

StackChan Mic

Microphone-related APIs and sample programs for the StackChan.

| Sample Program

| Build Requirements

- M5Stack board manager version $\geq 3.2.2$
- Development board selection = M5CoreS3
- M5Unified library version $\geq 0.2.11$

```
1  #include <M5Unified.h>
2
3  static constexpr const size_t record_number      = 256;
4  static constexpr const size_t record_length      = 320;
5  static constexpr const size_t record_size        = record_number * record_length;
6  static constexpr const size_t record_samplerate  = 17000;
7  static int16_t prev_y[record_length];
8  static int16_t prev_h[record_length];
9  static size_t rec_record_idx = 2;
10 static size_t draw_record_idx = 0;
11 static int16_t *rec_data;
12
13 void setup(void) {
14     auto cfg = M5.config();
15
16     M5.begin(cfg);
17
18     M5.Display.startWrite();
19     M5.Display.setRotation(1);
20     M5.Display.setTextDatum(top_center);
21     M5.Display.setTextColor(WHITE);
22     M5.Display.setFont(&fonts::FreeSansBoldOblique12pt7b);
23
24     rec_data = (typeof(rec_data))heap_caps_malloc(record_size *
25                                                    sizeof(int16_t),
26                                                    MALLOC_CAP_8BIT);
27     memset(rec_data, 0, record_size * sizeof(int16_t));
28     M5.Speaker.setVolume(255);
29
30     /// Since the microphone and speaker cannot be used at the same time,
31     /// turn
32     /// off the speaker here.
33     M5.Speaker.end();
34     M5.Mic.begin();
35     M5.Display.fillCircle(130, 15, 8, RED);
36     M5.Display.drawString("REC", 180, 3);
37 }
38
39
40 void loop(void) {
41     M5.update();
42
43     if (M5.Mic.isEnabled()) {
44         static constexpr int shift = 6;
45         auto data                  = &rec_data[rec_record_idx * record_length];
46         if (M5.Mic.record(data, record_length, record_samplerate)) {
47             data = &rec_data[draw_record_idx * record_length];
48
49             int32_t w = M5.Display.width();
50             if (w > record_length - 1) {
```

```

51     w = record_length - 1;
52 }
53 for (int32_t x = 0; x < w; ++x) {
54     M5.Display.writeFastVLine(x, prev_y[x], prev_h[x],
55                               TFT_BLACK);
56     int32_t y1 = (data[x] >> shift);
57     int32_t y2 = (data[x + 1] >> shift);
58     if (y1 > y2) {
59         int32_t tmp = y1;
60         y1          = y2;
61         y2          = tmp;
62     }
63     int32_t y = ((M5.Display.height()) >> 1) + y1;
64     int32_t h = ((M5.Display.height()) >> 1) + y2 + 1 - y;
65     prev_y[x] = y;
66     prev_h[x] = h;
67     M5.Display.writeFastVLine(x, prev_y[x], prev_h[x], WHITE);
68 }
69
70 M5.Display.display();
71 M5.Display.fillCircle(130, 15, 8, RED);
72 M5.Display.drawString("REC", 180, 3);
73 if (++draw_record_idx >= record_number) {
74     draw_record_idx = 0;
75 }
76 if (++rec_record_idx >= record_number) {
77     rec_record_idx = 0;
78 }
79 }
80 }
81
82 int state = M5.BtnPWR.wasClicked();
83
84 if (state) {
85     auto cfg = M5.Mic.config();
86     cfg.noise_filter_level = (cfg.noise_filter_level + 8) & 255;
87     M5.Mic.config(cfg);
88     M5.Display.clear();
89     M5.Display.fillCircle(130, 15, 8, GREEN);
90     M5.Display.drawString("NF:" + String(cfg.noise_filter_level), 180, 3);
91     delay(2000);
92     M5.Display.clear();
93 }
94
95 if (M5.Touch.getCount() && M5.Touch.getDetail(0).wasClicked()) {
96     if (M5.Speaker.isEnabled()) {
97         M5.Display.clear();
98         while (M5.Mic.isRecording()) {
99             delay(1);
100         }
101         /// Since the microphone and speaker cannot be used at the same
102         /// time, turn off the microphone here.
103         M5.Mic.end();

```

```

104     M5.Speaker.begin();
105
106     M5.Display.fillTriangle(130 - 8, 15 - 8, 130 - 8, 15 + 8,
107                             130 + 8, 15, 0x1c9f);
108     M5.Display.drawString("PLAY", 180, 3);
109     int start_pos = rec_record_idx * record_length;
110     if (start_pos < record_size) {
111         M5.Speaker.playRaw(&rec_data[start_pos],
112                             record_size - start_pos,
113                             record_samplerate, false, 1, 0);
114     }
115     if (start_pos > 0) {
116         M5.Speaker.playRaw(rec_data, start_pos, record_samplerate,
117                             false, 1, 0);
118     }
119     do {
120         delay(1);
121         M5.update();
122     } while (M5.Speaker.isPlaying());
123
124     /// Since the microphone and speaker cannot be used at the same
125     /// time, turn off the speaker here.
126     M5.Speaker.end();
127     M5.Mic.begin();
128
129     M5.Display.clear();
130     M5.Display.fillCircle(130, 15, 8, RED);
131     M5.Display.drawString("REC", 180, 3);
132 }
133 }
134 }

```

Once the firmware is successfully flashed, you can record and play back audio on the StackChan. Tap the screen to start recording, and press the power button (BtnPWR) to adjust the noise filter level. Tapping the screen again will play back the audio that was just recorded.



| API

The StackChan Mic functionality is implemented using the **Mic_Class** from the M5Unified library. For more related APIs, refer to the documentation below:

- [M5Unified - Mic Class](#)