Technical Datasheet: USB Type C Power Delivery Dummy Breakout - HUSB238

Adafruit Product ID: 5807

Product Overview

The Adafruit HUSB238 USB Type C Power Delivery Dummy Breakout (Product ID: 5807) is a compact breakout board built around the HUSB238 USB PD sink IC. It allows users to negotiate and select a fixed DC output voltage from any USB-C Power Delivery compliant power adapter — either via hardware jumper configuration (no microcontroller required) or dynamically over I2C. Selectable output voltages range from 5V to 20V, with maximum output up to 100W (20V at 5A). The board is suitable for powering projects that require higher voltages than standard USB 5V, converting DC-powered devices to USB-C, or retrofitting battery-powered devices with USB-C input.

Technical Specifications

Specification	Value
Product ID	5807
Onboard IC	HUSB238 USB PD Sink Controller
I2C Address	0x08 (default, fixed)
Input	USB Type-C receptacle (PD 2.0 / 3.0 compliant adapter required)
Output Voltages (Selectable)	5V, 9V, 12V, 15V, 18V, 20V
Maximum Output Power	100W (20V at 5A)
Configuration Modes	Jumper (hardware) or I2C (dynamic)

Onboard Voltage Regulator	Not included
Onboard I2C Pull-up Resistors	Not included
Included Accessories	1x 7-pin header strip, 1x 2-pin terminal block
Dimensions	24.5mm (L) x 20.2mm (W) x 4.9mm (H)
Imperial Dimensions	1.0" x 0.8" x 0.2"
Weight	2.2g

Pinout Reference

The following header pins are broken out along the bottom edge of the PCB (left to right as labeled on board silkscreen):

Pin Label	Type	Description
SDA	I2C Data	I2C data line for HUSB238 communication. No onboard pull-up resistor — external pull-up to logic voltage required.
SCL	I2C Clock	I2C clock line for HUSB238 communication. No onboard pull-up resistor — external pull-up to logic voltage required.
V+	Power Output	DC voltage output (Vout). Outputs the PD-negotiated voltage. Also available on the terminal block (+) pin.
GND	Ground	Common ground for power and logic. Also available on the terminal block (-) pin.
D-	USB Data	USB 2.0 Data Minus, broken out from the USB-C connector.

D+	USB Data	USB 2.0 Data Plus, broken out from the USB-C connector.

Note: Since Vout can be as high as 20V, there is no onboard 3.3V or 5V regulator. When using I2C, the microcontroller must have its own power supply and provide I2C pull-up resistors to the desired logic voltage level.

Jumper Configuration Guide

All voltage and current settings are controlled by solder jumpers on the PCB. Only one voltage jumper and one current jumper should be closed at a time.

Voltage Jumpers

Jumper Label	Default State	Action to Select
5V	Closed	Already selected by default. Cut to change voltage.
9V	Open	Solder closed to select 9V output.
12V	Open	Solder closed to select 12V output.
15V	Open	Solder closed to select 15V output.
18V	Open	Solder closed to select 18V output.
20V	Open (labeled "open")	Leave ALL voltage jumpers open to select 20V (or highest available adapter voltage).

Current Jumpers

Jumper Label	Default State	Action to Select
1A	Closed	1A requested by default. Cut to change current.

2A	Open	Solder closed to select 2A.
3A (max)	Open (labeled "open")	Leave both 1A and 2A open to request 3A.

Important: If the adapter cannot supply the requested current, incorrect voltage output may result. If voltage is wrong, try reducing the current setting to 2A or 1A.

Wiring Guide

Mode 1 — Jumper-Only (No Microcontroller Required):

1. Connect a USB-C PD-compliant power adapter to the USB-C port using a USB-C cable.
2. Cut the 5V voltage jumper (default). Solder closed the desired voltage jumper (e.g. 12V).
3. Optionally adjust the current jumper (default 1A; solder 2A closed or leave both open for 3A).
4. Connect your load to V+ and GND header pins, or to the terminal block (+ and -).
5. The HUSB238 will negotiate the requested voltage over the CC lines at power-up.

Mode 2 — I2C Dynamic Control:

1. Connect V+ and GND to your load circuit.
2. Connect SDA and SCL to your microcontroller's I2C bus.
3. Add external I2C pull-up resistors (typically 4.7k ohm) from SDA and SCL to your microcontroller's logic voltage (e.g. 3.3V).
4. Power the microcontroller from a separate power source.
5. Use the Arduino library (github.com/adafruit/Adafruit_HUSB238) or CircuitPython library to query available voltages and set the desired output dynamically.
6. On startup, the HUSB238 uses the jumper settings until I2C commands are received.

Important Operational Notes

• Separate Microcontroller Power Required for I2C: Because Vout can be up to 20V, there is no onboard 3.3V regulator. The microcontroller must be independently powered. Do not connect Vout directly to a 3.3V or 5V microcontroller power pin.

• External I2C Pull-ups Needed: The board has no onboard pull-up resistors on SDA/SCL. Pull-ups must be added externally to your microcontroller's logic voltage level before I2C will function.

• Jumper Settings Active at Boot: When using I2C mode, the jumper-configured voltage is negotiated first at power-on. I2C commands can override this after the microcontroller initializes.

• Only One Voltage Jumper Closed at a Time: Closing multiple voltage jumpers simultaneously will cause incorrect behavior. Only one voltage jumper should be closed at a time.

• PD Adapter Required: A USB-C Power Delivery compliant adapter is required to negotiate voltages above 5V. Standard 5V-only USB chargers will only output 5V regardless of jumper settings.

• Terminal Block for High-Current Loads: For high-power applications (e.g. 20V/5A), prefer the terminal block over the header pins to ensure adequate current-carrying capacity.