Atom VoiceS3R Button

APIs and example programs related to button input on Atom VoiceS3R.

> **Notes**
>
> When using this feature, you need to include the `M5.update()` function in the main loop to read state updates. Also, try to minimize blocking operations, otherwise button changes may not be detected in time.

Example Program

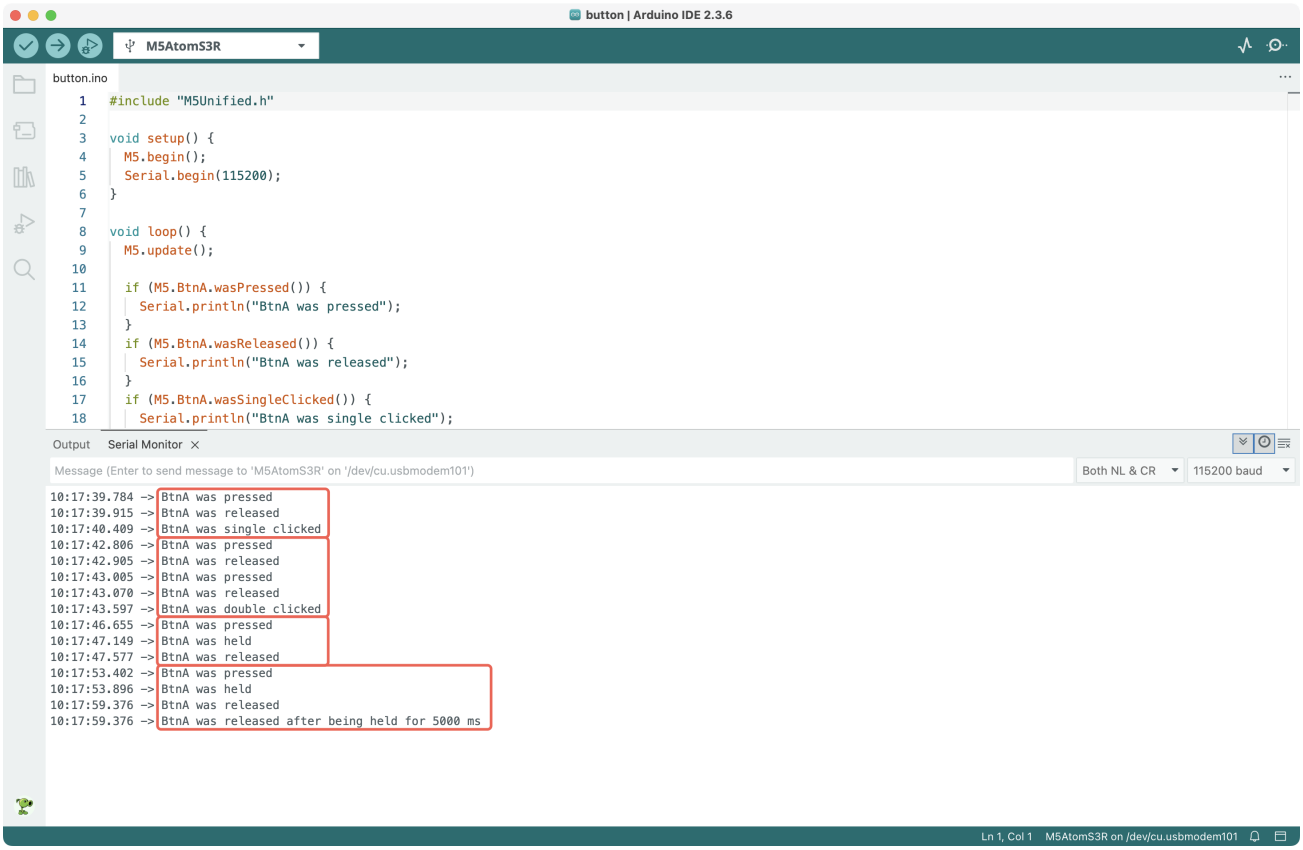
Compilation Requirements

- M5Stack board manager version >= 3.2.2
- Board option = M5AtomS3R
- M5Unified library version >= 0.2.8

cpp

```cpp
#include "M5Unified.h"

void setup() {
  M5.begin();
  Serial.begin(115200);
}

void loop() {
  M5.update();

  if (M5.BtnA.wasPressed()) {
    Serial.println("BtnA was pressed");
  }
  if (M5.BtnA.wasReleased()) {
    Serial.println("BtnA was released");
  }
  if (M5.BtnA.wasSingleClicked()) {
    Serial.println("BtnA was single clicked");
  }
  if (M5.BtnA.wasDoubleClicked()) {
    Serial.println("BtnA was double clicked");
  }
  if (M5.BtnA.wasHold()) {
    Serial.println("BtnA was held");
  }
  if (M5.BtnA.wasReleaseFor(5000)) {  // ms
    Serial.println("BtnA was released after being held for 5000 ms");
  }

  delay(5);
}
```

This program will detect the status of the front button on the device (pressed, released, single short press, double short press, long press, released after a long press of 5000 ms), and print messages in the Serial Monitor:



API

The button part of Atom VoiceS3R uses the `Button_Class` from the `M5Unified` library. For more related APIs, please refer to the following documentation:

- M5Unified - Button Class