

Unit Mini Scales Arduino チュートリアル

1. 準備

- 環境設定：[Arduino IDE 入門チュートリアル](#)を参照して IDE をインストールし、使用する開発ボードに対応したボードマネージャーと必要なドライバライブラリをインストールします。
- 使用するドライバライブラリ：
 - [M5Unified](#)
 - [M5GFX](#)
 - [M5Unit-Miniscale](#)
- 使用するハードウェア製品：
 - [CoreS3](#)
 - [Unit Mini Scales](#)



2. 注意事項

ピン互換性

ホストデバイスごとにピン構成が異なります。M5Stack 公式のピン互換性表を確認し、実際のピン接続に合わせてサンプルプログラムを変更してください。

3. サンプルプログラム

- 本チュートリアルでは、CoreS3 と Unit Mini Scales を使用します。Unit Mini Scales は I2C 通信を使用します。接続後に使用するピンは `G2 (SDA)` と `G1 (SCL)` です。

注意

使用前に Unit Mini Scales に計量プレートを取り付けてください。最大計量範囲は `5kg` です。センサーの破損を防ぐため、定格範囲を超えないでください。初回測定前にゼロ点校正または風袋引きを行うことを推奨します。

cpp

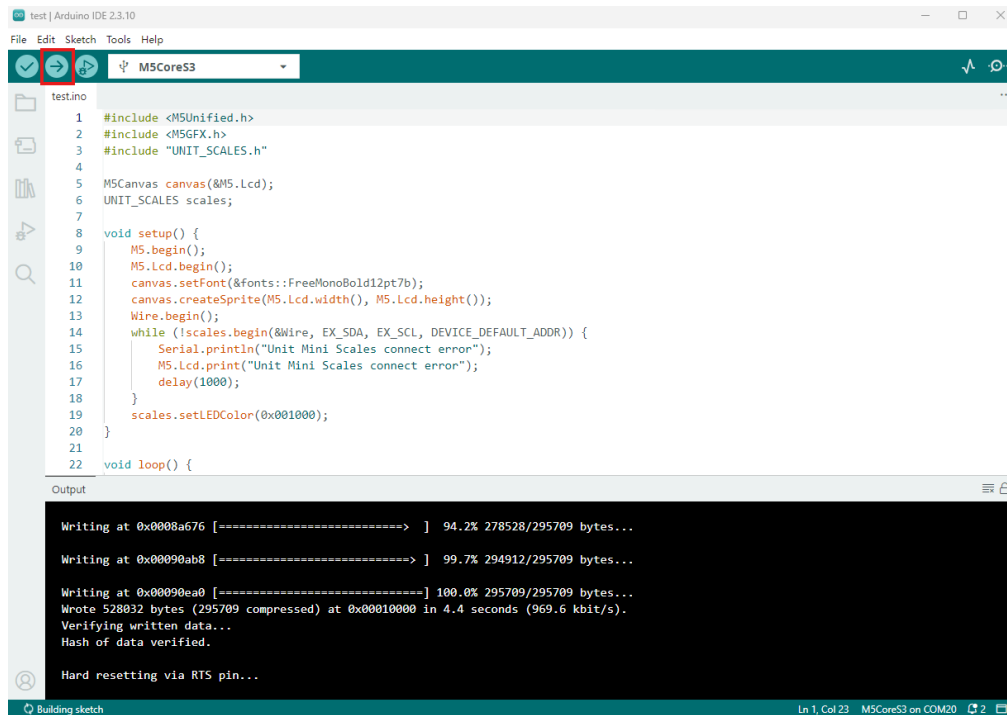
```

1  #include <M5Unified.h>
2  #include <M5GFX.h>
3  #include "UNIT_SCALES.h"
4
5  M5Canvas canvas(&M5.Lcd);
6  UNIT_SCALES scales;
7
8  void setup() {
9      M5.begin();
10     M5.Lcd.begin();
11     canvas.setFont(&fonts::FreeMonoBold12pt7b);
12     canvas.createSprite(M5.Lcd.width(), M5.Lcd.height());
13     // Initialize PORT.A I2C and retry until the unit responds.
14     Wire.begin();
15     while (!scales.begin(&Wire, EX_SDA, EX_SCL, DEVICE_DEFAULT_ADDR)) {
16         Serial.println("Unit Mini Scales connect error");
17         M5.Lcd.print("Unit Mini Scales connect error");
18         delay(1000);
19     }
20     // Set the unit status LED to dim green.
21     scales.setLEDColor(0x001000);
22 }
23
24 void loop() {
25     float weight = scales.getWeight();
26     float gap     = scales.getGapValue();
27     int  adc      = scales.getRawADC();
28
29     // Redraw the weight and raw ADC values.
30     canvas.fillRect(BLACK);
31     canvas.drawCenterString("Unit Mini Scale Test", 160, 10);
32     canvas.setTextColor(WHITE);
33     canvas.setCursor(10, 50);
34     canvas.printf("WEIGHT:");
35     canvas.setTextColor(GREEN);
36     canvas.printf("%.2fg", weight);
37     canvas.setCursor(10, 100);
38     canvas.setTextColor(WHITE);
39     canvas.printf("ADC:");
40     canvas.setTextColor(GREEN);
41     canvas.printf("%d", adc);
42     canvas.setTextColor(ORANGE);
43     canvas.drawCenterString("Touch Screen to Offset", 160, 200);
44     canvas.pushSprite(0, 0);
45
46     // Tap the screen to use the current load as the zero offset.
47     M5.update();
48     if (M5.Touch.getDetail().wasClicked()) {
49         scales.setOffset();
50     }
51 }

```

4. コンパイルとアップロード

- 上記のサンプルコードをプロジェクトのコード領域に貼り付け、デバイスポートを選択します。詳細は[プログラムのコンパイルと書き込み](#)を参照してください。Arduino IDE 左上のコンパイルおよびアップロードボタンをクリックし、プログラムのコンパイルとデバイスへのアップロードが完了するまで待ちます。



5. 計量とゼロ点設定

- Unit Mini Scales の接続に成功すると、モジュールのステータス LED が暗い緑色に点灯し、CoreS3 の画面には現在の重量と ADC 生データが継続的に表示されます。画面の任意の場所をタップすると `setOffset()` が呼び出され、現在の荷重をゼロ点に設定できます。風袋引き後に物体を載せたり取り除いたりすると、重量値と ADC 値の変化を確認できます。

