

# M5Paper 環境モニターソース

美都

2021 年 3 月 11 日

## 目次

1	メイン	2
2	バッテリーメーター	3
2.1	battery.h . . . . .	3
2.2	battery.cpp . . . . .	3

# 1 メイン

```
1  #include <M5EPD.h>
2  #define LGFX_M5PAPER
3  #include <LovyanGFX.hpp>
4
5  #include "battery.h"
6
7  static LGFX lcd;
8
9  void setup()
10 {
11     M5.begin(false, true, true, true, true);
12     M5.BatteryADCBegin();
13     M5.RTC.begin();
14     lcd.init();
15     lcd.setRotation(1);
16
17
18 }
19
20
21 void loop()
22 {
23     int bat = drawBattery(960-120-5, 5, &lcd);
24     if (bat > 90 || bat==1) {
25         delay(5000);
26     } else {
27         delay(2000);
28         M5.shutdown(8); // 一旦停止
29     }
30 }
```

## 2 バッテリメーター

### 2.1 battery.h

```
1 #include <M5EPD.h>
2 #define LGFX_M5PAPER
3 #include <LovyanGFX.hpp>
4
5 // バッテリー残量を(x,y)に表示する。
6 int drawBattery(int x, int y, LGFX *lcd) ;
```

### 2.2 battery.cpp

```
1 #include "battery.h"
2
3 // バッテリー残量の取得
4 static int get_rest_battery() {
5     const int max_vol = 4350;
6     const int min_vol = 3300;
7     //M5.BatteryADCBegin();
8     int voltage = M5.getBatteryVoltage();
9     voltage = max(voltage, min_vol);
10    voltage = min(voltage, max_vol);
11    float rest_battery_raw = (float)(voltage - min_vol) / (float)(max_vol - min_vol);
12    rest_battery_raw = max(rest_battery_raw, 0.01f);
13    rest_battery_raw = min(rest_battery_raw, 1.f);
14    return (int)(rest_battery_raw * 100);
15 }
16
17 // バッテリー残量計の表示
18 int drawBattery(int x, int y, LGFX *lcd) {
19     LGFX_Sprite battery_meter(lcd);
20     int rest_battery = get_rest_battery();
21
22     // バッテリー矩形の表示
23     battery_meter.setColorDepth(4);
24     battery_meter.createSprite(120, 30);
25     battery_meter.fillSprite(15);
26     battery_meter.setColor(0);
27     battery_meter.drawRect(10, 10, 45, 20);
28     battery_meter.fillRect(55, 17, 5, 5);
29     battery_meter.fillRect(10, 10, (int)((45*rest_battery)/100), 20);
30
31     // バッテリー残量文字の表示
32     battery_meter.setFont(&font::lgfxJapanMinchoP_20);
33     battery_meter.setTextSize(1, 1); // 縦,横 倍率
34     battery_meter.setTextColor(0, 15); // 文字色,背景
35     battery_meter.setCursor(62, 10);
36     battery_meter.printf("%d%%", rest_battery);
37
38     lcd->startWrite();
39     battery_meter.pushSprite(x, y);
40     lcd->endWrite();
41
42     return rest_battery;
```

