Open Sense Module Datasheet

DI-C3B0 v1.0

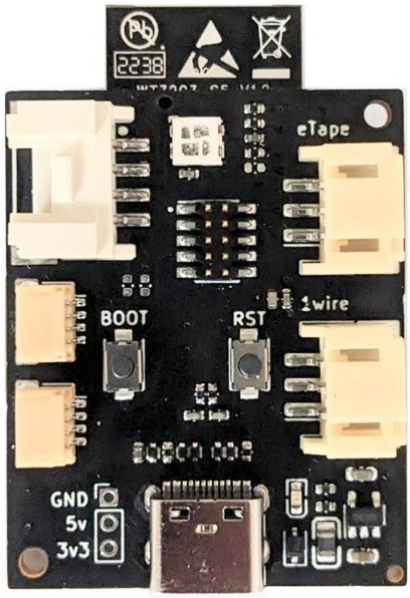


Table of Contents

1 Description

The **DI-C3B0** is a compact breakout board meticulously crafted to provide the perfect foundation for creating sensor-based modules using open automation platforms like Tasmota, ESPHome or ESPEasy. Its compact design ensures stability and reliability, setting it apart from other development kits on the market.

By offering a selection of standard connectors, the **DI-C3B0** enables seamless integration with a wide range of sensors, OLED displays, real-time clocks, EEPROMs, IO expanders, motor controllers, and more. Unlike conventional development kits that often rely on breadboards or unreliable dupont jumper connectors prone to loose connections, our breakout board guarantees solid connections, eliminating any concerns of compromised wiring.

The **DI-C3B0** is optimised for compatibility with ESP32-C3 modules, such as the `WT32C3-S5` and **`ESP8685-WROOM-01`** modules. However, with minor modifications, it can also accommodate most other modules available in SMD-22 form factors, including the widely popular `ESP-12F`.

The ESP32-C3 was selected, as it provides some key advantages over the older ESP8266 modules

- 32-bit 160MHz RISC-V core with improved performance, increased memory and better power efficiency
- Bluetooth Low Energy(BLE) support
- USB programming and JTAG debugging
- Improved ADC’s with up to 6 channels
- I2S support for audio devices

2 Features

- Compatible with most SMD-22 and SMD-16 ESP modules
- 2x QWIIC/StemmaQT I2C connectors
- Grove UART connector (optionally I2C)
- 1-wire connector (for sensors such as DS18B20 probes)
- 2x ADC inputs setup for resistive sensors (such as eTape)
- WS2812B RGB LED
- 1.27mm pin header with 4x GPIO (JTAG)
- 1x User Programmable Button
- 1x Reset Button
- USB Programming and Debug (ESP32-C3 only)

2.1 Specifications

- Dimensions: 32mm x 40mm x 7mm (excluding esp32 module)
- SDK: Arduino IDE or ESP-IDF
- Platforms: Tasmota, ESPHome, ESPEasy, WLED

2.2 Compatible Modules

The following ESP modules are compatible with this board. Not all features supported on ESP8266 modules. Some modules require the addition of extra resistors. Modules can easily be placed by hand soldering.

ESP32-C3

- WT32C3-S5
- ESP8585-WROOM-01
- ESP-C3-12F (Not recommended)

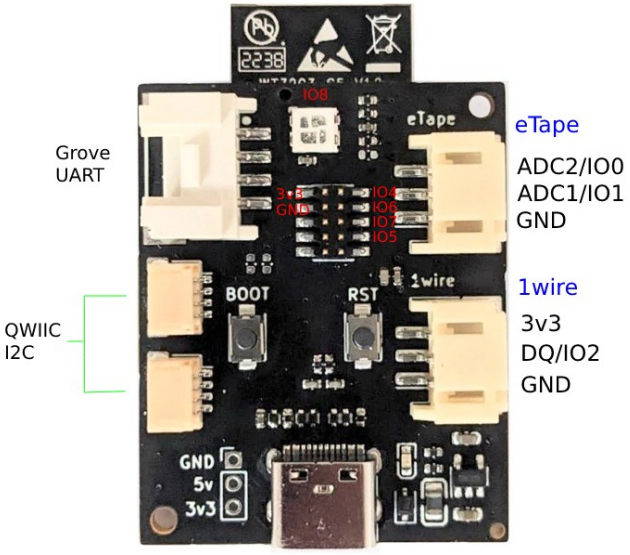
ESP8266

- ESP-12E, ESP-12F, WT8266-S5
- ESP-12S, WT8266-S6

OTHER

- ESP-12H [ESP32-S2]
- ESP8684-WROOM-01 [ESP32-C2] (requires esp-idf v5.1, not supported by other platforms yet)

2.3 Pin Mappings



Function	Connector	IO (ESP32-C3)	IO (ESP8266)
U0RX	Grove	GPIO20, RX	GPIO3
U0TX	Grove	GPIO21, TX	GPIO1
SDA	Qwiic	GPIO3	GPIO5
SCL	Qwiic	GPIO10	GPIO4
LED	RGB	GPIO8	GPIO2
Boot Button	Button	GPIO9	GPIO0
TMS	JTAG^	GPIO4	GPIO14
TDI	JTAG	GPIO5	GPIO12
TCK	JTAG	GPIO6	GPIO13
TDO	JTAG	GPIO7	GPIO15
DQ	1-wire	GPIO2	GPIO16
ADC2	eTape	GPIO0	N/A
ADC1	eTape	GPIO1	ADC
USB_D-	USB-C	GPIO18	N/A
USB_D+	USB-C	GPIO19	N/A

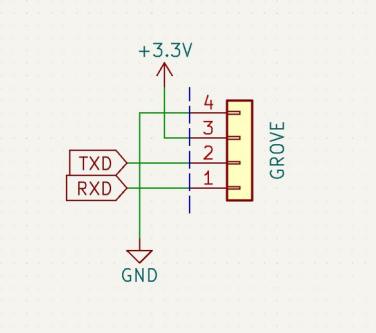
^ JTAG on ESP8266 is not officially supported on OpenOCD and requires a forked version of OpenOCD.

3 Connectors

3.1 Grove UART

Connector: GROVE

This connector is wired directly to UART0 on the ESP chip using standard Grove UART pinout and a 3.3v supply.



Some popular devices that connect via UART include Nextion HMI displays, Gas Sensors, Ultrasonic distance sensors, RFID Tag reader, Fingerprint scanner, LTE Modem and GPS units.

This port can be configured in software to instead be I2C, however in this case I2C must be disabled on QWIIC ports as the ESP32-c3 and ESP8266 modules only support a single I2C bus.

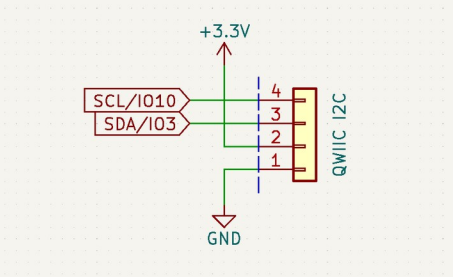
It can also be used to program the ESP chip using USB Serial adapter, you must use the boot button to put the device into bootloader mode in this case.

3.2 Qwiic I2C

Connector: JST SR(1mm pitch)

Quick connection of any QWIIC/StemmaQT compatible device or use adapter cables if you need to attach other connector types. You can further expand the number of available ports with a multi-port adapter board. Many devices can be supported on the I2C bus, however all connected devices must have unique addresses.

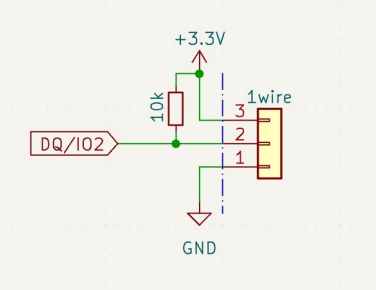
Suitable for the huge variety of senors,displays and other I2C devices on the market. Common sensors include temperature, humidity, pressure, light, gas and moisture sensors. Other devices that also use I2C include OLED displays, RGB leds, I/O Expanders, EEPROMs and many more.



3.3 1-Wire

Connector: 3-Pin JST PH (2mm)

1wire is protocol developed by Dallas Semi, that uses a single data wire to connect sensors. This is a JST PH Connector, with integrated 10k pull-up resistor.



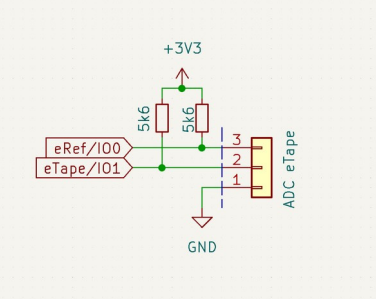
The most common 1-wire sensor is the Dallas DS18B20 temperature sensors or probes. Unlike I2C you can have many of the same sensor on the same bus.

3.4 ADC/eTape

Connector: 3-Pin JST PH (2mm)

This is a JST PH Connector, that has two ADC inputs with integrated 5.6k pull-up resistors.

This connector is setup to directly connect a Milone Tech eTape water level sensor, however can also be used other resistive based sensors including thermistors and photoresistors or as active low Digital IO pins.

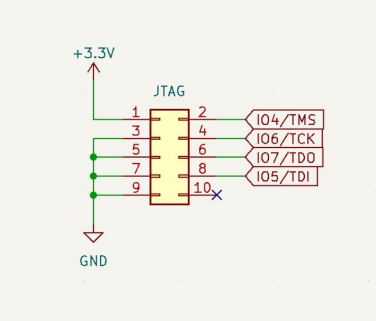


* IO0 must be left disconnected on all ESP8266 based modules

3.5 GPIO/JTAG Header

1.27mm pitch 10pin IDC header

This header is directly connected to the JTAG pins on the ESP module. Configured to connect directly to the `Espressif ESP-Prog` debugger for online debugging or you can use these 4 GPIO's as general purpose IO pins.



The WT32C3-S5 and ESP8685 modules use USB debugging by default, you must burn an eFuse if you prefer to use this connector for JTAG debugging.

3.6 Power Header

2.0mm pitch 3-pin header (not soldered)

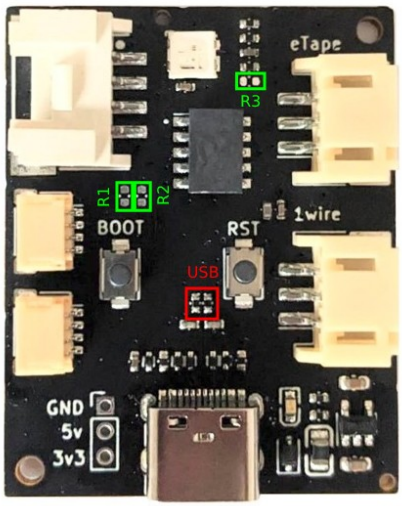
When the device is connected to USB this header provides access to the unregulated 5V (USB Vbus) and regulated 3.3V rails. Alternatively you can opt to power the board using 5V on this header instead of the USB port.

4 Modification for Other Modules

This module has been designed to be flexible and can be used with most modules utilising SMD-22 or SMD-16 form factor. Some features are not available when using ESP8266 modules compared to ESP32-C3. These modules have no support for USB programming/debugging, only a single ADC Channel, and no Bluetooth support.

GPIO Mapping varies depending on module, please check that datasheet for your chosen module. None of these alternate modules support USB programming and the ESP chip will need to be programmed using the Grove connector with USB UART adaptor + Boot button.

A number of resistors have been left unpopulated. To be able to program the ESP module in circuit via the Grove connector it is required to add these resistors, actual configuation is module dependant. If you pre-program the module before soldering you can skip these. The two USB resistors (marked in red) must be removed if you are using an ESP-12E or ESP-12F series module.



Module	R1	R2	R3	Remove USB	Other modules
ESP-12F	Y	Y	Y	Y	ESP-12E, WT8255-S5
ESP-12S	Y	N	Y	N	ESP-07(S), WT8266-S6
ESP8685	Y	N	N	N	

NB: Many of the ESP-12x chips on the market today are clone modules. Therefore we can’t guarantee the specific configuration of pull-up resistors for these modules.

Other available modules that have not been fully validated in our labs, but will work with addition of suitable resistors.

ESP8685-WROOM-01 - Forthcoming module that is 100% compatible with WT32C3-S5 and supporting full features.

ESP-12H - This module uses the ESP32-S2, this module should work without modification. Notably it offers 2x I2C busses. USB OTG will not be connected.

ESP8684-WROOM-01 - Brand new module based on ESP32-C2, requires esp-idf 5.1 not yet supported in Tasmota or ESPHome. No USB support but does include bluetooth, can be used without modication.

ESP-C3-12F - Alternate module based on ESP32-C3. Requires R1 to be added. No USB connection. Not recommend and poor availability. Completely different IO mapping.

