

Unit 8Angle Arduino ユーザーガイド

1. 準備作業

- 1. 環境設定: [Arduino IDE 入門チュートリアル](#) を参照して IDE のインストールを完了し、ご使用の開発ボードに対応するボードマネージャと必要なドライバライブラリをインストールしてください。
- 2. 使用するドライバライブラリ:
 - [M5Unified](#)
 - [M5GFX](#)
 - [M5Unit-8Angle](#)
- 3. 使用するハードウェア製品:
 - [CoreS3](#)
 - [Unit 8Angle](#)



2. サンプルプログラム

サンプル説明

Unit 8Angle は、8 チャンネルの可変ポテンショメータ、9 個のプログラム可能な RGB LED、および 1 つのトグルスイッチを統合した入力ユニットです。このサンプルでは、CoreS3 メインコントローラの PORT.A 拡張インターフェースを介して、各チャンネルのポテンショメータの入力状態とトグルスイッチの状態を読み取り、同時に RGB LED を制御します。

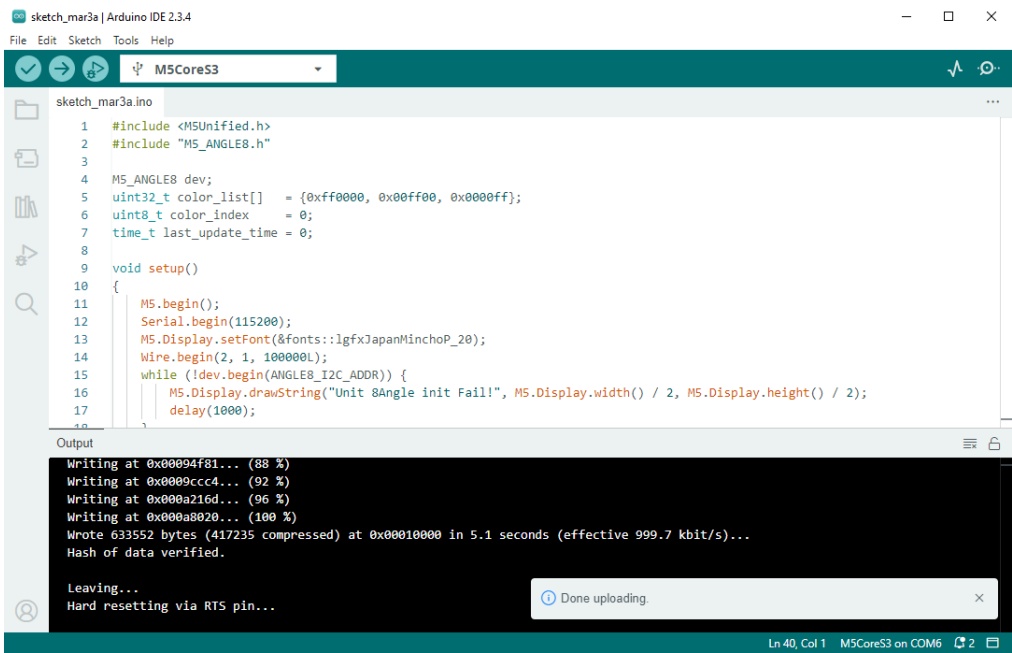
```
1  #include <M5Unified.h>
2  #include "M5_ANGLE8.h"
3
4  M5_ANGLE8 dev;
5  uint32_t color_list[] = {0xff0000, 0x00ff00, 0x0000ff};
6  uint8_t color_index = 0;
7  time_t last_update_time = 0;
8
9  void setup()
10 {
11     M5.begin();
12     Serial.begin(115200);
13     M5.Display.setFont(&fonts::lgfxJapanMinchoP_20);
14     Wire.begin(2, 1, 100000L);
15     while (!dev.begin(ANGLE8_I2C_ADDR)) {
16         M5.Display.drawString("Unit 8Angle init Fail!", M5.Display.width() / 2, M5.Display.height() / 2,
17             delay(1000);
18     }
19 }
20
21 void loop()
22 {
23     bool switch_status = dev.getDigitalInput();
24     M5.Display.clear();
25     M5.Display.setCursor(0, 0);
26     M5.Display.printf("Switch: %s\r\n", switch_status ? "ON" : "OFF");
27     for (uint8_t i = 0; i < ANGLE8_TOTAL_ADC; i++) {
28         uint16_t adc_v = dev.getAnalogInput(i, _12bit);
29         M5.Display.printf("CH[%d]: %d\r\n", i, adc_v);
30     }
31     if (millis() - last_update_time > 1000) {
32         color_index = (color_index + 1) % 3;
33         for (uint8_t i = 0; i < ANGLE8_TOTAL_LED; i++) {
34             dev.setLEDColour(i, color_list[color_index], 80);
35         }
36         last_update_time = millis();
37     }
38     delay(100);
39 }
```

3. コンパイルとアップロード

- 1. ダウンロードモード: 各デバイスはプログラム書き込み前にダウンロードモードに入る必要があり、この手順は主制御デバイスによって異なる場合があります。詳細は [Arduino IDE 入門チュートリアル](#) ページ下部のデバイスプログラム書き込みチュートリアル一覧を参照し、具体的な操作方法をご確認ください。
- CoreS3 の場合、リセットボタンを約 2 秒間長押しして、内部の緑色 LED が点灯するまで押し続けた後、ボタンを離すと、デバイスはダウンロードモードに入り、書き込み待機状態となります。



- 2. デバイスポートを選択し、Arduino IDE の左上にあるコンパイル&アップロードボタンをクリックして、プログラムがコンパイルされ、デバイスにアップロードされるのを待ちます。



4. 旋盤状態の読み取り

ポテンシオメータとスイッチの状態を読み取り表示すると同時に、RGB LED の色を 1 秒ごとに切り替えます。

