MetaShunt User Guide

Version 1.0, February 2024

MetaShunt is an electronic instrument designed for high-bandwidth, very high dynamic range measurements of the instantaneous current and power and accumulated energy use of low-power and IoT devices from 10s of nanoamps and 100s of milliamps

MetaShunt leverages a simple idea - by quickly and accurately measuring the voltage across a current shunt resistor and engaging additional shunt stages when needed, MetaShunt ensures that it can provide accurate current measurement across a range of approximately 10,000,000:1 without providing significant voltage burden. This range of current is specifically targeted for use with ultra-low power and IoT systems. By measuring current rapidly over time, the total energy use during a given portion of your code can be determined. MetaShunt acts as a virtual ground for your system, so that you can use your battery. Or, MetaShunt can provide 3.3V or 5V to the system under test if desired.

More information is available on the Hackaday.io page here:
https://hackaday.io/project/193628-metashunt-high-dynamic-range-current-measurement

Table of Contents

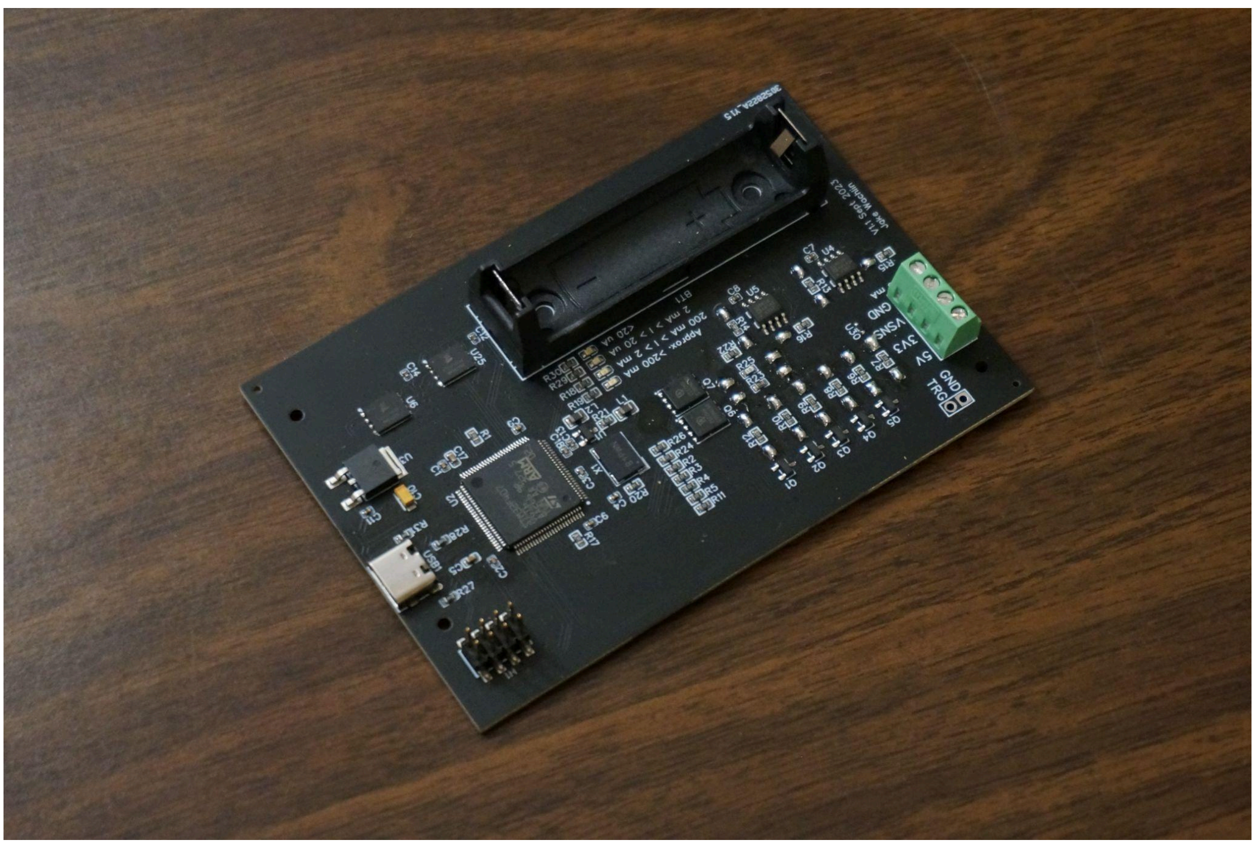
MetaShunt Versions

MetaShunt is currently sold in two versions. For both versions, you will need an Alkaline AA battery and a quality USB-C cable (both not included). Note that poor quality USB cables are prevalent and charging-only cables exist. MetaShunt may not work as expected without a high-quality cable. As a general rule, if the cable came with your phone or other quality electronic device, it will probably work with MetaShunt

Version 1: Pre-Soldered/Calibrated

This version of MetaShunt comes as shown in the image below. All solder jumpers have been connected and calibration has been performed so that MetaShunt is ready to use “out-of-the-box.” While all efforts are made to ensure accurate calibration, no guarantees are made as to the accuracy of calibration. If you need more assurance of calibration, please reach out to discuss options. This version should be selected if:

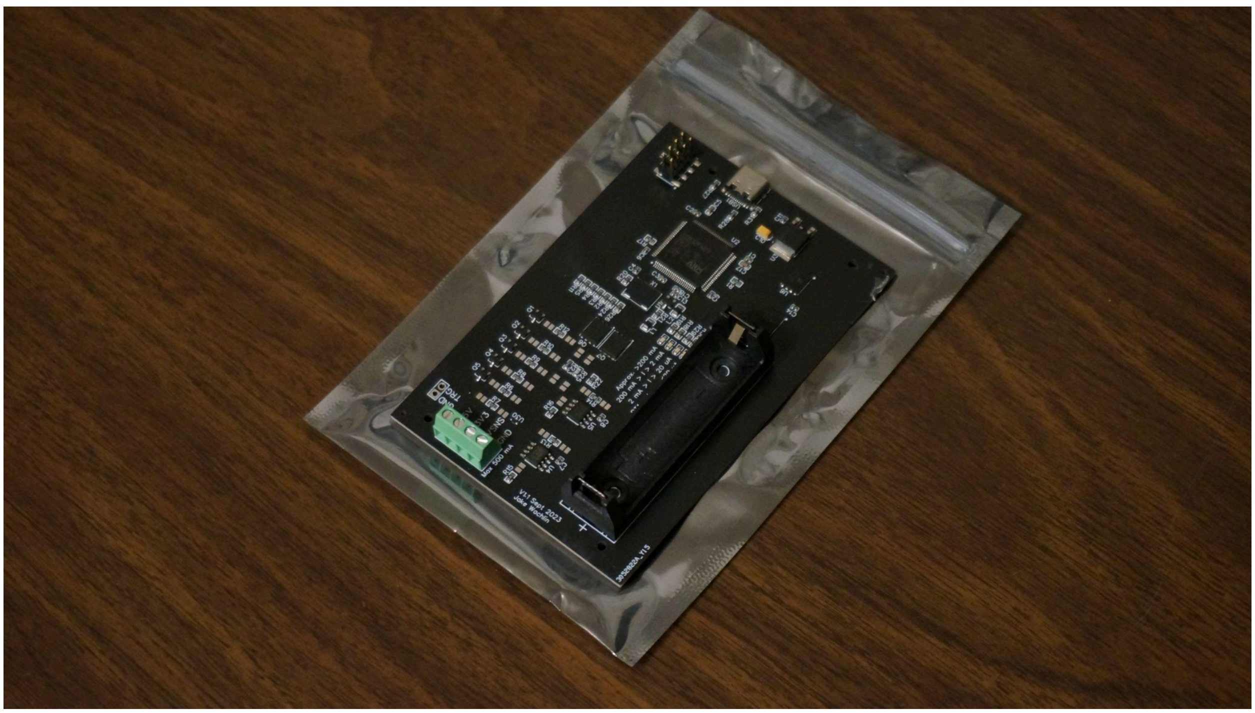
- You do not have a soldering iron, solder, and a multimeter available.
- You want the tool ready to use immediately.



Version 2: Basic

This version of MetaShunt is delivered as shown in the image below. Firmware has been programmed to MetaShunt, and nominal values for the various resistors have been configured onto it, but those values are not based on measurements. If you purchase this version you must have access to a soldering iron, solder, and a multimeter. You must feel confident in following along with the steps laid out in the calibration section of this user guide. This version should be selected if:

- You want to better understand how MetaShunt works.
- You enjoy soldering and can't wait to do more of it.
- You want to save some money on the purchase price.



Specifications

Specification	**Typical Value**	**Notes**
Maximum Burden Voltage	40mV	This is the maximum voltage across the "shunt" in MetaShunt before the gain stages change.
Continuous Data Output Rate	>3.9 kHz	
Maximum Burst Data Output Rate	25.5 kHz	

Burst Datapoints	32,000	This many measurements are recorded into a buffer on MetaShunt, then sent over the USB interface. This allows for faster measurements of key events and can be triggered immediately or on rising or falling current past a configured threshold. See the Use section for more information.
Control Loop Rate	>480 kHz	Rate at which burden voltage is measured and shunt gain stages are changed
Minimum Current Measured	0.4nA	Based on minimum resolution at highest gain
Maximum Current Measured	500mA	Higher currents up to ~1A are not expected to damage MetaShunt but have not been tested. Proceed at your own risk.
Resolution	0.4nA from 0.4nA to 0.4uA 4nA from 0.4uA to 4uA 40nA from 4uA to 40uA 0.4uA from 40uA to 400uA 4uA from 400uA to 4mA 40uA from 4mA to 40mA 320uA from 40mA to 320mA 1mA above 320mA	All values are approximate based on expected values of resistors in the measurement stage of MetaShunt. For reference use. May vary.
Size	4.125” x 2.75”	

Calibration

Resistor Measurement

All resistors in the measurement and amplification stages of MetaShunt are designed with solder jumpers on either side of them. This allows users of the tool to directly measure the resistance of each component, and provide this data to MetaShunt in order to increase the accuracy of its internal calculations of current. Each resistor is clearly marked on the PCB. The resistance of each resistor can be measured with a multimeter and recorded. Note that using a multimeter to measure the resistance of the lower resistance resistors will be inaccurate. Therefore, the next section covers how to compensate for that limitation.

Tools for sending the calibration to MetaShunt are included in the MetaShunt Interface repository on GitHub here:
https://github.com/jwachlin/metashunt_interface

Update the “metashunt_cfg.json” file with the measured values of the resistors (excluding R12, R23, R_LP_FET, and R_HP_FET, since those values are too low to measure accurately with a handheld multimeter). Then, with MetaShunt connected to your computer over USB, run the “metashunt_configure.py” script with your json’s filename as input. For example, if you edited the original file, run “python metashunt_configure.py metashunt_cfg.json”. This will set and check all of the parameters.

High-Power Stage

The high power stages (for measurements of ~40mA or more) must be calibrated in a different way. Resistors are to be connected between the 5V and VSNS pins of MetaShunt. These resistors should be >2W rated. One should be about 22 Ohm, and the other about 11 Ohm, which can be made with two 22 Ohm resistors in parallel. With a multimeter, measure the resistance of each resistor or pair of resistors. Also measure the voltage between 5V and VSNS. First, connect the 22 Ohm resistor, plug in MetaShunt, and read data for about 20 seconds. Record the average current. Next, follow the same process for the 11 Ohm Resistor (or pair of 22 Ohm resistors). Plug all of these values into the top of the “metashunt_hp_calibrate.py” Python script on the MetaShunt Interface repository. Then, run the script, providing the JSON file you used for the resistor measurement calibration in the previous step. This script will calculate updated values for R_HP_FET and R25. Put those into the JSON file in the appropriate places, and run the “metashunt_configure.py” script again. Now your MetaShunt is fully calibrated and ready to use!

Use

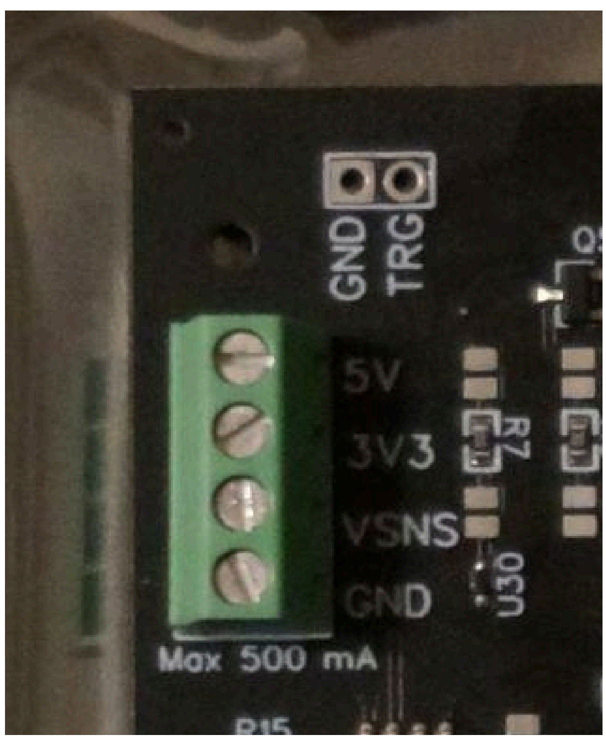
General Information and Recommendations

MetaShunt is designed to be simple to use. First, insert the AA battery into its holder. This battery is required even if MetaShunt is powered by the USB cable. This battery does not power the device; it is used to provide a stable negative voltage rail for an amplification circuit in MetaShunts front end.

Please note that MetaShunt is not isolated and the GND on the green connector is connected to the USB ground (and therefore, your computer). Do not use MetaShunt on high voltage systems. All use of MetaShunt is at your own risk. It is not a dangerous device, but always be careful when working with electronics.

If you are powering your device from the 5V or 3.3V connectors on MetaShunt, connect your Device Under Test (DUT) to MetaShunt before connecting MetaShunt to your computer. The 5V rail comes directly from USB, which can typically range from 4.5-5.5V, so be aware that it may not be very close to 5V. The 3.3V (3V3) rail is regulated to 3.3V with a linear regulator.

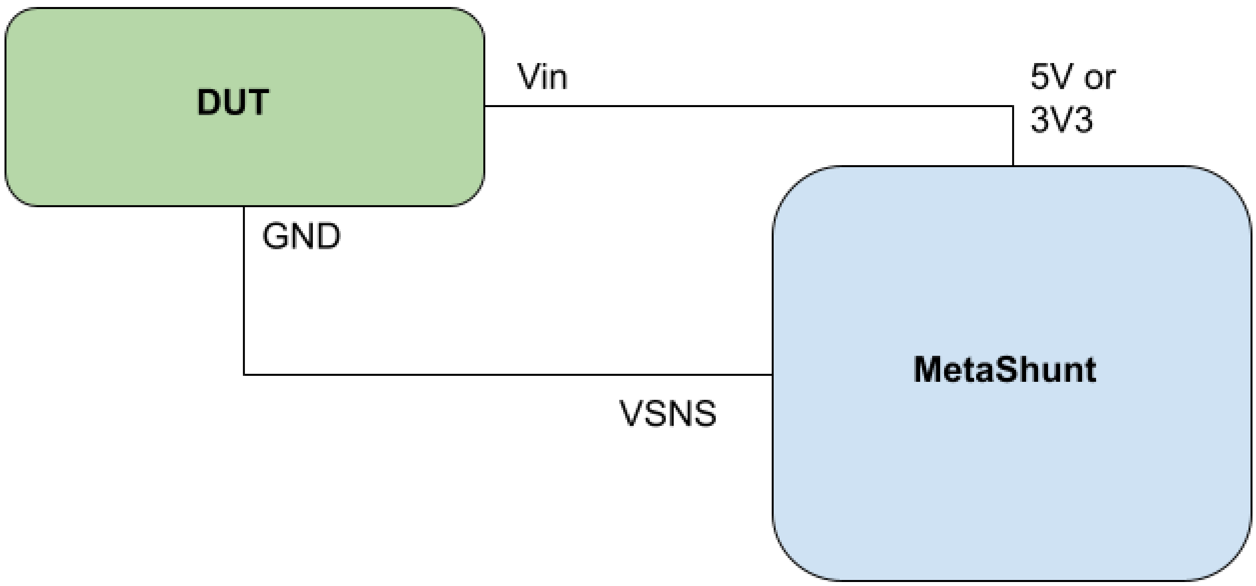
The VSNS pin on the green connector is the “virtual ground” that MetaShunt provides. MetaShunt maintains a small voltage between VSNS and GND, and uses this to measure the current of the DUT. Only positive current can be measured - negative current could damage MetaShunt. MetaShunt is not intended for use with AC circuits.



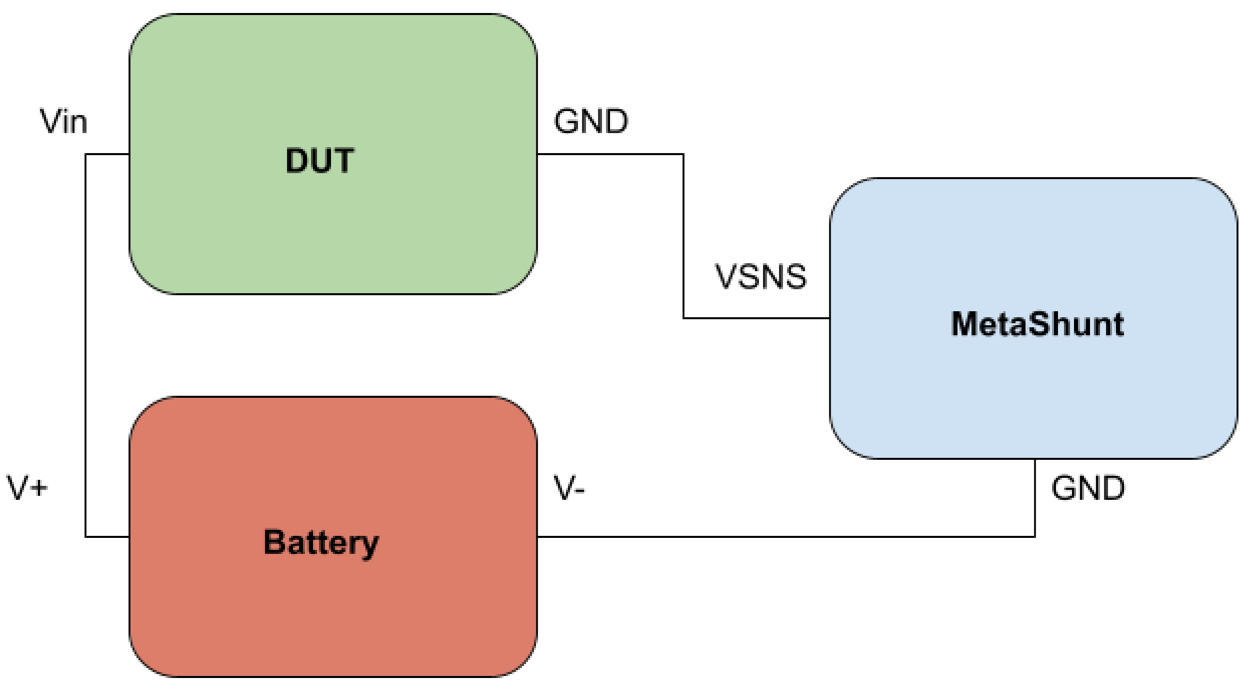
Connecting MetaShunt to Your Device

There are several ways that MetaShunt can be connected to your DUT.

First, if the DUT is powered by MetaShunt, connect the 5V or 3V3 output of MetaShunt to the Vin of your DUT. Then, connect the GND of your DUT to the VSNS pin of MetaShunt.



Second, if your DUT is powered by a battery, the battery's negative terminal can be connected to MetaShunt GND. The battery's positive terminal should be connected to the DUT Vin, and the DUT GND should be connected to MetaShunt VSNS.



Taking Measurements

MetaShunt communicates to your computer over USB serial using an open protocol. Open source, MIT licensed Python scripts for interfacing with MetaShunt are available on GitHub here:
https://github.com/jwachlin/metashunt_interface

To use, install Python and all required packages for the example. Connect MetaShunt to your computer, and then use the scripts to collect data. For measurements (recording data and showing plots of the data), see the "metashunt_realtime_interface.py" Python script in the “Realtime” Interface folder of the repository. To use, follow these rules:

python metashunt_realtime_interface.py h --- Provides helpful information

python metashunt_realtime_interface.py s [measurement_time_seconds] --- Get streaming data, by default for 10 seconds

python metashunt_realtime_interface.py b rate_hz --- Burst reads 32,000 samples immediately

python metashunt_realtime_interface.py b rate_hz r current_level_uA --- Burst reads 32,000 samples once current rises over the specified level

python metashunt_realtime_interface.py b rate_hz f current_level_uA --- Burst reads 32,000 samples once current falls below the specified level

python metashunt_realtime_interface.py b rate_hz s stage_index --- Burst reads 32,000 samples once system operates at specified stage index

Version History

Version	Date	Changes
1.0	February 23, 2024	Initial version