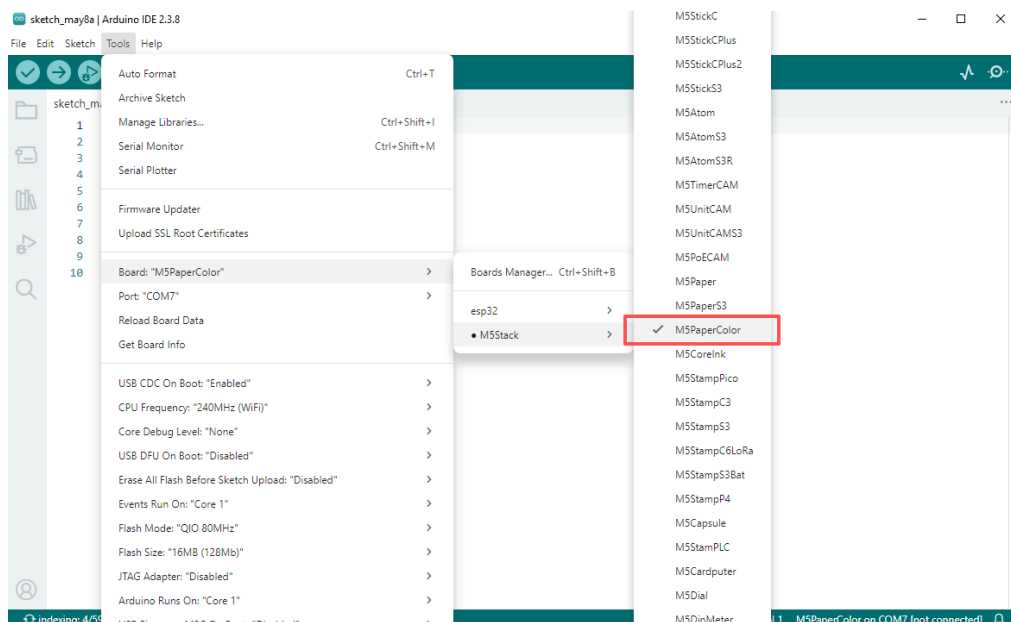


# PaperColor Arduino プログラムのコンパイルと書き込み

## 1. 準備作業

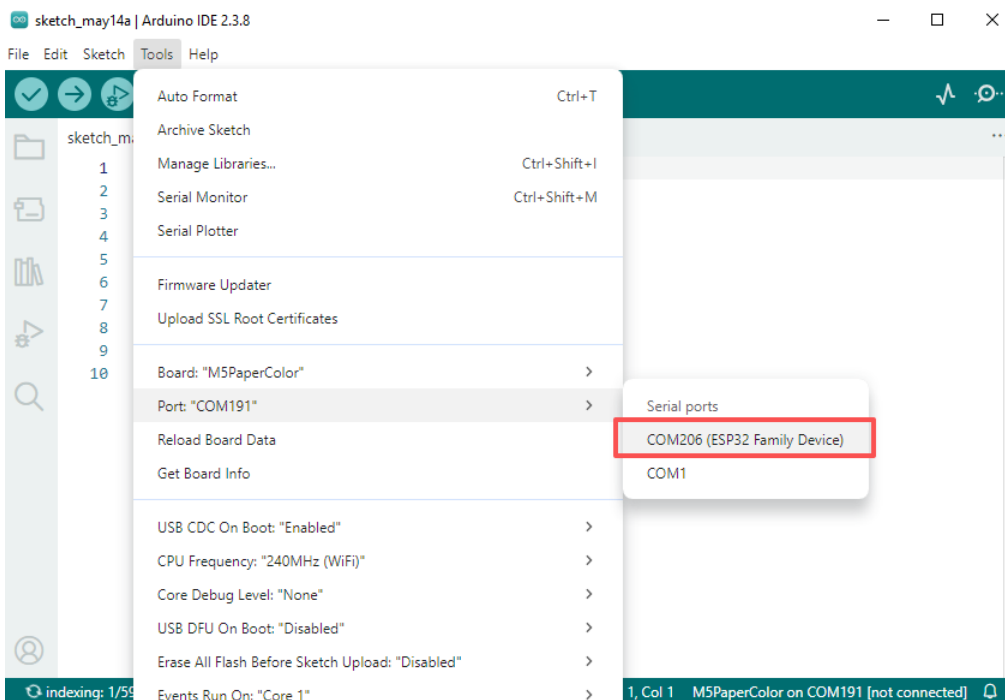
- 1. Arduino IDE のインストール: [Arduino IDE インストールチュートリアル](#)を参照し、IDE のインストールを完了してください。
- 2. ボードマネージャのインストール: [基本環境構築チュートリアル](#)を参照し、M5Stack ボードマネージャのインストールを完了し、開発ボード **M5PaperColor** を選択してください。



- 3. ドライバライブラリのインストール: [ライブラリ管理インストールチュートリアル](#)を参照し、**M5Unified** および **M5GFX** ドライバライブラリのインストールを完了し、表示されるプロンプトに従ってすべての依存ライブラリをインストールしてください。

## 2. ポートの選択

USB-C ケーブルでデバイスを PC に接続し、側面の電源ボタンを長押ししてデバイスがダウンロードモードに入ったら、Arduino IDE で対応するコントローラとデバイスポートを選択できます。



### 3.プログラムのコンパイル&書き込み

以下のサンプルプログラムを Arduino IDE にコピーし、アップロードボタンをクリックすると、プログラムがコンパイルされ PaperColor にアップロードされます。

```
1  #include <M5Unified.h>
2
3  M5Canvas Canvas(&M5.Display);
4
5  static void drawBoldText(M5Canvas& canvas, const String& text, int x, int y, uint16_t color)
6  {
7      // Draw the text in small offsets to emulate a bold weight on EPD.
8      canvas.setTextColor(color);
9      canvas.drawString(text, x, y);
10     canvas.drawString(text, x + 1, y);
11     canvas.drawString(text, x, y + 1);
12     canvas.drawString(text, x + 1, y + 1);
13 }
14
15 void setup()
16 {
17     auto cfg = M5.config();
18     cfg.clear_display = false;
19     M5.begin(cfg);
20     M5.Display.setEpdMode(epd_mode_t::epd_quality);
21
22     const int screen_w = M5.Display.width();
23     const int screen_h = M5.Display.height();
24
25     Canvas.createSprite(screen_w, screen_h);
26     Canvas.fillSprite(WHITE);
27
28     const uint16_t band_colors[6] = {YELLOW, RED, GREEN, BLUE, BLACK, WHITE};
29     const int band_h = screen_h / 6;
30
31     for (int i = 0; i < 6; ++i)
32     {
33         const int y = i * band_h;
34         const int h = (i == 5) ? (screen_h - y) : band_h;
35         Canvas.fillRect(0, y, screen_w, h, band_colors[i]);
36     }
37
38     const int white_band_y = 5 * band_h;
39     const int white_band_h = screen_h - white_band_y;
40
41     Canvas.setTextFont(4);
42     Canvas.setTextSize(2);
43     Canvas.setTextDatum(middle_left);
44     Canvas.setTextColor(BLACK);
45
46     const String text_paper = "PAPER";
47     const String text_color = "COLOR";
48     const int gap_w = Canvas.textWidth(" ");
49     const int total_w = Canvas.textWidth(text_paper) + gap_w + Canvas.textWidth(text_color);
50     const int start_x = (screen_w - total_w) / 2;
```

```

51     const int text_y          = white_band_y + (white_band_h / 2);
52
53     drawBoldText(Canvas, text_paper, start_x, text_y, BLACK);
54
55     int x = start_x + Canvas.textWidth(text_paper) + gap_w;
56     const uint16_t color_text_colors[5] = {RED, YELLOW, GREEN, YELLOW, BLUE};
57     for (int i = 0; i < text_color.length(); ++i)
58     {
59         const String ch = text_color.substring(i, i + 1);
60         drawBoldText(Canvas, ch, x, text_y, color_text_colors[i]);
61         x += Canvas.textWidth(ch);
62     }
63
64     Canvas.pushSprite(0, 0);
65 }
66
67 void loop()
68 {
69     M5.update();
70     delay(100);
71 }

```

The screenshot shows the Arduino IDE interface. The top menu bar includes File, Edit, Sketch, Tools, and Help. The toolbar shows icons for opening files, saving, and uploading. The sketch is named 'M5PaperColor'. The code editor displays the sketch\_may8a.ino file with the following content:

```

1  #include <M5Unified.h>
2
3  M5Canvas Canvas(&M5.Display);
4
5  static void drawBoldText(M5Canvas& canvas, const String& text, int x, int y, uint16_t color)
6  {
7      // Draw the text in small offsets to emulate a bold weight on EPD.
8      canvas.setTextColor(color);
9      canvas.drawString(text, x, y);
10     canvas.drawString(text, x + 1, y);
11     canvas.drawString(text, x, y + 1);
12     canvas.drawString(text, x + 1, y + 1);
13 }
14
15 void setup()
16 {
17     auto cfg = M5.config();
18     cfg.clear_display = false;
19     M5.begin(cfg);
20     M5.Display.setEpdMode(epd_mode_t::epd_quality);
21
22     const int screen_w = M5.Display.width();
23     const int screen_h = M5.Display.height();
24
25 }
26
27 void loop()
28 {
29 }

```

The Output window at the bottom shows the following messages:

```

Writing at 0x0008acd9 [=====] 91.2% 278528/305339 bytes...
Writing at 0x00090ecb [=====] 96.6% 294912/305339 bytes...
Writing at 0x00094c00 [=====] 100.0% 305339/305339 bytes...
Wrote 542744 bytes (305339 compressed) at 0x00010000 in 4.8 seconds (906.6 kbit/s).
Hash of data verified.
Hard resetting via RTS pin...

```

The status bar at the bottom indicates 'Ln 71, Col 2' and 'M5PaperColor on COM206'.

実行結果:



## 4.関連リソース

---

- [Arduino Library](#)
  - [M5Unified](#)
  - [M5GFX](#)
- [Arduino API & Examples](#)
  - [Button](#)
  - [Battery](#)
  - [Display](#)
  - [IR NEC](#)
  - [RGB LED](#)
  - [MIC](#)
  - [Speaker](#)
  - [microSD](#)
  - [RTC](#)
  - [SHT40](#)
  - [Wakeup](#)
  - [M5PM1](#)