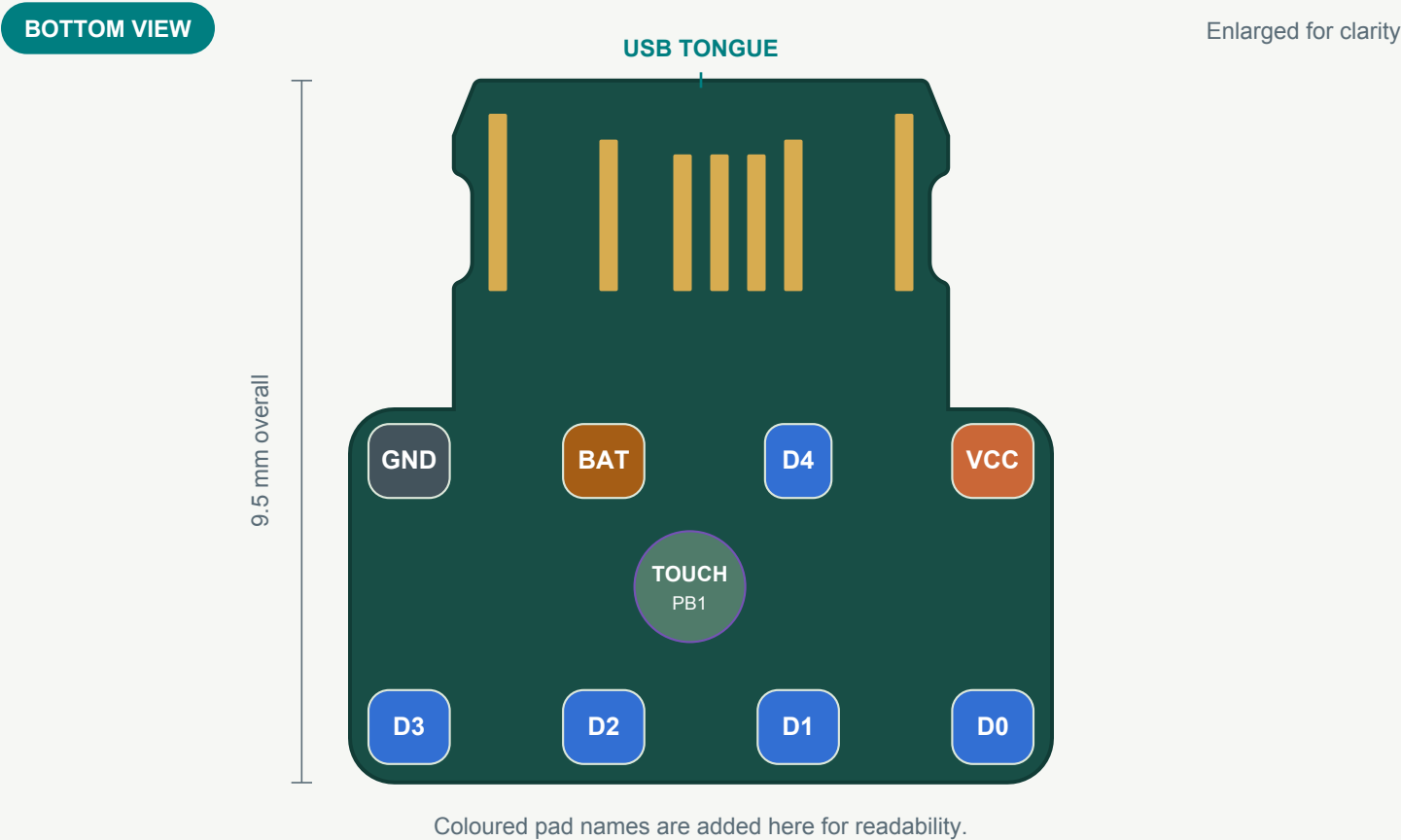


# Eight pads. One tiny board.

Direct underside view: turn the board over, keep the USB tongue pointing up.



- GPIO
- POWER
- GROUND
- TOUCH

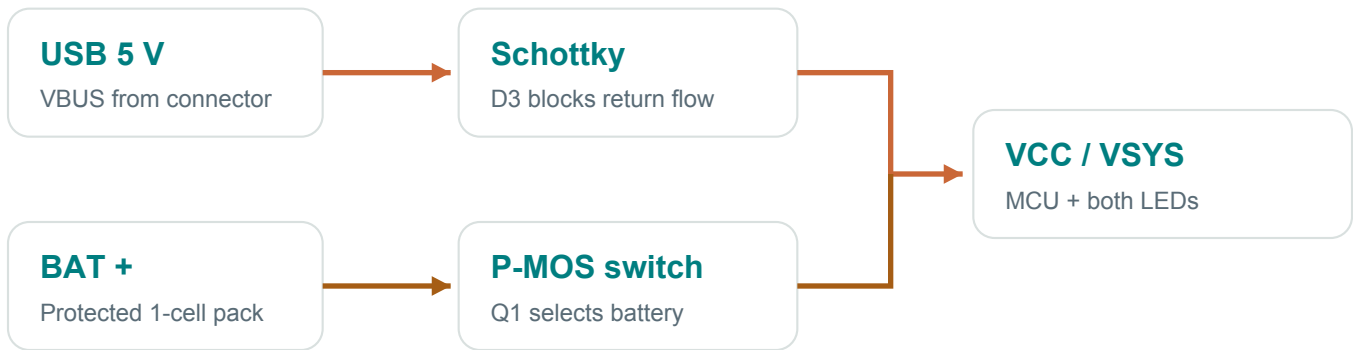
| PAD | MCU / NET     | USE                                                 |
|-----|---------------|-----------------------------------------------------|
| D0  | PA3 / pin 5   | Digital I/O, TIM2 CH4; ADC3 is lot-dependent        |
| D1  | PA6 / pin 8   | Digital I/O, ADC6, TIM3 CH1                         |
| D2  | PB0 / pin 10  | Digital I/O, ADC8, UART4 TX                         |
| D3  | PB3 / pin 12  | Digital I/O, UART3 TX                               |
| D4  | PC19 / pin 16 | Digital I/O; disable debug-clock function first     |
| VCC | VSYS          | Supply rail output; follows USB or battery          |
| BAT | BAT+          | Positive battery input; use a protected 1-cell pack |
| GND | GND           | Common ground / battery negative                    |

D0-D4 are board aliases, not Arduino core pin numbers. Alternate functions need configuration.

# Know your power.

VCC is the live system rail. There is no 3.3 V regulator on this board.

## HOW THE BOARD SELECTS POWER



USB connected: VCC is approximately USB voltage minus the diode drop. Battery only: VCC follows the cell voltage. GPIO HIGH levels follow VCC too.

## CONNECT A BATTERY



Underside / USB up

## Three rules for this power circuit

- 1 BAT is not a charging port. Charge the pack separately.
- 2 Match + to BAT and - to GND. Reverse polarity is not protected.
- 3 Use a protected pack; the board has no low-voltage cutoff. RGB operation below 3.3 V is outside the listed LED supply range.

# Program it. Make it respond.

Use the factory USB ISP bootloader and the CH32X035F8U6 firmware target.

## 01 Power off

Disconnect USB and the battery. Wait for VCC to fall.

## 02 Hold BOOT

Keep the button pressed while connecting a USB data cable.

## 03 Release + flash

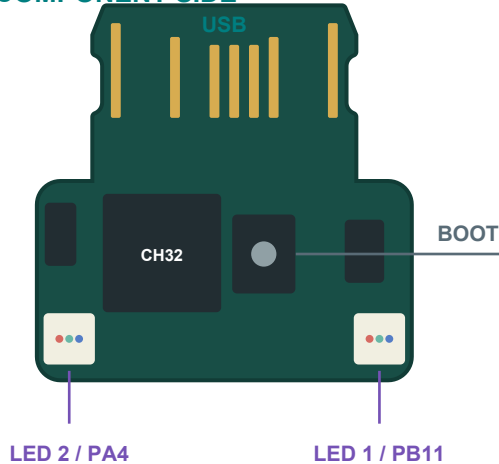
Release BOOT. Use WCHISPTool USB mode and the correct HEX/BIN.

## 04 Restart

Power-cycle with BOOT released to run your application.

BOOT temporarily pulls PC17 / USB D+ high through 4.7k. It is not a reset button. The USB connector still needs mechanical/electrical validation before this sequence can be tested on hardware.

### TOP / COMPONENT SIDE



The two LED data lines are independent.

### ONBOARD INTERACTIONS

#### RGB LEDs

D1 LED uses PB11; D2 LED uses PA4. Each is a separate pixel, not a two-pixel chain.

Use a driver for the exact WS2816A format. Ordinary GPIO blinking will not set its colour.

#### Capacitive touch

The masked centre electrode is PB1 / ADC9. Adapt the WCH TouchKey example to channel 9; calibrate with no finger present.

### Firmware naming matters

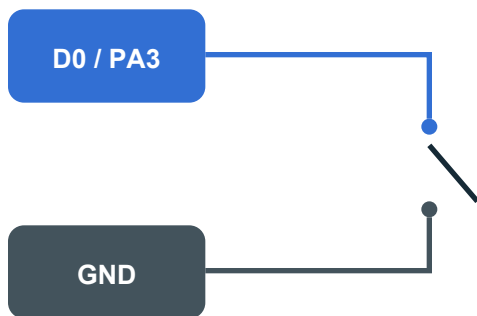
Pad D1 means PA6. Component D1 means the RGB LED on PB11. Use MCU port names in your code. No tested Arduino board package or application firmware is supplied with this prototype.

# Your first connections.

Simple wiring examples using the actual underside pads. Always share ground.

## A pushbutton input

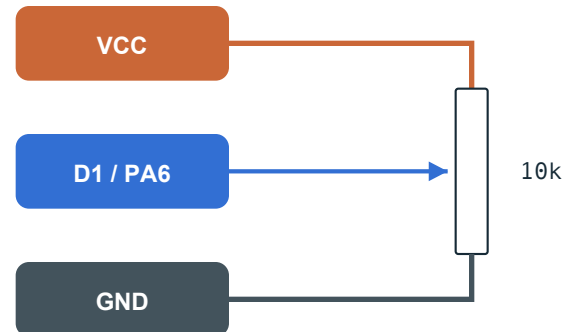
D0 / PA3 + GND



Enable PA3 input pull-up. Released reads HIGH; pressed reads LOW. Debounce in software.

## An analog control

D1 / PA6 + VCC + GND



Connect a 10k potentiometer across VCC and GND. Wiper to D1. Configure PA6 analog input / ADC6.

## Match sensor voltage to VCC

USB power can put VCC and GPIO HIGH near 5 V. Do not connect a 3.3 V-only sensor directly to VCC or a driven GPIO. Use a suitable regulator and level interface. Do not drive VCC externally while USB or BAT is connected.

## D4 / PC19 needs one extra step

PC19 starts as the debug clock. Enable the AFIO clock and disable the SDI remap before configuring it as GPIO. This board has no dedicated debug pads.

```
GPIO_PinRemapConfig(GPIO_Remap_SWJ_Disable, ENABLE);
```

## REFERENCE & VALIDATION

Pin positions and net names: current routed KiCad board. MCU functions: WCH CH32X035 datasheet v2.3, QFN20 pin map. LED supply: supplier listing. Examples describe intended use and are not bench-tested.

[WCH datasheets + examples](#)

[WCH USB programming tool](#)

[LED supplier specification](#)

Release hold: connector compliance / fit, LED land-pattern confirmation, and hardware bring-up.