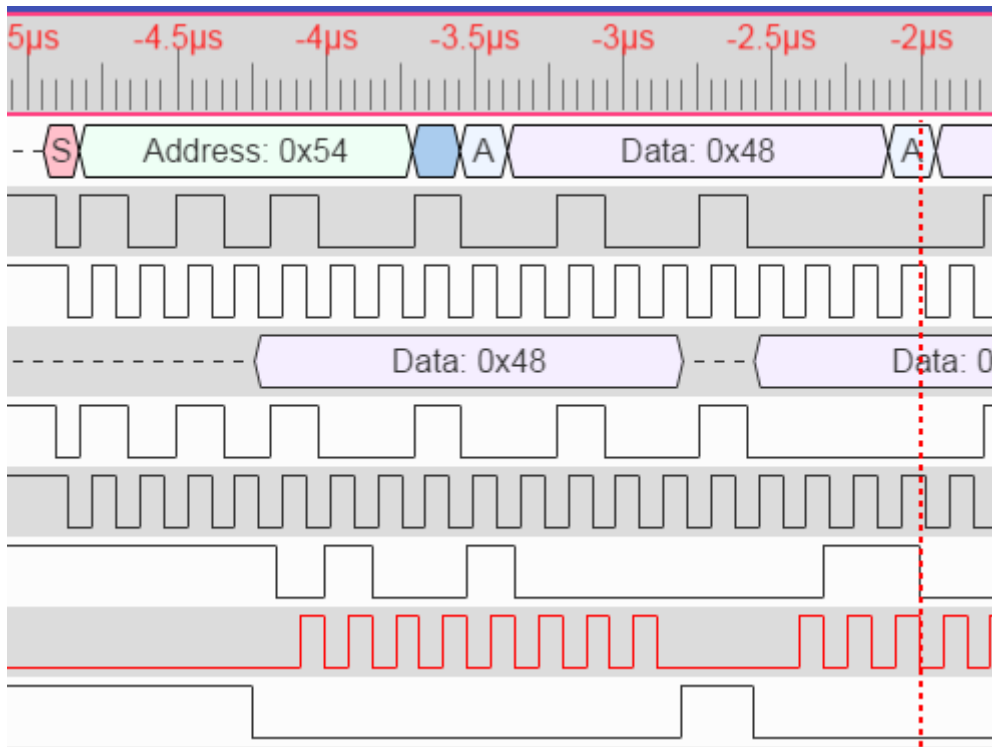


BX Logic Analyzer

Getting Started

This is the documentation for the Bugblat BX series of Logic Analyzers.



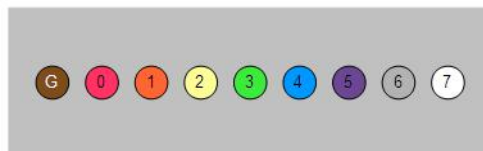
The installation guide for the BX software app is here: [InstallationGuide](#).

Pinouts

Many users will use hook-up cables built with standard multi-colored ribbon cable. This cable has a 10-color repeat, with the colors cycling in order through brown, red, orange, yellow, green, blue, purple, gray, white, black.

Default color assignments follow the same sequence, simplifying the assembly of hookup cables.

The pinning for the input connector on the BX8 is shown below. Note that this is the pinning when you look into the header with the component side uppermost. The pinout is also shown on the circuit board.



The *Trigger Out* connector is a loop behind the input connector.

Software Confidence Test

Now you need to connect to the target circuit via logic grabbers, set the acquisition parameters, and acquire some data. The simplest confidence test connects one of the logic analyzer input pins to the *Trigger Out* connection. Set *Trigger Out* to 100KHz or 1MHz via the software *Sample Capture* dialog (see the *Main Window* section).

Getting Started

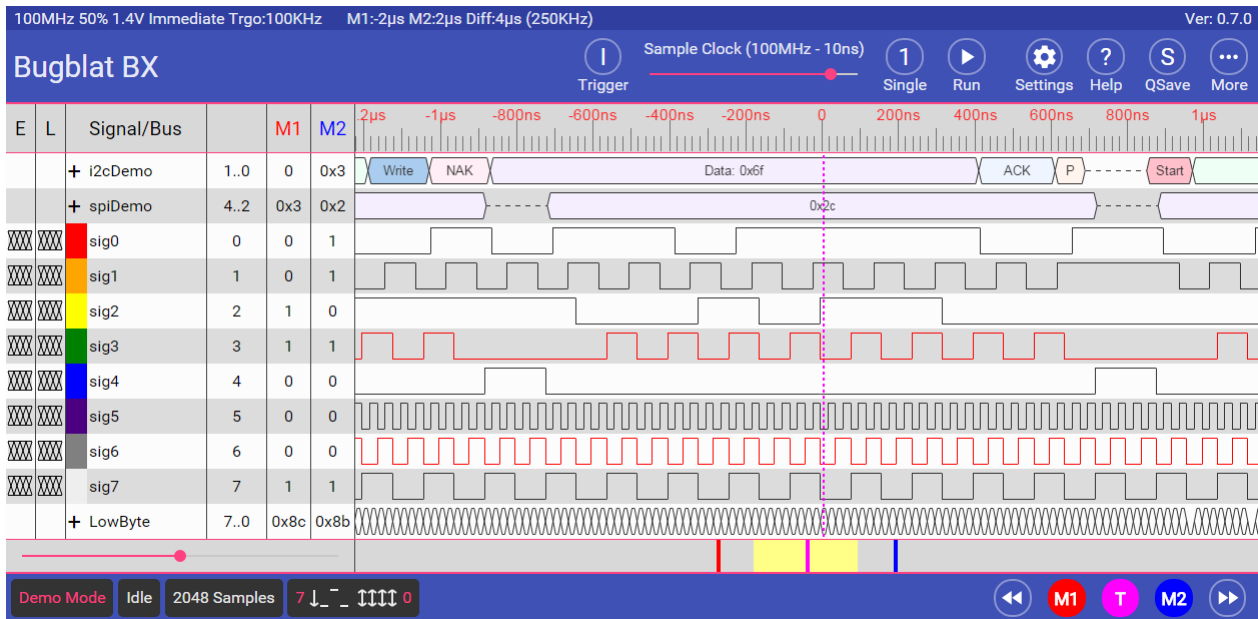
The default acquisition parameters, with all pins set to *don't care*, will sample the data on the probes. To make a more controlled acquisition you must set some or all of the controls, such as the E (edge) and L (level) patterns and the sample clock speed - see the *Main Window* section.

Then press *Run* and off you go.

Main Window

This is the main window of the program. It contains

- a top status bar
- a control bar
- a display area
- a bottom status bar



Top Status Bar

The top status bar displays

- the settings for the *next* run of the instrument. Note that two current display uses the settings that were in force when the *run* button was pressed. This part of the display is updated as the settings are changed. Clicking on this part of the display brings up the *Capture Settings* dialog.
- a measurement of the times of the *M1* and *M2* markers. This part of the display is updated as the markers are moved.

Control Bar

These are the main operating controls for the instrument. From left to right the controls are

Trigger Mode

Click to change the trigger mode. The trigger mode cycles through Immediate triggering, Edge triggering, Level triggering, and Advanced triggering.

Immediate, Edge, and Level cover the majority of captures. The detailed settings for Advanced triggering can be set via a dialog (see below).

Sample Rate

Drag the slider to change the acquisition sample rate.

Continuous Operation

Click to enable continuous sampling - C is displayed. Click again for one-shot sampling - 1 is displayed.

Run/Stop

Click to run, click again to stop. In normal operation the instrument takes a set of samples, displays the samples, and re-enables the run button. When a run is interrupted via the run/stop button any partial readings are discarded.

Settings

Click here to bring up a dialog display. Cycle through the dialogs by clicking on the round buttons on the top line of each dialog. You can also use left and right arrows on a keyboard to cycle through the dialogs. Close the dialog display by clicking outside.

Help

Display a short version of this information.

Configuration Quick Save

This control is only shown on desktop versions (Windows etc.) It is not shown if the software is running in a browser. The program save the configuration and exits. Data is not automatically saved.

More

- *Save Configuration* - save to a disk file, typically bx.ini
- *Restore Configuration* - restore from a disk file, typically bx.ini
- *Save Data* - save to a disk file, typically bx.bxd
- *Restore Data* - restore from a disk file, typically bx.bxd
- *About* - version information

Wave Display Area

Columns

E and L

For single-bit signals, the E and L columns show the Edge and Level Pattern Recognizer settings. Clicking on a cell cycles the pattern bit.

Color Stripe

- for single-bit signals: the wire color corresponding to the signal.
- for bus signals: clickable bus expand (+) and contract (-) controls.

Signal/Bus Name

Click to edit. Click down and drag to rearrange rows.

Signal/Bus Definition

Click on a definition to edit any field for the signal.

Bottom Status Bar

M1 and M2

Columns M1 and M2 correspond to the Marker 1 (red) and Marker 2 (blue). They show the signal values at the cursor position.

Waveforms

Waves are shown with transitions occurring between sample points. The red and blue markers can be dragged to reposition. The trigger position marker is fixed at the trigger point - it can not be moved.

Zooming and Panning

The zoom slider will zoom out until all samples fit on the screen.

The panning area is under the waveforms. It shows the positions of the markers, with a mimic rectangle to show the position of the waveform display in the sample set. The mimic rectangle has a minimum size and at large zooms it is slightly larger than the true waveform on display. It would be difficult to move if it were to shrink to one pixel wide.

Drag the yellow mimic rectangle to pan the display.

Bottom Status Bar

The status bar along the bottom edge of the screen contains a series of status panels plus a set of navigation buttons.

Demo Mode

Demo Mode is displayed when the software is in demonstration mode - the software cannot find a physical BX instrument on a USB port. If a BX instrument is connected, the serial number of the instrument is displayed, typically starting with "BX..."

Status

Shows the current operation being carried out by the program and the instrument - resetting the Logic Analyzer, configuring the Logic Analyzer, searching for a pattern, downloading data, and so on.

Sample Count

The sample depth panel shows the number of samples in the data buffer. BX instruments uses run length compression to squeeze up repeating data values so the total number can be very large.

Pins

The pin status panel is updated continuously (every 100ms) when the analyzer is idle, and freezes when the analyzer is taking a reading.

Navigation Buttons

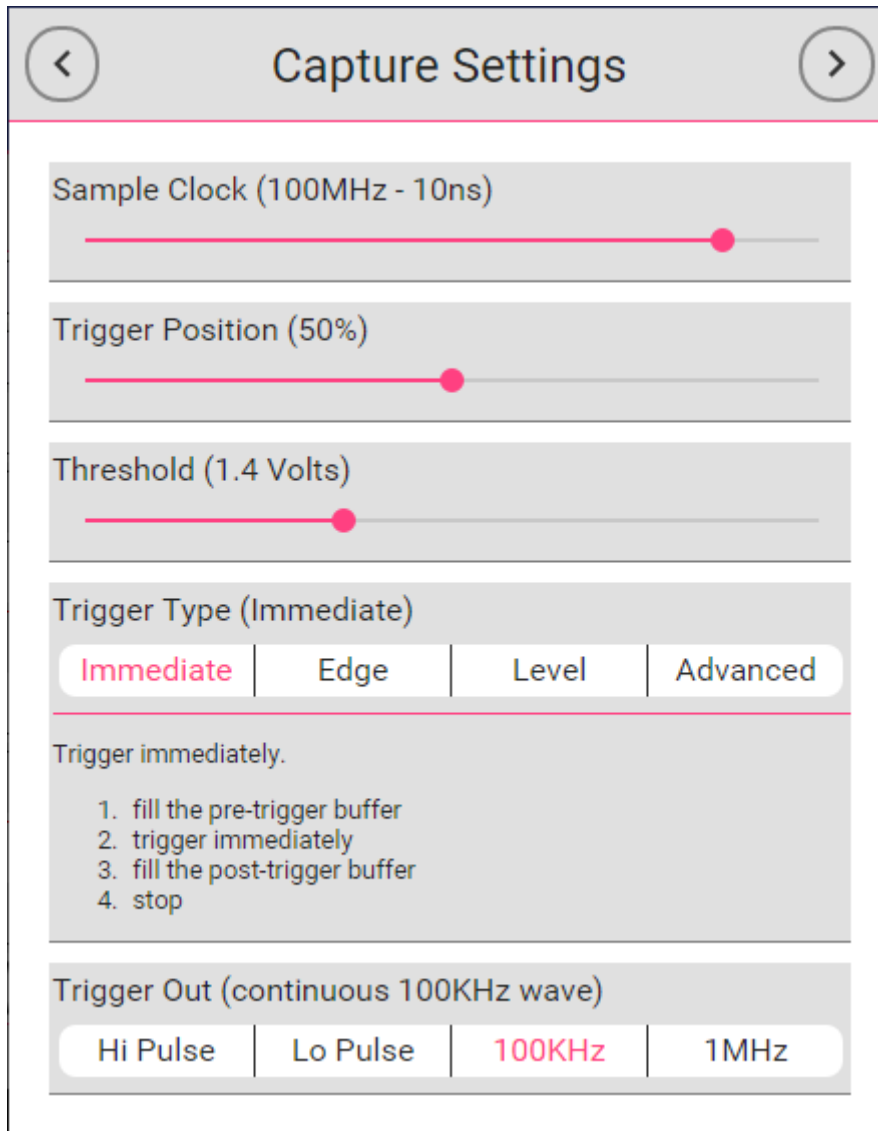
The five navigation buttons jump immediately to the first, trigger, and last samples, and to markers M1 and M2.

Dialogs

The *Settings* button brings up the last active dialog. The left and right arrows at the top of each dialog move between dialogs. Clicking outside a dialog will dismiss it.

Capture Settings

Usually these are the only vital settings. The default values cover many common cases.



The sampling clock is generated within the instrument. It is effectively a divide down from a master clock running at 800MHz or 1GHz. The default clock setting is 100MHz.

The trigger position slider divides the acquisition memory into the pre-trigger part and the post-trigger part. The pre-trigger part of memory is the minimum amount which must be filled with data samples before the logic analyzer starts searching for the trigger pattern. The post-trigger part of memory is filled after the trigger pattern is detected; when the post-trigger memory is full the analyzer stops. The trigger position can vary from 5% to 95%, in increments of 5%. The default is 50%.

The threshold slider selects the 0->1 and 1->0 threshold voltage. The threshold can be varied from 0.8Volt to 2.5Volt in steps of 0.1Volt. The default is 1.4Volt.

Signals and Buses

The trigger type can be chosen from:

- Trigger immediately. This is the default setting.
- Edge - trigger when an edge on **one** of the incoming signals matches the Edge (E) pattern recognizer. *Don't Care* selections in the E recognizer are ignored.
- Level - trigger when the levels on **all** the incoming signals match the Level (L) pattern recognizer. *Don't Care* selections in the L recognizer are ignored.
- Advanced - triggering is selected by the settings in the *Advanced Triggering* dialog. The default setting for Advanced Triggering corresponds to Edge triggering.

The *trigger out* signal can be chosen from:

- a Hi Pulse - the output is normally low and pulses high for approximately 250ns.
- a Lo Pulse - the output is normally high and pulses low for approximately 250ns.
- a continuous 100kHz signal. This is the default setting.
- a continuous 1MHz signal.

Signals and Buses

The *Signals and Buses* dialog is used to change, add, or delete the signal rows in the waveform display.

The screenshot shows the "Signals & Buses" dialog box. At the top, there are back and forward navigation arrows. The main content area is divided into several sections:

- Select a Signal or Bus:** A dropdown menu currently displays "i2cDemo".
- Signal/Bus Type (I2C):** Three buttons are present: "Signal or Bus", "I2C" (which is highlighted in red), and "SPI".
- Signal Wire(s):** A text input field contains "1..0".
- Display Color:** A color selection area showing a black swatch.
- Display Radix (hexadecimal):** Five buttons are shown: "Binary", "Octal", "Decimal", "Hex" (highlighted in red), and "ASCII".
- Buttons:** At the bottom, there are two buttons: "ADD SIGNAL" and "DELETE SIGNAL".

Select a Signal cycles through the defined signals. The signal or bus name can be edited. Alternatively, clicking the *Add* button puts a new signal on the end of the list. A new signal will start as a single bit (a single wire).

Advanced Triggering

For the *Signal/Bus Type* you can select between a regular parallel *Sig/Bus*, *I2C*, and *SPI*. *I2C* and *SPI* selections have no effect on triggering but they control the decoding of captured data.

Signal Wires. The signal wire(s) that make up a particular bus are defined via a scheme illustrated in the following examples:

- 1 : wire 1. A single wire signal.
- 1,2 : wires 1 and 2, with wire 1 being the most significant. This is the same as 1..2
- 2,1 : wires 2 and 1, with wire 2 being the most significant. This is the same as 2..1
- 3..0 : wires 3,2,1,0, with wire 3 being the most significant
- 0..3 : wires 0,1,2,3, with wire 0 being the most significant
- 7..5,1,0 : wires 7,6,5,1,0, with wire 7 being the most significant

For an *I2C Bus* specify two wires. For instance 1,0. The least significant wire (0 in this example) will be used as the *SDA* wire. The next wire (1 in this example) will be used as the *SCK* wire. There must be two wires.

For an *SPI Bus* specify three wires. For instance 4,3,2. The least significant wire (2 in this example) will be used as the data wire - it could be *MISO* or *MOSI*. The next wire (3 in this example) will be used as the *SCLK* wire. The next wire (4 in this example) will be used as the active low *SS* wire. There must be three wires. Also select the significant edge of *SCLK* (rising or falling - rising is the default) and whether data bytes are decoded MSB first or LSB first (MSB is the default).

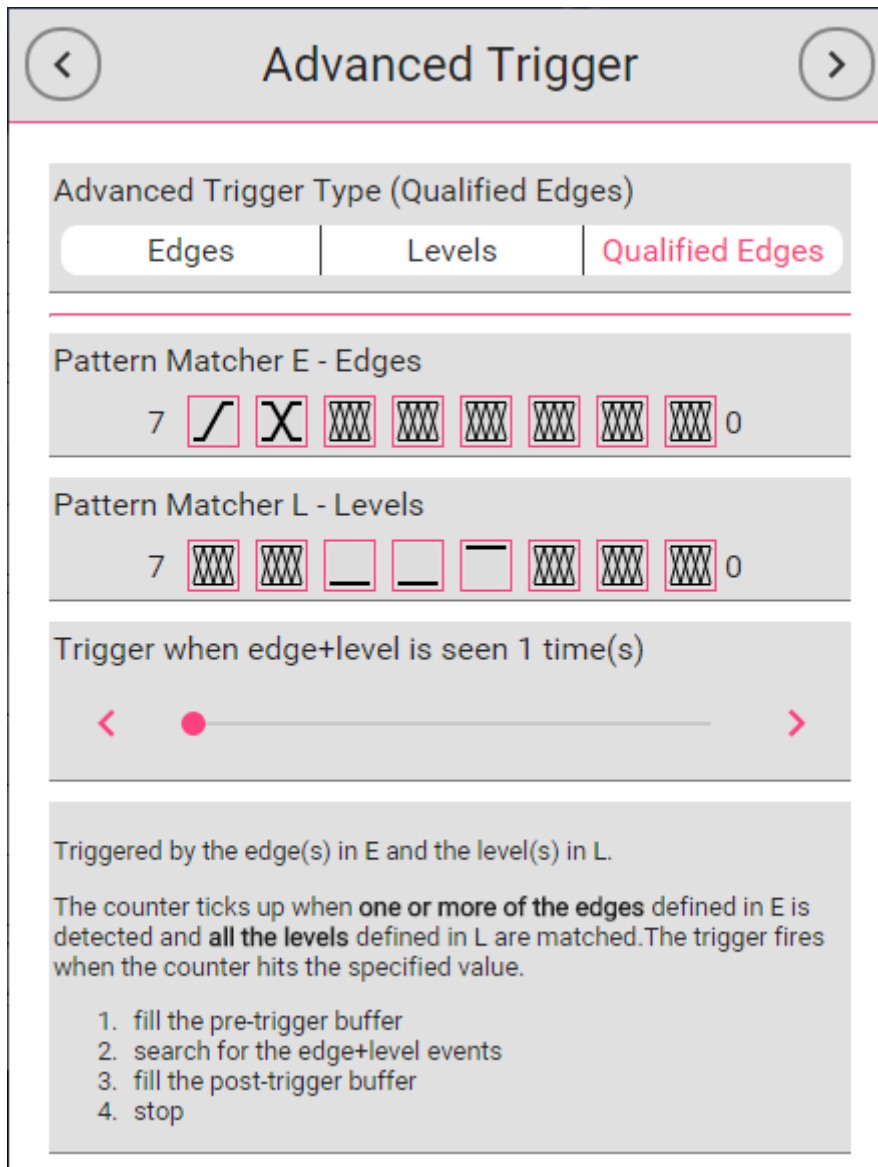
Click on *Display Color* to see a color selection popup. The display color is used to draw the waveforms in the main display.

The *Display Radix* is used in the main display. Binary, Octal, Decimal, and Hexadecimal values display as you would expect. ASCII values up to 127 display the usual printable characters and control codes. Non-printable ASCII values from 128 display hexadecimal values.

The *Delete Signal* button removes the current signal unless it is the last defined signal. This action cannot be undone.

Advanced Triggering

These triggering options are available if the *Advanced* trigger is selected in the *Capture Settings*.



Triggering is controlled by two pattern detectors and a counter.

- the *E* pattern detector searches for edges. Valid edge conditions can be *rising*, *falling*, *either*, or *don't care*. The detector fires when **one** of the edge conditions is matched, with the *don't care* condition ignored.
- the *L* pattern detector searches for levels. Valid level conditions can be *high*, *low*, or *don't care*. The detector fires when **all** of the level conditions are matched, with the *don't care* condition ignored.
- the counter

Advanced Trigger Type shows the available trigger types.

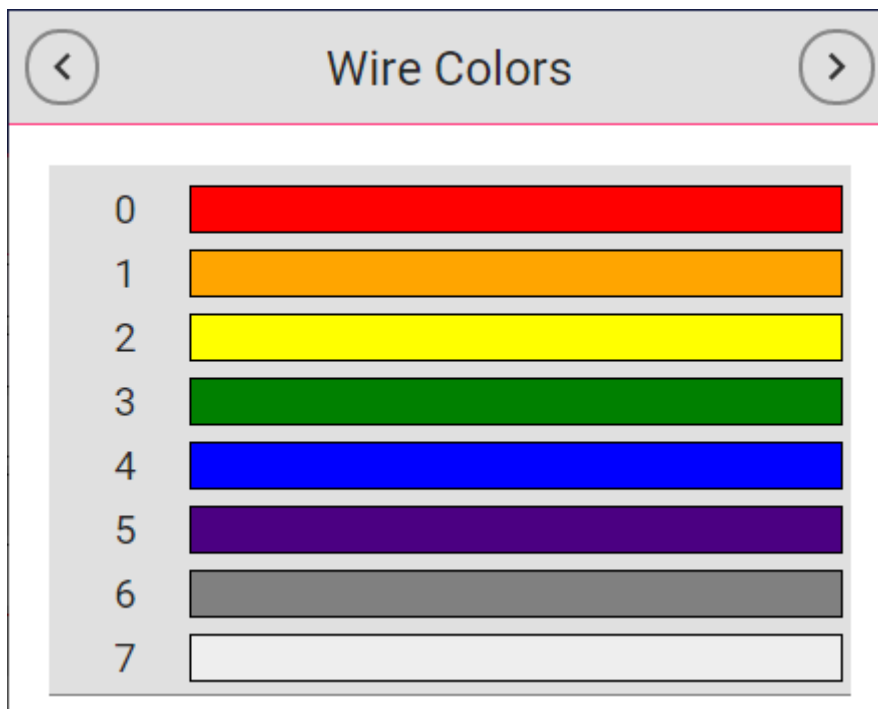
- *Edges* The instrument is triggered by hits on the E detector. The counter ticks up with each hit and the trigger fires when the counter matches the specified value.
- *Levels* The instrument is triggered by hits on the L detector. The counter ticks up while L fires (i.e. all the levels are matched simultaneously), resetting to 0 if L fails to fire. The trigger fires:
 - *greater than* - when the counter hits the specified value. This corresponds to searching for a pattern that lasts for at least the time defined by the counter value.

Wire Names and Colors

- *equals* - when the counter hits the specified value and immediately resets. This corresponds to searching for a pattern that lasts for exactly the time defined by the counter value.
- *less than* - when the counter resets before it hits the specified value. This corresponds to searching for a pattern that lasts for less than the time defined by the counter value.
- *Qualified Edges* The instrument is triggered by simultaneous hits on the E and L detectors. The counter ticks up when one of the edges defined in E is detected and all the levels defined in L are matched. The trigger fires when the counter hits the specified value. This corresponds to searching for an edge that occurs when a pattern is active - for instance for a write strobe when address bits are valid

Wire Names and Colors

The colors used to represent the probe wires correspond to the standard arrangement of colors on a *rainbow* ribbon cable. You can use this dialog to change the color assigned to a wire.



Configuration File

The app uses two types of storage for configuration data - HTML5 *Local Storage* and regular disk files. It stores configuration settings in the same format in both locations.

In *Local Storage* the configuration settings are stored with *bx.ini* as the key. In disk storage configuration settings can be stored with any name you choose, *bx.ini* is the default.

Startup

When the app starts it looks in *Local Storage* for configuration settings named *bx.ini*. If this configuration cannot be found a default configuration is used. You can override the choice by reading a new configuration file from disk.

The current configuration settings are not saved automatically on exit. Instead there is a *Quick Save* button which saves the current configuration to Local Storage. This configuration will then be read back the next time the app starts.

Example

The configuration file is a standard .INI file with sections as shown below. It is not case sensitive.

For the *signals* section the keys are

- name: no default. There are no restrictions on signal names, but if you want to restore a configuration the name must not have embedded blanks. "hello_world" is OK, "hello world" is not.
- val: the default is 0.
- type: reserved types are i2c and spi. Any other type is interpreted as a standard parallel bus.
- fg: foreground color, defined with the usual web rules. The default is black.
- base: *bin* or *oct* or *dec* or *hex* or *asc*. The default is binary.
- spiclockrising: the default is *true*.
- spimsbfirst: the default is *true*.

For the *trigger* section the keys are

- speed: the default is 100MHz.
- triggerpos: 5 to 95. The default is 50.
- threshold: threshold voltage x 10. The default is 14 (ie. 1.4V).
- triggertype: *imm* or *edge* or *level* or *advanced*. The default is immediate.
- advtriggertype: *edge* or *level* or *qual* The default is edge.
- counter: 1 to 1023. The default is 1.

```
[BX]
BXProgramVersion=18 August 2015
ConfigWriteDate=19 Aug 2015
ConfigWriteTime=13:08:31
Module=BXA3T6GVNE
WireCount=8

[WireColors]
0=red
```

Configuration File

```
1=orange
2=yellow
3=green
4=blue
5=indigo
6=gray
7=#eee

[Signals]
i2cDemo=val:1..0,type:i2c,base:hex
spiDemo=val:4..2,type:spi,base:hex
sig0=val:0
sig1=val:1
sig2=val:2
sig3=val:3,fg:red
sig4=val:4
sig5=val:5
sig6=val:6,fg:red
sig7=val:7,
LowByte=val:7..0,base:hex
HighByte=val:15..8,base:hex

[Trigger]
Speed=100MHz
TriggerPos=50
Thresholdx10=14
TriggerType=imm
AdvTriggerType=edge
Counter=5
```

Data File

Sample data is be saved in native (BXD) format.

The data files are run-length compressed. Identical consecutive data samples are represented by a (value, count) pair, known as a slot. For instance, if the first two slots contain (data=55, count=3), (data=16, count=2), the first five samples will be 55, 55, 55, 16, 16. The repetition count can range up to 250,000 for the BX8/BX8u.

The trigger position (TriggerPos) is a sample number, not a slot number.

Here is an example of the first few rows of an BXD file. There is one data+count value per acquisition slot. Data and count values are in decimal.

This format is a version of a standard .INI file. It is not case sensitive.

```
[BX]
...
...
; the same as the BX section of the configuration file
...
...

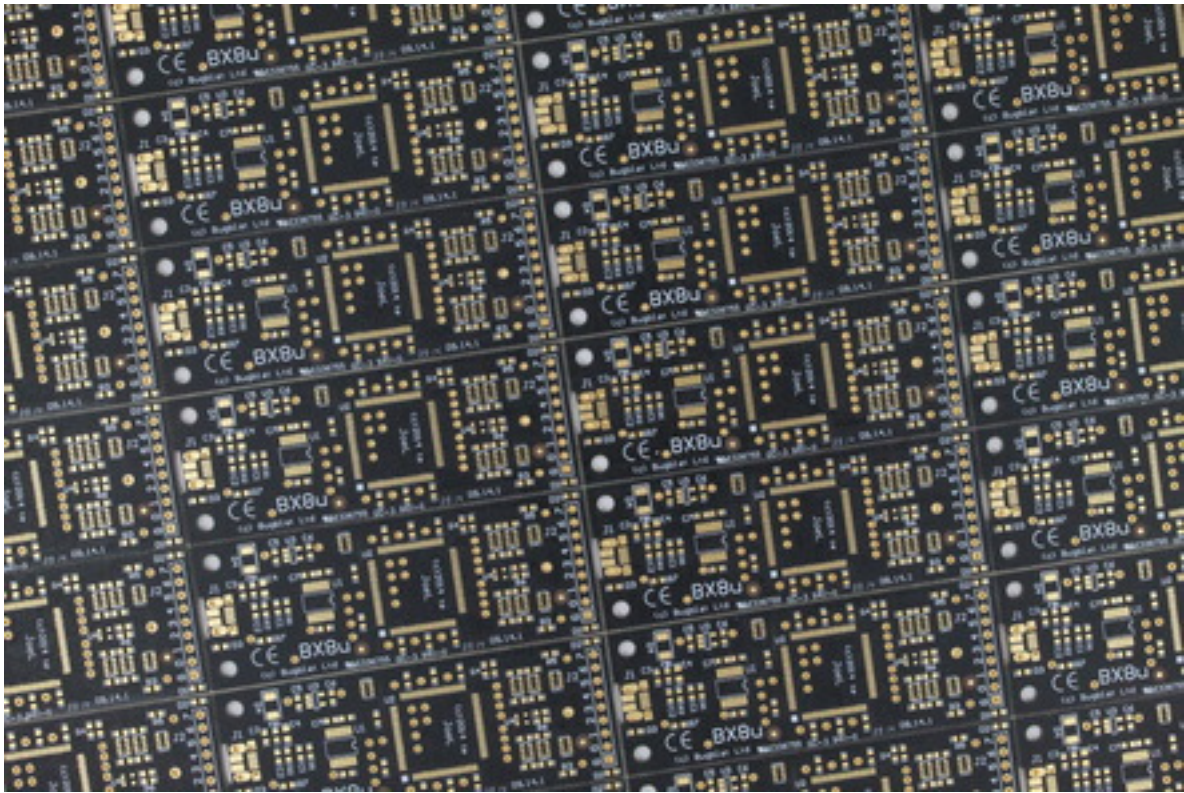
[Signals]
...
...
; the same as the signals section of the configuration file
...
...

[Trigger]
...
...
; the same as the trigger section of the configuration file
...
...

[Data]
0=23,18
1=55,23
2=87,2
3=119,8
4=151,8
5=183,19
```

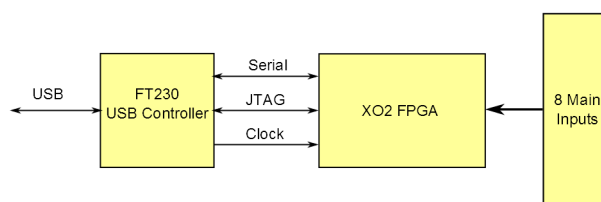
BX8 Hardware

The hardware in the bare-board version of the BX8 is the BX8u circuit board.



This is a brief introduction to the BX8u board. None of this information is required to run the logic analyzer software on the BX8. The BX8u is functionally identical to the Bugblat *Fan* board and much more detailed information is available here: [fan](#)

Block Diagram



Functional Description

The key components of a Bugblat *BX8u* board are

- XO2-2000HC: a Lattice Semiconductor MachXO2 FPGA (details at [Lattice](#))
- FT230X: an FTDI USB/UART interface (details at [FTDI](#)).

The XO2-2000HC FPGA contains 2112 four-input lookup tables (LUTs), on-chip SRAM and Flash, a flexible PLL, and a host of other features.

The FT230X is a USB to serial UART interface that drives the XO2's control pins.

Connectors

J2 - 9-pin single-row Logic Analyzer Inputs

Pin 1, indicated by a square pad, is a ground reference.

Pins 2 to 9 are the Logic Analyzer inputs, connected to the FPGA inputs via a protection circuit as shown on the schematic.

JJ1 - Trigger Out Loop

The *trigger out* loop is driven by a 74LVC04 inverting buffer, via a 200R series resistor. It can be used for any function programmed into the FPGA - the logic analyzer firmware uses the loop as a *trigger out* and *test clock* connector.

Legal Stuff

BX Logic Analyzers are for inquisitive minds with a basic understanding of electronics. You know what that means.

You assume full responsibility to determine and/or assure compliance with any regulatory and safety compliance standards and related certifications as may be applicable. You will employ reasonable safeguards to ensure that your use of a BX Logic Analyzer will not result in any property damage or injury or death even if the product should fail to perform as described or expected.

BX8u Note

The hardware in the bare-board version of the BX8 is the BX8u circuit board. The BX8u board is not a complete product and may not meet all the regulatory and safety compliance standards which may normally be associated with similar items.

The BX8u was designed as a logic analyzer plus an evaluation and development tool. It was not designed with any other application in mind. As such these design materials may or may not be suitable for any other purposes. If any design material is used it becomes your responsibility as to whether it meets your specific needs or the needs of your specific applications and the design material may require changes to meet your requirements.

Design Materials

Design materials referred to in this document are **not supported** and do **not** constitute a reference design.

To the extent permitted by applicable law there is no warranty for the design materials. Except when otherwise stated in writing the copyright holders and/or other parties provide the design materials as is without warranty of any kind either expressed or implied including but not limited to the implied warranties of merchantability and fitness for a particular purpose. The entire risk as to the quality and performance of the design materials is with you. Should the design materials prove defective you assume the cost of all necessary servicing repair or correction.

(2 September 2015)