

# Flea-Scope™ -- 18 Msps analog and more!

See: <https://rtestardi.github.io/pages>

by Rich Testardi (rtestardi@live.com -- e-mail if you want to help me build or sell this!)

Flea-Scope™ is a very low-cost (\$13 BoM + PCBA) and easy-to-use 18 million sample-per-second (Msps) USB oscilloscope and mixed-signal logic analyzer with a built-in waveform generator, and so much more! Flea-Scope can be controlled by any [Chromium-based web browser](#) that supports [WebUSB](#) API or [Web Serial](#) API running on a computer, tablet, or phone (!!!), with no need for further software install – just plug it in and open a web-page and you are up and running!



## Flea-Scope Specifications and Features

- 1 analog input with 12-bit ADC sampled at up to 18 million samples-per-second (18 Msps), with [roughly 4 MHz of usable analog bandwidth](#)
  - 1 full-size BNC connector with 1 Mohm, 14 pF input impedance
  - -6V to +6V input scale with 10 mV resolution (or -60V to +60V input scale with 100 mV resolution when used with x10 oscilloscope probe)
  - compatible with most x1 and x10 oscilloscope probes with full-size BNC connector
  - DC coupling or digital AC coupling
- 9 digital logic analyzer inputs co-sampled along with the analog input
  - compatible with 3.3V or 5V logic levels
- auto, level, rise, and fall analog/digital trigger modes
  - delayed trigger/capture capability (0us to 1s)
  - "trigger out" pin allows Flea-Scopes to work together simultaneously for more channels!
- 1 test waveform output
  - up to 40 kHz DAC-based sine, triangle, or ekg waves
  - up to 4 MHz square waves
- 2% accuracy goal for voltages and timing
- 1 micro-USB connector (full-speed USB isolator-compatible)
- simple web-page user interface
- runs on Windows, macOS, Linux, ChromeOS, and Android
- dimensions: 1" x 3" (2.54cm x 7.62cm)

## ...and So Much More!

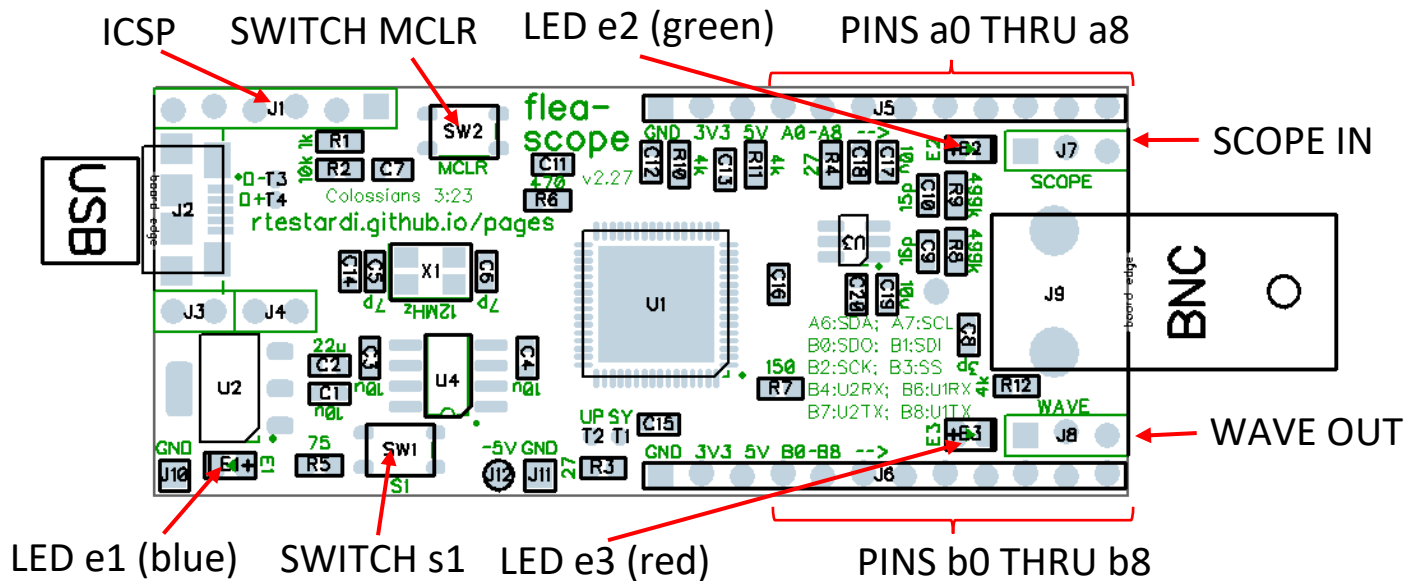
When used in "[deep-dive interactive mode](#)", Flea-Scope also features:

- StickOS® BASIC, an entirely MCU-resident interactive programming environment, with built-in:
  - program editor
  - line-by-line compiler
  - interactive debugger, including edit-and-continue
  - performance profiler
  - flash filesystem
- 120 MHz PIC32MK0512GPK064 MCU
  - 150,000 BASIC lines/second performance
- 18 fully programmable and interactive multi-function I/O pins, configurable for autonomous:
  - digital input/digital output
  - analog input/analog output (PWM)
  - frequency output
  - servo output (PWM)
  - I2C master mode
  - SPI master mode
  - UART input/output
- 2 user-controlled LEDs and 1 push-button switch
- compatible with 0.1" (2.54mm) solderless breadboards!
- simple web-page user interface
- customized hostname setting (to distinguish multiple Flea-Scopes on the USB and thru the user interface)
- firmware upgrade support

## Table of Contents

Flea-Scope Specifications and Features.....	1
...and So Much More! .....	1
Flea-Scope Board (shown four times actual size) .....	3
Controls.....	3
Pins .....	3
Connecting Flea-Scope.....	3
The Flea-Scope Web-Page User Interface.....	4
Calibrating Flea-Scope.....	5
Testing Flea-Scope .....	5
Setting Horizontal Scale (secs/div).....	6
Setting Analog Triggering (trigger and analog trigger voltage).....	6
Setting Digital Triggering (trigger and digital trigger bits b0 thru b8) .....	7
Setting Delayed Capture (delay sec) .....	7
Setting the Probe Type (probe).....	7
Setting the Input Signal Polarity or AC Coupling (signal) .....	7
Setting Vertical Scale (volts).....	8
Setting Waveform Generation (wave out and hertz out) .....	8
Pausing Trigger/Capture and Single Trigger/Capture (pause/resume/single) .....	8
Taking Display Area Measurements (measure) .....	8
Warnings/Limitations/Accuracy/Features .....	9
Analog Bandwidth.....	9
Compatibility .....	9
Deep Dive Interactive Mode (aka StickOS® BASIC Operating System) .....	10
Going Deeper Still! .....	10
How it Works.....	11
The Board:.....	11
The MCU .....	12
Firmware Upgrade .....	14
Firmware versions.....	15
Bugs.....	15
Github Repository .....	15
About the Author .....	15
Pins and Schematic .....	16

## Flea-Scope Board (shown four times actual size)



### Controls

- SWITCH MCLR -- resets the Flea-Scope microcontroller (MCU)
- SWITCH s1 -- (safemode switch used by StickOS in deep-dive interactive mode)
- LED e1 (blue) -- heartbeat LED (blinks 1Hz while Flea-Scope is idle and 2Hz while Flea-Scope is active)
- LED e2 (green) -- indicates trigger detected (also can be used by StickOS in deep-dive interactive mode)
- LED e3 (red) -- indicates waiting for trigger (also can be used by StickOS in deep-dive interactive mode)

### Pins

- PINS a0-a8 (J5) -- (3.3V multifunction and analog-capable pins used by StickOS in deep-dive interactive mode)
  - GND (square), 3V3, 5V, a0, a1, a2, a3, a4, a5, a6, a7, a8
  - *N.B. pins a1-a8 do not have built-in digital input pullups (they seem incompatible with analog inputs)*
- PINS b0-b8 (J6) -- 3.3V or 5V digital capture inputs (can also be used by StickOS in deep-dive interactive mode)
  - GND (square), 3V3, 5V, b0, b1, b2, b3, b4, b5, b6, b7, b8
- SCOPE IN pins (J7) -- scope trigger output and alternate scope analog input (instead of using BNC)
  - GND (square), TRIGGER OUT, SCOPE IN
- WAVE OUT pins (J8) -- 3.3V wave output from waveform generator
  - GND (square), WAVE OUT, WAVE OUT SERIES 4K
  - *N.B. WAVE OUT SERIES 4K has a 4k ohm series resistor which behaves as a bit of a low-pass filter*
- ICSP pins (J1) -- (only used for MCU initial system programming)
  - *N.B. beware that the square pin is MCLR, not GND, to be aligned with pin 1 of the programmer!!!*

## Connecting Flea-Scope

(For Ubuntu Linux, assuming user name "rich", to allow the browser to access USB serial ports, run these two commands: "sudo adduser rich dialout" and "sudo adduser root dialout", then log out and log back in.)

1. plug Flea-Scope's micro-USB connector into your computer, tablet, or phone, using a USB data cable
  - a. LED e1 (blue) should blink at 2Hz for 2 blinks, and then settle to a 1Hz blink while Flea-Scope is idle
2. open "[flea-scope.html](https://flea-scope.html)" from any Chromium-based web browser on your computer, tablet, or phone
  - a. click or tap the "Start Connect" button to authorize web browser access to a selected USB device
  - b. click or tap the desired Flea-Scope USB device or serial port (or customized hostname) to select it!**
  - c. click or tap "Connect" again to confirm the selected Flea-Scope USB device!

# The Flea-Scope Web-Page User Interface

The Flea-Scope web-page user interface contains the following components:



1. horizontal scale selection (secs/div) -- sets sample rate for input capture from 0.5us to 100ms per division
2. trigger mode selection (trigger) -- sets the trigger mode to one of: auto, level, rise, or fall
3. trigger voltage set (colored dashed line) -- displays the analog trigger level voltage against the analog waveform
4. digital trigger bits selection (b0 thru b8) -- set the digital trigger bits and start digital triggering
5. clear button (clear) -- clear all digital trigger bits to "don't care" ("-") and resume analog triggering at 0V "auto"
6. delayed capture seconds input (delay sec) -- sets the capture delay in milliseconds (#m) or microseconds (#u)
7. probe selection (probe) -- select x1 or x10 probe on BNC input (affects analog display area and volts selection)
8. signal selection (signal) -- selects unipolar (0 to +volts) or bipolar signal (-volts to +volts) or digital ac coupling
9. vertical scale selection (volts) -- vertical full scale of: 0.5, 1.0, 1.8, 2.5, 3.3, 5.0, or 6.6 volts (or 10x for x10 probe)
10. waveform output selection (wave out) -- sets the waveform type to one of: ekg, sine, square, or triangle
11. waveform output frequency input (freq out) -- sets the waveform frequency in hertz (#) or kilohertz (#k)
12. analog display area -- displays the captured analog waveform
13. digital display area -- displays the captured digital waveforms
14. calibrate zero volt button (cal\_zero) -- persistently set 0V calibration value for current probe type (x1 or x10)
15. calibrate 3.3 volt button (cal\_3v3) -- persistently set 3.3V calibration value for current probe type (x1 or x10)
16. pause/resume trigger and capture button (pause/resume) -- pause or resume triggering, capture, and display
17. single trigger and capture button (single) -- enable a single trigger and capture while paused
18. measure button -- measure one (absolute) or two (relative) points in the analog or digital display area
19. background button -- allow javascript to run in background window (make some noise on each display)
20. bw button -- switch screen between black-on-white with green and white-on-black with purple
21. horizontal scrollbar -- allows viewing of additional samples beyond those which are initially displayed
22. status area -- displays measured points time, frequency, and voltage, trigger voltage, or other status
23. voltage area -- displays average, peak-to-peak, and rms voltage of current analog capture
24. hostname area -- displays customized Flea-Scope hostname, if set (using "deep-dive interactive mode")

If you have multiple Flea-Scopes, you can set the "hostname" for each uniquely in deep-dive interactive mode, so that you can tell which is which in both the USB and web-page user interface.

## Calibrating Flea-Scope

Prior to use, Flea-Scope and its probe should be calibrated. The probe needs separate calibration values for x1 and x10 use. Calibration values are persistently stored in MCU flash. (Calibration values are cleared by “new” or “clear flash” commands in deep-dive interactive mode.) Flea-Scope will warn you whenever it is missing calibration values.

To calibrate Flea-Scope and its probe:

1. set the Flea-Scope web-page user interface to “1m” secs/div, “auto” trigger (the default)
2. set probe switch to x1 mode
3. select "x1" probe in Flea-Scope web-page user interface
  - a. connect the BNC input tip or scope input pin to ground
  - b. click or tap the "cal\_zero" button
  - c. connect the BNC input tip or scope input pin to 3.3 volts
  - d. click or tap the "cal\_3v3" button
4. set probe switch to x10 mode
5. select "x10" probe in Flea-Scope web-page user interface
  - a. repeat steps a-d above

You can find ground and 3.3 volts on the "a" or "b" pin strips.

Note that x1 and x10 will always have different calibration values, so you need to be sure your "probe" selection in the web-page user interface is consistent with your probe switch setting (this will also ensure the analog display area labels and volts selections are accurate -- don't multiply or divide by 10 in your head just to avoid setting the web-page user interface to match the probe switch!).

Periodic recalibration is recommended especially as temperature changes. You can force possibly wrong calibration values by selecting the “measure” button prior to calibration.

*Remember to also set your x10 probe compensation -- you can use the waveform output of Flea-Scope set to 1kHz square wave for that purpose! If you find your signal is always “over-compensated” (with sharp peaks and overshoot), you may add a 5-15 pF capacitor between SCOPE IN and ground (GND) pins (see above) to increase the overall capacitance of the Flea-Scope input circuit to match the higher capacitance of your x10 probe compensation range.*

## Testing Flea-Scope

By default, Flea-Scope generates a 1 kHz sine wave on its wave output pin, using a hundred digital-to-analog-converter points per wave cycle.

To test Flea-Scope, after calibrating it and its probe, simply connect Flea-Scope's wave output pin to its BNC input tip or scope input pin. If you also want to see digital activity, you can also connect the wave output to a digital input pin as well -- I used "b1" above (I also had some external activity on b7). If you don't have a probe, you can use a spring clip to connect the “WAVE OUT SERIES 4K” pin to the “SCOPE IN” pin, as found above.

You will now see the 1 kHz sine wave in the Flea-Scope analog display area that is being measured by the scope input pin, as above!

By default, the Flea-Scope "horizontal scale" is set to 1ms/division, and there are 10 labeled divisions across the display, resulting in 10ms total time from the left edge to the right edge of the display; the samples captured immediately after the "trigger" are on the left edge of the display, and time increases to the right.

By default, the Flea-Scope "vertical scale" is set to unipolar signal mode with 0V for the bottom of the display and to 3.3V for the top of the display.

## Setting Horizontal Scale (secs/div)

The "horizontal scale" determines how quickly or slowly Flea-Scope captures both analog and digital inputs for display after a "trigger" occurs. Flea-Scope captures roughly 1000 samples from the left edge to the right edge of the display -- so roughly 100 samples per display "division". Flea-Scope captures roughly an additional 1000 samples beyond the right edge of the display which can be viewed using the horizontal scrollbar.

The horizontal scale can be set from 100ms per division (for a total capture time over 10 divisions of 1 second, from left edge to right edge of the display) down to 0.5us per division (for a total capture time over 10 divisions of 5 microseconds, from left edge to right edge of the display).

Please be aware that the Flea-Scope hardware is capable of capturing up to 18 million samples-per-second on both analog and digital inputs simultaneously. At the fastest end of the horizontal scale range (2.5us/div or less, whose values are indicated with "\*" in the web-page user interface), Flea-Scope begins to "magnify" captured samples horizontally so that rather than seeing roughly 100 samples per display division, you may only see roughly 50, 20, or even 10 samples per display division.

## Setting Analog Triggering (trigger and analog trigger voltage)

Flea-Scope waits for a "trigger" event to occur before capturing and displaying both analog and digital sample data at the rate specified by the "horizontal scale" secs/division.

Flea-Scope supports the following trigger modes:

- auto -- like a "level" trigger, but if no trigger occurs within 100ms, trigger anyway (so the display is never blank)
- level -- trigger whenever the analog signal is above the specified trigger level (the colored dashed line on display)
- rise -- trigger whenever the analog signal transitions from below to above the trigger level
- fall -- trigger whenever the analog signal transitions from above to below the trigger level

To specify the analog trigger level, just click or tap in the analog display area at the height/voltage you are looking for, and you will see a green or purple dashed line appear indicating the current trigger level.

By default, the Flea-Scope trigger level is set to 0V and the mode is set to "auto", so the trigger is always active (since the signal is almost always above 0V, and if it is at exactly 0V, "auto" takes over every 100ms), so you will see the waveform display bouncing about on each capture. Flea-Scope can capture up to about 10 triggers each second, depending on the performance of the computer it is connected to...

To understand and experiment with the trigger setting, while examining a 1 kHz test sine wave, do the following:

- click or tap in the middle of the analog display area to set a mid-voltage trigger level -- now you will see that the waveform is always at or above that voltage on the left edge of the display where the trigger starts -- the display is still bouncing, but less than before.
- click or tap higher in the middle of the analog display area to set a higher voltage trigger level -- the higher you set the trigger voltage level, the less bounce, because we always start capture when the waveform is above the trigger level!
- click or tap below the analog display area to set a 0V trigger level again -- now you will be back to the default setting with the waveform display bouncing around randomly.
- again, click or tap in the middle of the analog display area to set a mid-voltage trigger level, and now change "auto" to "rise" -- you will see no bouncing at all because now the trigger occurs when the signal (after 3 samples of digital filtering) transitions from below to above the trigger level -- you will see the signal always starts at the trigger level and goes up from there!
- now change "rise" to "fall" -- again, you will see no bouncing at all because now the trigger occurs when the signal (after 3 samples of digital filtering) transitions from above to below the trigger level -- you will see the signal always starts at the trigger level and goes down from there!



If you ever see a completely blank (or stale) display, it is likely because you have the trigger mode set to "level" or "rise" or "fall", but the input waveform is not above or transitioning above or below the trigger level -- you will see a red "[waiting]" in the command status area -- set the trigger mode back to "auto" to see where your waveform is at, and then re-set the trigger level and mode appropriately.

## Setting Digital Triggering (trigger and digital trigger bits b0 thru b8)

By default, Flea-Scope waits for an analog trigger event on its analog input using a combination of the trigger mode (auto, level, rise, or fall) and the analog trigger level (colored dashed line).

Optionally you can specify that Flea-Scope should wait for a digital trigger event by setting one or more of the various trigger bits (b0 thru b8) at the top of the web-page user interface to "0" or "1". Bits that are not specified are "don't care". Waveforms in the digital display area corresponding to set digital trigger bits will be displayed in green or purple.

Note that digital triggering works much like analog with regard to the trigger mode:

- auto -- like a "level" trigger, but if no trigger occurs within 100ms, trigger anyway (so the display is never blank)
- level -- trigger whenever the specified digital bits "all match"
- rise -- trigger whenever the specified digital bits transition from "not all matching" to "all matching"
- fall -- trigger whenever the specified digital bits transition from "all matching" to "not all matching"

If you ever see a completely blank (or stale) display, it is likely because you have the trigger mode set to "level" or "rise" or "fall", but the input bits are not matching or transitioning to or from the trigger bits -- you will see a red "[waiting]" in the command status area -- set the trigger mode back to "auto" to see where your input bits are at, and then re-set the trigger bits and mode appropriately.

To return to analog triggering, click or tap the "clear" button to set all digital bits back to "don't care".

Note that by using the "trigger out" pin of one Flea-Scope as a digital trigger event of another Flea-Scope, you can capture multiple analog inputs (nearly) simultaneously!

## Setting Delayed Capture (delay sec)

By default, when a trigger is detected, Flea-Scope immediately starts capturing analog and digital samples at the rate specified by the "horizontal scale" secs/division.

When a "delayed capture" is specified by having a non-0 number in the "delay secs" input of the web-page user interface, Flea-Scope delays capture of the analog and digital samples by the specified time (up to a second) following the trigger. The time can be specified in milliseconds (by appending a "m" suffix to the number) or microseconds (by appending a "u" suffix to the number).

This allows you to "zoom in" to an event after the trigger by using a faster "horizontal scale" secs/division after the specified time has elapsed.

## Setting the Probe Type (probe)

Flea-Scope will work with most x1 or x10 probes. By default, Flea-Scope expects to be connected to an x1 probe, and the input signal range should be between -6 volts and +6 volts. Alternately, by using an x10 probe and selecting "x10" in the web-page user interface "probe" drop-down, Flea-Scope will measure and display signals between -60 volts and +60 volts.

## Setting the Input Signal Polarity or AC Coupling (signal)

By default, Flea-Scope displays unipolar signals -- that is, signals with a positive voltage relative to USB "ground". By selecting the "bi" in the "signal" drop-down, Flea-Scope will display signals with a positive or negative voltage relative to USB "ground". By selecting "ac", Flea-Scope will display signals relative to their digital average voltage.

## Setting Vertical Scale (volts)

The "vertical scale" determines what voltage range is visible in the Flea-Scope analog display area. By default, Flea-Scope measures and displays analog signals from 0V to 3.3V. Optionally, you can specify a different "full scale" display range from 0.5 to 6.6 volts using the web-page user interface "volts" drop-down.

Please note that decreasing the "full scale" display range does not increase the analog capture resolution of Flea-Scope. At the smallest end of the vertical scale range (0.33V or less, whose values are indicated with "\*" in the web-page user interface), Flea-Scope begins to "magnify" captured samples vertically (meaning the resolution is multiple pixels).

Digital logic analyzer inputs shown in the Flea-Scope digital display area represent signals from 0V to 0.6V as logic 0 and from 2.1V to 5V as logic 1.

## Setting Waveform Generation (wave out and hertz out)

Flea-Scope has two different waveform generation modes:

1. arbitrary waveforms generated up to 40 kHz using a digital-to-analog converter running at 100 points per wave cycle, repeating a 100 point waveform from a table over and over again (so if the arbitrary waveform is at 40 kHz, then the digital-to-analog converter is generating 100x more points than that, or 4 Msps).
2. square wave generation using a digital timer and output compare module running at 2 points per wave cycle (so if the square wave waveform is at 40 kHz, then the timer is essentially running at 40 kHz).

You can specify the waveform type using the "wave out" drop-down of the web-page user interface:

- ekg via arbitrary waveform generation
- sine via arbitrary waveform generation
- square via square wave waveform generation
- triangle via arbitrary waveform generation

And you can specify the frequency of the waveform by having a non-0 number in the "freq out" input of the web-page user interface. The frequency can be specified in hertz (by default) or kilohertz (by appending a "k" suffix to the number).

## Pausing Trigger/Capture and Single Trigger/Capture (pause/resume/single)

By default, Flea-Scope is always waiting for a trigger, and whenever the trigger occurs, it captures both analog and digital samples for display in the analog and digital display areas. It does this over and over again, giving an "animated" display of the analog and digital waveforms.

To pause trigger and capture (and freeze the display), click or tap the "Pause" button. While trigger and capture are paused, you can wait for a single trigger and capture analog and digital samples by clicking or tapping the "Single" button. To resume always waiting for a trigger over and over again and always capturing samples, click or tap the "Resume" button.

## Taking Display Area Measurements (measure)

To measure the absolute position of a point on the display area, click or tap the "Measure" button, and then click or tap the point. If the point is in the analog display area, both its voltage and time (relative to the left margin) will be displayed; if the point is in the digital display area, only its time will be displayed.

To measure the relative position of two points on the display area, click or tap the "Measure" button, and then click or tap the two points. If the points are in the analog display area, both their voltage difference and time difference will be displayed, along with their inverse time difference as a frequency; if the points are in the digital display area, only their time difference and inverse time difference as a frequency will be displayed.



## Warnings/Limitations/Accuracy/Features

If you have an older PC with 3-prong plug or are worried about ground loops at all, use a [full-speed USB isolator](#) to float the oscilloscope probe ground wire relative to PC ground.

Above all, Flea-Scope is made to be inexpensive and "good enough". This means we're using non-precision components and relying on calibration to get "good enough" voltage readings. Calibration values are stored in Flea-Scope flash memory.

There is only one input voltage range (-6V to 6V) even though there are multiple display ranges, and the ADC is running in 12-bit mode, resulting in roughly 4 mV theoretical vertical resolution, but with maybe 10-15 mV of noise floor.

The analog input uses "DC coupling" or "digital AC coupling"; "actual AC coupling" (for example with a DC offset beyond the ADC input range) can be achieved with an external 0.1uF capacitor.

Trigger delay is under 1us.

The digital-to-analog converter used for arbitrary waveform generation has a conversion speed of 1 Msps and no double-buffer, so there is some noise, especially at higher frequencies; WAVE OUT SERIES 4K has a 4k ohm series resistor which behaves as a bit of a low-pass filter.

### Analog Bandwidth

When pushing the limits using a 1 MHz square wave (whose bandwidth well exceeds 1 MHz, bypassing the probe and connecting WAVE OUT -- not WAVE OUT SERIES 4K -- directly to SCOPE IN!), we see decent signal fidelity, as expected for a true 18 Msps sampling oscilloscope:

#### FLEA-SCOPE™



Pushing further to 4 MHz, we see we are still well above the -3db (70.7%) "analog bandwidth" cutoff typically quoted for oscilloscopes, but, Nyquist is now limiting our signal fidelity:



### Compatibility

I test Flea-Scope with Windows, macOS, Linux, ChromeOS, and Android. On macOS, you might get better performance using the "manual" webpage to select Web Serial instead of WebUSB. I test with Google Chrome and Microsoft Edge browsers. For more information on WebUSB and Web Serial used by the Flea-Scope GUI web-page, see:

<https://wicg.github.io/webusb/> and <https://wicg.github.io/serial/>

I do not believe any iPhones or iPads yet expose USB devices thru WebUSB or Web Serial into the browser JavaScript, so they don't yet work with Flea-Scope. Some Samsung phones claim the USB device with their serial driver, but do not subsequently expose Web Serial into the browser, leaving the USB device inaccessible to JavaScript.

## Deep Dive Interactive Mode (aka StickOS® BASIC Operating System)

Rather than starting "flea-scope.html" to connect to the Flea-Scope, you can instead start "[stickos-basic.html](https://rtestardi.github.io/usbte/stickos-basic.html)", and get an interactive terminal emulator directly into the underlying StickOS® BASIC operating system running on the Flea-Scope MCU! (Be sure to close your flea-scope.html window first, as only one web-page at a time can open the Flea-Scope.)

To set the hostname to distinguish multiple Flea-Scopes on the USB and thru the user interface, use the command:

```
> hostname hostname
```

(Reset your Flea-Scope using the MCLR button after changing its hostname.)

From here you can also interactively configure pins, write and debug BASIC programs, and even configure a saved program to "auto run" when Flea-Scope is turned on, *turning Flea-Scope into the base for your embedded system!*

You can use 0.1" header pins to connect the Flea-Scope to a solderless breadboard or directly to the rest of your embedded circuit.

See the website: [CPUStick™ and StickOS® -- Embedded Systems Made Easy \(rtestardi.github.io\)](https://rtestardi.github.io)

And the StickOS User's Guide: [stickos.v1.90.pdf \(rtestardi.github.io\)](https://rtestardi.github.io/stickos.v1.90.pdf)

And the StickOS Quick Reference: [quickref.v1.90.pdf \(rtestardi.github.io\)](https://rtestardi.github.io/quickref.v1.90.pdf)

See examples of what you can do: [toaster.pdf \(rtestardi.github.io\)](https://rtestardi.github.io/toaster.pdf) and [simon.pdf \(rtestardi.github.io\)](https://rtestardi.github.io/simon.pdf)

Full online-help is available at the command prompt -- start with the "help" command:



## Going Deeper Still!

You can also use the 6-pin ICSP connector to reprogram the board in C directly from MPLAB X using a Microchip Pickit 3, Pickit 4, or "[Snap](#)" programmer! You can start with the "[hello](#)" repository on github, and take control of the whole board! You can even re-program back to Flea-Scope when you are done using board/pic32.X.27.production.hex in the same repository!

# How it Works

## The Board:



1. USB connector -- supplies 5V power and USB data connectivity to Flea-Scope board
2. 3.3V regulator -- regulates 5V to 3.3V for use by MCU and other components
3. DC-to-DC converter -- converts 5V to -5V for use by dual supply op amp
4. BNC connector -- high impedance input accepts -6.6V/+6.6V at 1 Mohm/14 pF
5. passive input circuitry -- divides -6.6V/+6.6V BNC input to -3.3V/+3.3V op amp input
6. dual supply op amp -- unity gain op amp to drive MCU op amp from high impedance BNC input
7. passive pullup circuitry -- divides bipolar -3.3V/+3.3V op amp output to unipolar 0V/+3.3V MCU op amp input
8. MCU -- manages USB interface and MCU timers, op amp, ADCs, comparators, DACs, etc., plus StickOS BASIC
9. Crystal -- provides 12MHz timing for MCU
10. passive reset circuitry -- resets MCU on power on or button press
11. ICSP connector -- programs MCU at board manufacture time

## The MCU

Flea-Scope uses a huge amount of automated hardware in the [PIC32MK0512GPK064 microcontroller](#) (MCU) to achieve its end result. The individual Analog-to-Digital converters (ADCs) in the MCU only run up to 3.75 million samples-per-second (MSPS), so Flea-Scope uses 5 of them interleaved together to measure the input signal in a time-synchronized fashion, thereby achieving 18 MSPS.

The notable PIC32 MCU hardware includes:

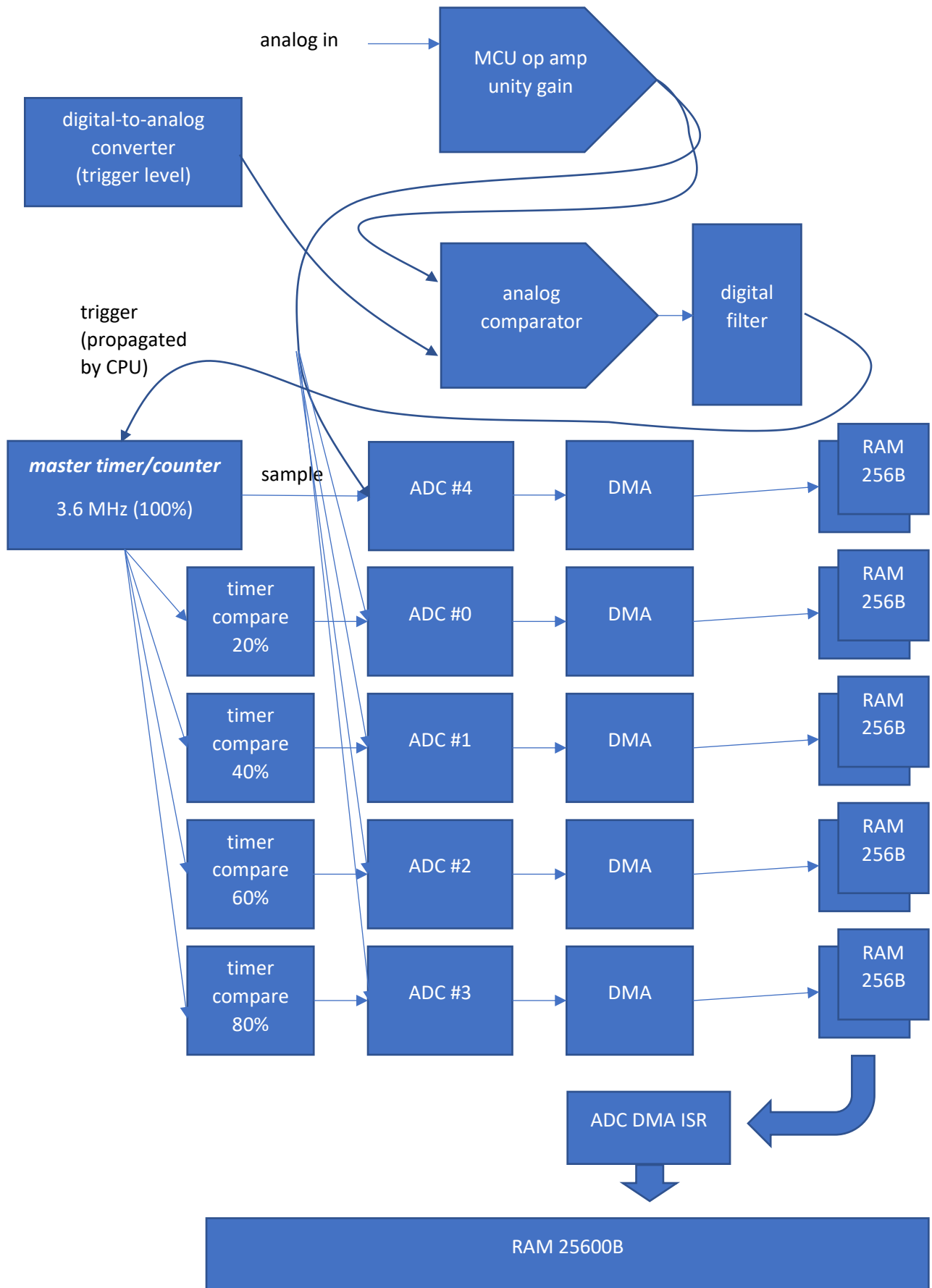
- one unity gain op-amp to drive the analog input signal to one input of the analog comparator as well as the five inputs of the five interleaved analog-to-digital converters (each of which has significant resistive and capacitive load).
- one control digital-to-analog converter (CDAC) to generate the requested analog trigger voltage level to drive the other input of the analog comparator.
- one analog comparator (CM) used to detect when the analog input signal is above or transitions above or below the analog trigger voltage level.
- one digital filter on the output of the analog comparator used to prevent spurious input signal noise from false-triggering a trace capture.
- one **master counter/timer** module (TMR) running at 1/5x the desired analog and digital sample rate, once we decide to capture a trace.
- four timer compare modules (OCx) to detect when the master counter/timer reaches 20%, 40%, 60%, and 80% of its maximum value.
- five interleaved analog-to-digital converters (ADCx) in 12-bit mode, whose measurement samples are triggered (i.e., started) at 20%, 40%, 60%, 80%, and 100% of the master counter/timer value.
- five dedicated ADC DMA (direct memory access) channels (ADCDx) to transfer the analog-to-digital converter results into ping-pong buffers in RAM once they are completed.
- one ADC DMA ISR (interrupt service routine) to reassemble the interleaved analog-to-digital converter ping-pong buffer results into time sequential results.
- five (more) general-purpose DMA channels (DCHx) to capture the digital port results concurrent to the analog-to-digital-converter results, again at 20%, 40%, 60%, 80%, and 100% of the master counter/timer value.

The microcontroller MIPS CPU running at 120 MHz is used before we decide to capture a trace simply to wait for the trigger event to occur, either polling the analog comparator for an analog trigger, or examining the digital port values for a digital trigger. Once the trigger event occurs, the CPU simply enables the **master counter/timer**, and hardware takes care of the actual trace capture. The CPU then waits for DMA to capture the desired number of samples, simply servicing the DMA ISR's as needed. Once the full trace has been captured, the CPU uploads the trace to USB where it is displayed by the web-page user interface.

Waveform generation is occurring concurrent with everything above using more hardware, including:

- one (more) counter/timer module (TMR) to generate the desired waveform frequency for square wave generation or 100x the desired waveform frequency for arbitrary waveform generation.
- for arbitrary waveform generation, one (more) general-purpose DMA channel (DCHx) to feed one (more) control digital-to-analog converter (CDAC) from a 100 point waveform table for the waveform output.

Below is a picture of the analog trigger/capture module configuration:



## Firmware Upgrade

Should you need to upgrade the firmware of your Flea-Scope, follow the procedure below.

1. *Note that a firmware upgrade clears all Flea-Scope flash memory, including probe calibration values, BASIC programs, flash variables, persistent modes, customized hostname, etc.*
2. log into “deep dive interactive mode”, as described above:

```
Welcome to StickOS for Microchip PIC32MK0512GPK v2.27g!  
Copyright (c) 2008-2023; all rights reserved. Patent U.S. 8,117,587.  
https://github.com/rtestardi/StickOS2  
rtestardi@live.com  
(checksum 0x7c8e)  
> echo on  
> prompt on  
> hostname  
Flea-Scope  
>
```

3. enter the “upgrade” command followed by <Enter>:

```
> upgrade
paste HEX upgrade file now...
```

4. copy the contents of the new firmware hex file to the clipboard (this will be about 5000 lines of text).
5. click the “Paste” button on the UI, and then “Allow” to allow the web-browser permission to access the clipboard.

```
> upgrade  
paste HEX upgrade file now...  
. . . . .  
. . . . .  
. . . . .  
. . . . .  
. . . . .  
. . . . .  
. . . . .  
. . . . .  
paste done!  
programming flash for upgrade...  
wait for heartbeat LED to blink!
```

A screenshot of a terminal window. At the top, there is a long, empty text input field. Below this field is a row of five buttons: 'Enter', 'Ctrl-C', 'Ctrl-D', 'Paste', and 'Clear'. A large red arrow points downwards from the top of the terminal window, specifically targeting the 'Paste' button.

[rtestardi's pages](#)   [StickOS® website](#)   [github.com/rtestardi/usbte](#)   [github.com/rtestardi/StickOS2](#)

6. After the paste is complete, wait for the blue heartbeat LED on the Flea-Scope to blink again -- this takes about 30 seconds.
7. *Note that if the Flea-Scope loses power during this 30 second window above, it will likely be bricked and can only be reprogrammed with the use of a "Snap" programmer as described in "going deeper still", above.*



## Firmware versions

Current firmware versions are here: <https://github.com/rtestardi/StickOS2/tree/main/board>

Known bugs in current firmware versions include:

**2.27h (checksum 0x990f)** -- fix bugs #2 and #3, 1 May, 2024

- (none)

**2.27g (checksum 0x7c8e)** -- initial release, 1 January, 2024

- #3 weak pull-ups (7k-20k ohm) are enabled by default on logic analyzer inputs, which can interfere with some pull-up dependent protocols like I2C
- #2 StickOS BASIC uart read/write statements garble data when read into or written from array variables

## Bugs

Submit Flea-Scope questions or bugs here: <https://github.com/rtestardi/usbte/issues>

## Github Repository

Rich's main github landing page is here: <https://rtestardi.github.io/pages/>

The Flea-Scope GUI software repository is at: <https://github.com/rtestardi/usbte>

The Flea-Scope MCU firmware and PCB hardware repository is at: <https://github.com/rtestardi/StickOS2>

A "deeper still" MPLAB X "hello world" LED blink application is at: <https://github.com/rtestardi/hello>

## About the Author



Richard Testardi has a wife and 17yo daughter and lives in Colorado. He is grateful most of the time and Christian. He loves anything outdoors or math/science related. In the future, he hopes to be teaching high school students. He lives without a cell phone (well, except a sim-less phone for interoperability testing!!)

## Pins and Schematic

board	a0	a1	a2	a3	a4	a5	a6/sda	a7/scl	a8	e2	e3
PIC32	RB7	AN12/RE12	AN13/RE13	AN14/RE14	AN15/RE15	AN16/RG9	AN17/RG8	AN18/RG7	AN19/RG6	RA4	RA10
board	b0/sdo	b1/sdi	b2/sck	b3/ss	b4/u2rx	b5	b6/u1rx	b7/u2tx	b8/u1tx	e1	s1
PIC32	RB4	RB5	RB6	RB10	RB11	RB12	RB13	RB14	RB15	RC6	RC7

