CoreMatrix microSD 卡

CoreMatrix microSD 卡相关 API 与案例程序。

案例程序

编译要求

- M5Stack 板管理版本 >= `3.3.10-a`
- 开发板选项 = `M5CoreMatrix`
- M5Unified 库版本 >= `0.2.29`
- M5GFX 库版本 >= `0.2.36`

CoreMatrix 的 microSD 使用 SPI `G25`（SCK）、`G26`（MISO）、`G27`（MOSI）、`G28`（CS），卡槽电源由 M5IOE1 `IO1` / `PYB_TF_EN` 控制，插卡检测由 `IO10` / `PYB_TFCARD_DET` 读取，低电平表示已插入。M5Unified 初始化时默认打开卡槽电源，请准备格式化为 FAT32 的 microSD 卡。

cpp

```cpp
#include <Arduino.h>
#include <SPI.h>
#include <SD.h>
#include <M5Unified.h>
#include <utility/M5IOE1_Class.hpp>

static constexpr uint8_t SD_CS_PIN = 28;
static constexpr uint8_t SD_SCK_PIN = 25;
static constexpr uint8_t SD_MISO_PIN = 26;
static constexpr uint8_t SD_MOSI_PIN = 27;

static constexpr auto TF_DET = m5::M5IOE1_Class::gpio10;

static void drawResult(bool ok)
{
  M5.Display.startWrite();
  M5.Display.fillScreen(TFT_BLACK);
  M5.Display.drawRect(0, 0, 16, 16, TFT_WHITE);

  if (ok) {
    M5.Display.drawLine(2, 7, 6, 11, TFT_WHITE);
    M5.Display.drawLine(7, 10, 13, 4, TFT_WHITE);
  } else {
    M5.Display.drawLine(3, 3, 12, 12, TFT_WHITE);
    M5.Display.drawLine(12, 3, 3, 12, TFT_WHITE);
  }

  M5.Display.endWrite();
  M5.Display.display();
}

static bool prepareTfCard()
{
  auto& ioe = M5.getIOExpander(0);

  // Card detect is active low: M5IOE1 IO10 -> PYB_TFCARD_DET
  ioe.setDirection(TF_DET, false);
  ioe.setPullMode(TF_DET, m5::IOExpander_Base::pull_up);
  delay(50);

  bool level = true;
  if (!ioe.getInputLevel(TF_DET, &level)) {
    Serial.println("TF card detect read failed");
    return false;
  }

  Serial.printf("TF card detect level: %s\n", level ? "HIGH" : "LOW");
  return !level;
}

void setup()
{
  auto cfg = M5.config();
  cfg.serial_baudrate = 115200;
  M5.begin(cfg);
```

```
  M5.Display.setColorDepth(8);
  M5.Display.setRotation(1);
  M5.Display.setBrightness(128);

  if (!prepareTfCard()) {
    Serial.println("TF card not inserted");
    drawResult(false);
    return;
  }

  SPI.begin(SD_SCK_PIN, SD_MISO_PIN, SD_MOSI_PIN, SD_CS_PIN);
  if (!SD.begin(SD_CS_PIN, SPI, 25000000)) {
    Serial.println("SD mount failed");
    drawResult(false);
    return;
  }

  File file = SD.open("/corematrix.txt", FILE_WRITE);
  if (!file) {
    Serial.println("File open for write failed");
    drawResult(false);
    return;
  }

  file.println("Hello CoreMatrix");
  file.close();

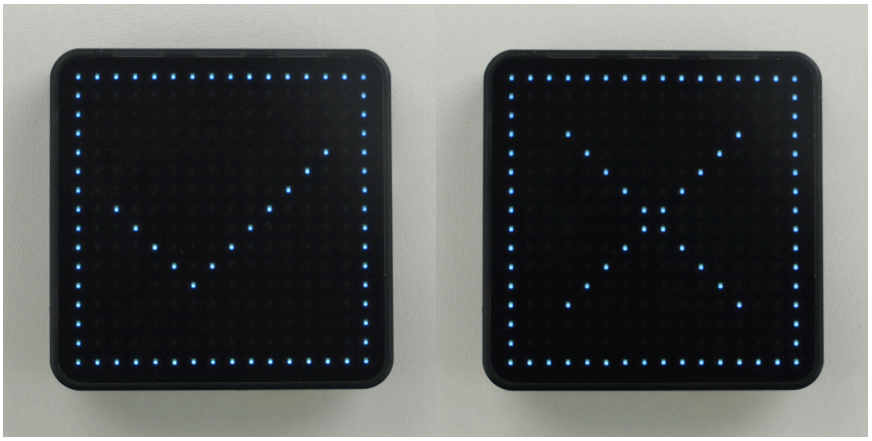
  file = SD.open("/corematrix.txt", FILE_READ);
  if (!file) {
    Serial.println("File open for read failed");
    drawResult(false);
    return;
  }

  const String line = file.readStringUntil('\n');
  file.close();

  Serial.printf("Read back: %s\n", line.c_str());
  drawResult(true);
}

void loop()
{
  delay(1000);
}
```

程序检测插卡、挂载文件系统、创建 /corematrix.txt 并重新读取。成功时点阵显示对勾，失败时显示叉号，详细结果通过 115200 波特率串口输出。若启动时未插入卡，插入后复位设备即可重新检测。



API

CoreMatrix microSD 部分使用 Arduino 标准 SD 标准库，更多相关的 API 可以参考下方文档：

- SD | Arduino Doc
- Guide to SD Storage | Arduino Doc