

---

# AdafruitNTP Library Documentation

*Release 1.0*

**Brent Rubell**

**Mar 15, 2020**



---

## Contents

---

<b>1</b>	<b>Dependencies</b>	<b>3</b>
<b>2</b>	<b>Installing from PyPI</b>	<b>5</b>
<b>3</b>	<b>Usage Example</b>	<b>7</b>
<b>4</b>	<b>Contributing</b>	<b>9</b>
<b>5</b>	<b>Documentation</b>	<b>11</b>
<b>6</b>	<b>Table of Contents</b>	<b>13</b>
6.1	Simple test . . . . .	13
6.2	adafruit_ntp . . . . .	14
6.2.1	Implementation Notes . . . . .	14
<b>7</b>	<b>Indices and tables</b>	<b>15</b>
	<b>Python Module Index</b>	<b>17</b>
	<b>Index</b>	<b>19</b>



Network Time Protocol (NTP) helper for CircuitPython.



# CHAPTER 1

---

## Dependencies

---

This driver depends on:

- [Adafruit CircuitPython](#)

Please ensure all dependencies are available on the CircuitPython filesystem. This is easily achieved by downloading the [Adafruit library and driver bundle](#).





## CHAPTER 2

---

### Installing from PyPI

---

On supported GNU/Linux systems like the Raspberry Pi, you can install the driver locally [from PyPI](#). To install for current user:

```
pip3 install adafruit-circuitpython-ntp
```

To install system-wide (this may be required in some cases):

```
sudo pip3 install adafruit-circuitpython-ntp
```

To install in a virtual environment in your current project:

```
mkdir project-name && cd project-name  
python3 -m venv .env  
source .env/bin/activate  
pip3 install adafruit-circuitpython-ntp
```



## CHAPTER 3

---

### Usage Example

---

```
import time
import board
import busio
from digitalio import DigitalInOut
from adafruit_esp32spi import adafruit_esp32spi
from adafruit_ntp import NTP

# If you are using a board with pre-defined ESP32 Pins:
esp32_cs = DigitalInOut(board.ESP_CS)
esp32_ready = DigitalInOut(board.ESP_BUSY)
esp32_reset = DigitalInOut(board.ESP_RESET)

# If you have an externally connected ESP32:
# esp32_cs = DigitalInOut(board.D9)
# esp32_ready = DigitalInOut(board.D10)
# esp32_reset = DigitalInOut(board.D5)

spi = busio.SPI(board.SCK, board.MOSI, board.MISO)
esp = adafruit_esp32spi.ESP_SPIcontrol(spi, esp32_cs, esp32_ready, esp32_reset)

print("Connecting to AP...")
while not esp.is_connected:
    try:
        esp.connect_AP(b"WIFI_SSID", b"WIFI_PASS")
    except RuntimeError as e:
        print("could not connect to AP, retrying: ", e)
        continue

# Initialize the NTP object
ntp = NTP(esp)

# Fetch and set the microcontroller's current UTC time
ntp.set_time()
```

(continues on next page)

(continued from previous page)

```
# Get the current time in seconds since Jan 1, 1970
current_time = time.time()
print("Seconds since Jan 1, 1970: {} seconds".format(current_time))
```

## CHAPTER 4

---

### Contributing

---

Contributions are welcome! Please read our [Code of Conduct](#) before contributing to help this project stay welcoming.



## CHAPTER 5

---

### Documentation

---

For information on building library documentation, please check out [this guide](#).





## 6.1 Simple test

Ensure your device works with this simple test.

Listing 1: examples/ntp\_simpletest.py

```
1 import time
2 import board
3 import busio
4 from digitalio import DigitalInOut
5 from adafruit_esp32spi import adafruit_esp32spi
6 from adafruit_ntp import NTP
7
8 # If you are using a board with pre-defined ESP32 Pins:
9 esp32_cs = DigitalInOut(board.ESP_CS)
10 esp32_ready = DigitalInOut(board.ESP_BUSY)
11 esp32_reset = DigitalInOut(board.ESP_RESET)
12
13 # If you have an externally connected ESP32:
14 # esp32_cs = DigitalInOut(board.D9)
15 # esp32_ready = DigitalInOut(board.D10)
16 # esp32_reset = DigitalInOut(board.D5)
17
18 spi = busio.SPI(board.SCK, board.MOSI, board.MISO)
19 esp = adafruit_esp32spi.ESP_SPIcontrol(spi, esp32_cs, esp32_ready, esp32_reset)
20
21 print("Connecting to AP...")
22 while not esp.is_connected:
23     try:
24         esp.connect_AP(b"WIFI_SSID", b"WIFI_PASS")
25     except RuntimeError as e:
26         print("could not connect to AP, retrying: ", e)
27     continue
```

(continues on next page)

(continued from previous page)

```

28
29 # Initialize the NTP object
30 ntp = NTP(esp)
31
32 # Fetch and set the microcontroller's current UTC time
33 # keep retrying until a valid time is returned
34 while not ntp.valid_time:
35     ntp.set_time()
36     print("Failed to obtain time, retrying in 5 seconds...")
37     time.sleep(5)
38
39 # Get the current time in seconds since Jan 1, 1970
40 current_time = time.time()
41 print("Seconds since Jan 1, 1970: {} seconds".format(current_time))
42
43 # Convert the current time in seconds since Jan 1, 1970 to a struct_time
44 now = time.localtime(current_time)
45 print(now)
46
47 # Pretty-parse the struct_time
48 print(
49     "It is currently {}/{} at {}:{}:{} UTC".format(
50         now.tm_mon, now.tm_mday, now.tm_year, now.tm_hour, now.tm_min, now.tm_sec
51     )
52 )

```

## 6.2 adafruit\_ntp

Network Time Protocol (NTP) helper for CircuitPython

- Author(s): Brent Rubell

### 6.2.1 Implementation Notes

**Hardware:**

**Software and Dependencies:**

- Adafruit CircuitPython firmware for the supported boards: <https://github.com/adafruit/circuitpython/releases>

**class** `adafruit_ntp.NTP` (*esp*)

Network Time Protocol (NTP) helper module for CircuitPython. This module does not handle daylight savings or local time.

**Parameters** `esp` (*adafruit\_esp32spi*) – ESP32SPI object.

**set\_time** ()

Fetches and sets the microcontroller's current time in seconds since since Jan 1, 1970.

## CHAPTER 7

---

### Indices and tables

---

- `genindex`
- `modindex`
- `search`



**a**

adafruit\_ntp, 14



## A

`adafruit_ntp` (*module*), 14

## N

`NTP` (*class in `adafruit_ntp`*), 14

## S

`set_time()` (*`adafruit_ntp.NTP` method*), 14