

CoreInk Button

CoreInk Button input related APIs and case programs.

Note:

Include `M5.update()` function in the main loop to read status update and minimize blocking, otherwise you may not be able to retrieve the status of key changes in time.

Example

cpp

```
1  #include "M5Unified.h"
2
3  void setup()
4  {
5      auto cfg = M5.config();
6      M5.begin(cfg);
7      Serial.begin(115200);
8  }
9
10 void loop()
11 {
12     M5.update();
13     if (M5.BtnA.wasPressed()) {
14         Serial.println("BtnA Pressed");
15     }
16     if (M5.BtnB.wasPressed()) {
17         Serial.println("BtnB Pressed");
18     }
19     if (M5.BtnC.wasPressed()) {
20         Serial.println("BtnC Pressed");
21     }
22     if (M5.BtnPWR.wasPressed()) {
23         Serial.println("BtnPWR Pressed");
24     }
25     if (M5.BtnEXT.wasPressed()) {
26         Serial.println("BtnEXT Pressed");
27     }
28 }
```

API

The button part of CoreInk uses the `Button_Class` in the M5Unified library. For more APIs related to buttons, you can refer to the document below:

- [M5Unified - Button Class](#)