

# StickS3 Mic 麦克风

StickS3 Mic 相关 API 与案例程序。

## | 案例程序

---

### | 编译要求

- M5Stack 板管理版本  $\geq 3.2.5$
- 开发板选项 = M5StickS3
- M5Unified 库版本  $\geq 0.2.12$
- M5GFX 库版本  $\geq 0.2.18$

```
1  #include "M5Unified.h"
2
3  static constexpr const size_t record_number      = 256;
4  static constexpr const size_t record_length      = 256;
5  static constexpr const size_t record_size        = record_number * record_length;
6  static constexpr const size_t record_samplerate  = 18000;
7  static size_t rec_record_idx = 2;
8  static size_t draw_record_idx = 0;
9  static int16_t *rec_data;
10 static int32_t w;
11
12 static bool indicator_visible = true;
13 static uint32_t indicator_timer = 0;
14 static constexpr uint32_t indicator_interval = 500; //ms
15 static uint16_t indicator_color = RED;
16
17 static void updateIndicator(void) {
18     uint32_t now = millis();
19     if (now - indicator_timer >= indicator_interval) {
20         indicator_timer = now;
21         indicator_visible = !indicator_visible;
22     }
23     if (indicator_visible) {
24         M5.Lcd.fillCircle(85, 10, 8, indicator_color);
25     } else {
26         M5.Lcd.fillCircle(85, 10, 8, BLACK);
27     }
28 }
29
30 void setup(void) {
31     M5.begin();
32     M5.Lcd.setRotation(1);
33     M5.Lcd.setTextDatum(middle_center);
34     M5.Lcd.setTextColor(WHITE);
35     M5.Lcd.setFont(&fonts::FreeMonoBold9pt7b);
36     w = M5.Lcd.width();
37     rec_data = (typeof(rec_data))heap_caps_malloc(record_size * sizeof(int16_t), MALLOC_CAP_8BIT);
38     memset(rec_data, 0, record_size * sizeof(int16_t));
39     M5.Speaker.setVolume(200);
40     // Since the microphone and speaker cannot be used at the same time,
41     // turn off the speaker here.
42     M5.Speaker.end();
43     M5.Mic.begin();
44     indicator_color = RED;
45     updateIndicator();
46     M5.Lcd.drawString("REC", w / 2, 12);
47     M5.Lcd.drawString("BtnA to play record", w / 2, 80);
48 }
49
50 void loop(void) {
```

```

51 M5.update();
52 if (M5.Mic.isEnabled()) {
53     static constexpr int shift = 6;
54     auto data = &rec_data[rec_record_idx * record_length];
55     updateIndicator();
56     if (M5.Mic.record(data, record_length, record_samplerate)) {
57         data = &rec_data[draw_record_idx * record_length];
58         M5.Lcd.display();
59         updateIndicator();
60         M5.Lcd.drawString("REC", w / 2, 12);
61         M5.Lcd.drawString("BtnA to play record", w / 2, 80);
62         if (++draw_record_idx >= record_number) {
63             draw_record_idx = 0;
64         }
65         if (++rec_record_idx >= record_number) {
66             rec_record_idx = 0;
67         }
68     }
69 }
70 if (M5.BtnA.wasPressed()) {
71     if (M5.Speaker.isEnabled()) {
72         while (M5.Mic.isRecording()) {
73             delay(1);
74         }
75         M5.Lcd.clear();
76         indicator_color = GREEN;
77         indicator_visible = true;
78         indicator_timer = millis();
79         updateIndicator();
80         M5.Lcd.drawString("PLAY", w / 2, 12);
81         // Since the microphone and speaker cannot be used at the same time,
82         // turn off the microphone here.
83         M5.Mic.end();
84         M5.Speaker.begin();
85         int start_pos = rec_record_idx * record_length;
86         if (start_pos < record_size) {
87             M5.Speaker.playRaw(&rec_data[start_pos],
88                               record_size - start_pos,
89                               record_samplerate, false, 1, 0);
90         }
91         if (start_pos > 0) {
92             M5.Speaker.playRaw(rec_data, start_pos, record_samplerate,
93                               false, 1, 0);
94         }
95         do {
96             delay(1);
97             M5.update();
98             updateIndicator();
99         } while (M5.Speaker.isPlaying());
100        // Since the microphone and speaker cannot be used at the same time,
101        // turn off the speaker here.
102        M5.Speaker.end();
103        M5.Mic.begin();

```

```

104         M5.Lcd.clear();
105         indicator_color = RED;
106         indicator_visible = true;
107         indicator_timer = millis();
108         updateIndicator();
109         M5.Lcd.drawString("REC", w / 2, 12);
110     }
111 }
112 }

```

该程序的功能是，屏幕上会显示麦克风录音状态，并在按下按键 A 时播放录音，录音播放完毕后会自动返回录音状态。



## API

StickS3 Mic 部分使用了 M5Unified 库中的 Mic\_Class 及 Speaker\_Class, 更多相关的API可以参考下方文档:

- [M5Unified - Mic Class](#)
- [M5Unified - Speaker Class](#)