M5Paper 環境モニターソース

美都

2021 年 3 月 12 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

3 温湿度計 5

3.1 thermometer.hpp 5

3.2 thermometer.cpp 5

1 メイン

```
#include <M5EPD.h>
#define LGFX_M5PAPER
#include <LovyanGFX.hpp>

#include "battery.h"
#include "thermometer.hpp"

static LGFX lcd;

void setup()
{
    M5.begin(false, true, true, true, true);
    M5.BatteryADCBegin();
    M5.RTC.begin();
    M5.SHT30.Begin();
    lcd.init();
    lcd.setRotation(1);
}


void loop()
{
    int bat = drawBattery(960-120-5, 5, &lcd);
    Thermometer t = Thermometer(350,350);
    t.drawString(10, 10, &lcd);
    t.drawTempMeter(87, 100, &lcd);
    t.drawHumMeter(524, 100, &lcd);

    if (bat > 90 || bat==1) {
        delay(5000);
    } else {
        delay(2000);
        M5.shutdown(58); // 一旦停止
    }
}
```

2 バッテリーメーター

2.1 battery.h

```
#include <M5EPD.h>
#define LGFX_M5PAPER
#include <LovyanGFX.hpp>

// バッテリー残量を(x,y)に表示する。
int drawBattery(int x, int y, LGFX *lcd) ;
```

2.2 battery.cpp

```
#include "battery.h"

// バッテリー残量の取得
static int get_rest_battery() {
    const int max_vol = 4350;
    const int min_vol = 3300;
    //M5.BatteryADCBegin();
    int voltage = M5.getBatteryVoltage();
    voltage = max(voltage, min_vol);
    voltage = min(voltage, max_vol);
    float rest_battery_raw = (float)(voltage - min_vol) / (float)(max_vol - min_vol);
    rest_battery_raw = max(rest_battery_raw, 0.01f);
    rest_battery_raw = min(rest_battery_raw, 1.f);
    return (int)(rest_battery_raw * 100);
}

// バッテリー残量計の表示
int drawBattery(int x, int y, LGFX *lcd) {
    LGFX_Sprite battery_meter(lcd);
    int rest_battery = get_rest_battery();

    // バッテリー矩形の表示
    battery_meter.setColorDepth(4);
    battery_meter.createSprite(120, 30);
    battery_meter.fillSprite(15);
    battery_meter.setColor(0);
    battery_meter.drawRect(10, 10, 45, 20);
    battery_meter.fillRect(55, 17, 5, 5);
    battery_meter.fillRect(10, 10, (int)((45*rest_battery)/100), 20);

    // バッテリー残量文字の表示
    battery_meter.setFont(&fonts::lgfxJapanMinchoP_20);
    battery_meter.setTextSize(1, 1); // 縦,横　倍率
    battery_meter.setTextColor(0, 15); // 文字色,背景
    battery_meter.setCursor(62, 10);
    battery_meter.printf("%d%%", rest_battery);

    lcd->startWrite();
    battery_meter.pushSprite(x, y);
    lcd->endWrite();

    return rest_battery;
```

```
43 }
```

3 温湿度計

3.1 thermometer.hpp

```
#include <M5EPD.h>
#define LGFX_M5PAPER
#include <LovyanGFX.hpp>

class Thermometer {
    private:
        float temp;
        float hum;
        int sizex;
        int sizey;
        float radius;
        LGFX_Sprite face;
        LGFX_Sprite scale[2];
        LGFX_Sprite hand;

        void makeMeterFace(int min, int max, const char* unit);
        void makeScale();
        void makeHand();
    public:
        Thermometer(int sizex=200, int sizey=200);

        float get_temp();
        float get_hum();

        void drawTempMeter(int x, int y, LGFX *lcd);
        void drawHumMeter(int x, int y, LGFX *lcd);
        void drawString(int x, int y, LGFX *lcd);
};
```

3.2 thermometer.cpp

```
#include "thermometer.hpp"

Thermometer::Thermometer(int sizex, int sizey)
    : sizex(sizex), sizey(sizey) {
    M5.SHT30.Begin();
    radius = min(sizex, sizey)/2*0.95;
    makeScale();
    makeHand();
};

void Thermometer::makeScale() {
    scale[0].setColorDepth(4);
    scale[0].createSprite(radius/25, radius/5);
    scale[0].fillSprite(0);
    scale[0].setPivot(scale[0].width()/2, scale[0].height());
    scale[1].setColorDepth(4);
    scale[1].createSprite(radius/40, radius/7);
    scale[1].fillSprite(0);
    scale[1].setPivot(scale[1].width()/2, scale[1].height());
}
```

```
21
22  void Thermometer::makeHand() {
23      float height, width;
24      height = radius * 0.8f;
25      width = height * 0.1f;
26
27      hand.setColorDepth(4);
28      hand.createSprite(width, height);
29      hand.fillSprite(15);
30      hand.setColor(0);
31      hand.fillTriangle(width/2.f, 0, 0, height/4.f, width, height/4.f);
32      hand.fillTriangle(0, height/4.f, width, height/4.f, width/2.f, height);
33      hand.setPivot(width/2., height);
34  }
35
36  void Thermometer::makeMeterFace(int min, int max, const char* unit) {
37      face.setColorDepth(4);
38      face.createSprite(sizex, sizey);
39      face.fillSprite(15);
40      face.setColor(0);
41      face.setFont(&fonts::lgfxJapanMinchoP_36);
42      face.setTextColor(0, 15);
43      face.setTextDatum(middle_center);
44      float center[2] = {sizex/2.0f, sizey/2.0f};
45      face.fillCircle(center[0], center[1], radius);
46      face.fillCircle(center[0], center[1], radius*0.95, 15);
47      float angleInterval = 270.f / (float)(max-min);
48      for (int i = min ; i <= max ; i+=2) {
49          LGFX_Sprite *use_scale = (i%10==0) ? &scale[0] : &scale[1];
50          float angle = (270.f-45.f) - (float)(i-min) * angleInterval;
51          float angleRad = angle * 3.14159265f / 180.f ;
52          float startx = (radius - use_scale->height()) * cos(angleRad) + center[0];
53          float starty = -1.0f * ((radius - use_scale->height()) * sin(angleRad)) + center[1];
54          use_scale->pushRotateZoom(&face, startx, starty, 90.f-angle, 1.f, 1.f);
55          if (i%10==0) {
56              float charsize = (float)scale[0].height() / 36.f;
57              float charx = (radius - use_scale->height() * 1.5f) * cos(angleRad) + center[0];
58              float chary = -1.f * (radius - use_scale->height() * 1.5f) * sin(angleRad) +
                  center[0];
59              face.setTextSize(charsize);
60              face.drawNumber(i, charx, chary);
61          }
62          face.drawString(unit, center[0], sizey/5.f*3.f);
63      }
64
65  }
66
67  void Thermometer::drawTempMeter(int x, int y, LGFX *lcd) {
68      makeMeterFace(0, 50, "℃");
69      float center[2] = {(float)sizex/2.f, (float)sizey/2.f};
70      float angle = 270.f - 45.f;
71      angle -= 270.f / 50.f * get_temp();
72      hand.pushRotateZoom(&face, center[0], center[1], 90.f - angle, 1.f, 1.f);
73      face.pushSprite(lcd, x, y);
74  }
75
76  void Thermometer::drawHumMeter(int x, int y, LGFX *lcd) {
77      makeMeterFace(20, 80, "%");
```

```
    float center[2] = {(float)sizex/2.f, (float)sizey/2.f};
    float angle = 270.f - 45.f;
    angle -= 270.f / 60.f * (get_hum() - 20.f);
    hand.pushRotateZoom(&face, center[0], center[1], 90.f - angle, 1.f, 1.f);
    face.pushSprite(lcd, x, y);
}

void Thermometer::drawString(int x, int y, LGFX *lcd) {
    M5.SHT30.UpdateData();
    LGFX_Sprite meter(lcd);
    meter.setColorDepth(4);
    meter.createSprite(250, 100);
    meter.fillSprite(15);
    meter.setColor(0);
    meter.setTextColor(0, 15);
    meter.setFont(&fonts::lgfxJapanMinchoP_36);
    meter.setCursor(10,10);
    meter.printf("温度:%5.1f℃", this->get_temp());
    meter.setCursor(10,50);
    meter.printf("湿度:%5.1f%%", this->get_hum());
    meter.pushSprite(x, y);
}

float Thermometer::get_temp() {
    M5.SHT30.UpdateData();
    this->temp = M5.SHT30.GetTemperature();
    return this->temp;
}

float Thermometer::get_hum() {
    M5.SHT30.UpdateData();
    this->hum = M5.SHT30.GetRelHumidity();
    return this->hum;
}
```