sensors will be in the garden and I am toying with having a digital temperature display and a clock.

I guess I need to get the things working before I start to complicate it beyond my limited expertise

Cheers Clive

[Reply](#)



Kamil

December 17, 2020 at 7:51 pm

Hallo Richie

concerning problem during compilation “gmtime was not declared in this scope” I did a small investigation:

“struct tm” is a function type of C++ library . There should be added line #include “time.h” at beginning of the code. Than compilation is OK.

Kamil

[Reply](#)

**iamjucy**

February 16, 2021 at 11:58 am

install new version of esp8266
arduino menu -> tools -> board -> board manager

[Reply](#)**David Duehren**

November 15, 2020 at 7:08 pm

Please comment on this"
<https://forum.arduino.cc/index.php?topic=655222.0>

[Reply](#)**Joachim Z.**

November 26, 2020 at 3:54 pm

In the explanation above you write “You can use the setTimeOffset() method to adjust the time for your timezone in milliseconds.” Is that right, or isn’t it adjusted in seconds?

[Reply](#)**Sara Santos**

November 26, 2020 at 4:44 pm

Hi.

You are right.

It is in seconds.

The tutorial is fixed now.

Regards,

Sara

[Reply](#)



Kamil

December 17, 2020 at 6:04 pm

Hello Sara, Rui,

Your website is very good, i like it.

But in this case I tried to verify this ntp code, it said: 'gmtime' was not declared in this scope. It is the same problem as described in comment in September by Richie.

I have selected "Generic ESP8266 Module", than NodeMcu 1.0 (esp-12E), than Adafruit ESP8266 with the same problem. In your article there is not mentioned an exact board type, just ESP8266. Can you provide info, for which board type is project dedicated?

Thank you,

Kamil

[Reply](#)



Kamil

December 17, 2020 at 7:57 pm

Hallo Sara, Rui,

one more info to my previous comment. Do not care about it, problem is now solved:

concerning error during compilation “gmtime was not declared in this scope” I did a small investigation:

“struct tm” is a function type of C++ library . There should be added line #include “time.h” at beginning of the code. After this update, compilation is OK.

Kamil

Kamil

[Reply](#)



Luiz

February 2, 2021 at 2:15 pm

Hi Sara and Rui,

Is there the equivalent in Micropython?

thanks, Luiz

[Reply](#)



mikail

February 10, 2021 at 7:36 pm

Hi Sara and Rui,

I used your code. its run well. But whenn I integrated code in to my code and

If I specify the IP number, the codes return blank. (1970).

[Reply](#)**Sara Santos**

February 11, 2021 at 10:47 am

Hi.

That's because it wasn't able to get the time from the server.

Reset the board and try again.

Regards,

Sara

[Reply](#)**Sagara Dissanayake**

June 6, 2021 at 4:30 pm

Hi,

I am using this program for some time now and it was working well. But now i see some errors in the date and the formatting. I even reuploading the this sketch to my NodeMcu but no luck. Following is the serial out put i get.

Epoch Time: 1622996660

Formatted Time: 16:24:20

Hour: 16

Minutes: 24

Seconds: 20

Week Day: Sunday

Month day: 26

Month: 3

Month name: March

Year: 1340177

Current date: 1340177-3-26

As you can see date, the date format and the year is incorrect. Any idea where it has gone wrong ?

Regards

Sagara

[Reply](#)



Sara Santos

June 7, 2021 at 9:17 am

Hi.

Downgrade your ESP8266 boards installation to version 2.7.4.

In your Arduino IDE, go to Tools > Board > Boards Manager. Search for ESP8266. Downgrade to version 2.7.4.

Regards,

Sara

[Reply](#)



Irakli

August 17, 2021 at 11:54 am

Works for me, thank you.

[Reply](#)

**Emil**

August 17, 2021 at 9:33 pm

Thanks. This fixed my issue!

[Reply](#)**Christian**

June 18, 2021 at 7:32 am

Hi

Thanks for yours tutoriels

I used your code in order to get date, but it'snt run well in may case.

The date change every second (see below)

I am using an ESP8266 card and I program on VSCode + Plateformio.

Thank you in advance for your help.

Code:

```
timeClient.update();
```

```
unsigned long epochTime = timeClient.getEpochTime();
```

```
//Get a time structure
```

```
struct tm *ptm = gmtime ((time_t *)&epochTime);
int monthDay = ptm->tm_mday;
int currentMonth = ptm->tm_mon+1;
int currentYear = ptm->tm_year+1900;
```

Result:

epochTime Date Times

1624008484 — 5/5/1884 177 09:28:04 – 1655565 – Lum > 797

1624008485 — 9/6/2020 279 09:28:05 – 1656565 – Lum > 795

1624008486 — 14/7/2156381 09:28:06 – 1657565 – Lum > 797
1624008487 — 17/8/2292483 09:28:07 – 1658565 – Lum > 798
1624008488 — 21/9/2428585 09:28:08 – 1659565 – Lum > 798
1624008489 — 27/10/2564687 09:28:09 – 1660566 – Lum > 796
1624008490 — 30/11/2700789 09:28:10 – 1661565 – Lum > 796
1624008491 — 4/1/2836892 09:28:11 – 1662565 – Lum > 796
1624008492 — 8/2/2972994 09:28:12 – 1663565 – Lum > 797
1624008493 — 14/3/3109096 09:28:13 – 1664565 – Lum > 797
1624008494 — 19/4/3245198 09:28:14 – 1665565 – Lum > 797

[Reply](#)



Ogursoft

October 30, 2021 at 5:01 pm

```
try to replace struct tm *ptm = gmtime((time_t *)&epochTime);  
to  
time_t rawtime = epochTime;  
struct tm *ptm = gmtime(&rawtime);
```

[Reply](#)



Parrena

November 13, 2021 at 6:23 am

Thank you I had a kind issue, and solve with this (even I dont understand exactly why!)

[Reply](#)

**Selçuk Özbayraktar**

June 27, 2021 at 7:00 am

Hi Rui, Hi Sara,

I attempted to use NTP as you described in your book ESP32/8266 WEB SERVERS, and I also tried this example.

But epochTime starts with zero each time. Then continues to increment each second. Any clue?

BR

[Reply](#)**Sara Santos**

June 29, 2021 at 5:50 pm

Hi.

Were you able to solve the problem?

Regards,

Sara

[Reply](#)**Selçuk Özbayraktar**

June 29, 2021 at 5:56 pm

No Sara, it hasn't been solved. By the way, I forgot to tell you that I am using ESP8266_01 1M chip. On each reset, epoch time starts with zero and then counts on. Years start with tens of thousands BC, days and months with random values.

Your help shall be most appreciated.

Kind Regards,

[Reply](#)



Sara Santos

June 29, 2021 at 6:08 pm

Hi.

What is the version of the ESP8266 boards that you are using?

Downgrade to version 2.7.4 (it works well, I've tested it) while we cannot find the "real" solution.

Regards,

Sara

[Reply](#)



Selçuk Özbayraktar

June 29, 2021 at 6:12 pm

Thanks for the instant response Sara,

How shall I know the version of my ESP8266-01? It has been sitting in my drawer for 2 years or more. Then next question, how to downgrade it?

BR



Selçuk Özbayraktar

June 29, 2021 at 6:22 pm

I am using platformIO. The following lines are printed when I connect for uploading code. It seems my board is 3.0.0 version. Then ignore my first question. But the second question still persists. How to downgrade it? Shall I sacrifice any feature by downgrading it?

Verbose mode can be enabled via -v, --verbose option

CONFIGURATION:

https://docs.platformio.org/page/boards/espressif8266/esp01_1m.html

PLATFORM: Espressif 8266 (3.0.0) > Espressif Generic ESP8266
ESP-01 1M

HARDWARE: ESP8266 80MHz, 80KB RAM, 1MB Flash



Sara Santos

June 30, 2021 at 1:34 pm

Hi.

In your VS Code, go to PlatformIO Home.

Then, at the left-side bar, select Platforms -> Embedded -> Espressif 8266

Then, next to the “Installation” title there’s a drop-down menu where you can select the version.

Regards,

Sara



Selçuk Özbayraktar

July 4, 2021 at 6:05 am

Hi Sara,

I wanted to give you feedback and ask for additional support.

I tried to change the version in VS following your prescription. Although the process seems to progress smoothly, espressif 8266 version still appears to be 3.0.0 on the console. As a result, NTP does not work.

So I tried the ezTime library that worked perfectly. But that too works with only dynamic IP allocation in my Async Webserver application. When I attempt to use static IP using the following lines, NTP servers start to timeout.

```
// Connect to Wi-Fi  
IPAddress ip(192,168,1,180); //Node static IP  
IPAddress gateway(192,168,1,1);  
IPAddress subnet(255,255,255,0);  
  
WiFi.mode(WIFI_STA);  
WiFi.begin(ssid, password);  
WiFi.config(ip, gateway, subnet);
```

There may be a problem with the subnet mask, but I am a newbie with Internet protocols, maybe you can help.

Static IP is a real need since I don't know a way to know the IP of my ESP when it is not connected to my PC. I must be able to connect to the ESP server without worrying about the changed IP each time.

Thanks in advance,

Kind regards,



Sara Santos

July 5, 2021 at 10:04 am

Hi.

Can you try that snippet of code for the static IP address without the

new NTP code?

This way you can figure out if there is some sort of “incompatibility” between the two or if there is something wrong with the static IP. At the moment, I still couldn’t find a way to solve the NTP issue other than using an old version of the ESP8266 boards.

Regards,

Sara



Selçuk Özbayraktar

July 5, 2021 at 1:08 pm

Hi Sara,

I have been using those couple of lines on all of my ESP projects without exception for years. My projects are from your books, with that small modification to establish static IP on my connections. I like to have a predetermined IP when playing with ESP8266 and ESP32. That worked also with the project “async web server with charts from file” from your book of WEB SERVERS. I can display charts nicely on clients’ screens. But their time axis’ displaying wrong date/time values since NTPClient is not working properly.

In conclusion; Async web servers are working with my static IP initialization lines.

But ezTime has a problem with it, the NTP server time-outs with static IP.

The issue of the NTPClient is different because it does not work with dynamic IP also. I can not study it more because I couldn’t manage to downgrade the version of espressif 8266.

Kind regards,

**Selçuk Özbayraktar**

July 8, 2021 at 5:30 pm

Hi Sara,

In return for your kind attention to my questions, I would like to inform you how I resolved this issue:

I still use the ezTime library, ESP8266 Espressif 8266 version 3.0.0, and VS Platform IO.

After lots of trials and errors, finally I added primary and secondary DNS definitions to my wifi.config call, as follows:

```
IPAddress DNS_1(8, 8, 8, 8);  
IPAddress DNS_2(8, 8, 4, 4);  
WiFi.config(ip, gateway, subnet, DNS_1, DNS_2);
```

Now ezTime works as a charm.

Kind regards.

[Reply](#)**Sara Santos**

July 9, 2021 at 9:27 am

Great!

Thank you so much for taking the time to share this.

Regards,

Sara

[Reply](#)

**Andre**

August 25, 2021 at 7:50 pm

Dear Selçuk,

Thank you so much for sharing this. Had exactly the same issue with my ESP8266 and static IP and your proposed fix solved it so perfectly !
bib thanks:-)

Andre

[Reply](#)**Selçuk Özbayraktar**

August 25, 2021 at 9:19 pm

Glad to hear it was useful Andre. Kind regards.

[Reply](#)**Jeff Blavat**

September 9, 2021 at 3:02 am

Year: 1340177

Current date: 1340177-3-26

As you can see date, the date format and the year is incorrect. Any idea where it has gone wrong ?

Regards

Sagara

I was having this same problem with a NodeMCU 1.0 under MacOS 11.5 (Big Sur). I downgraded to ESP8266 2.7.4 and then started having upload failures. Compile and upload ended with “pyserial or esptool directories not found next to this upload.py tool.

An error occurred while uploading the sketch”.

The solution to this was found at: <https://forum.arduino.cc/t/pyserial-and-esptools-directory-error/671804/5>

PySerial and EspTools Directory Error

This involved editing a library file:

1.- Open

~/Library/Arduino15/packages/esp8266/hardware/esp8266/2.7.4/tools/pyserial/serial/tools/list_ports_osx.py

2.- Comment out lines 29 and 30 and append these lines:

iokit =

```
ctypes.cdll.LoadLibrary('/System/Library/Frameworks/IOKit.framework/IOKit')
```

cf =

```
ctypes.cdll.LoadLibrary('/System/Library/Frameworks/CoreFoundation.framework/CoreFoundation')
```

The code should look like this:

```
#iokit = ctypes.cdll.LoadLibrary(ctypes.util.find_library('IOKit'))
#cf = ctypes.cdll.LoadLibrary(ctypes.util.find_library('CoreFoundation'))
iokit =
ctypes.cdll.LoadLibrary('/System/Library/Frameworks/IOKit.framework/IOKit')
cf =
ctypes.cdll.LoadLibrary('/System/Library/Frameworks/CoreFoundation.framework/CoreFoundation')
```

Is there a better way to handle this problem?

[Reply](#)

**hasan**

October 25, 2021 at 11:31 pm

i had never get the date right
i use the same code on nodemcu 1.0
i have arduino ide with this verision 1.8.15
and this is the output i got

Epoch Time: 1635204557

Formatted Time: 23:29:17

Hour: 23

Minutes: 29

Seconds: 17

Week Day: Monday

Month day: 14

Month: 8

Month name: August

Year: 1340177

Current date: 1340177-8-14

[Reply](#)**Sonz Verzosa**

November 4, 2021 at 8:34 am

Hi hasan,

i had the same issue with your output, but after downgrade to version 2.7.4 the year issue was solved.

previous i used the esp8266 board version 3.0.1, then downgrade to 2.7.4 as per Ms. Sara information i got the correct date..

hope it will solve the issue on your side

cheer

sonz

[Reply](#)



Dragos Chelan

November 27, 2021 at 8:29 am

How do I know if ESP is not receiving time from the ntp server? I need this to reset the ESP from the program. After a power failure, the router works when it returns, but the internet can return after a few minutes and ESP no longer manages to receive the time data unless I reset it while the router is functional and connected to the internet.

[Reply](#)



ZaGa

November 27, 2021 at 9:02 am

This circuit requires a backup RTC (ds3231) in case the NTP communications drop away for whatever reason.

[Reply](#)



Dragos Chelan

November 27, 2021 at 4:16 pm

I don't want to use rtc, I accept that when there is no internet the clock doesn't work, I just want as long as there is no internet esp8266 to feel

this and keep resetting.

[Reply](#)



Dave Mucha

December 26, 2021 at 2:38 pm

Hi, Rui, Sara,

This has been very helpful, but like others, I want to be near a WiFi, get the time, then move my device out into the garden where there is no WiFi. I have only started on this, I have not done any extensive testing.

I am not a good programmer, but am cobbling together a work-around. Maybe Rui/Sara, you can make a follow-up tutorial with fix ?

C++ has some system help here. This small code should work without any external help.

But, it should return the epoch time starting at 00:00 Jan 1 1970 and count up using the internal time.

It does not have any link to a WiFi, it is a bare minimum code and uses system defined variables.

```
void setup() {  
  Serial.begin(115200);  
}  
  
void loop() {  
  time_t current = time(nullptr); // increments internal clock  
  Serial.print(ctime(&current));  
  delay(500);  
}
```

The above shows the variables are system variables.

Assume that when you upload, you are near a WiFi and can get NTP time.

in setup()

```
WiFi.begin(ssid, pass); // starts the connection
while (WiFi.status() != WL_CONNECTED) {
delay(500);
Serial.print(".");
}
```

in loop()

```
if (WiFi.status() == WL_CONNECTED) {
unsigned long epochTime = timeClient.getEpochTime();
}

while (WiFi.status() != WL_CONNECTED) {

// lastUpdateTime = 1640488470 // get from NTP attempt 1 use hard
number
lastUpdateTime = epochTime // get from NTP attempt 2 use epochTime.

// time_t current = time(nullptr); // increments internal clock attempt 1 verify
system clock
// time_t current = 1640488470+time(nullptr); // adds NTP time attempt 2
use hard code
time_t current = lastUpdateTime+time(nullptr); // adds NTP time attempt 3
use epoch time

Serial.print(ctime(&current)); // prints formatted as Sun Dec 26 11:15:31
2021
delay(500);
} // ===== END if(while ! Connected) =====
```

If power is lost, then the system clock and epochTime will both be lost but, the internal timer will start at 00:00 Jan1 1970 so you will be able to know when power was lost and when epochTime is available, it will correct for future readings. It would allow you to go back, and fix the time for the data.

MISSING :

`lost_Time = epochTime – lastUpdateTime ;`
to know the time lost/gained when not connected.

I have one device I use to test things while out in the garden. Usually less than one hour.

Take readings every 15 seconds

I have a device I put out and leave for a week, then bring in and download the data.

Take readings every 30 minutes

WISH LIST : possibly a separate tutorial
go to the garden with the Android phone, connect,
pass EPOCH_TIME
and download the data file saved in SPIFFS/LITTLEfs

in my case, my readings are temperature, sunlight, soil and are done every 30 minutes.

[Reply](#)



Ferenc Kovacs

January 18, 2022 at 10:38 am

Hello

I had the issue of:

Epoch Time: 1642509190

Formatted Time: 12:33:10

Hour: 12

Minutes: 33

Seconds: 10

Week Day: Tuesday

Month day: 6

Month: 4

Month name: April

Year: 1340722

Current date: 1340722-4-6

After a lot of research (without result) I made the following change in the code in line 59:

instded of

```
unsigned long epochTime = timeClient.getEpochTime();
```

I used:

```
unsigned long long epochTime = timeClient.getEpochTime();
```

After this the app start to calculate correctly the date.

Thanks for the app.

[Reply](#)



orionz

March 8, 2022 at 2:26 am

working WELL!

Thanks Mr. Ferenc Kovacs. Have good days!

[Reply](#)



Sara Santos

March 8, 2022 at 10:25 am

Hi.

You can use

```
time_t epochTime = timeClient.getEpochTime();
```

I updated the code just now.

Regards,

Sara

[Reply](#)



Luc Berger

January 19, 2022 at 7:55 pm

A get a wrong year, I have to change
from

```
//unsigned long epochTime = timeClient.getEpochTime();  
to  
time_t epochTime = timeClient.getEpochTime(); // was unsigned long
```

I do not kow the update who ignite that.

[Reply](#)



Gabryx

February 22, 2022 at 8:48 pm

Look at my example here:

<https://forum.lvgl.io/t/a-precision-table-clock-with-wind-advisor/8304>

Not need any external library, just use #include "time.h"

you can set your time zone and day saving with this lines:

String TimeZone = "CET-1CEST,M3.5.0,M10.5.0/3"; /* Rome

https://github.com/nayarsystems posix_tz_db/blob/master/zones.csv */

and in this link you can see all world time zones string

ESP32 at start show 1970 how current year, so i use this approach for wait NTP server response ok

```
while (Year == "1970") {  
    getLocalTime(&timeinfo);  
    strftime(TT, sizeof(TT), "%Y", &timeinfo);  
    Year = String(TT);  
}
```

where TT is a char array

[Reply](#)



crashed information

March 22, 2022 at 8:22 am

Hello, I'd like to share my idea to you. I had a problem with crashed or incomplete information from NTP printed to the serial monitor and sometimes also with showing the downloaded time on my 7segment display – the time started from 01:00:00. It seems like a problem with very fast communication to me, so i put some delay(50) instructions among the lines wifi.begin, timeClient.begin and Serial.print. Now the connecting process is more stable and problem with crashed information is really fixed.

[Reply](#)



Sara Santos

March 25, 2022 at 6:15 pm

Thanks for sharing.

Regards,

Sara

[Reply](#)

Leave a Comment

Name *

Email *

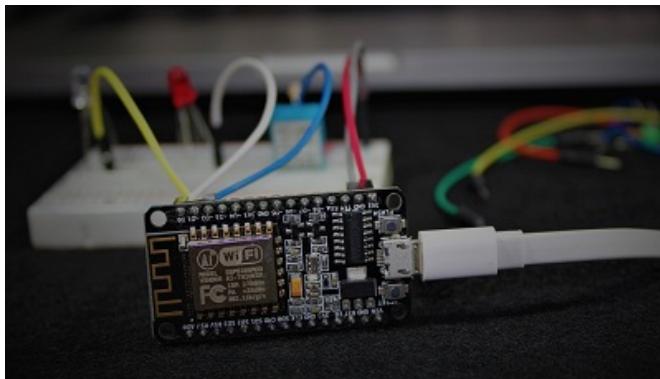
Website

- Notify me of follow-up comments by email.
- Notify me of new posts by email.

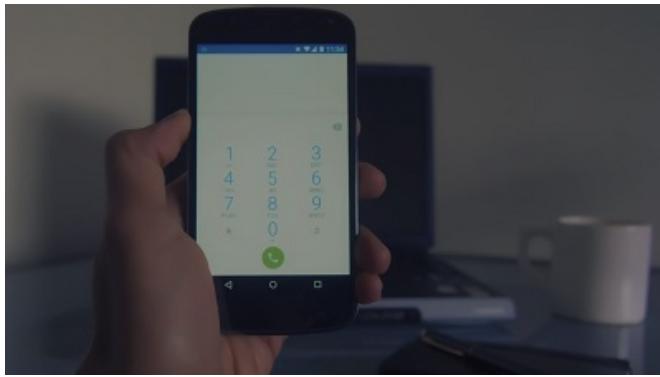
[Post Comment](#)



[Visit Maker Advisor – Tools and Gear for makers, hobbyists and DIYers »](#)



[Home Automation using ESP8266 eBook and video course »](#) Build IoT and home automation projects.



[Build Web Servers with ESP32 and ESP8266 »](#) boards to control outputs and monitor sensors remotely.