

PowerHub RTC 实时时钟

PowerHub RTC 实时时钟相关 API 与案例程序。

| 案例程序

| 编译要求

- M5Stack 板管理版本 $\geq 3.2.3$
- 开发板选项 = M5PowerHub
- M5Unified 库版本 $\geq 0.2.11$

```
1  #define WIFI_SSID "*****"
2  #define WIFI_PSWD "*****"
3
4  #define NTP_TIMEZONE "UTC-8" // POSIX standard, in which "UTC+0" is UTC London, "UTC-8" is UTC+8 Be
5  #define NTP_SERVER1 "0.pool.ntp.org"
6  #define NTP_SERVER2 "1.pool.ntp.org"
7  #define NTP_SERVER3 "2.pool.ntp.org"
8  #include <WiFi.h>
9
10 // Different versions of the framework have different SNTP header file names and availability.
11 #if __has_include(<esp_sntp.h>)
12 #include <esp_sntp.h>
13 #define SNTP_ENABLED 1
14 #elif __has_include(<sntp.h>)
15 #include <sntp.h>
16 #define SNTP_ENABLED 1
17 #endif
18
19 #ifndef SNTP_ENABLED
20 #define SNTP_ENABLED 0
21 #endif
22
23 #include <M5Unified.h>
24
25 static constexpr const char* const wd[7] = { "Sun", "Mon", "Tue", "Wed", "Thr", "Fri", "Sat" };
26
27 void setup() {
28     M5.begin();
29     Serial.begin(115200);
30     Serial.println("PowerHub RTC Test");
31
32     if (!M5.Rtc.isEnabled()) {
33         Serial.println("RTC not found");
34         while (true) {
35             delay(500);
36         }
37     }
38     Serial.println("RTC found");
39
40     Serial.print("WiFi: ");
41     WiFi.begin(WIFI_SSID, WIFI_PSWD);
42     while (WiFi.status() != WL_CONNECTED) {
43         Serial.print(".");
44         delay(500);
45     }
46     Serial.println("\nWiFi connected");
47
48     configTzTime(NTP_TIMEZONE, NTP_SERVER1, NTP_SERVER2, NTP_SERVER3);
49
50     Serial.print("NTP: ");
```

```

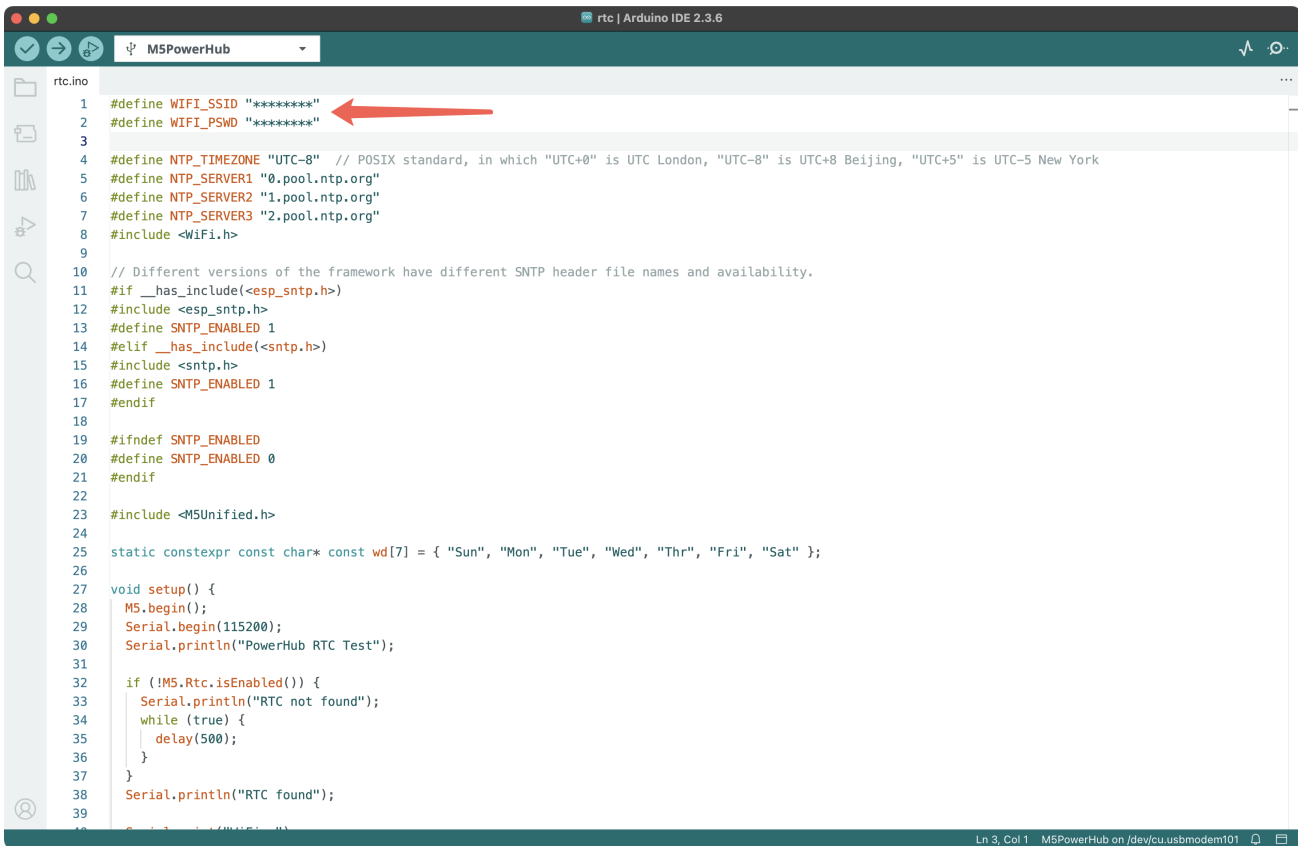
51 #if SNTP_ENABLED
52     while (sntp_get_sync_status() != SNTP_SYNC_STATUS_COMPLETED) {
53         Serial.print(".");
54         delay(500);
55     }
56 #else
57     struct tm timeInfo;
58     while (!getLocalTime(&timeInfo, 1000)) {
59         Serial.print(".");
60         delay(500);
61     }
62 #endif
63     Serial.println("\nNTP connected");
64
65     time_t t = time(nullptr) + 1; // Advance one second
66     while (t > time(nullptr))
67         ; // Synchronization in seconds
68     M5.Rtc.setDateTime(gmtime(&t));
69
70     delay(1000);
71 }
72
73 void loop() {
74     M5.update();
75
76     // RTC timer
77     auto dt = M5.Rtc.getDateTime();
78     Serial.printf("RTC   UTC   : ");
79     Serial.printf("%04d/%02d/%02d(%s) ", dt.date.year, dt.date.month, dt.date.date, wd[dt.date.weekDay]);
80     Serial.printf("%02d:%02d:%02d\n", dt.time.hours, dt.time.minutes, dt.time.seconds);
81
82     // ESP32 internal timer
83     auto t = time(nullptr);
84     {
85         auto tm = gmtime(&t); // for UTC
86         Serial.printf("ESP32 UTC   : ");
87         Serial.printf("%04d/%02d/%02d(%s) ", tm->tm_year + 1900, tm->tm_mon + 1, tm->tm_mday, wd[tm->tm_wday]);
88         Serial.printf("%02d:%02d:%02d\n", tm->tm_hour, tm->tm_min, tm->tm_sec);
89     }
90     {
91         auto tm = localtime(&t); // for local timezone
92         Serial.printf("ESP32 %s: ", NTP_TIMEZONE);
93         Serial.printf("%04d/%02d/%02d(%s) ", tm->tm_year + 1900, tm->tm_mon + 1, tm->tm_mday, wd[tm->tm_wday]);
94         Serial.printf("%02d:%02d:%02d\n", tm->tm_hour, tm->tm_min, tm->tm_sec);
95     }
96
97     delay(1000);
98 }

```

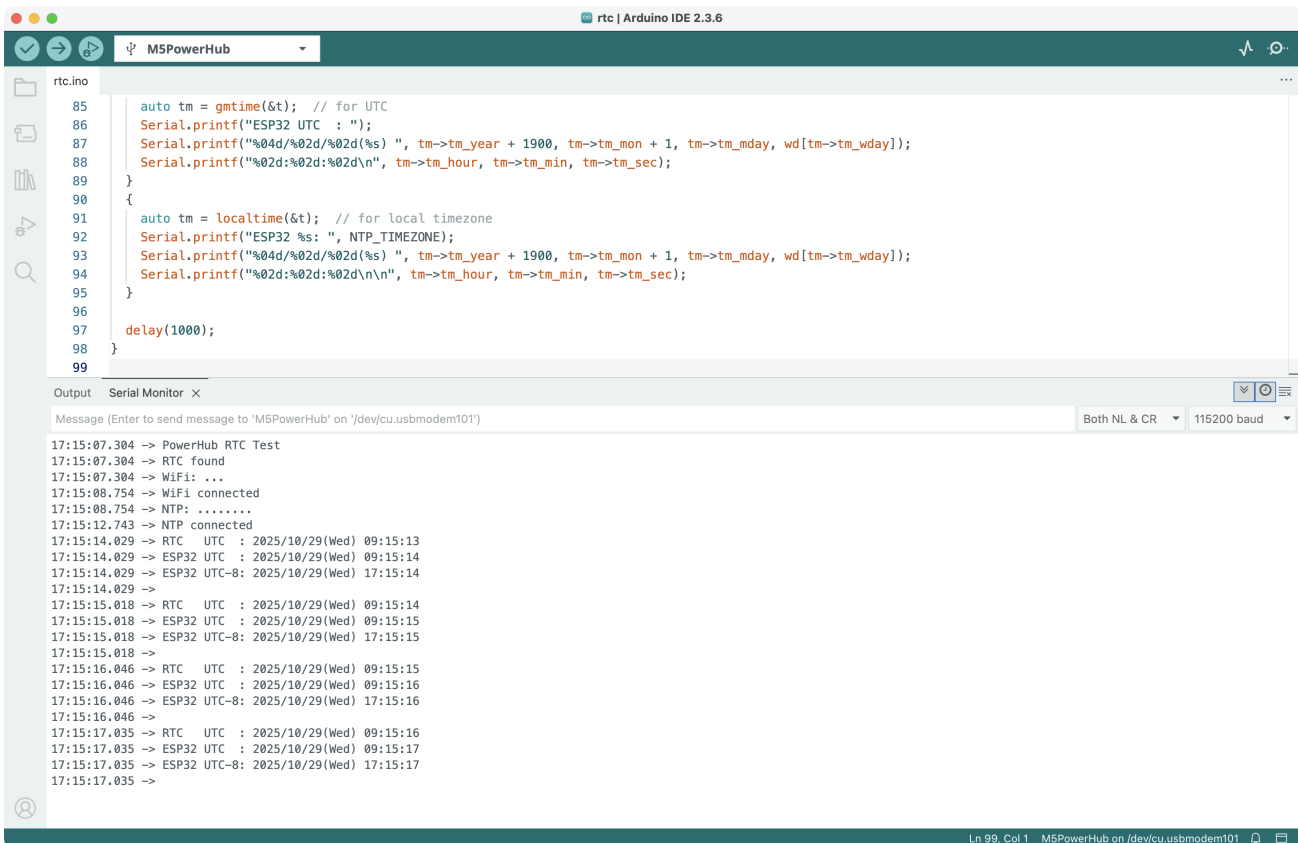
在程序中填写要连接的 Wi-Fi 名称及密码、你的当地时区，点击上传按钮将程序上传至 PowerHub，打开串口监视器即可显示实时时钟。

注意

程序中的时区定义 `NTP_TIMEZONE` 遵循 POSIX 标准，与常见的人类易读格式相反，比如 `"UTC+0"` 表示 UTC London，`"UTC-8"` 表示 UTC+8 Beijing，`"UTC+5"` 表示 UTC-5 New York。



```
1 #define WIFI_SSID "*****"
2 #define WIFI_PSWD "*****"
3
4 #define NTP_TIMEZONE "UTC-8" // POSIX standard, in which "UTC+0" is UTC London, "UTC-8" is UTC+8 Beijing, "UTC+5" is UTC-5 New York
5 #define NTP_SERVER1 "0.pool.ntp.org"
6 #define NTP_SERVER2 "1.pool.ntp.org"
7 #define NTP_SERVER3 "2.pool.ntp.org"
8 #include <WiFi.h>
9
10 // Different versions of the framework have different SNTP header file names and availability.
11 #if __has_include(<esp_sntp.h>)
12 #include <esp_sntp.h>
13 #define SNTP_ENABLED 1
14 #elif __has_include(<sntp.h>)
15 #include <sntp.h>
16 #define SNTP_ENABLED 1
17 #endif
18
19 #ifndef SNTP_ENABLED
20 #define SNTP_ENABLED 0
21 #endif
22
23 #include <M5Unified.h>
24
25 static constexpr const char* const wd[7] = { "Sun", "Mon", "Tue", "Wed", "Thr", "Fri", "Sat" };
26
27 void setup() {
28   M5.begin();
29   Serial.begin(115200);
30   Serial.println("PowerHub RTC Test");
31
32   if (!M5.Rtc.isEnabled()) {
33     Serial.println("RTC not found");
34     while (true) {
35       delay(500);
36     }
37   }
38   Serial.println("RTC found");
39 }
```



```
85 auto tm = gmtime(&t); // for UTC
86 Serial.printf("ESP32 UTC : ");
87 Serial.printf("%04d/%02d/%02d(%s) ", tm->tm_year + 1900, tm->tm_mon + 1, tm->tm_mday, wd[tm->tm_wday]);
88 Serial.printf("%02d:%02d:%02d\n", tm->tm_hour, tm->tm_min, tm->tm_sec);
89 }
90 {
91   auto tm = localtime(&t); // for local timezone
92   Serial.printf("ESP32 %s: ", NTP_TIMEZONE);
93   Serial.printf("%04d/%02d/%02d(%s) ", tm->tm_year + 1900, tm->tm_mon + 1, tm->tm_mday, wd[tm->tm_wday]);
94   Serial.printf("%02d:%02d:%02d\n\n", tm->tm_hour, tm->tm_min, tm->tm_sec);
95 }
96
97 delay(1000);
98 }
99 }
```

Output Serial Monitor X

Message (Enter to send message to 'M5PowerHub' on '/dev/cu.usbmodem101')

Both NL & CR 115200 baud

```
17:15:07.304 -> PowerHub RTC Test
17:15:07.304 -> RTC found
17:15:07.304 -> WiFi: ...
17:15:08.754 -> WiFi connected
17:15:08.754 -> NTP: .....
17:15:12.743 -> NTP connected
17:15:14.029 -> RTC UTC : 2025/10/29(Wed) 09:15:13
17:15:14.029 -> ESP32 UTC : 2025/10/29(Wed) 09:15:14
17:15:14.029 -> ESP32 UTC-8: 2025/10/29(Wed) 17:15:14
17:15:14.029 ->
17:15:15.018 -> RTC UTC : 2025/10/29(Wed) 09:15:14
17:15:15.018 -> ESP32 UTC : 2025/10/29(Wed) 09:15:15
17:15:15.018 -> ESP32 UTC-8: 2025/10/29(Wed) 17:15:15
17:15:15.018 ->
17:15:16.046 -> RTC UTC : 2025/10/29(Wed) 09:15:15
17:15:16.046 -> ESP32 UTC : 2025/10/29(Wed) 09:15:16
17:15:16.046 -> ESP32 UTC-8: 2025/10/29(Wed) 17:15:16
17:15:16.046 ->
17:15:17.035 -> RTC UTC : 2025/10/29(Wed) 09:15:16
17:15:17.035 -> ESP32 UTC : 2025/10/29(Wed) 09:15:17
17:15:17.035 -> ESP32 UTC-8: 2025/10/29(Wed) 17:15:17
17:15:17.035 ->
```

PowerHub RTC 实时时钟部分驱动使用了**M5Unified**库中的**RTC8563_Class**，更多相关的 API 可以参考下方文档：

- [M5Unified - RTC8563 Class](#)