

Unit Mini BPS Arduino 使用教程

| 1.准备工作

- 1.环境配置：参考[Arduino IDE上手教程](#)完成IDE安装, 并根据实际使用的开发板安装对应的板管理, 与需要的驱动库。
- 2.使用到的驱动库:
 - [M5Unified](#)
 - [M5GFX](#)
 - [M5UnitUnified](#)
 - [M5Unit-ENV](#)

驱动库

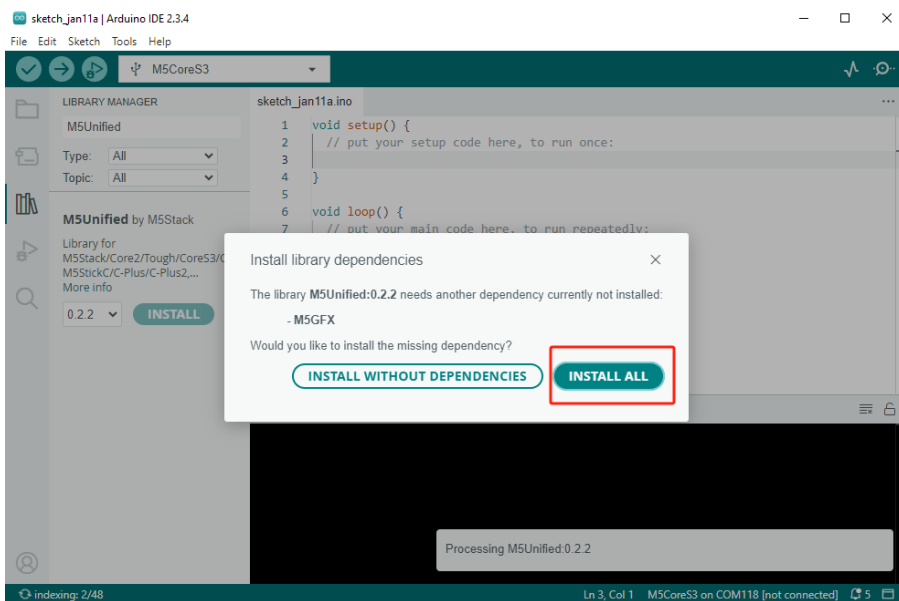
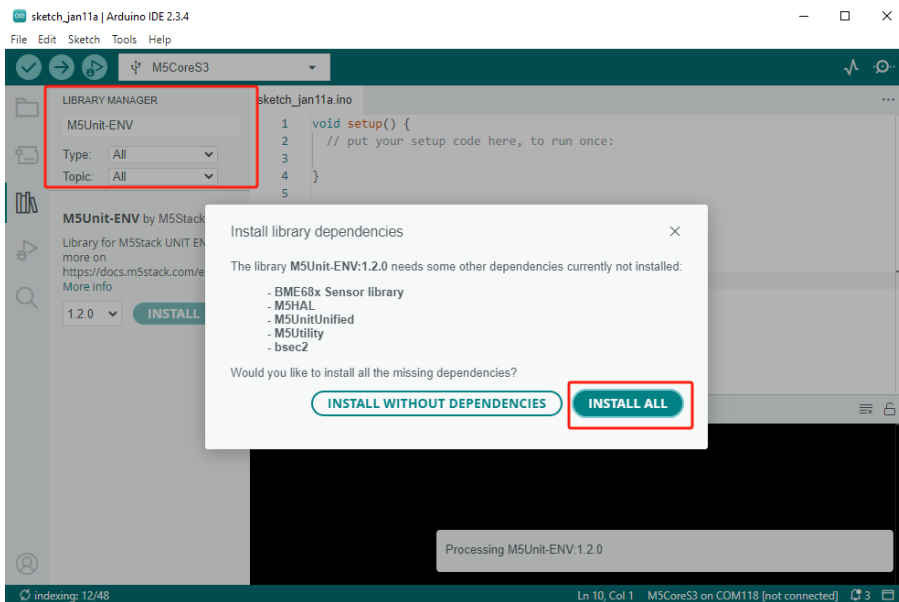
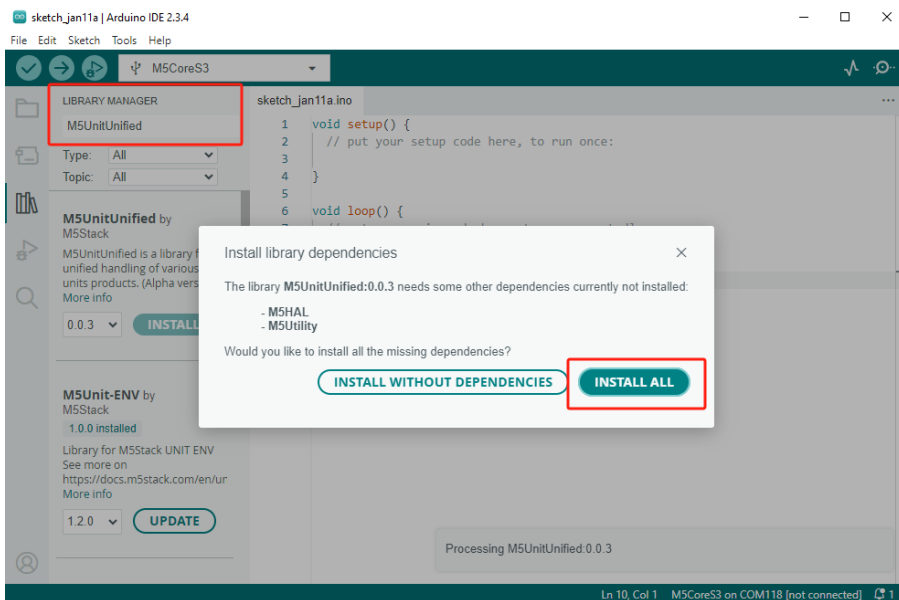
在 M5Unit-ENV 库中包含了对 Unit Mini BPS / Unit Mini BPS v1.1 传感器的驱动与案例，能够非常方便的实现传感器数据读取。

- 3.使用到的硬件产品:
 - [CoreS3](#)
 - [Unit Mini BPS v1.1](#)



依赖库

以上驱动库如M5UnitUnified, M5Unit-ENV等在安装时需要其他的依赖库(如[M5HAL](#), [M5Utility](#)等), 如果通过Arduino库管理进行安装时, 请根据提示, 安装全部依赖。



2.案例程序

案例说明

本案例以 Unit Mini BPS v1.1 为例，实现气压数据读取。如果使用的是 Unit Mini BPS (BMP280款) 可通过切换 `USING_BPS` 注释来启用不同的实例。

cpp



```
1  #include <M5Unified.h>
2  #include "M5UnitENV.h"
3
4  M5Canvas canvas(&M5.Display);
5
6  // #define USING_BPS
7  #define USING_BPS_V11
8
9  #if defined(USING_BPS)
10
11  BMP280 bmp;
12
13  #elif defined(USING_BPS_V11)
14
15  QMP6988 qmp;
16
17  #endif
18
19  float calculate_altitude(const float pressure, const float seaLvHpa = 1013.25f)
20  {
21      return 44330.f * (1.0f - pow((pressure / 100.f) / seaLvHpa, 0.1903f));
22  }
23
24  void setup()
25  {
26      M5.begin();
27      Serial.begin(115200);
28      M5.Display.setFont(&fonts::lgfxJapanMinchoP_20);
29      M5.Display.setTextSize(1);
30      auto pin_num_sda = M5.getPin(m5::pin_name_t::port_a_sda);
31      auto pin_num_scl = M5.getPin(m5::pin_name_t::port_a_scl);
32      M5_LOGI("getPin: SDA:%u SCL:%u", pin_num_sda, pin_num_scl);
33
34      Wire.begin(pin_num_sda, pin_num_scl, 400000U);
35
36      #if defined(USING_BPS)
37          if (!bmp.begin(&Wire, BMP280_I2C_ADDR, pin_num_sda, pin_num_scl, 400000U)) {
38              Serial.println("Couldn't find BMP280");
39              M5.Display.clear(TFT_RED);
40              while (1) delay(1);
41          }
42          /* Default settings from datasheet. */
43          bmp.setSampling(BMP280::MODE_NORMAL, /* Operating Mode. */
44                        BMP280::SAMPLING_X2, /* Temp. oversampling */
45                        BMP280::SAMPLING_X16, /* Pressure oversampling */
46                        BMP280::FILTER_X16, /* Filtering. */
```

```

47         BMP280::STANDBY_MS_500); /* Standby time. */
48
49 #elif defined(USING_BPS_V11)
50     if (!qmp.begin(&Wire, QMP6988_SLAVE_ADDRESS_L, pin_num_sda, pin_num_scl, 400000U)) {
51         Serial.println("Couldn't find QMP6988");
52         M5.Display.clear(TFT_RED);
53         while (1) delay(1);
54     }
55
56 #endif
57 }
58
59 void loop()
60 {
61     M5.update();
62
63     #if defined(USING_BPS)
64
65     if (bmp.update()) {
66         M5.Display.setCursor(0, 0);
67         M5.Display.clear(TFT_BLACK);
68         M5.Display.println("-----BMP280-----");
69         M5.Display.print(F("Temperature: "));
70         M5.Display.print(bmp.cTemp);
71         M5.Display.println(" degrees C");
72         M5.Display.print(F("Pressure: "));
73         M5.Display.print(bmp.pressure);
74         M5.Display.println(" Pa");
75         M5.Display.print(F("Approx altitude: "));
76         M5.Display.print(bmp.altitude);
77         M5.Display.println(" m");
78         M5.Display.println("-----\r\n");
79     }
80
81     #elif defined(USING_BPS_V11)
82
83     if (qmp.update()) {
84         M5.Display.setCursor(0, 0);
85         M5.Display.clear(TFT_BLACK);
86         M5.Display.println("-----QMP6988-----");
87         M5.Display.print(F("Temperature: "));
88         M5.Display.print(qmp.cTemp);
89         M5.Display.println(" *C");
90         M5.Display.print(F("Pressure: "));
91         M5.Display.print(qmp.pressure);
92         M5.Display.println(" Pa");
93         M5.Display.print(F("Approx altitude: "));
94         M5.Display.print(qmp.altitude);
95         M5.Display.println(" m");

```

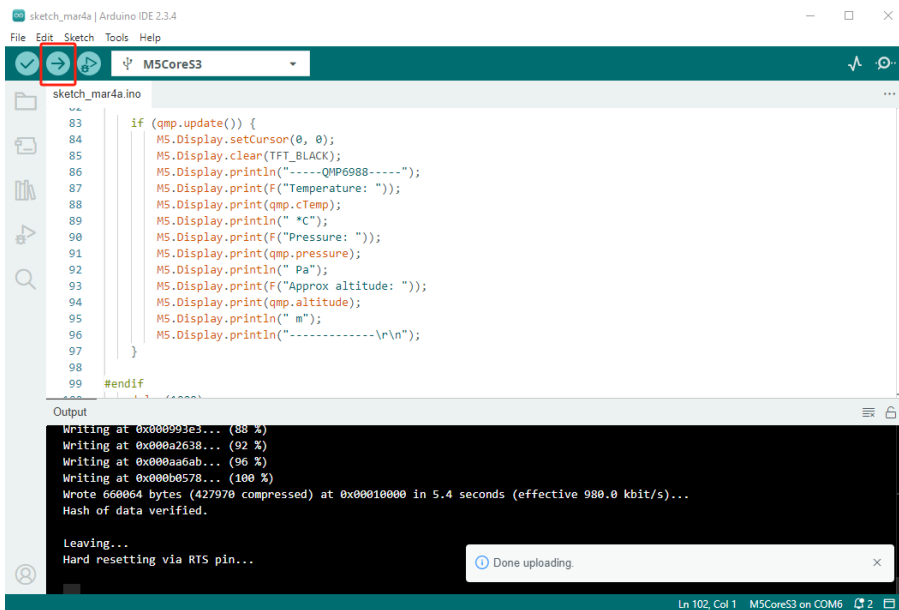
```
96     M5.Display.println("-----\r\n");
97 }
98
99 #endif
100     delay(1000);
101 }
```

3.编译上传

- 1.下载模式: 不同设备进行程序烧录前需要下载模式, 不同的主控设备该步骤可能有所不同。详情可参考[Arduino IDE上手教程](#) 页面底部的设备程序下载教程列表, 查看具体的操作方式。
- CoreS3长按复位按键(大约2秒)直到内部绿色LED灯亮起, 便可松开, 此时设备已进入下载模式, 等待烧录。



- 2.选中设备端口, 点击Arduino IDE左上角编译上传按钮, 等待程序完成编译并上传至设备。



4.气压值读取

