

Figure 1: Xprotolab, Xminilab, and the Portable models

Description:

The XScopes (Xminilab and Xprotolab) are a combination of three electronic instruments: a mixed signal oscilloscope, an arbitrary waveform generator, and a protocol sniffer; all housed in a small breadboard friendly module. The XScopes can also be used as development boards for the AVR XMEGA microcontroller.

Main Features:

- **Mixed Signal Oscilloscope:** Simultaneous sampling of 2 analog and 8 digital signals.
- **Arbitrary Waveform Generator** with advanced sweep options on all the wave parameters.
- **Protocol Sniffer:** SPI, I²C, UART
- **Advanced Triggering System:** Normal / Single / Auto / Free, with many trigger modes; adjustable trigger level, and ability to view signals prior to the trigger.
- **Meter Mode:** VDC, VPP and Frequency readout.
- **XY Mode:** For plotting Lissajous figures, V/I curves or checking the phase difference between two waveforms.
- **Spectrum Analyzer** with different windowing options and selectable vertical log and IQ visualization.
- **Channel Math:** add, multiply, invert, and average.
- **Horizontal and Vertical Cursors** with automatic waveform measurements, and waveform references.

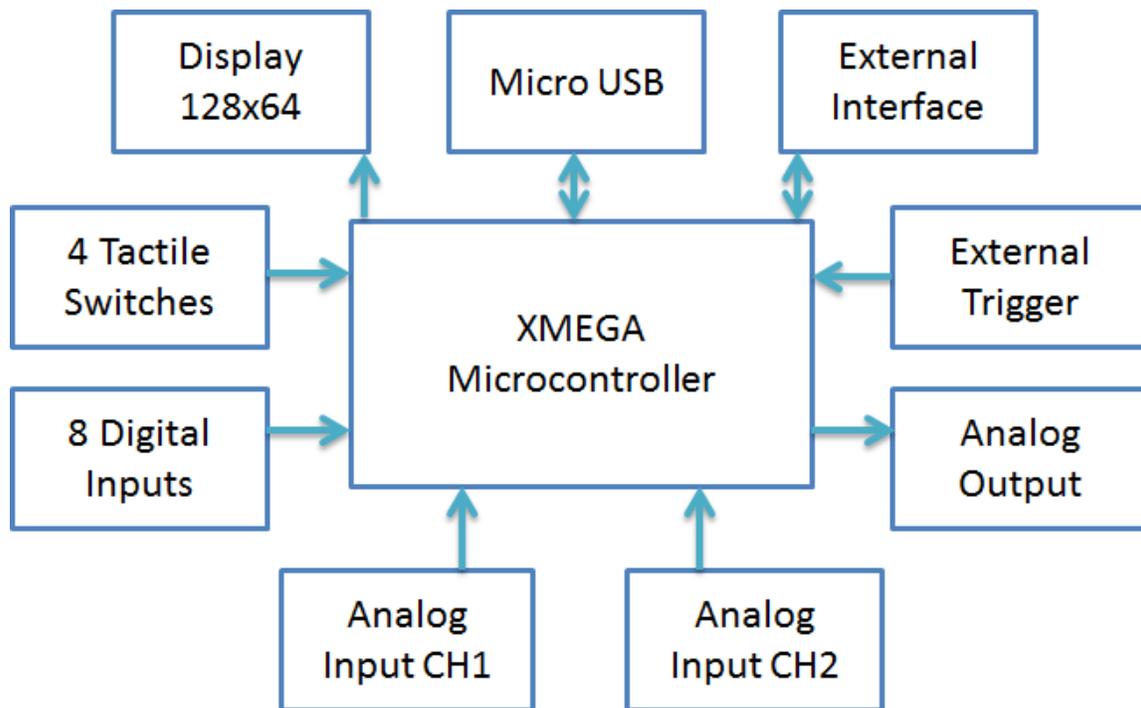


Figure 2: XScopes Block Diagram



About this manual

This manual targets both novice and advanced users, providing a full resource for everyone. However, for a full understanding of the operation of the XScopes, the user should be familiar with the operation of a regular oscilloscope.

The features documented in this manual are for units with firmware version 2.36+.

Conventions

XScope: Xprotolab, Xminilab, Xprotolab Portable or Xminilab Portable.

Portables: Xprotolab Portable or Xminilab Portable

CH1: Analog Channel 1

CH2: Analog Channel 2

CHD: Logic Inputs

Fast Sampling: 10ms/div or faster time base

Slow Sampling: 20ms/div or slower time base



Helpful tip



Warning



Technical Detail

Manual Revision History



Version	Date	Notes
3.0	December 2013	Updated protocol interface information
3.1	February 2014	Minor error corrections
3.2	May 2014	Firmware update on Linux
3.3	October 2014	Updated interface protocol
3.4	November 2014	Counter mode

TABLE OF CONTENTS

1. General Overview.....	7
1.1 Xprotolab & Xminilab Pin Description	7
1.2 Xprotolab Portable and Xminilab Portable Overview	8
1.2.1 Input Coupling Switch	8
1.2.2 Curve Tracer Switch	8
1.2.3 MENU / Power button	8
1.2.4 USB Port / Device Charging.....	8
1.3 Specifications	9
1.4 Dimensions	10
1.5 Absolute Maximum Ratings.....	11
1.6 Factory Setup	11
1.7 Quick Start Guide	11
1.8 User Interface	12
1.9 Saving the settings.....	12
2. Mixed Signal Oscilloscope.....	13
2.1 Horizontal Settings.....	13
2.1.1 Time Base	13
2.1.2 Technical Details.....	13
2.1.3 Explore Wave	13
2.1.4 Auto Setup	13
2.2 Vertical Settings.....	14
2.2.1 Disable Channel	14
2.2.2 Channel Gain.....	14
2.2.3 Channel Position	14
2.2.4 Channel Invert.....	14
2.2.5 Channel Math	14
2.3 Trigger Settings	15
2.3.1 Trigger Types	15
2.3.2 Trigger Modes.....	16



2.3.3	Trigger Hold	17
2.3.4	Post Trigger	17
2.3.5	Trigger Source	17
2.4	Device Modes	18
2.4.1	Oscilloscope Mode	18
2.4.1.1	Roll Mode	18
2.4.1.2	Elastic Traces	18
2.4.1.3	XY Mode	19
2.4.2	Meter Mode	19
2.4.2.1	Frequency Measurements	19
2.4.2.2	Pulse Counter	19
2.4.3	Spectrum Analyzer	20
2.4.3.1	IQ FFT Mode	20
2.4.3.2	Logarithm display	20
2.4.3.3	FFT Windows	20
2.5	Cursors	21
2.5.1	Vertical Cursors	21
2.5.2	Horizontal Cursors	21
2.5.3	Automatic Cursors	21
2.5.4	Track Horizontal Cursors	21
2.5.5	Reference Waveform	21
2.5.6	Cursors in XY Mode	21
2.6	Display Settings	22
2.6.1	Persistent Display	22
2.6.2	Line / Pixel Display	22
2.6.3	Show scope settings	22
2.6.4	Grid Type	22
2.6.5	Flip Display	22
2.6.6	Invert Display	22
3.	Logic Analyzer and Protocol Sniffer	23



3.1	Input Selection	23
3.2	Channel Position	23
3.3	Invert Channel.....	23
3.4	Thick Logic '0'	23
3.5	Parallel Decoding	24
3.6	Serial Decoding.....	24
3.7	Protocol Sniffer	24
3.8	Sniffers Modes.....	24
3.9	I2C Sniffer	25
3.10	UART Sniffer	25
3.11	SPI Sniffer	25
4.	Arbitrary Waveform Generator	26
4.1	Predefined Waveforms	27
4.2	Parameter Sweep	27
4.2.1	Sweep Modes	27
4.3	Technical Details	27
5.	PC Interface	28
5.1	Connecting the device	28
5.1	Controlling the device	28
5.2	Custom AWG Waveform	29
5.3	Saving Waveforms and Screenshots.....	29
6.	Interface Protocol	29
6.1	Interface settings	29
6.2	Control Data.....	29
6.2.1	Bitfield variables	31
6.3	Vendor ID and Product ID	32
6.4	Command Set	32
6.4.1	Auto Send	33
6.4.2	METER measurement data	33



7. BMP Screen Capture	34
7.1 To send a BMP screen capture to a PC:.....	34
7.2 To send a BMP screen capture to Linux:.....	35
8. XScope's Examples	36
8.1 Resistor Voltage Divider	36
8.2 Measurement of an RC time constant	36
8.3 Half Wave Rectifier with Smoothing Capacitor.....	36
8.4 BJT Amplifier.....	37
8.5 Component V/I Curves.....	37
8.6 Frequency Plots	37
9. Firmware Updating	38
9.1 Firmware upgrade using an external programmer	38
9.1.1 Tools required	38
9.1.2 Instructions to install the tools	38
9.1.3 Instructions to update the firmware.....	38
9.2 Firmware upgrade using the bootloader.....	39
9.2.1 Tools required	39
9.2.2 Activating the bootloader.....	39
9.2.1 Firmware update in Windows: FLIP.....	39
9.2.2 Firmware upgrade in Linux: AVRDUde.....	39
10. Frequently Asked Questions	40
11. Troubleshooting	41
12. XScope Design	42
12.1 System Architecture	42
12.2 Schematics	43



1. General Overview



1.1 Xprotolab & Xminilab Pin Description

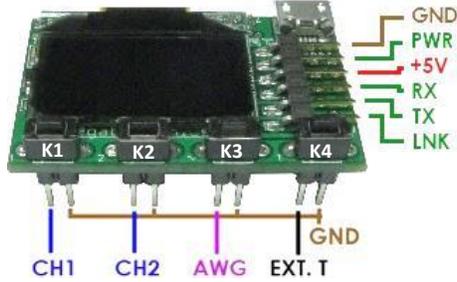


Figure 4: Front and Top Signals

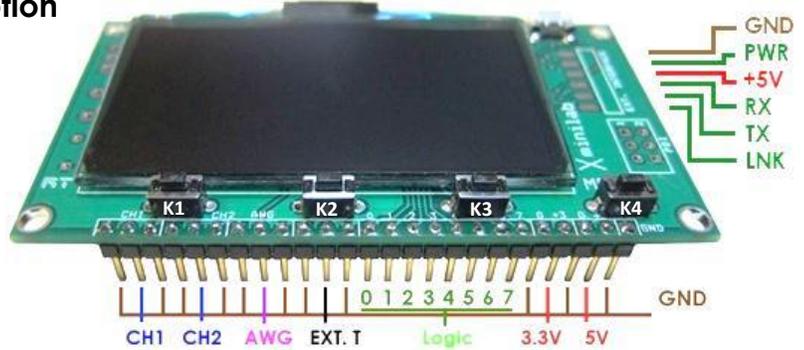


Figure 3: Xminilab HW 2.1 & 2.2 Front Signals

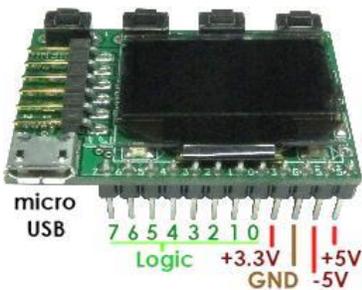


Figure 5: Back Signals

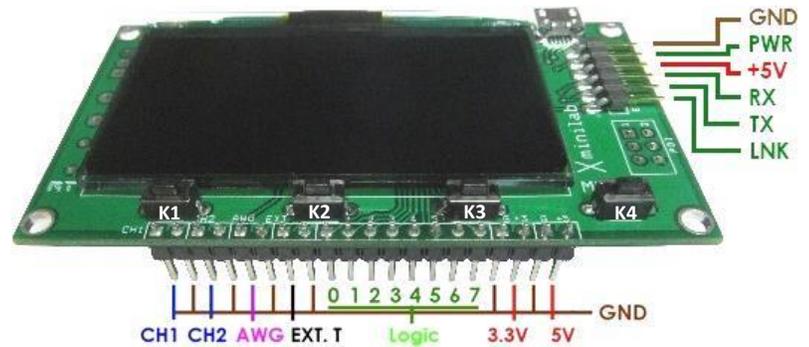


Figure 6: Xminilab HW 2.3 Front Signals

Name	Description	Comment
+5V	+5V Input voltage	Do not apply +5V if using the USB port
-5V	-5V Output voltage	50mA max output
GND	Ground	It is recommended use all ground pins to reduce voltage offset errors.
+3.3V	+3.3V Output voltage	200mA max output
Logic 0	Digital Channel 0	I2C Sniffer signal: SDA
Logic 1	Digital Channel 1	I2C Sniffer signal: SCL
Logic 2	Digital Channel 2	UART Sniffer signal: RX
Logic 3	Digital Channel 3	UART Sniffer signal: TX
Logic 4	Digital Channel 4	SPI Sniffer signal: /SS
Logic 5	Digital Channel 5	SPI Sniffer signal: MOSI
Logic 6	Digital Channel 6	SPI Sniffer signal: MISO
Logic 7	Digital Channel 7	SPI Sniffer signal: SCK
EXT. T	External Trigger	Digital input, max 5.5V
AWG	Arbitrary Waveform Generator	Output range: +/- 2V
CH2	Analog Channel 2	Input range: -14V to +20V
CH1	Analog Channel 1	Input range: -14V to +20V
PWR	Power up output signal	3.3V signal, 10mA max output
RX	Interface RX input	Connect to host's TX
TX	Interface TX output	Connect to host's RX
LNK	Interface link input	3.3V level input, with internal pull up

Table 1: Pin description

1.2 Xprotolab Portable and Xminilab Portable Overview

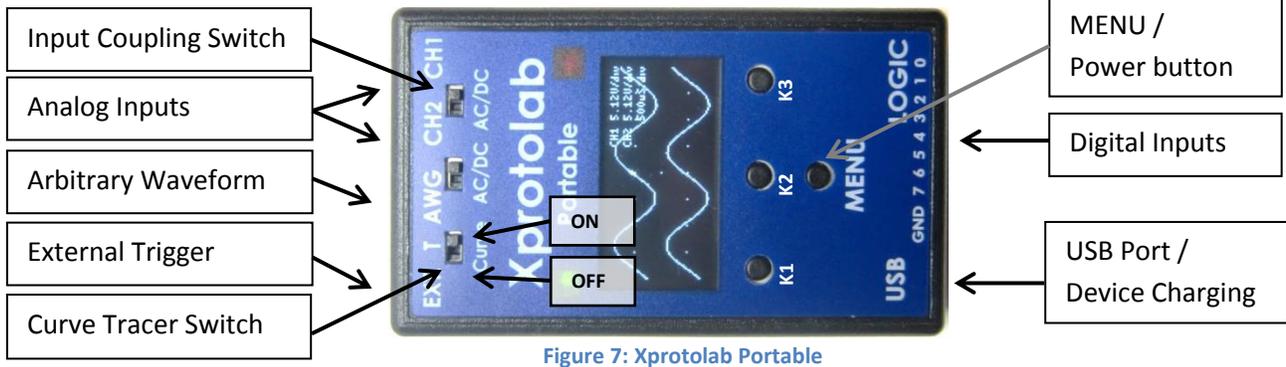


Figure 7: Xprotolab Portable

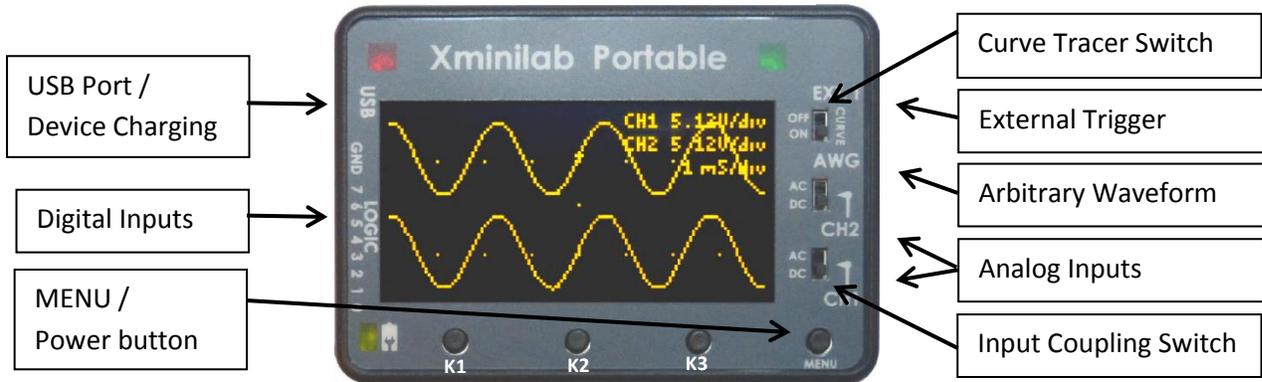


Figure 8: Xminilab Portable

1.2.1 Input Coupling Switch

The coupling switch is electrically placed between the input connector and the oscilloscope's input amplifier. The switch selects a direct path for DC or AC measurements, or a path through a capacitor, for AC only measurements.

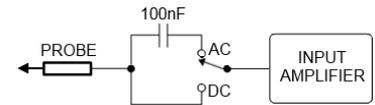


Figure 9: Input Coupling Switch

1.2.2 Curve Tracer Switch

This switch connects the AWG to the input channels, this is used in particular for creating V/I curve traces. An example of setting the device for curve tracing is shown in section 8.5.

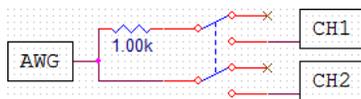


Figure 10: Curve Tracer Switch

Do not connect CH2 to a voltage source while the CURVE switch is on. Damage to the device will occur.

1.2.3 MENU / Power button

The device is powered on by pressing the MENU button. To power off, press the MENU button for 2 seconds. Some of the device modes disable this command, so to power off, set the device in Scope mode. The device is also powered off when the shutdown timer expires (regardless of the device mode).

1.2.4 USB Port / Device Charging

The USB port is also used to charge the device. The device can be charged either when the powered on or off. When the device is powered off, the battery can be fully charged in about 2 hours.



1.3 Specifications

		Xprotolab 	Xminilab 	Xprotolab Portable 	Xminilab Portable 
General	Microcontroller	ATXMEGA32A4U 32KB+4KB Flash, 4KB SRAM, 1KB EEPROM			
	Display Type	Graphic OLED, 128x64 pixels, max. refresh rate 122Hz			
	Display Size	0.96 inches	2.42 inches	1.3 inches	2.42 inches
	Display Life Time	10,000 hours min.	40,000 hours min.	10,000 hours min.	40,000 hours min.
	Device size	1.615" x 1.01"	3.3" x 1.75"	1.83" x 3.13" x 0.7"	3.17" x 2.22" x 0.7"
	Weight	8.6 grams	25 grams	60 grams	75 grams
	Interfaces	4 Tactile Switches, USB (Micro USB connector), UART, PDI for debugging			
	Battery	N/A		Li-Ion 3.7V 600mAh	Li-Ion 3.7V 750mAh
	Active current ¹	40mA to 60mA	45mA to 75mA	40mA to 60mA	45mA to 75mA
	Sleep current	3.6mA		1uA	
Logic Analyzer	Logic Inputs	8 Digital Inputs			
	Logic Input levels	3.3V only		3.3V, 5V tolerant	
	Input Pull	None, 24kΩ Pull Up, or 24kΩ Pull Down		820kΩ Pull Down	
	Max. Sampling rate	2Msps			
	Buffer Size	256			
	Frequency Counter	16MHz, 1Hz resolution, +/- 100ppm accuracy			
	Sniffer Protocols	UART, I2C, SPI			
Oscilloscope	Analog Inputs	2 Analog Inputs			
	Max. Sampling rate	2Msps			
	Analog Bandwidth	200kHz			
	Resolution	8 bits			
	Input Impedance	1MΩ			
	Buffer size	256 on each channel			
	Input Voltage Range	-14V to +20V			
	Vertical Sensitivity	80mV/div to 5.12V/div			
AWG	Analog Outputs	1 Analog Output			
	Max. Conversion rate	1Msps			
	Resolution	8bits			
	Buffer Size	256			
	Output current	> +/- 7mA			
	Output Voltage	+/- 2V		+/- 4V	
	Low Pass Filter	44.1kHz		53kHz	

Table 2: XScopes Specifications

Notes: 1. The Active current varies depending on the number of pixels lit on the display.



1.4 Dimensions

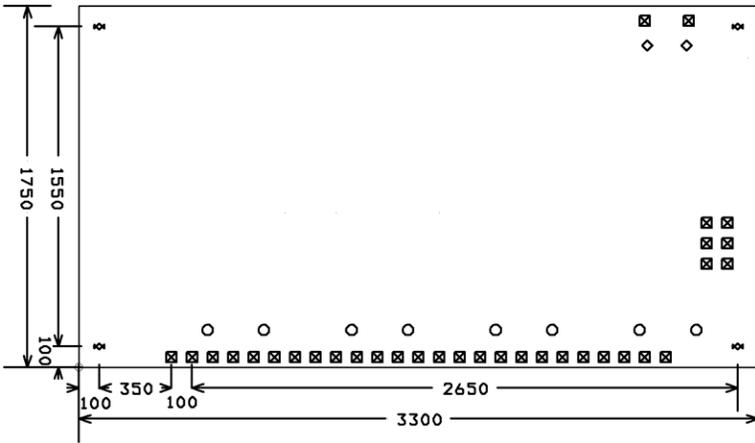


Figure 12: Xminilab 2.1 & 2.2 Dimensions

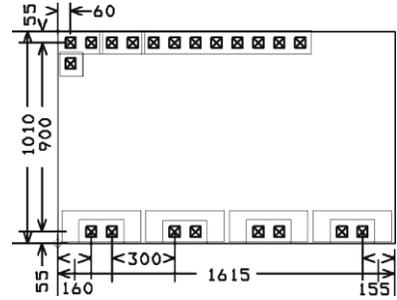


Figure 11: Xprotolab Dimensions

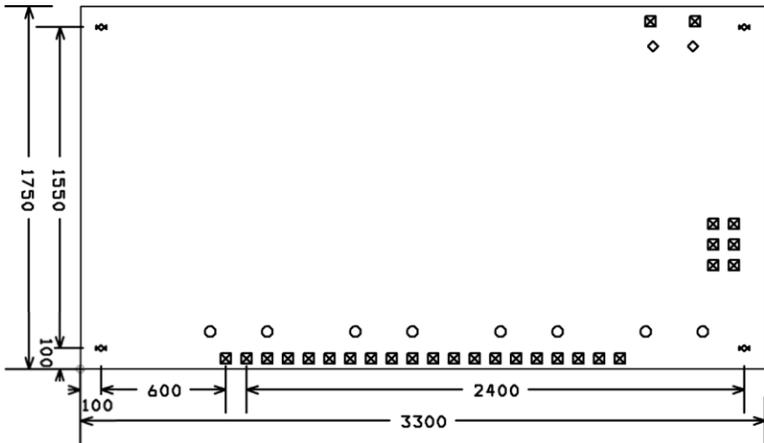


Figure 13: Xminilab 2.3 Dimensions

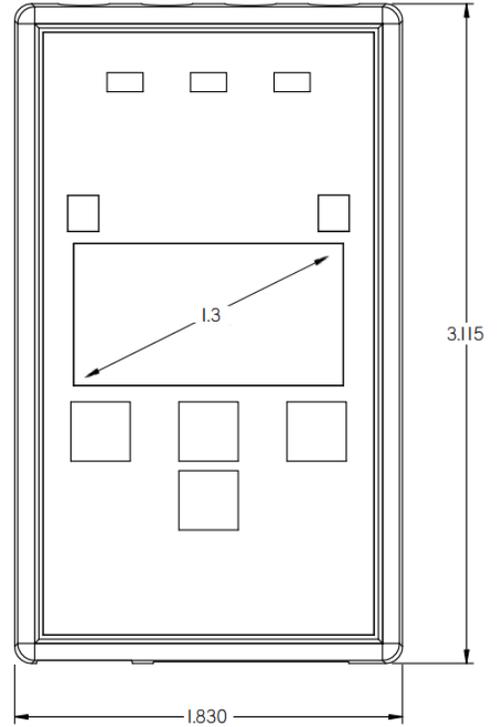


Figure 14: Xprotolab-Portable Dimensions

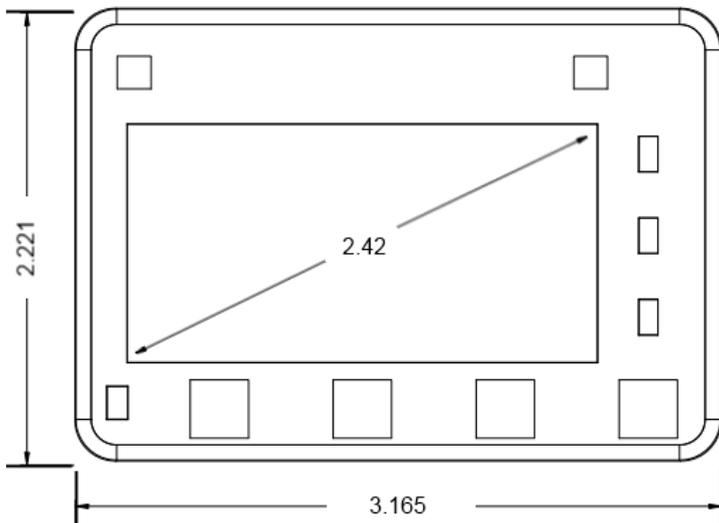


Figure 15: Xminilab-Portable Dimensions



1.5 Absolute Maximum Ratings

Parameter	Xprotolab & Xminilab		Portable Variants		Unit
	Minimum	Maximum	Minimum	Maximum	
Supply Voltage (+5V)	-0.5	5.5	N/A (Battery Powered)		V
Analog Inputs ¹	-30	30	-180	180	V
Digital Inputs	-0.5	3.8	-0.5	5.5	V
External Trigger	-3.5	6.8	-3.5	6.8	V
Operating Temperature	-40	70	-40	70	°C
Storage Temperature	-40	80	-40	80	°C

Table 3: Absolute Maximum Ratings

1. The maximum voltage on the analog inputs has only been tested to these limits. The device may tolerate higher voltages, but it is not recommended for safety.

1.6 Factory Setup

The device can enter factory options if the MENU key is pressed during power up. The following options are available:

- 1) **Offset calibration:** The unit is calibrated before being shipped, but calibration is required again if the firmware is updated. During calibration, two graphs are shown that represent the calibration on each channel.
- 2) **Shut off timeout:** Sets the time to shut down the device after the last key has been pressed. On the non-portable devices, it will only shut off the display and put the microcontroller to sleep.
- 3) **Restore defaults:** Select this function to restore to the default the settings. There are many settings on the device, if you are not familiar with them, this function is useful to set the device to a known state.

1.7 Quick Start Guide

- Take the device out of the packaging. There is a protective film on the display which can be removed.
- Power on the device. The non-portable devices can be powered with either the USB or with an external power supply, by applying +5V on the corresponding pin. Double check your connections because the device WILL get damaged if applying power on the wrong pin. The portable variants are powered with the MENU button.
- Connect the AWG pin to CH1. On the portable variants, you can flip the Curve switch instead.
- The tactile switches are named (from left to right) K1, K2, K3 and K4. The K4 is the Menu button.
- Press and hold the K1 key (auto setup). The screen should look like figure 16.
- Pressing K2 or K3 will change the sampling rate.

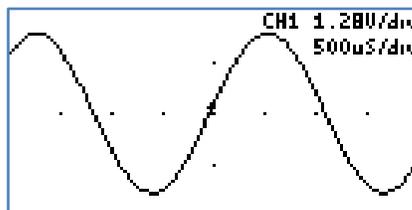


Figure 16: Quick start



Additional examples on how to use the device are presented in chapter 8.

1.8 User Interface

The K4 button is the MENU button, used to navigate thru all the menus. The K1 - K3 buttons action depend on the current menu. The green arrows represent the flow when pressing the MENU button. When the MENU button is pressed on the last menu, the device settings are saved and the menu goes back to the default. Figure 17 shows the main menus in blue and some secondary menus in yellow.

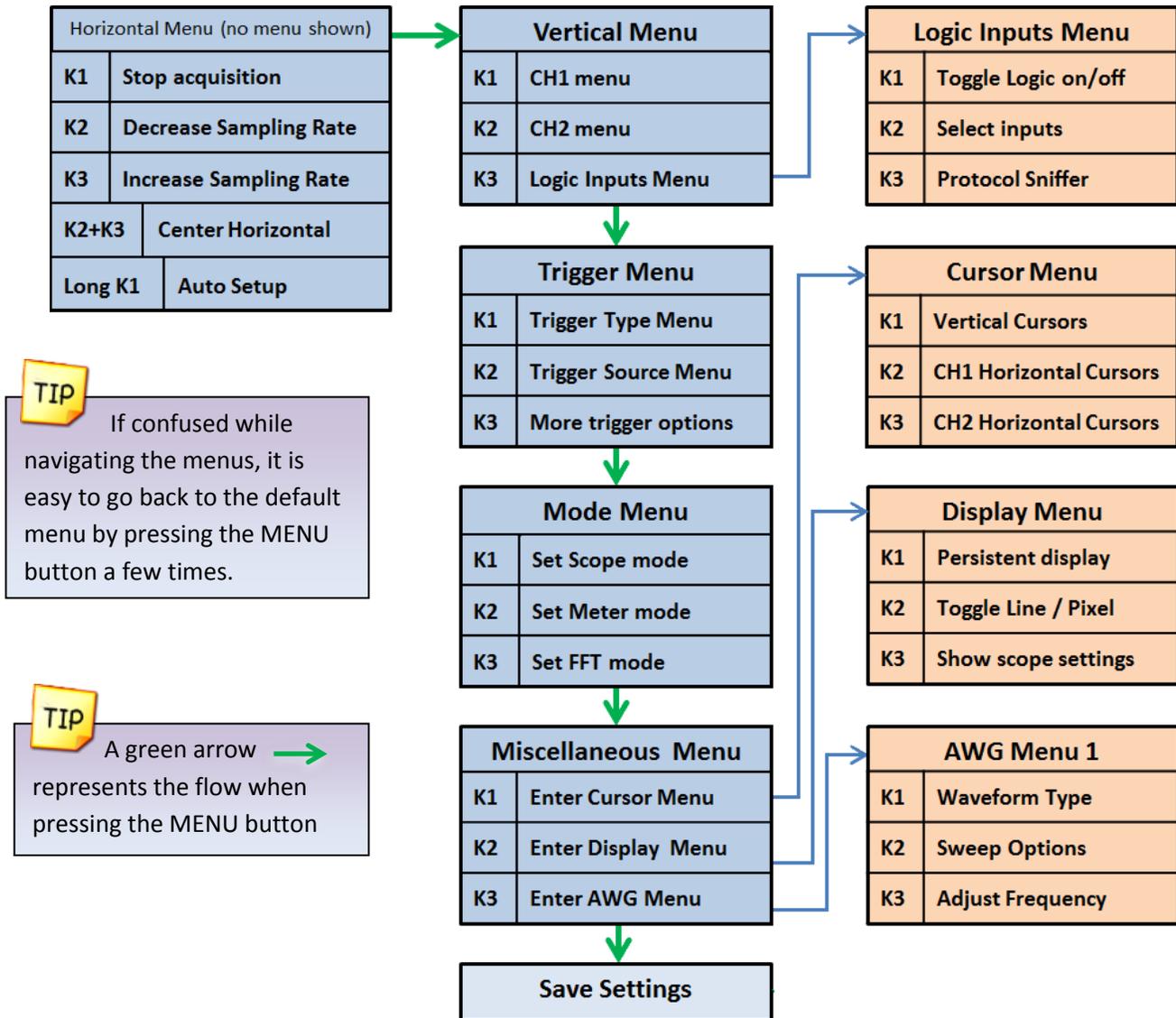


Figure 17: Main Menus

1.9 Saving the settings

All settings are stored to non-volatile memory only when exiting from the last menu. This method is used to reduce the number of write cycles to the microcontroller's EEPROM. The settings are not saved if the VCC voltage is under 3.15V.

2. Mixed Signal Oscilloscope

The XScope is a mixed signal oscilloscope; it has 2 analog channels and 8 digital channels. This chapter will focus on the analog signals. More information about the digital channels is presented in chapter 3.

2.1 Horizontal Settings

The horizontal settings are controlled on the default menu. The menu is shown on figure 18.

2.1.1 Time Base

The time base can be varied from 8µs/div to 50s/div. Table 4 shows all the possible time bases. One time division consists of 16 pixels. Example: 8µs / division = 8µs / 16 pixels → 500ns / pixel.

Horizontal Menu (no menu shown)	
K1	Stop acquisition
K2	Decrease Sampling Rate
K3	Increase Sampling Rate
K2+K3	Center Horizontal
Long K1	Auto Setup

Explore Wave	
K1	Continue acquisition
K2	Explore wave -
K3	Explore wave +

Figure 18: Horizontal Menus

Time Base (s / div)	Fast	*8µ	16µ	32µ	64µ	128µ	256µ	500µ	1m	2m	5m	10m
	Slow	20m	50m	0.1	0.2	0.5	1	2	5	10	20	50

Table 4: Time divisions *At 8µs/div, CH2 is not displayed.

2.1.2 Technical Details

There are two distinct sampling methods: Fast Sampling and Slow Sampling.

- **Fast Sampling** (10ms/div or faster): All samples are acquired to fill the buffer, and then they are displayed on the screen.
 - o Pre-trigger sampling (ability to show samples before the trigger) is available only with fast sampling.
 - o Only 128 samples are visible at a time, varying the horizontal position allows exploring the full buffer.
- **Slow Sampling** (20ms/div or slower): Single samples are acquired and simultaneously displayed on the display.
 - o The ROLL mode (waveform scrolls to the left during acquisition) is only available with the slow sampling.
 - o All 256 samples are visible on the display (each vertical line will have at least two samples)

2.1.3 Explore Wave

The horizontal position can be varied on the Fast Sampling time bases. There are 256 samples for each channel, but only 128 are displayed on the screen. When the acquisition is stopped, the full sample buffer can be explored with the K2 and K3 buttons. Pressing K2 and K3 simultaneously on the default menu will center the horizontal position.

When setting the XY mode, the graph can be moved vertically (Section 2.4.1.3).

2.1.4 Auto Setup

The Auto Setup feature will try to find the optimum gain and time base for the signals being applied on CH1 and CH2.



2.2 Vertical Settings

The analog channel controls are discussed in this section. Figure 19 shows the Vertical menu flow.

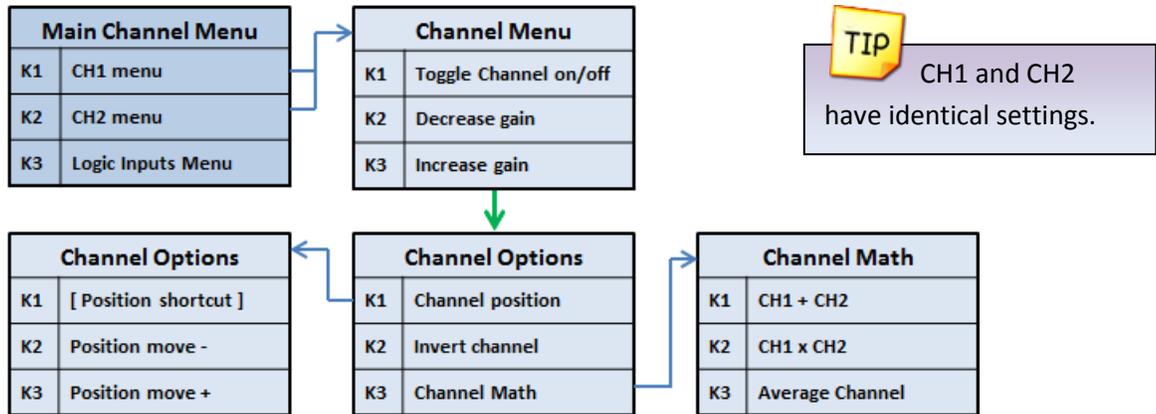


Figure 19: Vertical menus

2.2.1 Disable Channel

Any channel can be disabled; this is useful to reduce clutter on the display.

2.2.2 Channel Gain

Table 5 shows the possible gain settings for the analog channels. One gain division consists of 16 pixels. The current gain settings for the analog channels are shown in the top right part of the display (If the SHOW setting of the display is enabled).

Gain Settings (Volts / Division)
5.12
2.56
1.28
0.64
0.32
0.16
80m

Table 5: Gain Settings

2.2.3 Channel Position

The position of the waveform can be moved up or down in the Channel Position menu.

2.2.4 Channel Invert

The channel can be inverted. The displayed waveform and channel calculations will be affected.

2.2.5 Channel Math

- Subtract: The channel trace will be replaced with the difference.
- Multiply: The channel trace will be replaced with the product.
- Average: The channel samples will be averaged to reduce aliasing. (See Figure 20).

Channel Math Examples:

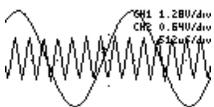


Figure 22: Two signals



Figure 21: CH1+CH2

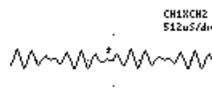


Figure 23: CH1xCH2

8 μ s/div:	1 sample (no average)
16 μ s/div:	1 sample (no average)
32 μ s/div to	
10 ms/div:	2 samples are averaged
20 ms/div:	1 sample (no average)
50 ms/div:	2 samples are averaged
100 ms/div:	4 samples are averaged
200 ms/div:	8 samples are averaged
500 ms/div:	20 samples are averaged
1 s/div:	40 samples are averaged
2 s/div:	80 samples are averaged
5 s/div:	200 samples are averaged
10 s/div:	400 samples are averaged
20 s/div:	800 samples are averaged
50 s/div:	2000 samples are averaged

Figure 20: Number of samples averaged when enabling the channel AVERAGE option. The device's sampling rate is normally faster than needed to be able to average samples



To display CH1+CH2, first invert CH2 and then select the SUBTRACT



2.3 Trigger Settings

The XScope has an advance triggering system, it has most of the trigger controls of a professional oscilloscope. Figure 24 shows the trigger menus.

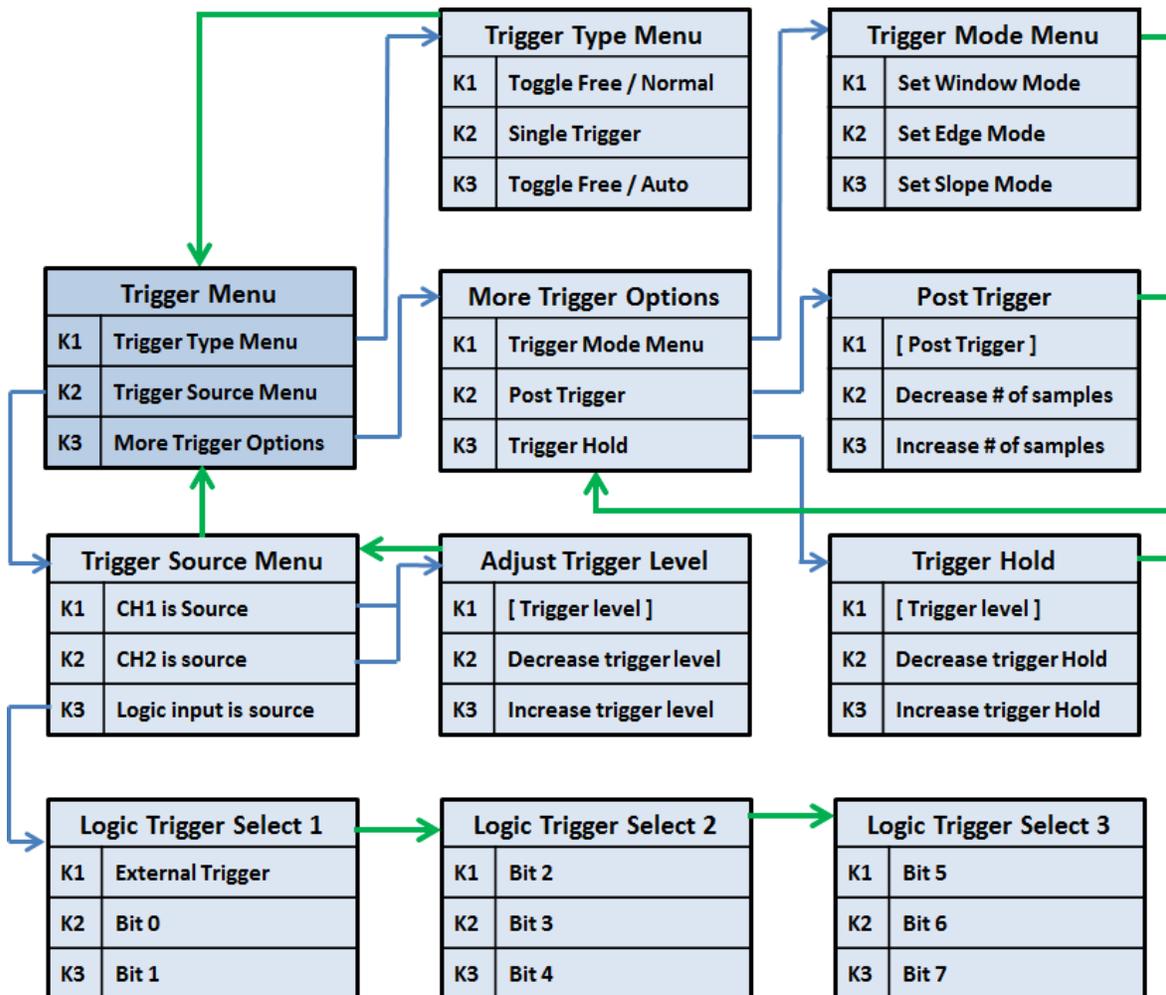


Figure 24: Trigger menus

2.3.1 Trigger Types

There are four different trigger types, which determine when to display the trace on the screen:

- **Normal:** Trace only when the trigger event occurs.
- **Single:** Only one trace is displayed when the trigger event occurs.
- **Auto:** Trace when the trigger event occurs, or after a timeout.
- **Free:** Trace continuously ignoring the trigger.

2.3.2 Trigger Modes

Three triggering modes are available: Edge, Window, and Slope. The Edge and Slope have selectable direction. When selecting an analog trigger source, the trigger direction is changed in the “Adjust Trigger Level” menu, by moving up or down the trigger level. When selecting a digital trigger source, the trigger direction is toggled on every button press.

- Edge Trigger:** The trigger occurs when the signal crosses the trigger level in a certain direction. The trigger level is represented on the display as a rising (↑), falling (↓) or dual arrow (↕).
 - Rising edge:** The trigger occurs when the signal crosses the level from below to above.
 - Falling Edge:** The trigger occurs when the signal crosses the level from above to below.
 - Dual Edge:** The trigger occurs when the signal crosses the trigger level in any direction. To select the Dual Edge mode, deselect Window, Edge, and Slope in the “Trigger Mode Menu”, the trigger mark will change to a dual arrow: ↕

TIP
Edge Trigger:
 The signal crosses a level.

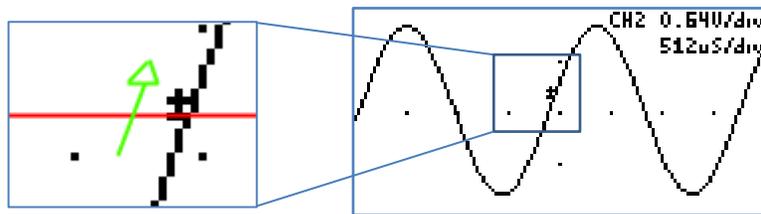


Figure 25: Edge Trigger

- Window Trigger:** The trigger occurs when the signal leaves a voltage range. This mode is useful for detecting overvoltages or undervoltages. Two arrow trigger marks represent the window levels.

TIP
Window Trigger:
 The signal is outside a range.

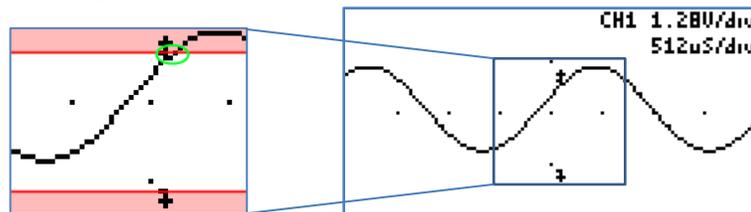


Figure 26: Window Trigger

- Slope Trigger:** The trigger occurs when the difference between two consecutive samples is greater or lower than a predefined value. This is useful for detecting spikes or for detecting high frequency signals. The trigger mark is represented on the screen as two small lines, with a separation proportional to the trigger value.

TIP
Slope Trigger:
 The difference of two points in the signal is above a value.

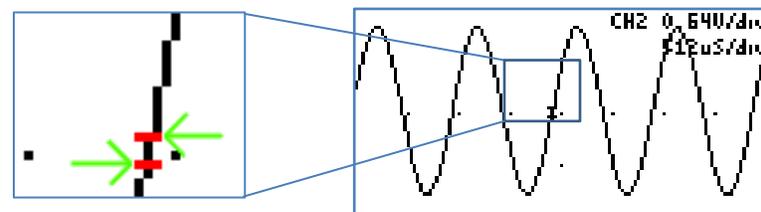


Figure 27: Slope trigger

2.3.3 Trigger Hold

The trigger hold specifies a time to wait before detecting the next trigger. It is useful when the signal can have multiple trigger events occurring close to each other, but you only want to trigger on the first one.

2.3.4 Post Trigger

The oscilloscope is continuously acquiring samples in a circular buffer. Once the trigger event occurs, the oscilloscope will acquire more samples, specified by the Post Trigger value. The ability to show samples before or after the trigger is one of the most powerful features of a digital sampling oscilloscope. The post trigger is only available on the fast sampling rates.

Depending on the post trigger settings, different parts of a signal can be displayed. Consider the signal on figure 28:



Figure 28: Sample signal

Even though the buffer sample is relatively small, any section of the shown figure can be analyzed by varying the post trigger value. Examples:

- Post trigger = 0 (don't acquire more signals after the trigger).
Only the signals that occurred before the trigger event are shown.

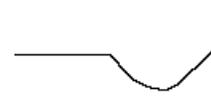


Figure 29: Post trigger value equal zero

- Post trigger = 50% of the sample buffer (default setting).
Half of the buffer contains samples before the trigger, and half contains the samples after the trigger.

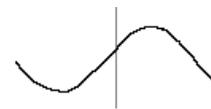


Figure 30: Post trigger = 50% of sample buffer

- Post trigger = 100% of the sample buffer
Only signals immediately after the trigger event are shown.



Figure 31: Post trigger = 100% of buffer

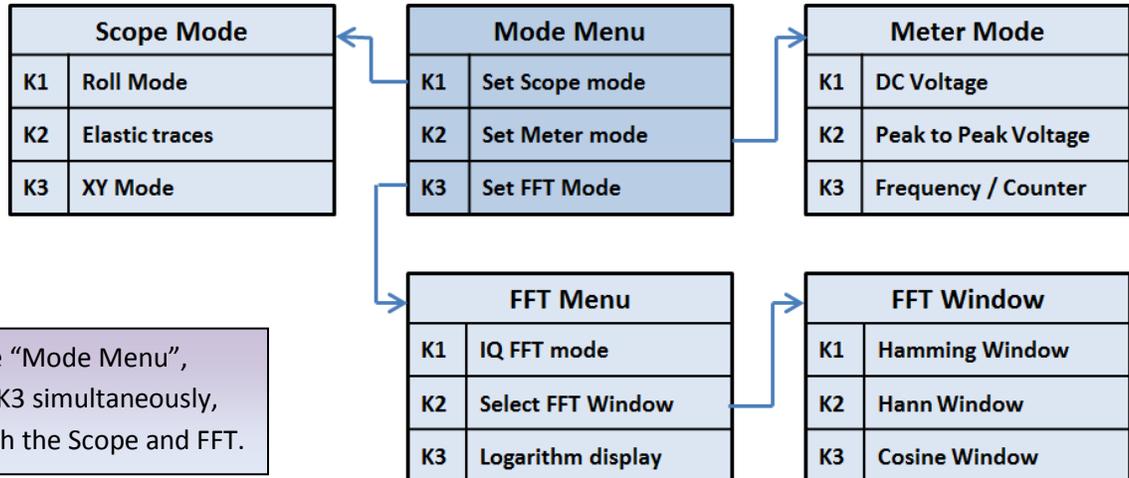
The actual post trigger value can vary between 0 and 32768 samples, so you can explore the signal after a very long time after the trigger event has occurred, but with a high post trigger value, the refresh rate of the scope will be reduced.

2.3.5 Trigger Source

Any analog or digital channel can be the trigger source. If selecting a digital channel as trigger source, the slope and window modes are not applicable; the device will use edge triggering. The external trigger input is an additional digital trigger source which tolerates voltages up to 5.5V.

2.4 Device Modes

There are multiple device modes that can be selected; the menus shown on figure 32 allow selecting the Scope Mode, the Meter Mode or the Spectrum Analyzer Mode (FFT). Another device mode is the Protocol Sniffer, which is discussed in section 3.8.



TIP In the “Mode Menu”, press K1 and K3 simultaneously, to display both the Scope and FFT.

Figure 32: Device mode menus

2.4.1 Oscilloscope Mode

This is the default mode of the XScope. The 2 analog and 8 digital channels are sampled simultaneously. Any of these 10 channels can be shown on the display. Figure 33 shows the oscilloscope mode and the various sections of the display are detailed.

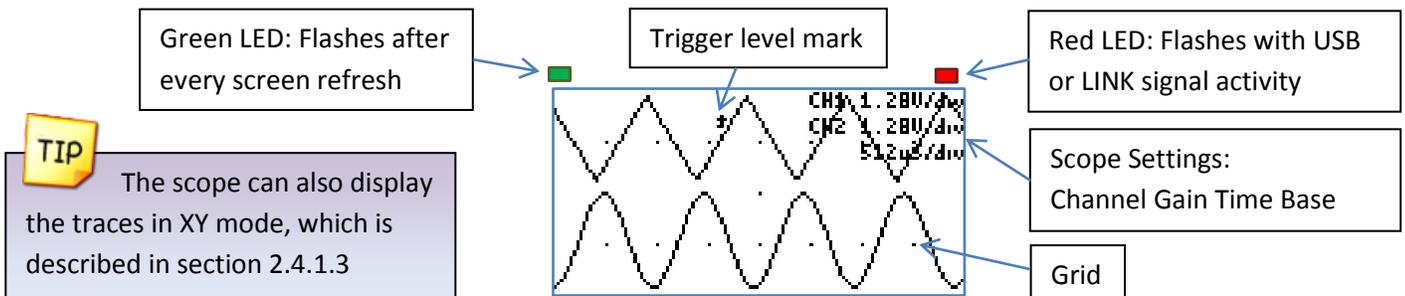


Figure 33: Oscilloscope Mode

2.4.1.1 Roll Mode

The data on the display is scrolled to the left as new data comes in. This is only available on the Slow Sampling rates. The Roll mode and Elastic mode cannot be selected simultaneously. The Roll mode disables the triggering.

2.4.1.2 Elastic Traces

This is also called “Display average” on other digital oscilloscopes. It works by averaging the trace data with the new data. The result is a more stable waveform displayed on the screen. However, using this setting only makes sense when the scope is properly triggered on a periodic signal. The Elastic trace computes this equation for every point in the trace:

$$NewTrace = \frac{OldTrace + NewData}{2}$$



2.4.1.3 XY Mode

The XY mode changes the display from volts vs. time, to volts vs. volts. You can use XY mode to compare frequency and phase relationships between two signals. The XY mode can also be used with transducers to display strain versus displacement, flow versus pressure, volts versus current, or voltage versus frequency. Lissajous figures can be plotted using the XY Mode. Component V/I curves can also be plotted, see section 8.5.



When using the XY modes with a Slow Sampling rate, activating the ROLL mode will display a continuous “beam”.



You can use the “Explore Wave” menu to move the graph vertically.

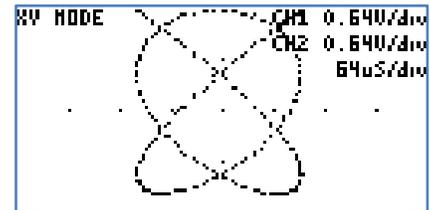


Figure 34: XY Mode

2.4.2 Meter Mode

The XScope can function as a dual digital voltmeter. The font used is bigger in meter mode to facilitate reading. The available measurements in meter mode are: Average Voltage (DC), Peak to Peak Voltage, Frequency, and Pulse counting. A small trace of the analog signals is displayed below the voltage measurements.

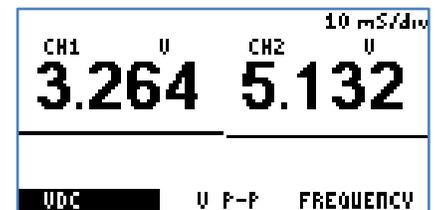


Figure 35: Meter Mode

2.4.2.1 Frequency Measurements

The device can measure frequencies on any channel (analog or digital). The measurements on the analog channels are always shown, and the trigger source (see section 2.3) selects which digital channel to measure.

Frequency measurements on the analog channels are done using the FFT of the acquired data, so measured frequencies have discrete steps. The frequency range is determined by the highest frequency of the analog channels. If there is a high frequency on one channel and a low frequency on the other, the channel with the lowest frequency will have low resolution. Frequency measurements with the FFT are best suited for analog signals. Frequency measurements on the digital inputs are done counting the pulses on the pin over one second. The resolution of the measurement is 1Hz. Frequency measurements with the Frequency Counter are best suited for digital signals.

	FFT (Analog channels)	Frequency Counter (Digital channels)
Maximum voltage range	-14V to 20V	Logic level range (or Ext. Trig -2.2V to 5.5V)
Maximum Frequency	500kHz	16MHz (or 12MHz on the Ext. Trig)
Resolution	Variable, depending on frequency range. From 6.25Hz to 7.812kHz	1Hz
Signal is noisy, or is mixed with other signals	Finds the fundamental frequency	Not suitable
Signal has a high offset	Still works	Stops working when the offset is above the logic threshold.

Table 6: FFT vs. Frequency Counter

2.4.2.2 Pulse Counter

The device can count the number of pulses on a digital input. The counter will roll over after reaching 99,942,399. The counting can be stopped by stopping the acquisition (K1 button on the main menu). Pressing any button will reset the counter. A stopwatch displays the time since the Pulse Counter started, and can reach 255 hours, 59 minutes, 59 seconds.



2.4.3 Spectrum Analyzer

The spectrum analyzer is done by calculating the Fast Fourier Transform (FFT) of the selected analog channels (or the channel math functions if enabled). When the FFT is enabled, the spectrum is plotted as frequency vs. magnitude. The horizontal axis represents the frequency (Hertz), and the vertical axis represents the magnitude. Figure 36 shows the XScope in Spectrum Analyzer Mode. The Nyquist frequency is shown on the top right corner of the display.

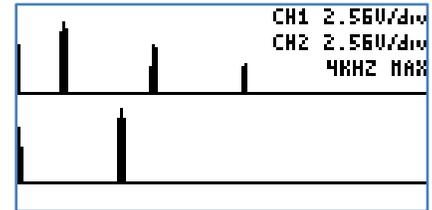


Figure 36: Spectrum Analyzer Mode



If only interested in one channel, turn off the other channel to maximize the vertical display.

2.4.3.1 IQ FFT Mode

When the IQ FFT is disabled, the XScope calculates two independent 256 point FFTs of the analog channels, the Real and Imaginary components of the FFT have the same data. The output of the FFT is symmetrical, but only half of the result is shown on the display. When the IQ FFT is enabled, only one FFT is calculated, the Real component is filled with the CH1 data, and the Imaginary component is filled with the CH2 data. The result is a 256 point FFT, you can use the horizontal controls described in section 2.1.3 to explore all the data (since only 128 points can be shown on the display). The IQ FFT is useful to monitor RF Spectrums with the proper hardware mixer.

2.4.3.2 Logarithm display

The log is useful when analyzing low level components on the signal. When analyzing audio, it is also very useful as it maps more directly to how humans perceive sound. The actual function performed is: $y = 16 * \log_2(x)$.

Example:



Figure 37: Triangle Wave



Figure 39: FFT without Log



Figure 38: FFT with Log

2.4.3.3 FFT Windows

To reduce the spectral leakage, an FFT window function can be applied. Four FFT window types are available:

- **Rectangular:** No window applied
- **Hamming:** $0.53836 - 0.46164 * \cos\left(\frac{2*\pi*n}{FFT_N-1}\right)$
- **Hann:** $0.5 * \left(1 - \cos\left(\frac{2*\pi*n}{FFT_N-1}\right)\right)$
- **Blackman:** $0.42 - 0.5 * \cos\left(\frac{2*\pi*n}{FFT_N-1}\right) + 0.08 * \cos\left(\frac{4*\pi*n}{FFT_N-1}\right)$

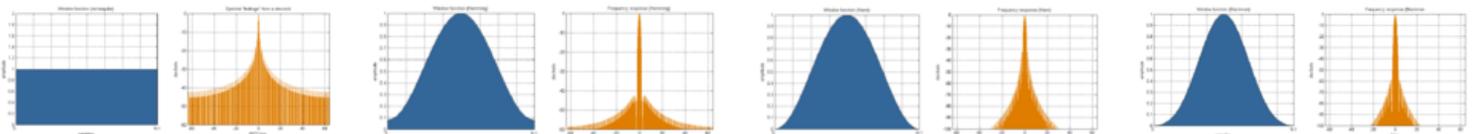


Figure 40: Window and sine frequency response, from left to right: Rectangular, Hamming, Hann and Blackman



2.5 Cursors

You can measure waveform data using cursors. Cursors are horizontal and vertical markers that indicate X-axis values (usually time) and Y-axis values (usually voltage) on a selected waveform source. The position of the cursors can be moved on the respective menu. Figure 41 shows the cursor menus.

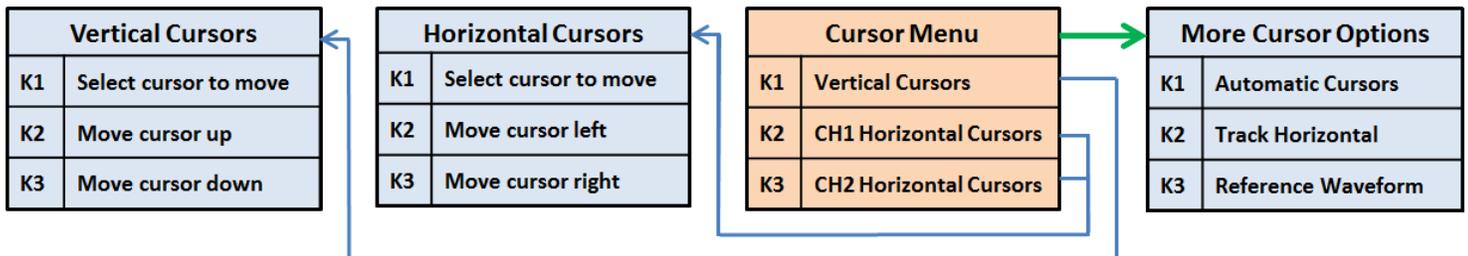


Figure 41: Cursor menus

2.5.1 Vertical Cursors

Time interval measurements are made with a pair of time markers. The oscilloscope automatically calculates the time difference between the two markers and displays the difference as a delta time. Additionally, the oscilloscope calculates the inverse of the delta time, which is the frequency of the selected period.

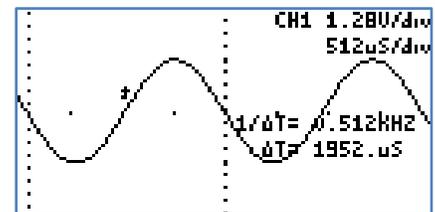


Figure 42: Vertical Cursors

2.5.2 Horizontal Cursors

Voltage measurements are made with a pair of voltage markers to determine 1 or 2 specific voltage points on a waveform. The oscilloscope automatically calculates the voltage difference between the two markers and displays the difference as a delta voltage value.

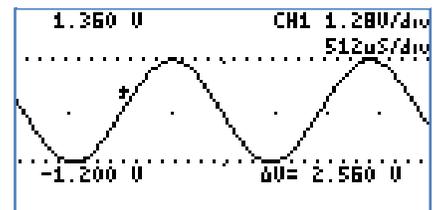


Figure 43: Horizontal Cursors

2.5.3 Automatic Cursors

When the automatic cursors are enabled, the device will try to automatically make measurements on the waveform.

- **Vertical Cursors:** The device will try find a full or half cycle of the selected waveform. If both CH1 and CH2 are enabled, the channel with the most amplitude will be used.
- **Horizontal Cursor:** The selected horizontal cursor will be set with the maximum and minimum points of the waveform.

2.5.4 Track Horizontal Cursors

When Track is enabled, the location of the horizontal cursor will track the signal located on the vertical cursor.

2.5.5 Reference Waveform

A snapshot is taken of the analog waveforms to be used as reference waveforms (the captured waveforms stay on the screen). The reference waveforms are stored in non-volatile memory.

2.5.6 Cursors in XY Mode

When the XY mode is active, the vertical cursors are disabled, and the pair of horizontal cursors represent the X and Y position.

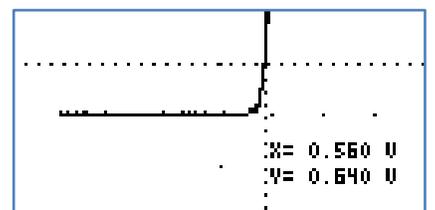


Figure 44: Cursors in XY Mode



2.6 Display Settings

These menus control various characteristics of the display. Figure 45 shows the display menus.



Figure 45: Display menus

2.6.1 Persistent Display

When the persistent display is enabled, the waveform traces are not erased. The persistent display is useful as a simple data logger or to catch glitches in the waveform. The persistent mode can also be used to make frequency plots in combination with the AWG frequency sweep.

2.6.2 Line / Pixel Display

This menu item selects the drawing method.

- **Line:** A line is drawn from one sample to the next.
- **Pixel:** A single pixel represents a sample. The pixel display is useful at slow sampling rates or when used in combination with the persistent mode. Figure 46 shows the pixel display.

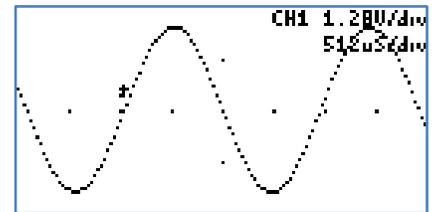


Figure 46: Pixel Display

2.6.3 Show scope settings

Toggles the display of the scope settings (Channel gain and time base).

2.6.4 Grid Type

There are 4 different grid types:

- **No grid.**
- **Dots for each division:** Vertical dots represent the scale divisions. Horizontal dots represent the time base setting and the ground level of each channel.
- **Vertical grid line follow trigger:** Vertical dots represent the position of the trigger, the location of the vertical dots follow the trigger position. Horizontal dots represent the time base setting and the ground level of each channel.
- **Dot graticule:** The screen is filled with dots that represent the vertical and horizontal divisions.

2.6.5 Flip Display

The display orientation is flipped. This is useful when mounting the XScope on a panel, and the display's orientations is backwards.

2.6.6 Invert Display

When enabled, the display's pixels are inverted (the display will have a white background).

3. Logic Analyzer and Protocol Sniffer



The XScope has an 8 bit logic analyzer and can do sniffing on standard protocols: I2C, UART and SPI. The logic inputs are 3.3V level, only the Portable devices are 5V tolerant. If you need to connect 5V signals to the logic analyzer, you could add a 3K resistor in series with the signal, or use a 5V to 3.3V level converter chip. Figure 47 shows the logic menus.

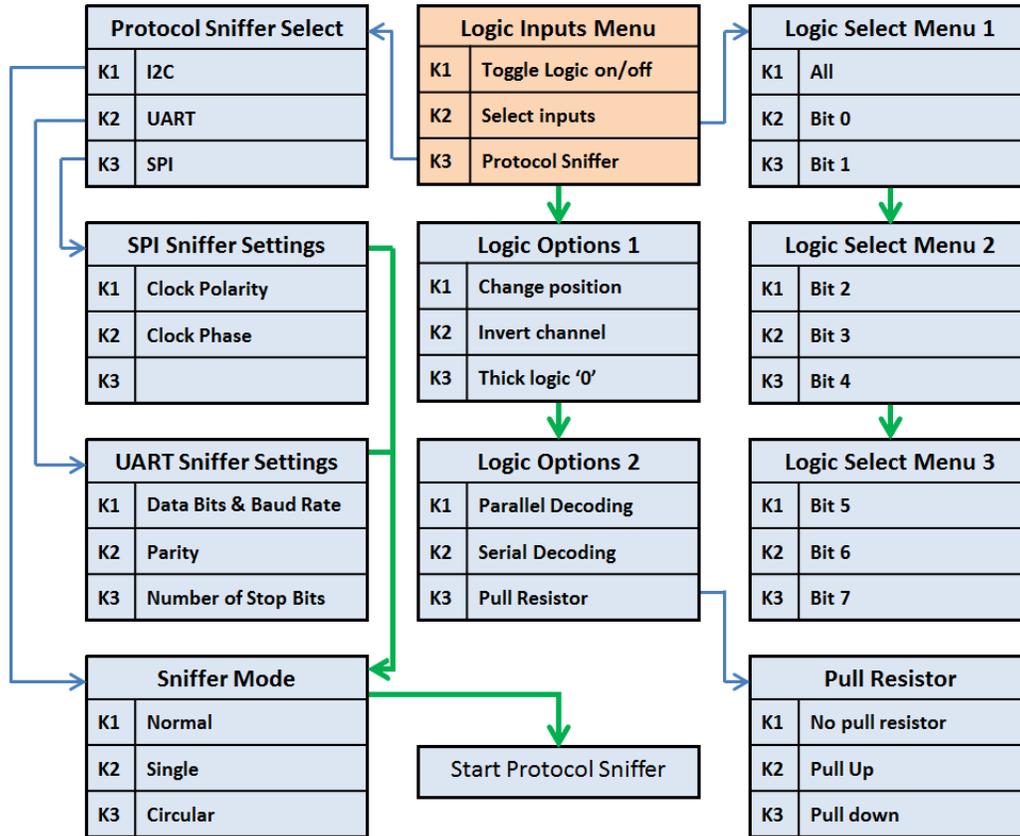


Figure 47: Logic Analyzer Menus

3.1 Input Selection

A subset of the 8 digital signals can be selected. Any digital signal can be enabled or disabled.

3.2 Channel Position

The selected digital channels can be moved up or down. Only applicable if less than 8 digital signals are selected.

3.3 Invert Channel

All digital channels are inverted. This setting also affects the protocol sniffer!

3.4 Thick Logic '0'

A thick line is drawn when the signal is at logic '0'. This is useful to quickly differentiate a '0' from a '1'.

3.5 Parallel Decoding

Shows the hexadecimal value of the 8 bit digital input lines. The hexadecimal number is shown below the last digital trace. If all the 8 digital traces are enabled, then there is no space to show the parallel decoding. Figure 48 shows an example of the parallel decoding with 4 logic lines enabled.

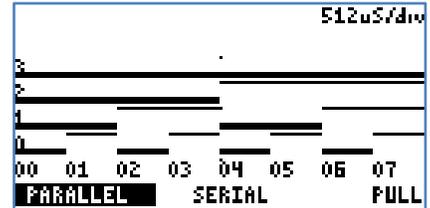


Figure 48: Parallel Decoding

3.6 Serial Decoding

Shows the hexadecimal value of the stream of bits on each channel. The decoding starts at the first vertical cursor and ends at the second vertical cursor, 8 bits are decoded. If the cursors are disabled, then the decoding is done from the start of the screen, to the end. The data can be decoded MSB first or LSB first, depending on the position of the first vertical cursor.

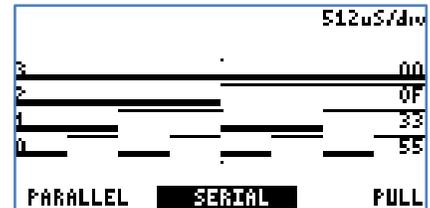


Figure 49: Serial Decoding

3.7 Protocol Sniffer

When the XScope is in Sniffer mode and before any data is received, a brief text appears on the screen to indicate where to hook up the signals. As soon as data is received, the data is displayed in "pages". There are 16 pages of data. To browse thru the pages, use the buttons K2 and K3. To stop and start the sniffer, press the K1 button. Figure 50 shows the device in sniffer mode.

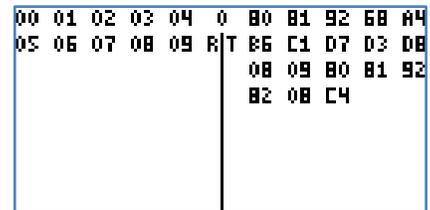


Figure 50: Sniffer

In the UART and SPI sniffers, the data can be displayed in HEX or ASCII, press K2 and K3 simultaneously to toggle between them. If using ASCII, only codes 0x20 thru 0x7A will show valid characters. Figure 51 shows the 3x6 font.

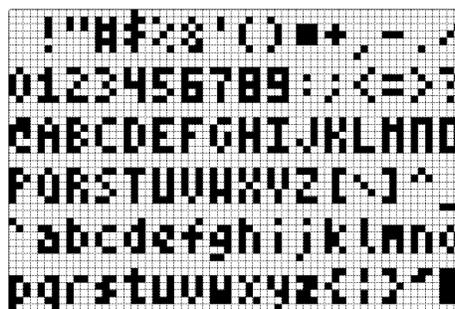


Figure 51: Small 3x6 font

3.8 Sniffers Modes

- **Normal** mode: Continuous operation, when the buffer is filled, all pages are erased, the index goes back to page 1
- **Single** mode: The sniffer will stop when the buffer is filled.
- **Circular** mode: New data will be placed at the end of the last page, older data will be shifted towards the first page. At the beginning, the device will show 0x00 on all pages, and the last page will be set. The circular mode is useful if you are only interested in the most recent data received.

3.9 I2C Sniffer

Connect SDA to Bit 0, SCL to Bit 1

The XScope implements the I2C sniffing in a bit-bang fashion. The maximum tested clock frequency is 400kHz (Standard I2C Fast Speed). As the data is decoded, the data in HEX will appear on the screen, accompanied by a symbol:

When the Master initiates a read, < is an ACK and (is a NACK

When the Master initiates a write, > is an ACK and) is a NACK

Subsequent data in the frame will be accompanied by + for ACK or a - for NACK.

There are 16 pages of data, each page shows 64 bytes => the total memory for the I2C sniffer is 1024 bytes.

Example communicating to a Si570 Programmable oscillator:

```
55> 07+ (Master initiates Write to slave 55, byte address 7)
55< 05+ 42+ B6+ 04+ 79+ 9A- (Master initiates Read to slave 55, then reads 6 bytes)
```

3.10 UART Sniffer

Connect RX to Bit 2, TX to Bit 3

The XScope can decode both the TX and RX lines of the UART at the standard baud rates, and with selectable data bits:

```
5,6,7,8 Data bits / 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200 bps
```

When the sniffer begins, the screen is split in two, the left side is used for the RX line, and the right side is used for the TX line. Each side can show 40 bytes per page. With 16 pages, a total of 640 bytes can be stored for each decoded line.

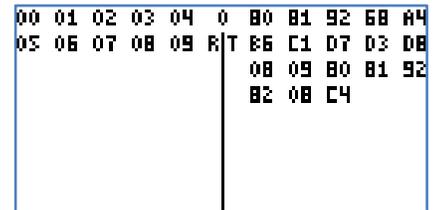


Figure 52: UART Sniffer screen

3.11 SPI Sniffer

Connect the Slave Select to Bit 4, MOSI to Bit 5, MISO to Bit 6, SCK to Bit 7

The XScope can decode both the MOSI and MISO lines of an SPI bus. The SPI's MOSI pin decoding is done in hardware, so it can decode data at high speed (up to 8MHz), but the SPI's MISO pin decoding is implemented in software using bit-banging, the maximum clock is 439kHz. Decoding starts when a falling edge on the SS pin is detected.

Configuration	Leading edge	Trailing edge
CPOL CPHA	Rising, sample	Falling, setup
CPOL CPHA	Rising, setup	Falling, sample
CPOL CPHA	Falling, sample	Rising, setup
CPOL CPHA	Falling, setup	Rising, sample

Table 7: SPI Configuration

The screen is split in two, the left side is used for the MOSI line, and the right side is used for the MISO line. Each side can show 40 bytes per page. With 16 pages, a total of 640 bytes can be stored for each decoded line. Table 7 shows the SPI configuration.



4. Arbitrary Waveform Generator



The XScope has an embedded arbitrary waveform generator. The waveform generator output is independent from the data acquisition and is always running in the background. You can adjust all the parameters of the waveform: frequency, amplitude, offset and duty cycle. You can sweep the frequency, amplitude and duty cycle. Figure 53 shows the AWG Menus.

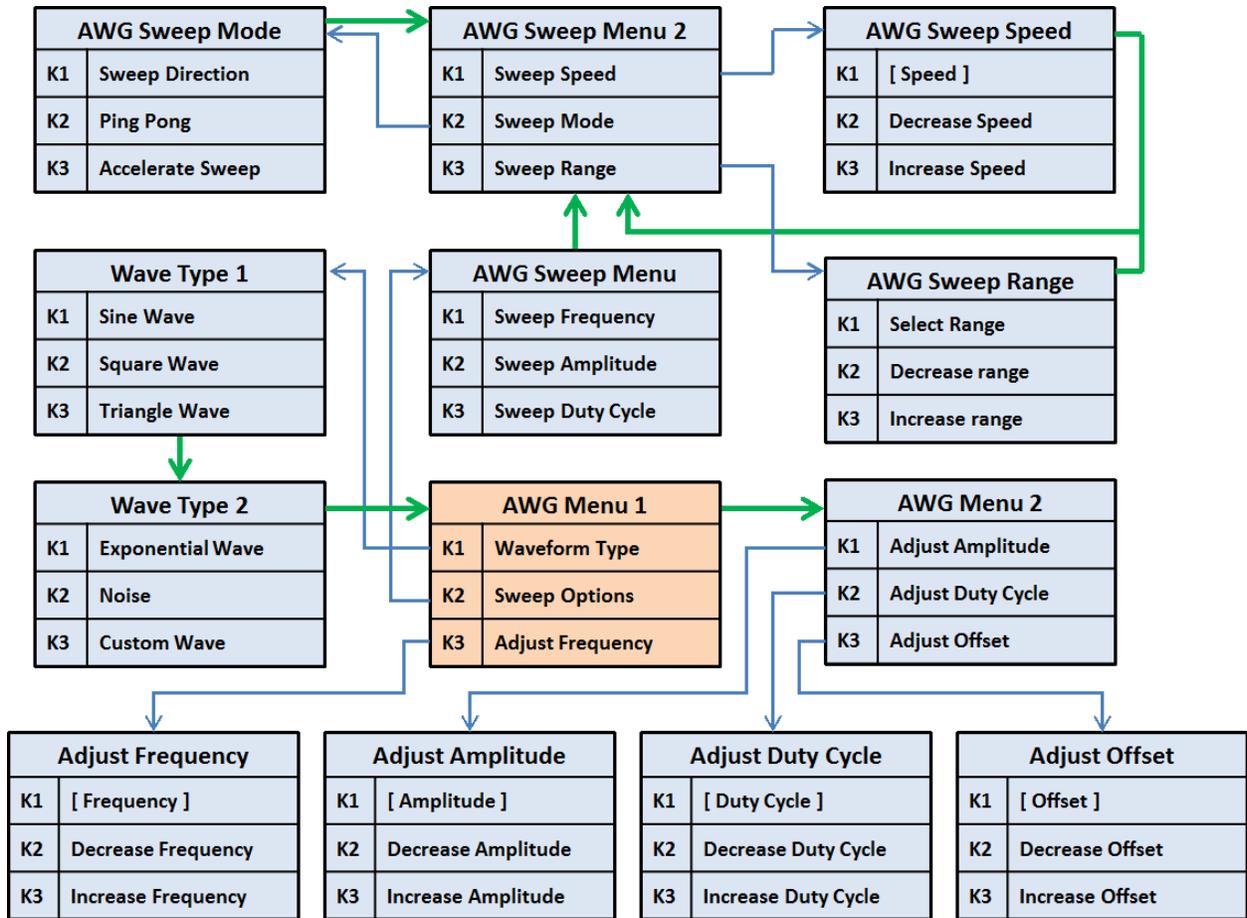


Figure 53: AWG Menus



When adjusting the parameters, the K1 button serves as a shortcut key, which sets predefined values.



When enabling the Sweep, the waveform will be updated only on a screen refresh. For a smooth sweep, set the scope with a high speed sampling, or stop the oscilloscope.

4.1 Predefined Waveforms

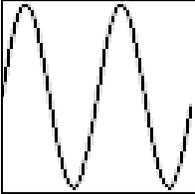
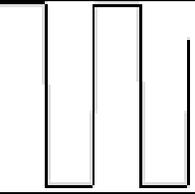
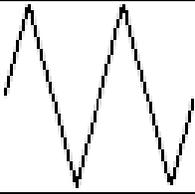
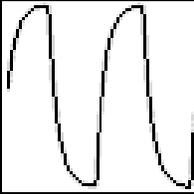
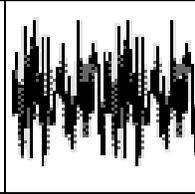
					
Sine Wave	Square Wave	Triangle Wave	Exponential	Periodic Noise	Custom Wave *

Table 8: AWG Preprogrammed Waveforms

The XScope can output the following waveforms: Sine, Square, Triangle and Exponential. There is a “Periodic Noise” option that fills the AWG buffer with random data, it is periodic because the same data is output over and over, but each time the Noise wave is selected, new random data will be generated. There is also a custom waveform which is initially set with an ECG wave, but can be changed with the PC interface (see section 5.2).

4.2 Parameter Sweep

The XScope has a SWEEP feature, which increases one or more parameter values automatically **on each screen refresh of the oscilloscope**. When the sweep is enabled, three dots will appear at the bottom of the screen, representing the start, end, and current sweep value. When doing a Frequency sweep, the frequency range is determined by the current time base. Since the frequency sweep is synchronized with the oscilloscope, displaying perfect frequency plots is easy. To make a frequency plot, set the mode to FFT, and set the display to persistent. See section 8.6 for an example.

4.2.1 Sweep Modes

In the Sweep Mode menu, the sweep direction can be changed. Automatic change of the direction is done by enabling the Ping Pong mode. The sweep acceleration increases or decreases the sweep speed, the sweep speed is reset when reaching the start or end of the sweep.

4.3 Technical Details

The waveform is stored in a 256 byte long buffer, this buffer is fed to the XMEGA's DAC thru the DMA. Once the waveform is set, the waveform will be generated without any CPU intervention. The maximum conversion rate of the DAC is 1Msps, this limits the maximum output frequency of the AWG as a system. For example, if the AWG is generating a sinewave with 256 points, the maximum frequency is 3906.25Hz. If generating a sinewave with only 32 points, the maximum frequency is 31.25KHz. The AWG amplifier has a low pass filter of 44.1KHz. The predefined AWG Frequency range is: 1Hz thru 125 kHz

The resolution of the waveform generator varies depending on the frequency range: the lower the frequency, the higher the resolution. Note that the possible frequencies are discrete:

$$Frequency = Cycles * \frac{125000}{Period}$$

Cycles: Integer number, with these possible values: 1, 2, 4, 8, 16, 32

Period: Integer number, with values between 32 and 65535

5. PC Interface

The device can interface with a PC (Linux, Mac, or Windows). The device needs to be updated with the latest firmware. The source code for the PC interface is open source under the [GPLv3](#) license and is available on GitHub.

5.1 Connecting the device

The XScope can communicate to a PC thru the USB port. When the application starts, it will try to connect automatically. The USB connection is not a virtual serial port, but there is also an option to connect thru a serial port. To establish a serial port connection you can use a UART adapter or the Bluetooth module on the device's external port (see section 1.1).

Data Bits: 8
Baud rate: 115200
Parity: None
Stop Bits: One
Handshaking: None

Table 9: Serial settings

5.1 Controlling the device

All of the functions described previously in this manual can be controlled with the PC interface. The main operating modes of the device can be selected by switching to the corresponding tab of the interface. Figure 54 shows a snapshot of the device in the Oscilloscope mode.



Figure 54: PC Interface



5.2 Custom AWG Waveform

You can define your own waveforms for the AWG. On the Waveform Generator tab, click “Open CSV” and select your CSV file, then click “Save as Custom”, the waveform will be permanently saved in the device’s EEPROM memory. When creating a CSV file, the range of the data must be [-127, 127], and there should be only 256 numbers.

5.3 Saving Waveforms and Screenshots

On the Options tab there are buttons to save and load waveforms, as well as saving screenshots of the graph area in PNG format. When saving a waveform or screenshot, the program will ask for a destination folder only once, and then use this folder for subsequent files. When opening a waveform, the “Clear” button must be pressed to continue normal operation.



Figure 55: File operations

The Record button allows saving waveforms continuously. The Play button is used to playback a recording.

6. Interface Protocol



The XScope can communicate to external devices thru the USB or the external port. Each interface can access the Xscope’s main settings. Follow the protocols to make your own applications, or to make devices that attach to the XScope.

6.1 Interface settings

The settings for communicating with the serial port are shown in Table 9. If using the USB interface, you can use WinUSB or LibUSB libraries. The USB device’s endpoints have a size of 64 bytes. The device uses BULK IN transfers on endpoint 1 for transferring data (770 bytes in SCOPE mode: CH1, CH2, CHD, FRAME and INDEX; 1289 bytes in SNIFFER mode), BULK OUT transfers on endpoint 1 to write to the AWG RAM buffer (256 bytes), and CONTROL READ transfers on endpoint 0 for changing and reading settings. The FRAME and INDEX are variables used to check the flow of the data. They are also needed if you want to implement the ROLL mode.

The **FRAME** will increase whenever a full buffer has been acquired. For the fast sampling rates, this will occur all the time. Ideally, when you read data, the FRAME will increment by 1 every time. If you see that the FRAME has incremented by more than 1, then you are not reading data fast enough and missing frames of data. In the slow sampling rates, since you can read faster than the acquisition, the FRAME will help you determine if you are reading from the same frame.

The **INDEX** is not used in the fast sampling rates. For the slow sampling rates, it indicates the current position in the buffer. For example, if you set the Xprotolab at 1S/div, you will see on the display that the samples start filling the screen from left to right. The index represents this position. So for the slow sampling rates, the FRAME and INDEX can help you determine if you need to erase the screen, and you are able to show more data on the screen (similar to the Xprotolab screen).

6.2 Control Data

All XScope’s settings are stored in 44 bytes, table 10 shows these variables, table 11 describes the bitfield variables.

Index	Name	Data Type	Description	Notes
0	Srate	Unsigned 8bit	Sampling Rate	Range: [0, 21] 8 us/div to 50 s/div
1	CH1ctrl	Bit Field 8bit	Channel 1 controls	
2	CH2ctrl	Bit Field 8bit	Channel 2 controls	
3	CHDctrl	Bit Field 8bit	Logic Analyzer Options 1	
4	CHDmask	Bit Field 8bit	Logic enabled bits	Selects which logic channels are displayed
5	Trigger	Bit Field 8bit	Trigger control	
6	Mcursors	Bit Field 8bit	Cursor Options	
7	Display	Bit Field 8bit	Display Options	
8	MFFT	Bit Field 8bit	FFT Options	
9	Sweep	Bit Field 8bit	AWG Sweep Options	
10	Sniffer	Bit Field 8bit	Sniffer Controls	
11	MStatus	Bit Field 8bit	Scope Status	
12	CH1gain	Unsigned 8bit	Channel 1 gain	Range: [0,6] 5.12V/div to 80mV/div
13	CH2gain	Unsigned 8bit	Channel 2 gain	Range: [0,6] 5.12V/div to 80mV/div
14	HPos	Unsigned 8bit	Wave Position	Range: [0,127] pixels
15	VcursorA	Unsigned 8bit	Vertical Cursor A	Range: [0,127] pixels
16	VcursorB	Unsigned 8bit	Vertical Cursor B	Range: [0,127] pixels
17	Hcursor1A	Unsigned 8bit	CH1 Horizontal Cursor A	Range: [0,127] pixels
18	Hcursor1B	Unsigned 8bit	CH1 Horizontal Cursor B	Range: [0,127] pixels
19	Hcursor2A	Unsigned 8bit	CH2 Horizontal Cursor A	Range: [0,127] pixels
20	Hcursor2B	Unsigned 8bit	CH2 Horizontal Cursor B	Range: [0,127] pixels
21	Thold	Unsigned 8bit	Trigger Hold	Range: [0,255] 0 to 255 milliseconds
22	Tpost L	Unsigned 16bit	Post Trigger	Range: [0, 32767] Indicates how many samples to wait after the trigger. Default is 128.
23	Tpost H			
24	Tsource	Unsigned 8bit	Trigger Source	0: CH1; 1: CH2; 2-9: CHD; 10: External Trigger
25	Tlevel	Unsigned 8bit	Trigger Level	Range: [3,252]
26	Window1	Unsigned 8bit	Windows Trigger Level 1	Range: [0,255]
27	Window2	Unsigned 8bit	Windows Trigger Level 2	Range: [0,255]
28	Ttimeout	Unsigned 8bit	Trigger Timeout	Range: [0,255] 0.04096s to 10.48576s
29	CH1pos	Signed 8bit	Channel 1 Position	Range: [-128,0] pixels
30	CH2pos	Signed 8bit	Channel 2 Position	Range: [-128,0] pixels
31	CHDpos	Unsigned 8bit	Logic Analyzer position	Range: [0,7]
32	CHDdecode	Unsigned 8bit	Selected Protocol	0: SPI; 1: I2C; 2: RS232
33	Sweep1	Unsigned 8bit	Sweep Start	Range: [0,255]
34	Sweep2	Unsigned 8bit	Sweep End	Range: [0,255]
35	SWSpeed	Unsigned 8bit	Sweep Speed	Range: [0,127]
36	AWGamp	Signed 8bit	AWG Amplitude	Range: [-128,0] 4V to 0V
37	AWGtype	Unsigned 8bit	AWG Wave Type	0: Noise; 1: Sine; 2: Square; 3: Triangle; 4: Custom
38	AWGduty	Unsigned 8bit	AWG Duty Cycle	Range: [1,255] 0.391% to 99.61%
39	AWGoffset	Signed 8bit	AWG Offset	Range: [-128,127] +2V to -1.985V
40	desiredF LLB	Unsigned 32bit	AWG Desired Frequency multiplied by 100	Range: [100, 12500000] 1Hz to 125kHz
41	desiredF LHB			
42	desiredF HLB			
43	desiredF HHB			

Table 10: Xscope's settings



6.2.1 Bitfield variables

Name	Bits	Notes
CH1ctrl and CH2ctrl	Bit 0: Channel on	
	Bit 1: x10 probe	For future hardware
	Bit 2: Bandwidth limit	For future hardware
	Bit 3: AC/DC select	For future hardware
	Bit 4: Invert channel	
	Bit 5: Average samples	
	Bit 6: Math Active	Enables math (addition or multiplication)
	Bit 7: Math operation	Subtract (1) or Multiply (0)
CHDctrl	Bit 0: Channel on	
	Bit 1: Pull	Pull resistor enabled
	Bit 2: Pull Up	Pull up (1) or pull down (0)
	Bit 3: Low	Thick line when logic '0'
	Bit 4: Invert channel	
	Bit 5: Serial Decode	
	Bit 6: Parallel Decode	
	Bit 7: ASCII Sniffer display	
Trigger	Bit 0: Normal Trigger	
	Bit 1: Single Trigger	The Normal Trigger bit must also be enabled for Single Trigger
	Bit 2: Auto Trigger	
	Bit 3: Trigger Direction	
	Bit 4: Round Sniffer	
	Bit 5: Slope Trigger	
	Bit 6: Window Trigger	
	Bit 7: Edge Trigger	Dual Edge Trigger is enabled by clearing bits 5,6,7.
Mcursor	Bit 0: Roll Scope	
	Bit 1: Automatic Cursors	
	Bit 2: Track Cursors	
	Bit 3: CH1 Horizontal Cursors on	CH1 and CH2 Horizontal cursors are mutually exclusive
	Bit 4: CH2 Horizontal Cursors on	CH1 and CH2 Horizontal cursors are mutually exclusive
	Bit 5: Vertical Cursor on	
	Bit 6: Reference waveform on	
	Bit 7: Single Sniffer Capture	
Display	Bit 0: Grid 0	00: No Grid, 01: Dots per division
	Bit 1: Grid 1	10: Follow trigger, 11: Graticule
	Bit 2: Elastic Display	
	Bit 3: Invert Display	
	Bit 4: Flip Display	
	Bit 5: Persistent Display	
	Bit 6: Line / Pixel Display	Line (1), Pixels (0)
	Bit 7: Show Settings	



Name	Bits	Notes	
MFFT	Bit 0: Hamming Window	Only one window must be selected, or none for No Window.	
	Bit 1: Hann Window		
	Bit 2: Blackman Window		
	Bit 3: Vertical Log		
	Bit 4: IQ FFT		
	Bit 5: Scope Mode	Multiple modes can be selected simultaneously. If no bits are set, the Meter mode is displayed.	
	Bit 6: XY Mode		
	Bit 7: FFT Mode		
Sweep	Bit 0: Acceleration Direction		
	Bit 1: Accelerate Sweep		
	Bit 2: Sweep Direction		
	Bit 3: Ping Pong Mode		
	Bit 4: Sweep Frequency		
	Bit 5: Sweep Amplitude		
	Bit 6: Sweep Offset		
	Bit 7: Sweep Duty Cycle		
Sniffer	Bit 0: Baud 0	UART Sniffer Baud Rates: 000: 1200 , 001: 2400, 010: 4800, 011: 9600, 100: 19200, 101: 38400, 110: 57600, 111: 115200	
	Bit 1: Baud 1		
	Bit 2: Baud 2		
	Bit 3: Uart 0	UART Data bits:	
	Bit 4: Uart 1	00: 5 Bits , 01: 6 Bits , 10: 7 Bits , 11: 8 Bits	
	Bit 5: Parity Mode / SS Invert	Enables UART parity check	SPI Invert Slave Select
	Bit 6: Parity / CPOH Clock Polarity	UART Parity Odd (1), Parity Even (0)	SPI Clock Polarity
	Bit 7: Stop Bit / CPOH Clock Phase	1 Stop bit (0), 2 Stop bits (1)	SPI Clock Phase
MStatus	Bit 0: Update	Exits triggering if the bit is set	
	Bit 1: Update AWG	The AWG parameters must be updated if the bit is set	
	Bit 2: Update MSO	The MSO parameters must be updated if the bit is set	
	Bit 3: Go Sniffer	Enters the Sniffer mode if the bit is set	
	Bit 4: Stop	Oscilloscope Stopped	
	Bit 5: Triggered	Oscilloscope Triggered	
	Bit 6: Meter VDC	If both bits are cleared, the Meter mode measures Frequency. If both bits are set, the Meter mode counts Pulses.	
	Bit 7: Meter VPP		

Table 11: Bitfield variable description

6.3 Vendor ID and Product ID

If you are using LibUSB to interface with the device, you need: VID=16D0 PID=06F9

If you are using WinUSB, you will need the GUID defined on the driver's .inf file:

GUID= 88BAE032-5A81-49f0-BC3D-A4FF138216D6

6.4 Command Set

When using the serial port, the commands are sent to the XScope in ASCII format, further data sent or received is in binary. When using the USB interface, the commands are sent as CONTROL READ requests, where the packet's request byte is the command, and the packet's Index and Value are additional parameters sent to the XScope. If the PC is requesting data, it will be returned in the endpoint 0 IN buffer. Table 12 shows the XScope Interface Protocol Commands.



Command	Description	Device Response / Notes
a	Request firmware version	The device returns 4 characters with the version.
b	Writes a byte to the XScope's Settings, at the specified index. If the Index is below 14, the updatemso bit is automatically set. If the Index is above 34, the updateawg is automatically set.	When using the USB interface, the setup packet's Index contains the index, and the setup packet's Value contains the data. When using the using the Serial interface, two additional bytes must be sent containing the index and data.
c	Sets the desired AWG Frequency (32bits).	When using the USB interface, the Index contains the lower 16bits, the Value contains the high 16bits. When using the Serial interface, 4 additional bytes must be sent in little endian format.
d	Save XScope's Settings in EEPROM	
e	Save AWG wave stored in RAM to EEPROM	
f	Stop Scope	
g	Start Scope	
h	Force Trigger	
i	Auto Setup	
j	Sets the desired Post Trigger value (16bits)	USB interface: the Value contains the 16bits. When using the Serial interface, 2 additional bytes must be sent in little endian format.
k	Restore factory settings	
m	Request METER measurement data	The device sends 4 bytes in little endian format. See the description in section 6.4.2 below.
p	Disable Auto send (Serial interface only)	
q	Enable Auto send (Serial interface only)	See the description in section 6.4.1 below.
u	Request settings	All the settings (44 bytes) are sent to the PC.
w	Request EE waveform (Serial interface only)	EE Wave data (256 bytes)
x	Send waveform data (Serial interface only)	'G' character, signals the PC that the device is ready, Then the PC sends the data (256 bytes), Then the device sends a 'T' character, which signals the PC that the data was received.
C	Request BMP (Serial interface only)	128x64 Monochrome BMP using the XModem protocol

Table 12: XScope Command Set

6.4.1 Auto Send

When the Auto send is active, the device will continuously send the channels data to the serial port. The "Channels Sent" byte is an

Fast Sampling	0x0D	0x0A	'F'	Data	0x0A	0x0D	Channels sent
Slow Sampling	0x0D	0x0A	'S'	Data	0x0A	0x0D	Channels sent

Table 13: Auto Send Packet Format

ASCII number that represents which channels were sent, Channel 1 will add 1 to the number, Channel 2 will add 2, and the logic port will add 4. For example, if all channels are sent, the "Channels Sent" will be the character '7'.

6.4.2 METER measurement data

The METER mode must be enabled in order to receive valid data; the data sent will depend on which METER mode is selected. In the VDC mode, the Channel data is in 16bit signed format, the value represents the millivolts divided by 1.25. In the Frequency Counter mode, the data is in 32bit unsigned format, which represents the counter value.

	VDC	FREQUENCY
Byte 0	Channel 1 LB	Freq. Cnt. byte 3
Byte 1	Channel 1 HB	Freq. Cnt. byte 2
Byte 2	Channel 2 LB	Freq. Cnt. byte 1
Byte 3	Channel 2 HB	Freq. Cnt. byte 0

Table 14: METER data



7. BMP Screen Capture



7.1 To send a BMP screen capture to a PC:

You can send a screen capture of the XScope to your PC using HyperTerminal. All the screen captures bitmaps in this manual were generated using this method. The screen capture is done thru the XScope's serial port.

- Open HyperTerminal.
- Enter a name for a new connection (example: scope).
- Enter the COM port where the device is connected.
- Select 115200 bits per second, 8 data bits, Parity None, 1 Stop bit, Flow control None. (See figure 56)

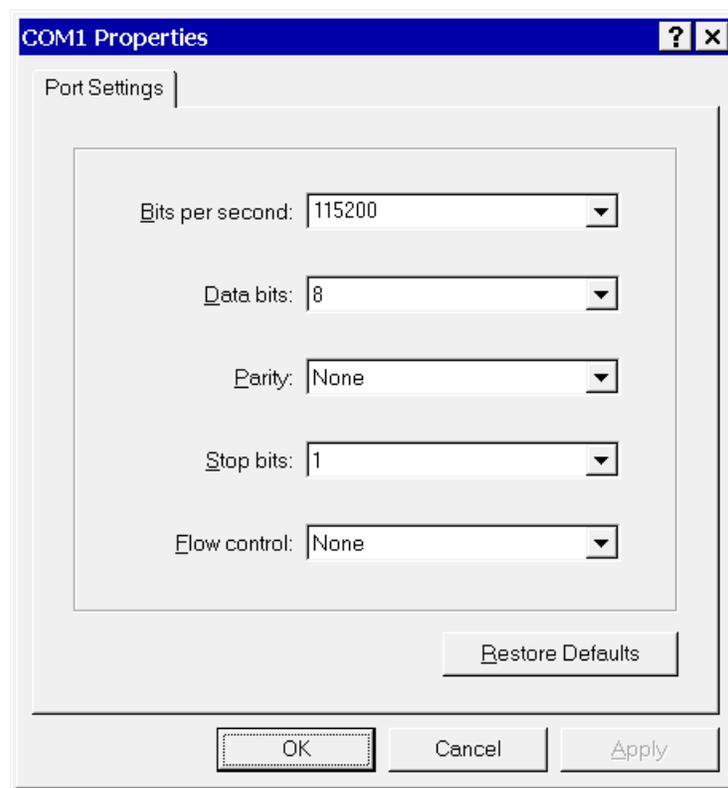


Figure 56: HyperTerminal Settings

- In the Transfer menu, select Receive File.
- Enter a folder where to save the file and use the XMODEM protocol. (See figure 56)

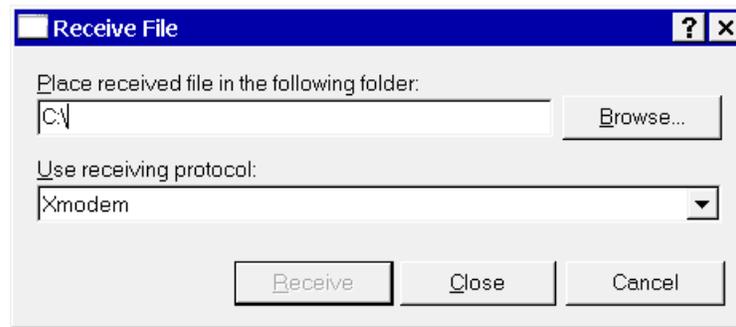


Figure 57: Receive File Settings

- Enter a file name with a BMP extension and press OK

7.2 To send a BMP screen capture to Linux:

Create the following script and save as capture.sh:



[capture.sh](#)

```
echo "Please enter filename. e.g  
capture.bmp"  
read name  
stty -F $1 115200  
rx -c $name < $1 > $1
```

To use, make the script executable with “`chmod +x capture.sh`”. Then enter “`./capture.sh`” into a terminal followed by the serial device for example “`./capture.sh /dev/ttyUSB0`”.

Then enter a name for the bmp image including the .bmp file extension.

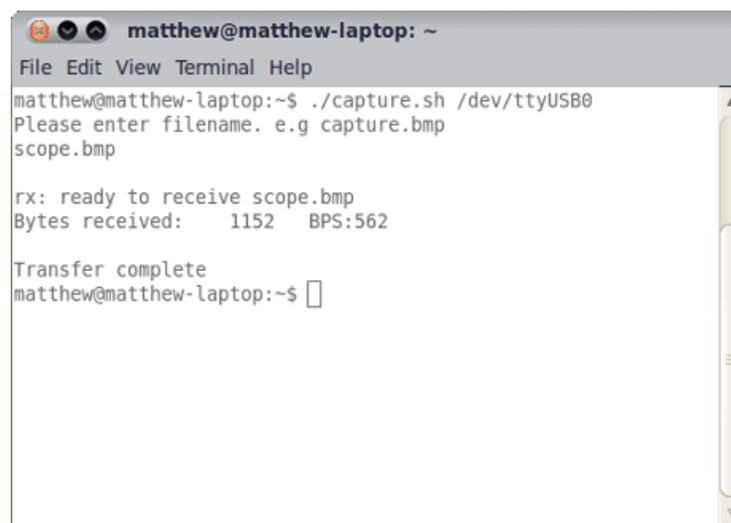


Figure 58: Screen capture in Linux



8. XScope's Examples

8.1 Resistor Voltage Divider

- 1) Build the circuit shown on figure 59.
- 2) Set the device to Meter mode
- 3) You should see similar voltages as shown on figure 60.

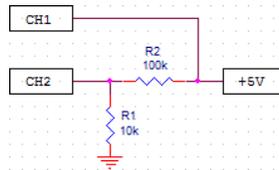


Figure 59: Resistor divider

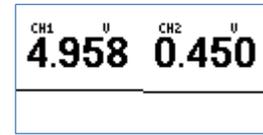


Figure 60: Meter mode

Theory of operation: The circuit is a voltage divider, where V_{in} is 5V, and V_{out} is the voltage at CH2: $V_{out} = \frac{V_{in} * R1}{R1 + R2}$

8.2 Measurement of an RC time constant

- 1) Build the circuit shown on figure 62.
- 2) Set the time base to 500 μ s/div.
- 3) Set the AWG to Square wave, 500Hz, 4V.
- 4) Set the gain on both channels to 2.56V/div.
- 5) The display should look similar to figure 61.
- 6) Now set the time base to 16 μ s/div.
- 7) Turn off CH1, set the CH2 gain to 1.28V/div.
- 8) Adjust the horizontal and CH2 positions so that the rising wave takes most of the screen.
- 9) Turn on the vertical and CH2 horizontal cursors.
- 10) Enable the cursors TRACK option.
- 11) Set the first vertical cursor at the corner of the wave, and the second cursor where the voltage equals 0V.
- 12) The display should look like figure 64. The measured time $T_{1/2} = 69\mu$ s, is the "half-life" time, so $RC = 99.95\mu$ s

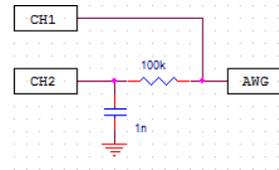


Figure 62: RC Circuit

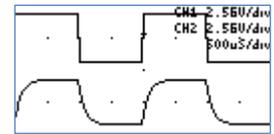


Figure 61: RC Measurements

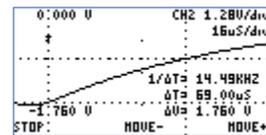


Figure 64: Half-life measurement

$$V_c = V_0(1 - e^{-\frac{t}{RC}})$$

$$RC = \frac{T_{1/2}}{\ln(2)}$$

Figure 63: RC Equations

Theory of operation: Circuit theory shows that if the RC circuit is fed with a step input, the output will approach a DC value exponentially; figure 63 shows the equation from which we can obtain RC when the half-life value is known.

8.3 Half Wave Rectifier with Smoothing Capacitor

- 1) Build the circuit shown on figure 66.
- 2) Set the time base to 2mS/div.
- 3) Set the AWG to Sine wave, 125Hz, 4V.
- 4) Set the gain on both channels to 1.28V/div.
- 5) The display should look like figure 65.
- 6) If the capacitor is removed, the display should look like figure 67.

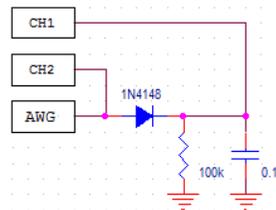


Figure 66: Half wave rectifier circuit

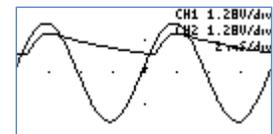


Figure 65: Half wave rectifier

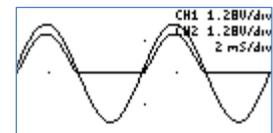


Figure 67: Removing the capacitor

Theory of operation: The diode will allow current to flow only during the positive half of the sine wave. The output voltage is a little bit lower because of the voltage drop of the diode. When the AWG voltage is negative, the diode acts like an open circuit and the capacitor discharges thru the resistor at an exponential rate.

8.4 BJT Amplifier

- 1) Build the circuit shown on figure 69.
- 2) Set the time base to 2ms/div
- 3) Move the position on both channels all the way down (GND reference grid is at the bottom of the screen).
- 4) Set CH1 to 0.32V/div, Set CH2 to 1.28V/div.
- 5) Set the AWG to Sine wave, 125Hz, 0.250V amplitude.
- 6) Increase the AWG offset until the CH2 wave is centered on the display. The display should look like figure 68.

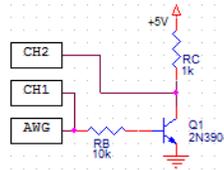


Figure 69: Amplifier circuit

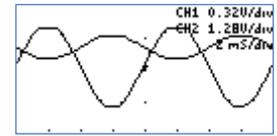


Figure 68: BJT Measurements

Theory of operation: The transistor needs to be biased in its forward active region; this is what the offset in the AWG is for. The output voltage will vary according to the BJT transfer curve: changes in the input make large changes in the output.

8.5 Component V/I Curves

- 1) Build the circuit shown in figure 72, or flip the CURVE switch on the Portables.
- 2) On the Portables, flip the input switches to the DC position.
- 3) Set the time base to 500 μ s/div.
- 4) Set the AWG to Sine wave, 125Hz, 4V.
- 5) Set the gain on both channels to: 0.64V/div, or 1.28V/div on the Portables.
- 6) Enter the CH2 options and select SUBTRACT.
- 7) Set the device oscilloscope in XY mode.

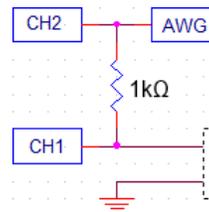


Figure 72: Component tester

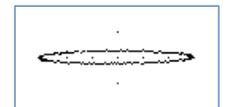


Figure 70: 100nF Capacitor curve

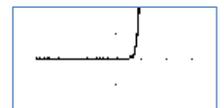


Figure 71: 1N4148 curve

Theory of operation: The goal is to plot the component's voltage, versus the component's current. Using the integrated waveform generator and a 1k Ω resistor, we can inject current into the component. The voltage is measured directly using CH1. The current thru the component is the same as the current thru the resistor, the voltage on the resistor is proportional to the current. The voltage on the resistor is equal to CH2-CH1. 1V on the scope will represent 1mA on the component. Figure 70 and figure 77 show examples of V/I curves on components.

8.6 Frequency Plots

The AWG sweep function can be used to plot the frequency response of a circuit. This method is not directly a BODE plot since the horizontal axis is not logarithmic, it is linear.

- 1) Connect the AWG to the input and CH1 to the output.
- 2) Set the device to FFT mode.
- 3) Change to the desired time base. The maximum frequency is shown on the top right of the display.
- 4) Set the AWG to Sine Wave.
- 5) Enable the Frequency Sweep.
- 6) Set the AWG Sweep range to 1:255
- 7) Set the display to persistent.

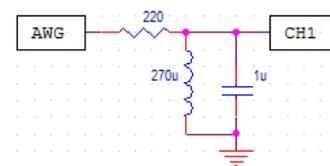


Figure 73: RLC Circuit

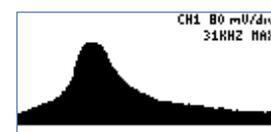


Figure 74: Frequency plot

Figure 73 shows an RLC circuit, and figure 74 shows the frequency response. This example shows the vertical scale with the LOG disabled.



9. Firmware Updating



This guide will show how to update the firmware on your AVR XMEGA based device. There are two updating methods; the first method requires an external programmer. You can use either method depending on your needs.

9.1 Firmware upgrade using an external programmer

9.1.1 Tools required

- AVRISP mkII, or similar PDI capable programmer
- AVR Studio 4 or Atmel Studio 6 IDE (Integrated Development Environment)
- HEX and EEP files for the device, found on the product's page (Look for the HEX icon).

A regular AVR programmer might not work, the programmer needs to be PDI capable. PDI is the new interface to program XMEGA microcontrollers. Many old AVR programmers use ISP, which is not compatible with the XMEGA.

9.1.2 Instructions to install the tools

- Install AVR Studio and USB driver
- Connect the programmer to the computer and auto install the hardware

A more detailed guide on how to install the tools is found [here](#):

http://www.atmel.com/dyn/resources/prod_documents/AVRISPMkII_UG.pdf

9.1.3 Instructions to update the firmware

1. Start AVR Studio
2. Connect the cable from the AVRISP to the PDI connector on the board
3. Power the board
4. Press the "Display the 'Connect' dialog" button: . Alternatively, you can go to this menu: Tools-> Program AVR -> Connect -
5. Select your programmer and port. (AVRISP mkII and AUTO or USB)
6. In the MAIN tab, select the device: ATXMEGA32A4U
7. In the programming mode, select PDI
8. To check that everything is ok, press the "Read Signature" button. You will see a message saying that the device matches the signature.
9. Go to the PROGRAM tab
10. In the Flash section, look for the .HEX file and click Program
11. In the EEPROM section, look for the .EEP file and click Program
12. Go to the FUSES tab and select:
 - BODPD: Sampled, BODACT: Continuous, BODLVL: 2.8V, SUT: 4ms
13. Click Program
14. After updating the firmware, make sure to recalibrate the device (See section 1.6).

9.2 Firmware upgrade using the bootloader

9.2.1 Tools required

- Standard USB type A to micro USB cable.
- **Windows:**
 - Atmel's FLIP software: <http://www.atmel.com/tools/FLIP.aspx>.
 - Flip Manual with driver installation procedure: <http://www.atmel.com/Images/doc8429.pdf>.
- **Linux:** AVRDUDE version 6.1 or above: <http://www.nongnu.org/avrdude/>
- HEX and EEP files for the device, found on the product's page (Look for the HEX icon).

9.2.2 Activating the bootloader

The device needs to be powered off first. The K1 button needs to be pressed while powering on the device:

- For the portable devices, you need to press K1 and also press the MENU button.
- For the non-portable, you need to press K1 while connecting the device to the computer with the USB cable.
- For devices without buttons, connect pins 1 and 6 of the external port, then connect the USB cable.

Once the XScope enters the bootloader, the red LED will be lit, and will blink with USB activity. The XScope will appear as a new device on the host computer, the drivers required are found in the FLIP application folder.

9.2.1 Firmware update in Windows: FLIP

- 1) Select ATXMEGA32A4U in the device selection list.
- 2) Select USB as communication medium.
- 3) Open the USB port to connect to the target.
- 4) Make sure the FLASH buffer is selected and check: ERASE, BLANK CHECK, PROGRAM, VERIFY.
- 5) Load the HEX file .hex
- 6) Press RUN.
- 7) Press SELECT EEPROM.
- 8) Load the HEX file .eep.
- 9) Uncheck ERASE and BLANK CHECK, only leave checked PROGRAM and VERIFY.
- 10) Press RUN.
- 11) Press START APPLICATION.
- 12) After updating the firmware, the device will enter

the calibration function on the first power up (See section 1.6).

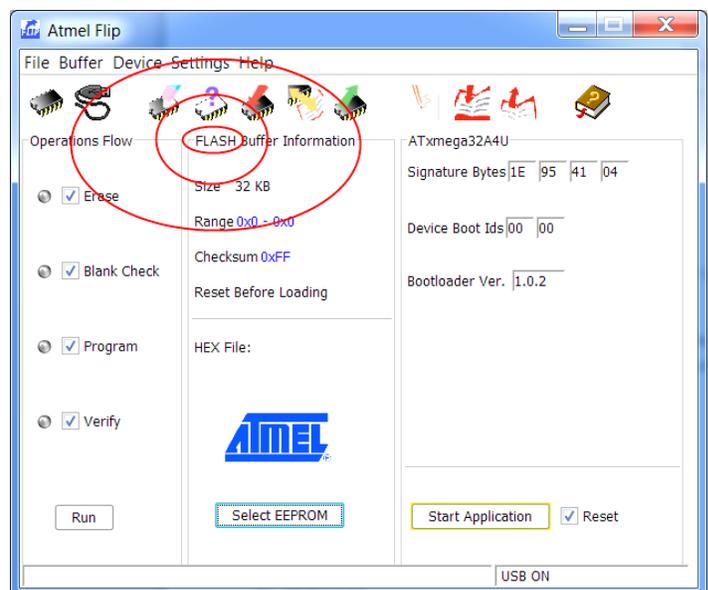


Figure 75: Flip application

9.2.2 Firmware upgrade in Linux: AVRDUDE

- 1) Run avrdude, for both the application and EEPROM contents:


```
sudo avrdude -c flip2 -p x32a4u -U application:w:xprotolab-p.hex:i
sudo avrdude -c flip2 -p x32a4u -U eep:w:xprotolab-p.eep:i
```

10. Frequently Asked Questions

1) What tools do I need to develop my own programs on the XScope?

If you don't need debugging capabilities, only a regular cable is needed to program the device. If you want to be able to debug your code, you need an external debugger, such as the [AVR JTAGICE mkII](#) or the [AVR ONE!](#).

Software Tools:

- Integrated Development Environment: [AVR Studio 4](#) or [Atmel Studio 6](#)
- If using AVR Studio 4, the C compiler is a separate package, found in the WinAVR package.

2) Can the waveform generator and the oscilloscope run simultaneously?

Yes, the waveform generator runs on the background. (The AWG uses the DMA, so it doesn't need any CPU intervention).

3) How do I power the non-portable XScopes?

The XScope can be powered thru the micro USB port. Alternatively, the XScope can be powered by connecting a 5V power supply on the 5V pin. Do not connect a 5V power supply and the USB at the same time.

4) Can I connect the XScope to the computer to control the oscilloscope and get the data?

Yes, you can use the [XScope PC Interface](#). A [UART to USB cable](#) will be required for old hardware revisions 1.4 and 1.5.

5) Can I connect the XScope to the computer using the USB for firmware updates?

Yes. Only the old hardware revisions (1.4 and 1.5) need a PDI programmer for firmware updates.

6) How much power can the non-portable XScopes supply?

The XScope can also power external devices. This is the maximum current on each voltage:

+5V: Will be the same as the power source minus 60mA.

-5V: Approximately 50mA, but this subtracts from the available current on the +5V line.

+3.3V: Approximately 200mA, but this subtracts from the available current on the +5V line.

7) What is the maximum frequency that I can measure with the XScope?

The analog bandwidth is set at 200kHz. However, you can still measure frequencies up to almost Nyquist/2, i.e. 1MHz. The FFT analysis will be particularly useful when measuring high frequencies. For digital signals, you can measure up to 16MHz.

8) Can I measure voltages above 20V?

The portable XScopes can use a 3.5mm to BNC adapter, and then you could use a standard 10:1 probe. On the non-portable devices, you can add a 9Mohm resistor in series to the input. Since the input impedance of the device is 1Mohm, the voltage will be divided by 10 (This is the equivalent of using a 10:1 probe).

9) Are the logic inputs 5V tolerant?

Only on the Portable devices. On the non-portable variants, the logic inputs are not 5V tolerant. An easy solution would be to place a 3K resistor in series with the 5V signal, this will work for signals with a frequency lower than 200kHz. Another solution would be to use a voltage translator chip, such as the 74LVC245.



10) The source code says "evaluation version", can I get the full version?

The full source code can be found on GitHub: <https://github.com/ganzziani/XScopes-Firmware>

11) I tried updating the firmware using the USB bootloader. It didn't work and now the device does not start.

The bootloader should still be functional. Try resetting the computer and try the procedure again, step by step.

12) How does the XScope compare to other digital oscilloscopes?

You can check this comparison table: <http://www.gabotronics.com/resources/hobbyists-oscilloscopes.htm>

11. Troubleshooting

1) The unit does not power up!

If using the USB connector to power, try applying 5V directly instead with another power supply.

2) Something went wrong during the USB updating, now my portable device doesn't even get recognized by the PC.

The device might be running invalid code. Open the case and short pins 5 and 6 of the PDI port to reset the device.

3) The unit powers up but the MSO is not working.

Try restoring the default settings: Press K4 during power up, then select "Restore"

4) The screen turns off after a certain amount of time.

This is the screen saver in action. The screen saver time can be changed in factory options (See Section 1.6).

5) I built my own Xprotolab, a particular button doesn't work.

There might be shorted pins on the microcontroller. Check for debris or solder bridges. Using a solder wick on the pins will help.

6) When powering up, the splash screen stays for a very long time (more than 4 seconds).

The crystal is defective or the traces on the crystal are shorted.

7) It still isn't working!

If you have a multimeter and want to try to fix yourself, check the following voltages. If any of the voltages are wrong, there might be a defective part.

- Voltage at +5V should be between +4.75 and +5.25V
- Voltage at -5V should be between -4.75 and -5.25V
- Voltage at +3.3V should be between +3.2 and +3.4V
- Voltage at pin 8 of U3 should be between +2.00 and +2.09V

TIP

If the unit is powered with more than 5.5V, the negative voltage generator would be the first component to get damaged.

TIP

If the screen is blank, some signs that the microcontroller is working are:

- Activity on the green LED.
- When connecting to a PC, you can hear the "Device connected" sound on Windows.

If all fails and if the device is under warranty, you can send it back for repairs.

12. XScope Design

The hardware and firmware are open source under the [GPLv3](https://www.gnu.org/licenses/gpl-3.0.html) license. The firmware for all the Xscopes is available on GitHub: <https://github.com/ganzziani/XScopes-Firmware>

12.1 System Architecture

The XScope uses many resources and peripherals of the XMEGA microcontroller. Figure 76 shows the XScope's Architecture block diagram. Figure 77 shows the Frequency Counter block diagram.

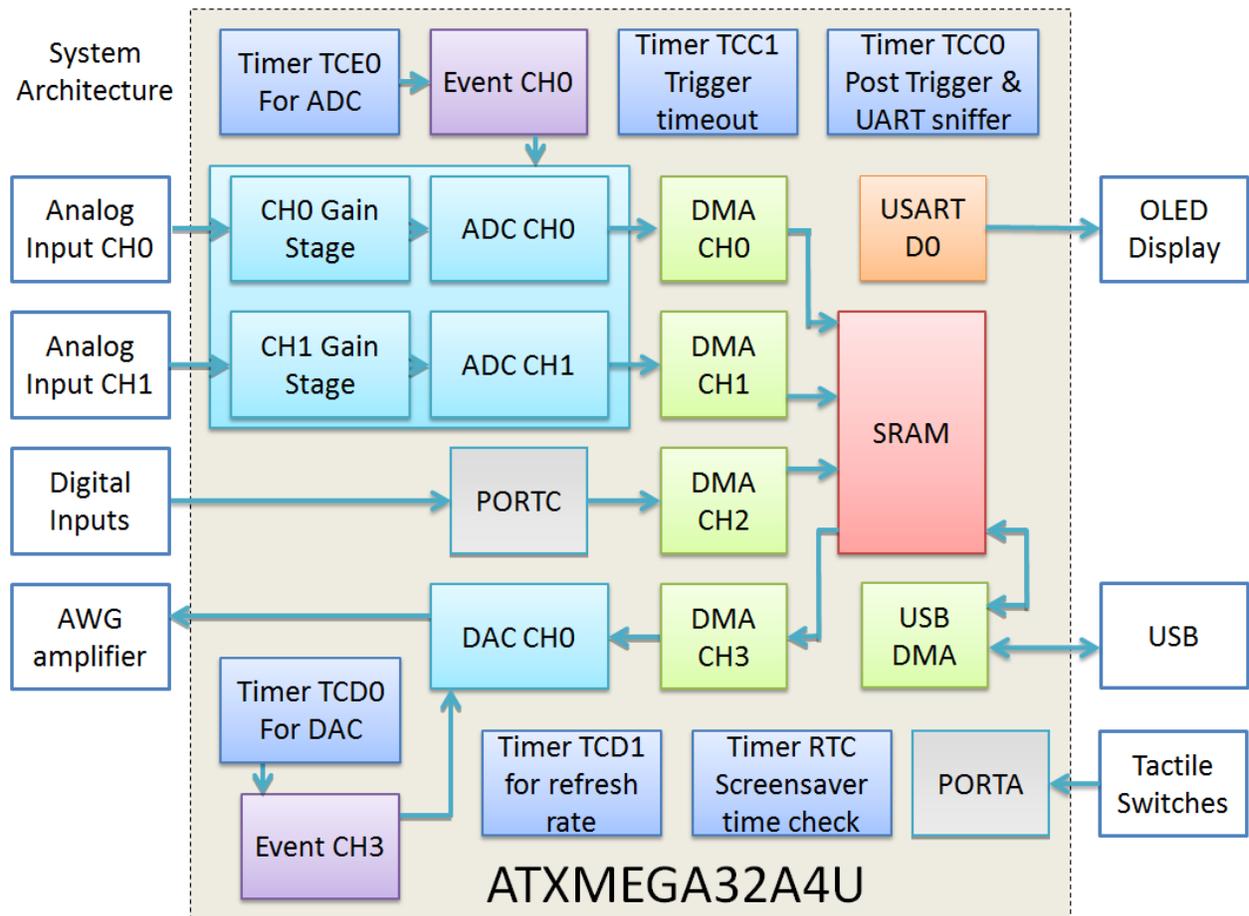


Figure 76: XScopes Architecture Block Diagram

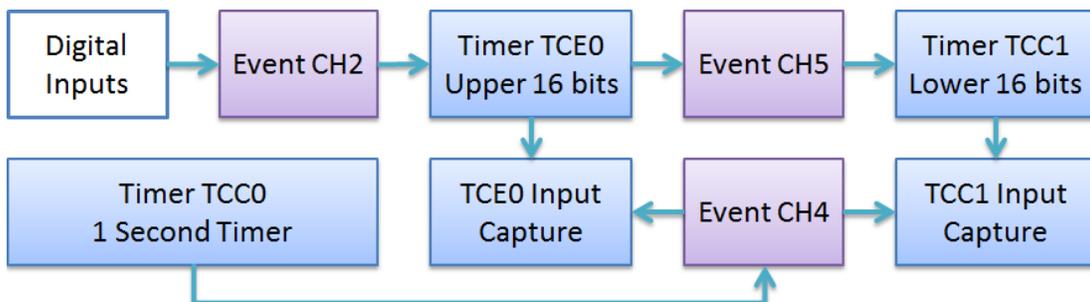


Figure 77: Frequency Counter Block Diagram



12.2 Schematics

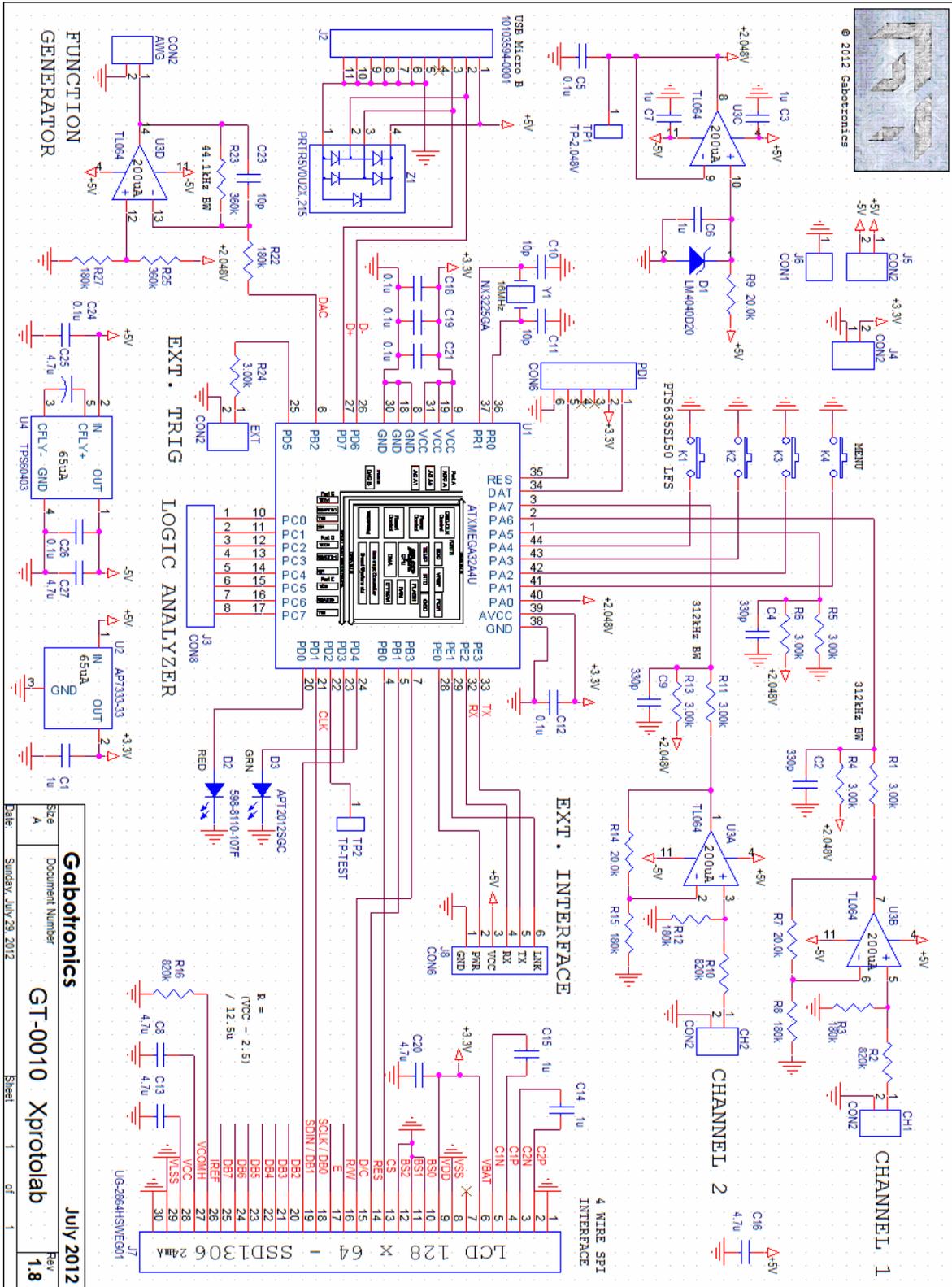


Figure 78: Xprotolab Schematic





www.gabotronics.com