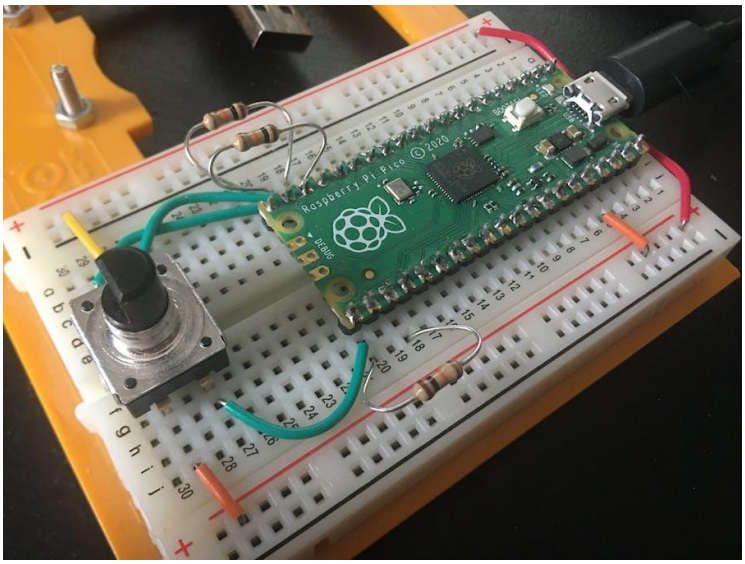
Pi Pico mbed RP2040 volume and macro keyboard

Note: This uses the Arduino mbed-based RP2040 board. The USB Keyboard supports both normal and consumer keys using the single #include <USBKeyboard.h> include.

https://www.hackster.io/Murchx/pi-pico-macro-keyboard-d0bd1c

Raspberry Pi Pico macro keyboard to simplify the workflow of some creative software or just for everyday use.



Things used in this project

Hardware components

	Raspberry Pi Pico	× 1
	Rotary Encoder with Push-Button My prototype has only 1	× 2
	Tactile Switch, Top Actuated Switches / buttons of your choice (not added to my prototype)	× 4
	Resistor 10k ohm 3 pieces are required for each rotary encoder with push-button, 2 if just rotary encoder, 1 piece are required for each switches / button	× 10

Story

Of course this is nothing unique, but I want to share my learning curve.

The aim of the project is to create a custom macro / shortcut / media keyboard. Comparing the microcontrollers available on the market, I chose the Raspberry Pi Pico because it is affordable and HID support is possible. I initially tried to write code in MicroPython in the Thonny IDE, but could not find a MicroPython compatible HID library. Then I tried the CircuitPython language in Thonny IDE - this option is acceptable with some drawbacks:

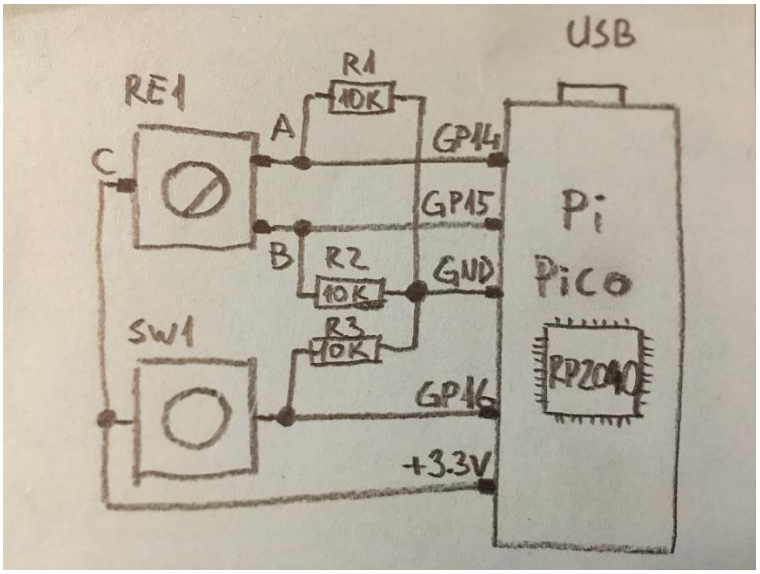
- when operating in HID mode, the Raspberry Pi Pico is also visible as a USB drive (not a big problem, but not a nice solution either);

- with my basic programming skills, I did not find an opportunity to use the RP2040 multicore option (to be honest, it is not necessary for a macro keyboard, but if there was an opportunity, it could be used in the future).

Since the Arduino also supports the RP2040 microcontroller, and I have some experience working with the Arduino Uno - I decided to try the Arduino IDE. First I tried Arduino IDE 1.8, but once again I encountered the lack of a HID library for the Pi Pico. However, the Arduino IDE 2.0 beta has Pi Pico HID support, but the disadvantage is that the Pi Pico built-in pulldown resistors does not work. With exactly the same Pi Pico board using the Python language the built-in pulldown resistors work as expected. In my prototype this problem was solved by adding resistors outside the Pi Pico board, but I hope that in the future in the Arduino IDE 2.0 this problem will be solved. By the way, INPUT_PULLDOWN does not issue a compilation error such as if it is replaced with INPUT_WHATSOEVER.

Schematics

Prototype scheme



Code

Raspberry Pi Pico macro keyboard

Arduino
Test code for Rotary encoder with push-button function test. Rotate knob for volume, press to play / pause.

```
#include <USBKeyboard.h>

USBKeyboard keyb;
int encA = 14;
int encB = 15;
int btn = 16;

void setup() {
  pinMode(encA, INPUT);
  pinMode(encB, INPUT);
  pinMode(btn, INPUT);
  // pinMode(LED_BUILTIN, OUTPUT);
}

int main(void) {
```

```
  while(1) {

    if(digitalRead(encA) == HIGH || digitalRead(encB) == HIGH) {
      digitalWrite(LED_BUILTIN, HIGH);
      delay(2);
      if(digitalRead(encA) == HIGH) {
        keyb.media_control(KEY_VOLUME_UP);
      }
      else if(digitalRead(encB) == HIGH) {
        keyb.media_control(KEY_VOLUME_DOWN);
      }
      else {}
      while(digitalRead(encA) == HIGH || digitalRead(encB) == HIGH) {
        delay(20);
      }
      digitalWrite(LED_BUILTIN, LOW);
    }

    else if(digitalRead(btn) == HIGH) {
      digitalWrite(LED_BUILTIN, HIGH);
      delay(5);
      keyb.media_control(KEY_PLAY_PAUSE);
      while(digitalRead(btn) == HIGH) {
        delay(50);
      }
      digitalWrite(LED_BUILTIN, LOW);
    }

    // keyb.printf("Hello World\r\n");
    // keyb.media_control(KEY_VOLUME_DOWN);
    // keyb.key_code('d', KEY_LOGO);
    // keyb.key_code('q', 101); // ALT 100; SHIFT 10; CTRL 1; CTRL+SHIFT 11
    // keyb.key_code('d');
    // keyb.key_code(0, KEY_LOGO);
    // keyb._putc(KEY_F1);
    // keyb._putc('d');
  }
}
```

Comments:

Hello, I am currently working on something simillar but can´t get the USBKeyboard library to work (it seems like it is not compatible with arm architecture). Did you use a specific version? Could you please send a link to that library? Thanks in advance

I wrote the code for my project in IDE 2.0 (beta 7). If I remember correctly the USB HID library is already included there, You need to add Arduino Mbed OS RP2040 board support (I am using version 2.1.0).