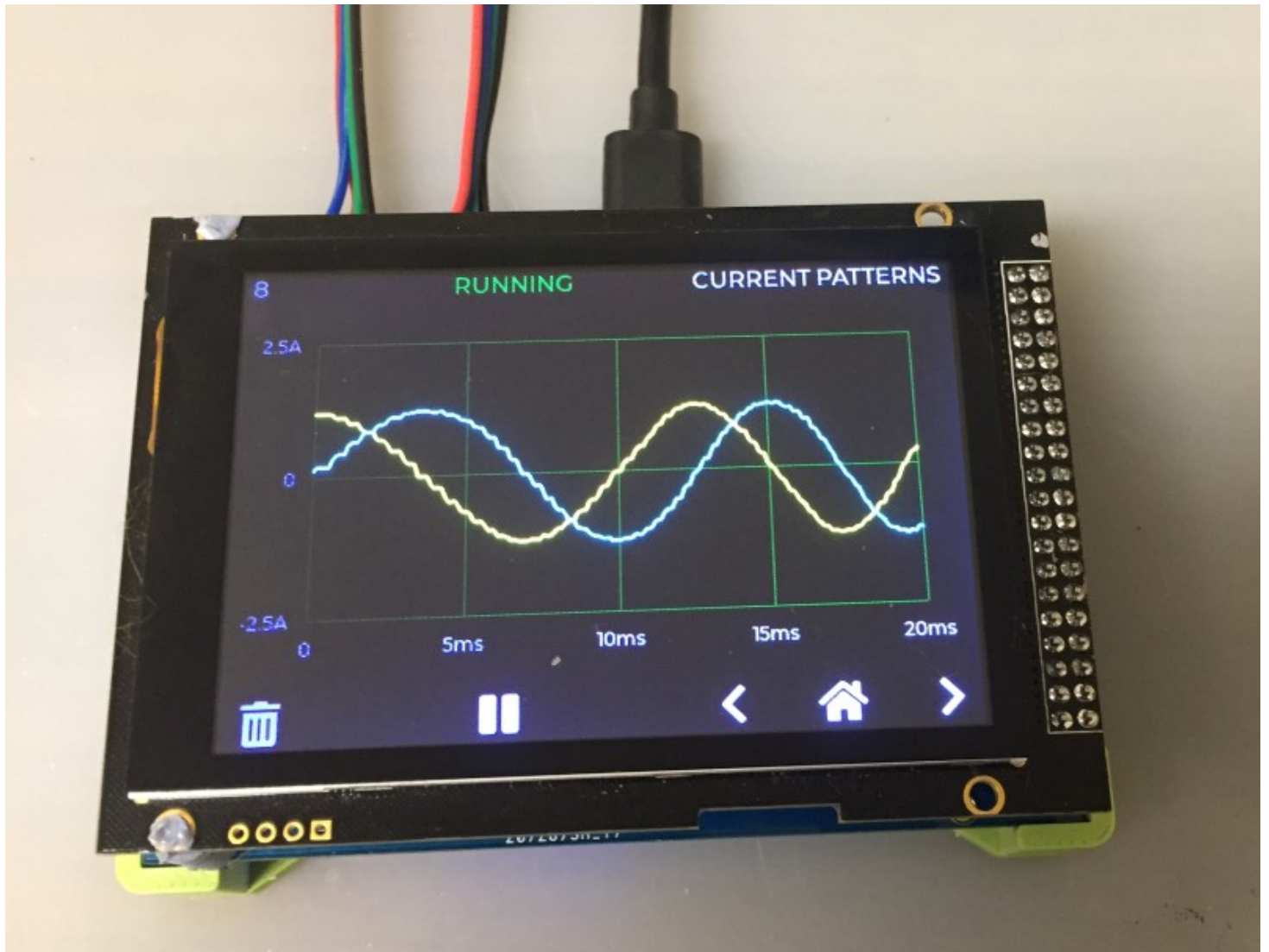


Simple Stepper Motor Analyzer



Overview

Diagnosing the operation of stepper motor in 3D printers and CNC machines used to require stepper motor analyzers that costs thousands of dollar. Not anymore! The Simple Stepper Analyzer ('Analyzer' in short) brings these capabilities to every maker for a tiny fraction of the price and with a minimalist DIY electronic design that leaves the heavy lifting to the firmware.

The Analyzer is self contained, has two 4-wires pass-through connectors connecting the stepper motor, is powered from a common 5V charger with a micro USB cable, provides a simple to use graphical user interface with a 3.5" color touch screen, and provides a plethora of stepper motor measurements.

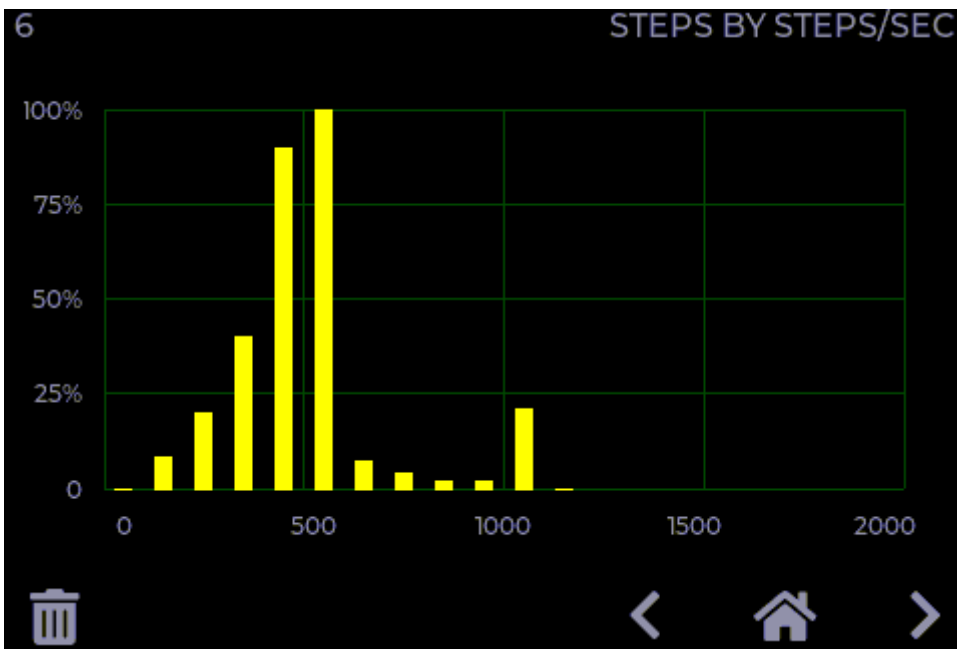
The Analyzer is provided to the community as a public domain reference design that can be built and used as is, or be customized with additional features or for a lower cost as desired. The hardware and firmware come with a liberal open source license (CC0) that allows commercial usages and doesn't require code source code sharing.

Disclaimer

THE DESIGN IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE DESIGN OR THE USE OR OTHER DEALINGS IN THE DESIGN.

How does it work?

The principle of operation is very simple, and so is the hardware design. The stepper motor is connected via 4 wire pass-through connectors that allow the Analyzer to monitor the current through the stepper's coils. The Analyzer's user low impedance current sensors that are galvanically isolated such does not interfere with the operation of the 3D printer or other equipment it monitors. The Analyzer measures the coils currents 100k times a second and uses firmware that analyzes the data, extracts the desired measurements, and present them on the screen. For example, this is how an analysis of the distribution of distance by speed distribution looks like:

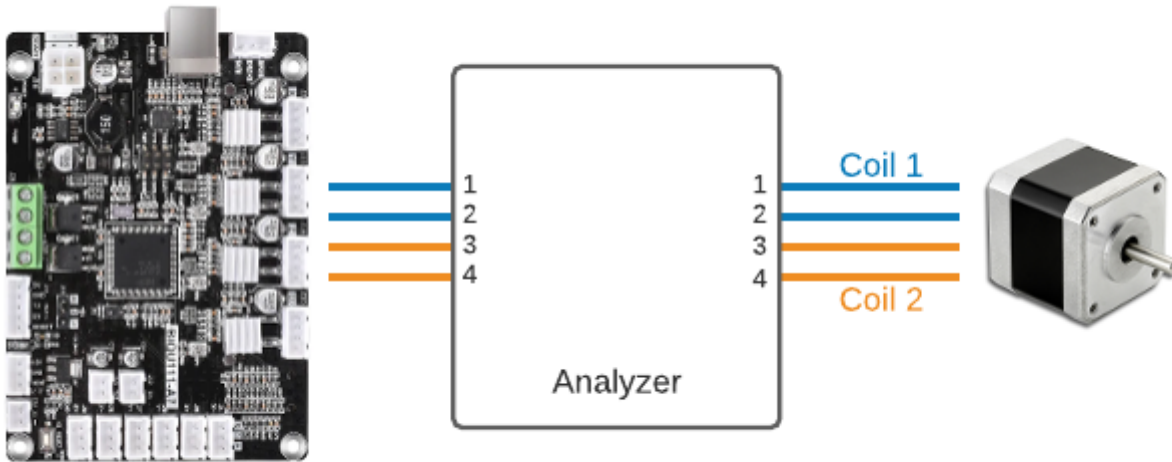


Operation is simple and intuitive

1. Turn off power to the equipment (e.g. 3D printer).
2. Disconnect the 4 wires of the stepper motor and connect them through the Analyzer two pass through connectors.
3. Power the analyzer using a 5V charger and a micro USB cable.
4. Turn on the equipment and get the stepper going.
5. Use the Analyzer's screen to view the measurements you want.

NOTE: The two pass-through connectors of the Analyzer are symmetric and each can act as an input or output.

CAUTION: Always connect and disconnect stepper motors when their controller is not energized, to reduce the risk of damage to the controller due to high voltage spikes. The analyser itself is not subject to that risk since its uses galvanically isolated current sensors.

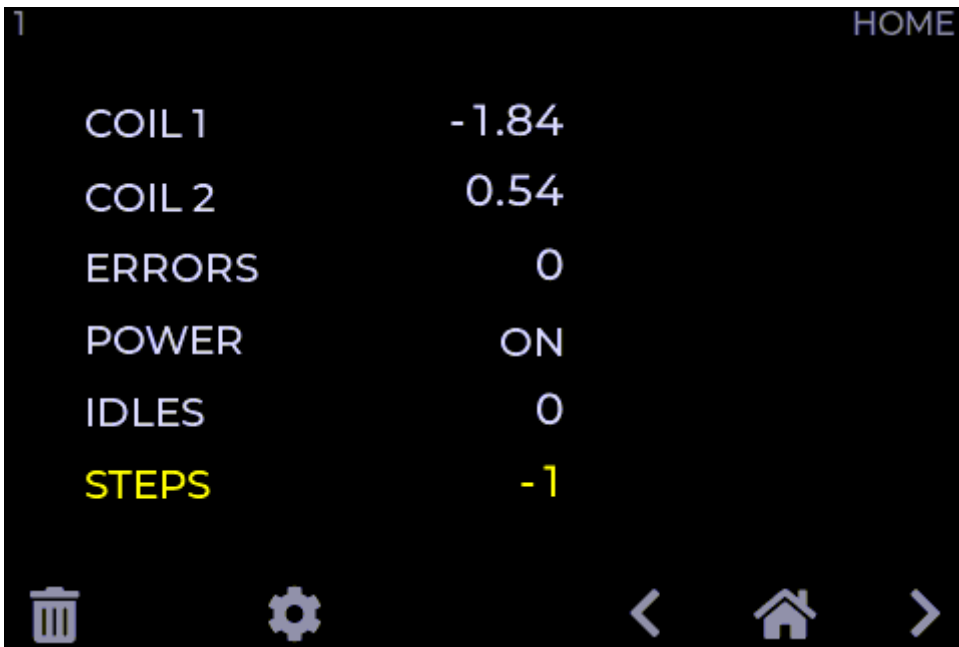


NOTE: The only requirement for proper wiring is that the two wires of one coil should go through pins pair 1,2 and the two wires of the other coil should go through pin pair 3,4. The Analyzer doesn't care about the +/- polarity of the wires in each pair, nor does it care which of the two pass-through connectors is on the controller side and which one is on the stepper side.

TRIVIA:: The pin out and the connector type of the Analyzer follows the Duet3D conventions, except that it uses right angle connector to save space.

The Home Page

The Home page is first page shown when the Analyzer is powered on. It provides summary information as well as access to the Settings page.



Page Data

| Field | Description |
|--------|--|
| COIL 1 | Momentary coil 1 current in Amps. |
| COIL 2 | Momentary coil 2 current in Amps. |
| ERRORS | The number of phase errors detected so far. Phase errors happens when the current patterns are so distorted that a forward/backward movement cannot be determined. |
| POWER | Indicates if the stepper motor is currently energized. This is determined by the sum of currents in they two coils. |
| IDLES | The number of times that stepper motor change status from energized to non energized. |
| STEPS | The distance the stepper moved so far. Each full step forward increments this value by one and each full step backward decrements it by one. |

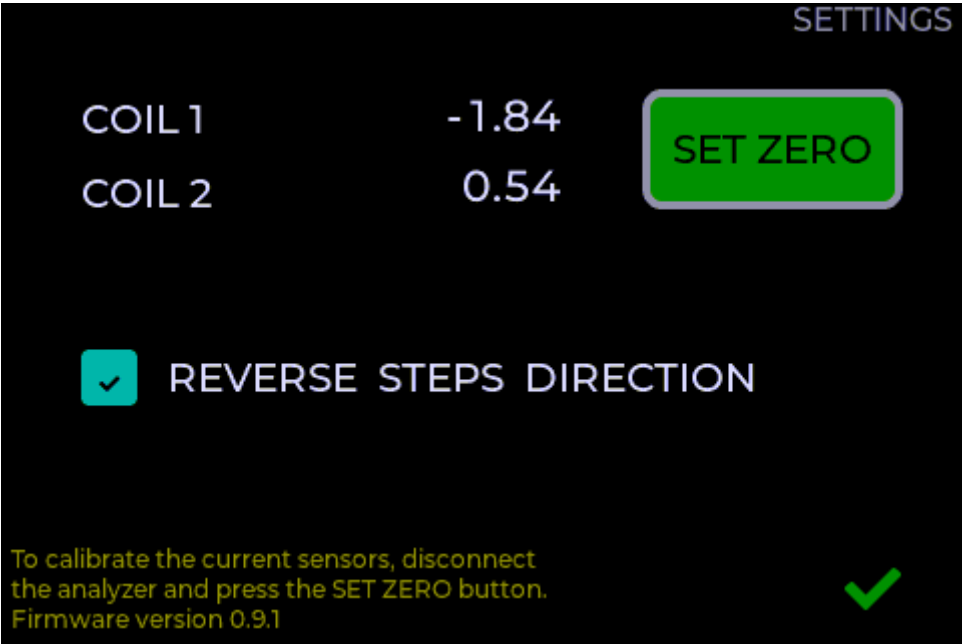
HINT: To swap forward/backward directions of the Analyzer use the Settings page. This does not affect the stepper motor itself.

Page Actions

| Action | Description |
|--------|-------------------------|
| | Clear all data. |
| | Open the Settings page. |

The Settings Page

This is a special page that is accessible from the Home page and allows to calibrate the Analyzer. The settings are stored in a non volatile memory and are preserved when the Analyzer is turned off.



Page Data

| Data | Description |
|--------|--|
| COIL 1 | The momentary current of stepper coil 1 in Amp units. Should be zero, or very close to zero, when the stepper motor is disconnected. |
| COIL 2 | The momentary current of stepper coil 2 in Amp units. Should be zero, or very close to zero, when the stepper motor is disconnected. |

Page Actions

| Action | Description |
|-------------------------|---|
| SET ZERO | Used to calibrate the zero reading of the current sensors. To do so, Disconnect the stepper motor and press this button. |
| REVERSE STEPS DIRECTION | Changes the forward/backward step directions of the Analyzer. Does not affect the stepper motor itself. Having the correct direction is important for measurements such as the Retraction analysis. |
| | Press exist the Settings page. |

The Speed Dial Page

This page measure the stepper speed in units of full steps per second.



Page Data

| Data | Description |
|------|-------------|
|------|-------------|

| Data | Description |
|-------------|--|
| Speed Dial | The speed dial shows the momentary stepper speed in full steps per second units. Forward speed is on the right side and backward speed is on the left. |
| Speed Value | This field shows the speed in a numeric form. IT is updated less frequently than the dial to allow easy reading. |

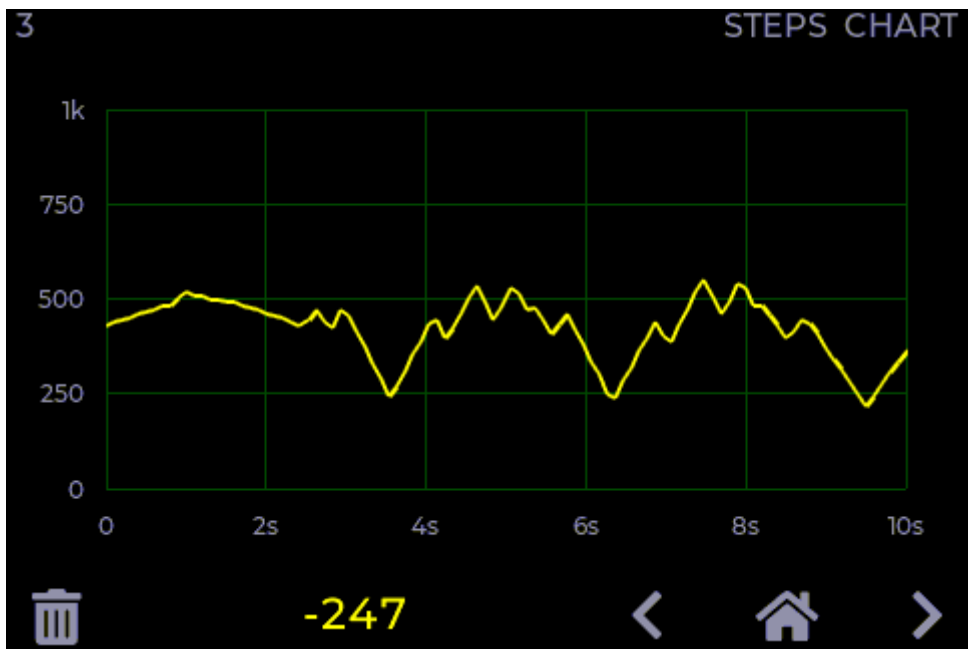
HINT: The Settings page allows to swap the forward/backward directions.

Page Actions

| Action | Description |
|--------------|---------------------------------|
| | Clears all data. |
| Tap the dial | Toggles the dial's speed range. |

The Distance Graph Page

This page shows scrolling graph with the total number of full steps vs. time.



Page Data

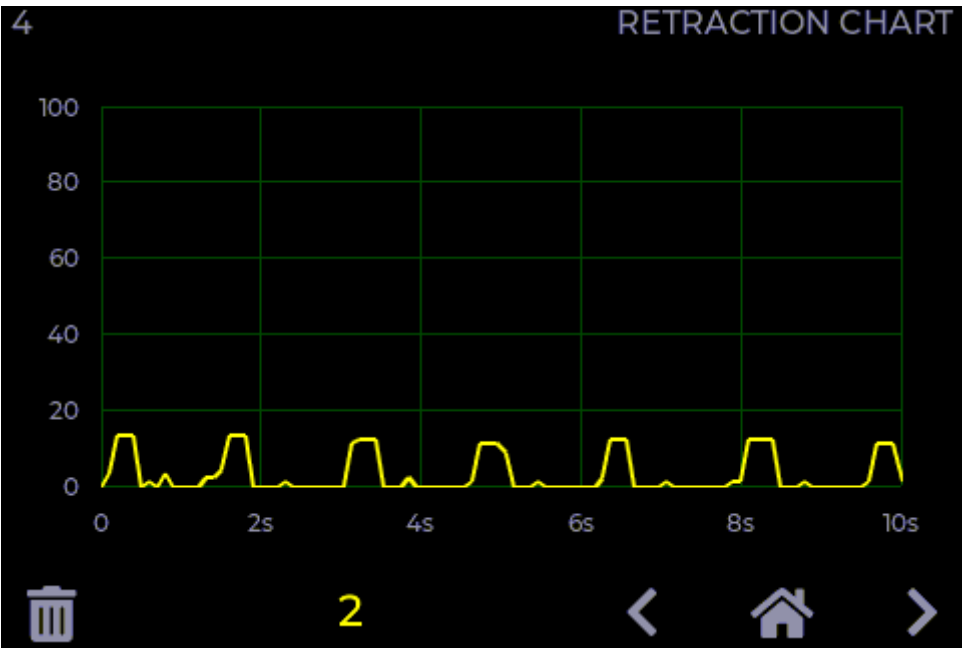
| Data | Description |
|----------------|--|
| Distance Graph | This graph shows how the distance, in units of full steps, changes over time. |
| Distance value | The numeric value of the distance in full steps. This fields is identical to the STEPS field in the Home page. |

Page Actions

| Action | Description |
|---------------|---|
| | Clears all data. |
| Tap the graph | Toggles the range of the vertical axis. |

The Retraction Graph Page

This page is useful to analyze the retraction in of 3D printer extruder.



TRIVIA: The graph above was taken while testing a direct extruder printer with [Thing #909901](#).

Page Data

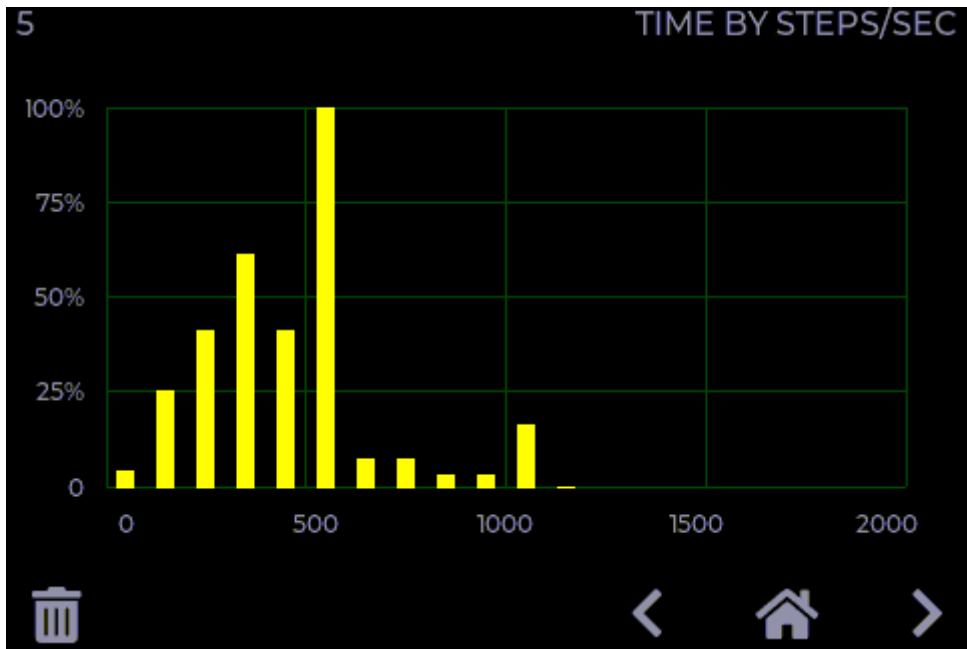
| Data | Description |
|------------------|---|
| Retraction Graph | This graphs tracks the retraction of 3D extruder full step units. Retraction is defined as the filament length that was inside the extruder and is now retracted outside of the extruder. |
| Retraction value | This field shows the current retraction level in a numeric form. |

Page Actions

| Action | Description |
|---------------|---|
| | Clear all data. |
| Tap the graph | Toggles the range of the vertical axis. |

The Time by Speed Page

This page shows an histogram with the distributing of time by stepper speed. That is how much time was spent in each speed range. The values are normalized such that the longest bar is always at 100%.



Page Data

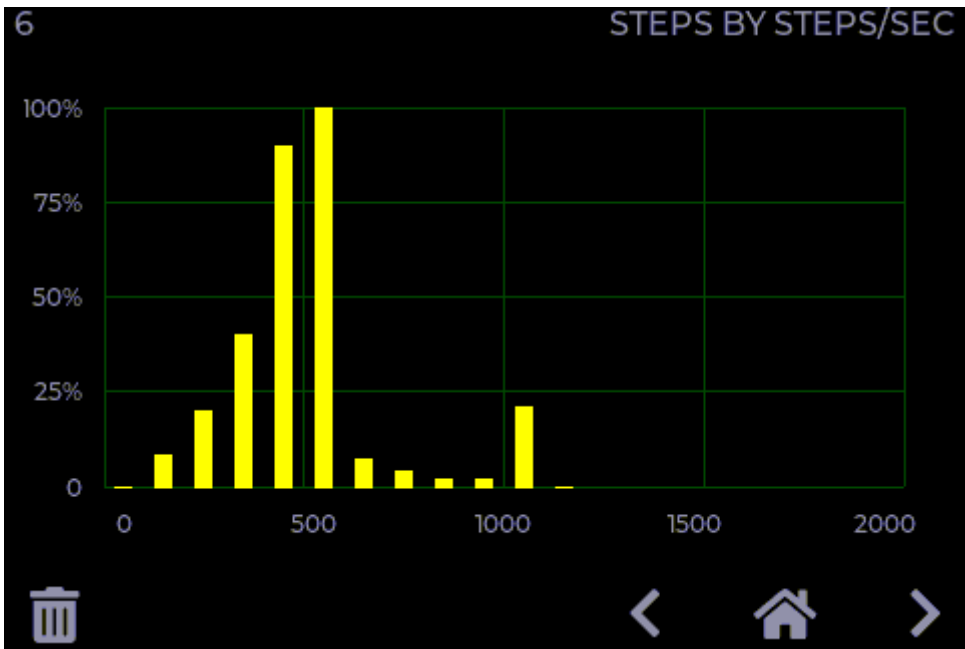
| Data | Description |
|----------------|---|
| Time Histogram | The horizontal axis indicates speed in full steps per second units. The height of each bar indicate the time of all steps in that speed range, with the length of the longest bar normalized to 100%. |

Page Actions

| Action | Description |
|--------|-----------------|
| | Clear all data. |

The Distance by Speed Page

This page shows an histogram with the the distribution of distance by stepper speed.



Page Data

| Data | Description |
|--------------------|--|
| Distance Histogram | The horizontal axis indicates speed in full steps per second units. The height of each bar indicate the distance or numberr of steps in that speed range, with the length of the longest bar normalized to 100%. |

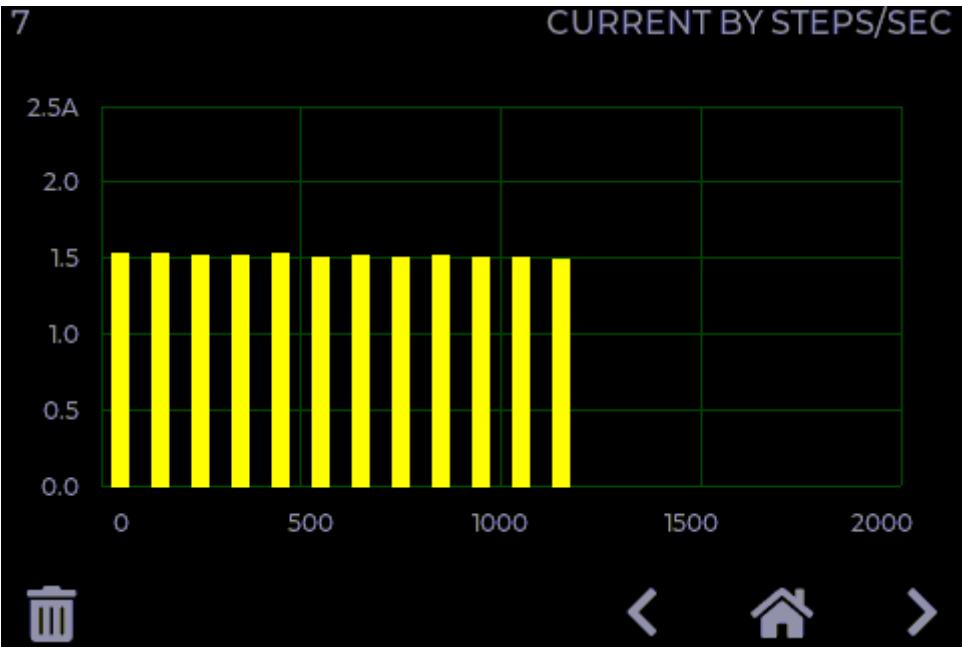
Page Actions

| Action | Description |
|--------|-----------------|
| | Clear all data. |

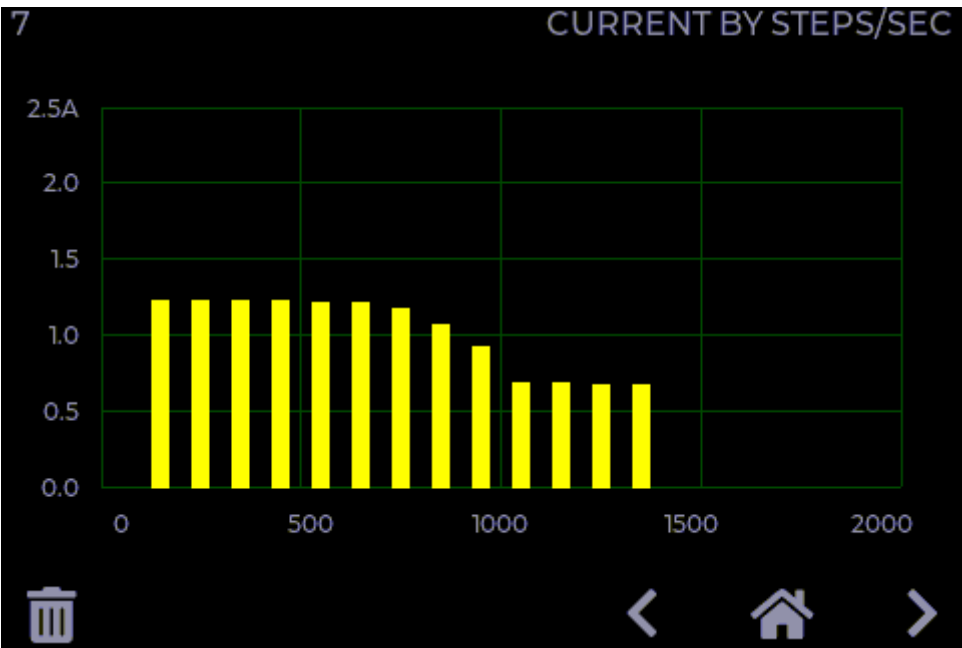
The Current by Speed Page

This pages measures the ability of the stepper motor's controller to maintain peak current at high speeds.

The screenshot below shows a healthy system where the peak current is virtually flat across the speed range.



The screenshot below shows a system that is not capable to maintain peak current and starves the stepper motor at high speeds, resulting in poor torque and possible skipped steps.



Page Data

| Data | Description |
|------|-------------|
|------|-------------|

| Data | Description |
|------------------------|--|
| Peak Current Histogram | The horizontal axis indicates speed in full steps per second units. The height of each bar indicates the average peak current in Amps of steps that speed range. |

Page Actions

| Action | Description |
|--------|-----------------|
| | Clear all data. |

Coil Current Patterns Page

This pages shows the current patterns in the two stepper coils.

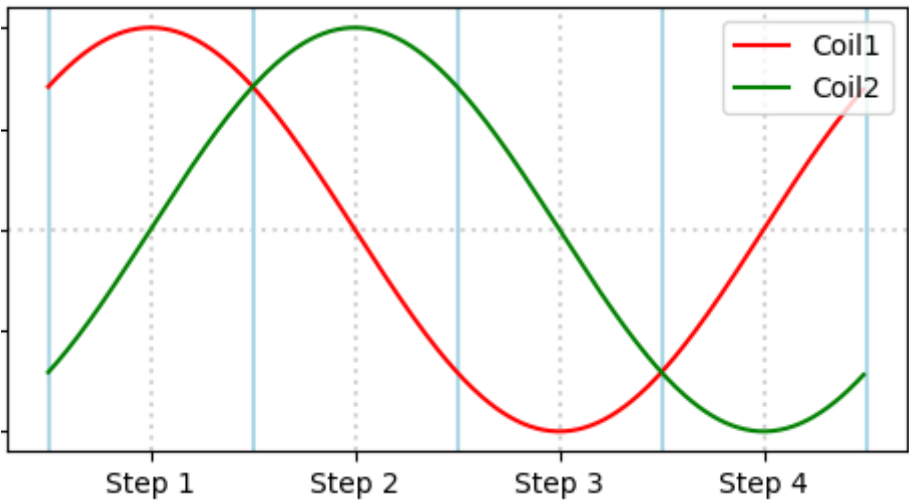
The screenshot below shows a healthy system where the signals are nice and clean and reaches the target peak current.



The screenshot below is from a poor system where high speed results in distorted and insufficient current, resulting in loss of torque and a potential for skipped steps.



GENERAL INFORMATION: Each cycle of the stepper current pattern represent 4 full steps. The centers of those steps are at the points any of the coils is at its maximal or minimal current as shown in the diagram below.



Page Data

| Data | Description |
|------|-------------|
|------|-------------|

| Data | Description |
|---------------|--|
| Current graph | The horizontal axis indicates time in milliseconds. The vertical axis indicate coil current level in Amps. The two graphs show the current patterns of the respective stepper coils. |

NOTE: The graph is updated only when sufficient stepper motor movement is detected.

Page Actions

| Action | Description |
|---------------|--|
| | Clear all data. |
| | When running, pauses the display updates. |
| | When paused, resume the display updates. |
| Tap the graph | This toggles the range of the horizontal axis. |

Specification

| Item | Specification |
|---------------------|--|
| Power consumption | 5V, 175ma. |
| Power connector | micro USB |
| MCU | 32bit ARM STM32F401CE @ 84Mhz. |
| Flashing | USB/DFU or SWD/Stlink V2. |
| Data link | USB/Serial link available but not used. |
| Current measurement | +/-2.5A per coil. +/-5A sensors available. |
| Dimensions | 97 x 66 mm |
| TFT Resolution | 480 x 320, Color. Landscape mode. |

| Item | Specification |
|---------------------------|--|
| Touch screen | Capacitive |
| Sensor isolation | See ACS70331 data sheet |
| Sampling rate | 100Khz per channel. |
| Sampling resolution | 12bits |
| Current accuracy | estimated at +/- 1% |
| Max speed | Software dependent. Currently 2K full steps/sec. |
| Partial steps measurement | Software dependent. Not implemented. |
| Settings storage | On board EEPROM chip. |
| GUI framework | LVGL library. R3G3B2 color depth. |
| Programming language | C++ |
| Programming IDE | Platform.io . Arduino/STM32. |
| Cad package | Kicad. |

I want to build me one

The electronic schematic, PCB layout, gerber files, and the firmware source files and released binary are all available on this github repository.

[Schematic](#)

BOM

| Part | Specification | Info |
|------|---------------|------|
|------|---------------|------|

| Part | Specification | Info |
|-----------------------------|---|---|
| TFT display | ER-TFTM035-6, Pin Header Connection-8080 16-bit Parallel, 5.0V, Capacitive Touch Panel, No font chip. | https://www.buydisplay.com/lcd-3-5-inch-320x480-tft-display-module-opti-touch-screen-w-breakout-board |
| STM32F401CE BlackPill | WeAct Studio STM32F401C, 25Mhz crystal. | https://github.com/WeActTC/WeAct-Studio-Product |
| 2 x current sensor. | Allegro +/-2.5A galvanic isolated current sensor. | Allegro ACS70331EESATR-2P5B3 |
| Settings EEPROM | Microchip AT24C02D-SSHM-T | 256 bytes, I2C, 8-SOIC. |
| TFT connector | 2x20 pins, 0.1" pitch, female header. | |
| 2 x stepper connector | 4 pin, 0.1" pitch, right angle, male, KF2510 connector. | Available from Aliexpress. |
| Resistors, capacitors, LEDS | Per the schematic diagram | Typical package is SMD 0603. |
| PCB | 97 x 66 mm, 2 layers, standard design rules, gerber files are available on github | JLCPCB, PCBWay, etc, or shop at https://pcbshopper.com/ |

Instructions: Build the PCB, use DFU or SWD protocols to program the released binary into the Blackpill, plugin the TFT screen, power and use. See instructions below on how to flash the firmware release you can find on this github repository.

Flashing a new firmware

Flashing the Analyzer with firmware can be done in a few ways. For software developers we recommend using StLink V2 and SWD which are well supported by the platformio IDE and enable debugging and single stepping. A simpler approach that is recommended for end users is to use the builtin DFU protocol of STM32 processor such as the one we use here. Adafruit has a [good description of the process](#) with the only difference that to enable the DFU bootloader mode you will need to perform the following sequence rather than the one specified by Adafruit:

1. Press and hold the NRST button.
2. Press and hold the BOOT0 button.
3. Release the NRST button.
4. Release the BOOT0 button.

How to modify the firmware?

The source code of the Analyzer is available on this github repository. It is a STM32 C++ Platformio project that uses the Arduino framework. The user interface is based on the LVGL library and the project includes its own port to support TFT/Touch used. Knowledge of C++ and Platformio is required and is outside the scope of this document.

Ideas for improvements

The Analyzer is provided as a fully functional reference design and changes and the developers encourage improvement/modifications of the hardware and/or firmware are encouraged. This can include new features, new measurements and pages, extending the design to support 2 or 3 steppers at once to track the x/y/z of a 3D printer, reducing the hardware cost using less expensive screen TFT, or even eliminating the screen altogether and using the Analyzer as a dongle tethered to a computer via the existing USB/Serial link.

FAQ

Q: Why not a SPI based TFT?

A: We could not achieve fast enough display refresh rate with SPI so switched to 16 bit parallel data transfers.

Q: Can a cheaper TFT/touch be used?

A: Yes, but it may require firmware changes.

Q: Can the TFT screen be eliminated?

A: Yes. The Analyzer has a USB/Serial output can a small Analyzer dongle can communicate with a computer program. This will require firmware changes.

Q: Can I connect the Analyzer to the TFT using a FFC cable?

A: Yes. The PCB was designed for both 0.1" header and 40 pins 1.0mm FFC cable connections. You will need a short (e.g. 60mm or less) type B ('reversed') FFC cable. Cables longer than let's say 100mm may introduce noises that will interfere with the operation of the display.

Q: Can you add feature X or customization Y?

A: Most likely not. The reference design is provided for the community to change and mix.

Q: I want to commercialize or crowd source it, can you help me?

A: Most likely not. The reference design is open source and you will need to find others to help you.

Q: Where do I file a bug?

A: Use the github repository.

Q: Is it ok to connect/disconnect power to the Analyzer while my 3D printer prints?

A: Of course. The Analyzer is isolated from the stepper wires and should not interfere with the operation of the printer.

Q: The Analyzer uses LVGL lib, what is the license of that library?

A: At the time of writing this, the LVGL license is available here <https://github.com/lvgl/lvgl/blob/master/LICENCE.txt>. The Analyzer's developers are not affiliated with LVGL.

Q: Is the LVGL library included in the Analyzer's repository?

A: It's not needed. LVGL is supported by platform.io so the Analyzer's project is set for platform.io to fetch LVGL automatically.

Q: Can I use a STM32F401CE MCU with a 8Mhz crystal.

A: The provided firmware assumes a 25Mhz crystal. For a 8Mhz crystal you will need to change the firmware.

Q: I can't find a STM32F401CE/25Mhz blackpill but I can find an STM32F411CE/25Mhz, will this work?

A: Most like it will work with the provided firmware as is (and with no performance benefits), but we do not guarantee it. We may switch the official reference design to the STM32F411CE/25Mhz if the STM32F401CE/25Mhz, which is less expensive, will not be available.

Q: Can I use a blackpill with different pinout than WeAct ?

A: No, unless if you are willing to change the PCB.