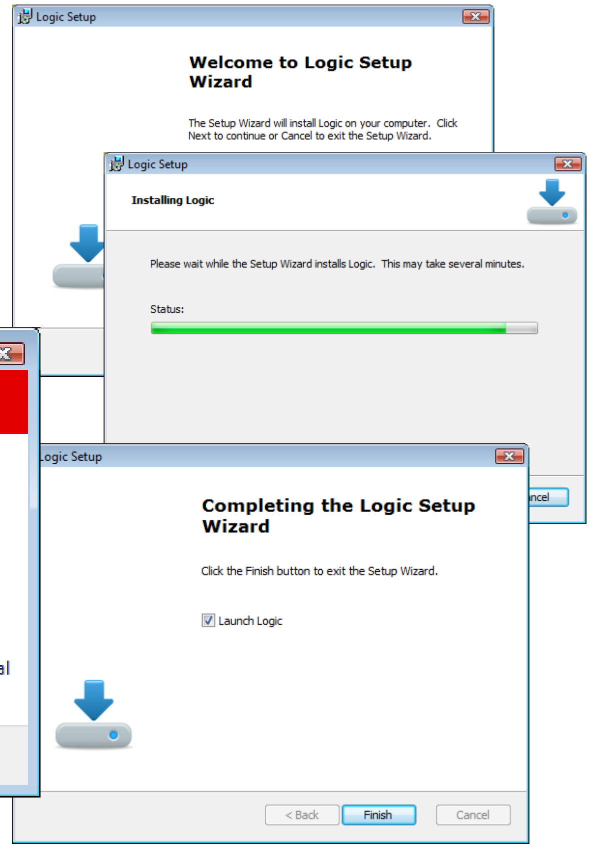
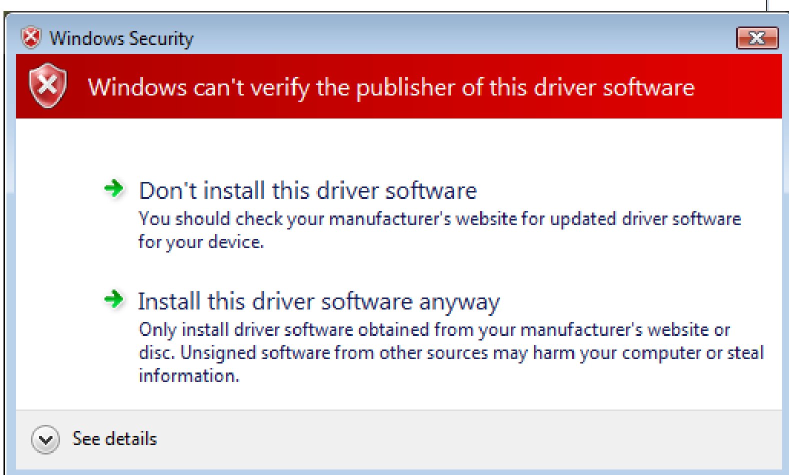
Table of Contents

INSTALL THE LOGIC SOFTWARE

- The latest software can be downloaded at http://www.saleae.com/downloads
- You will be prompted for whether or not you want to install unidentified driver software. **You must install the driver** for the installation to succeed. Rest assured that the driver is good (it's an unaltered version of a certification-ready driver provided by Cypress Semiconductor). If you are only installing temporarily to try out the software, don't worry; the driver will be removed when the application is uninstalled.
- After the software is installed, you can launch it from the desktop shortcut or the Start menu under





Start->All Programs->Logic->Logic. You can also uninstall the software from the same location (*Start->All Programs->Logic->Uninstall*).

CONNECTING TO THE CIRCUIT UNDER TEST

Logic comes with a 1x9 wire bundle. This is to assure that wire colors match up in the user interface. However, it can be plugged in two different ways, so it is important to have it plugged in the correct way.

- **Black wire = Input 1**
- **Gray wire = Ground**

There are three conditions to properly connect Logic to a circuit under test:

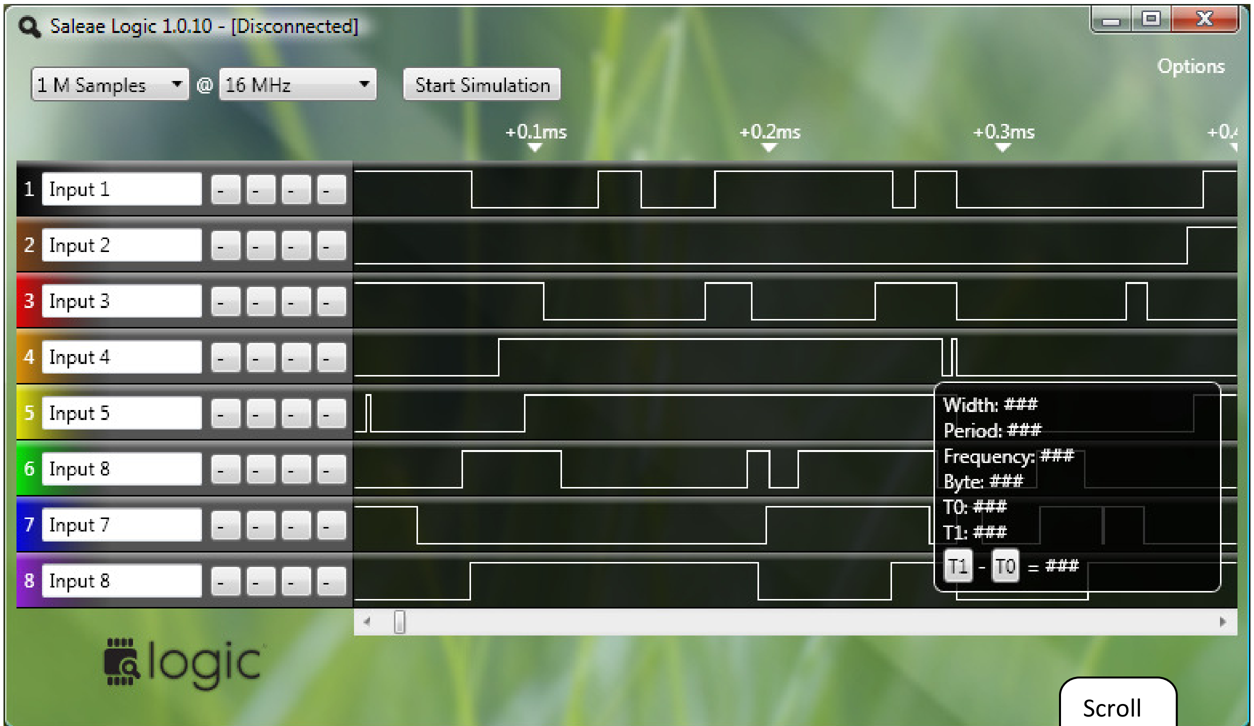
- You must connect the ground wire (gray) to the ground of the circuit under test.

- The input wires must be connected to voltages that do not exceed 5.25V or fall below -.5V to prevent damage to Logic.
- The digital frequency of the signals you are sampling should not exceed the sample rate you are using.
    - Note that generally, for accurate width measurement, the sample rate should be many times the measured frequency.
    - If the sample rate and the measured rate are very close, you may observe periodic anomalies in the data. This is due to the slightly different frequencies shifting in and out of phase.

USING THE SOFTWARE

- When you first start the Logic software, you will see something like the view above.
- At the top left, you will see if the Logic hardware is connected or not.
    - If Logic is not connected, the Start button will read “Start Simulation.” You are in simulation mode.
    - If Logic is connected, the Start button will read “Start.” Data will be collected from Logic.
- In the first drop-down, you can select the number of samples you would like to collect.
- In the second drop-down, you can select the sample rate you want to collect at.
- Notice that the colors of the inputs 1-8 correspond to the colors of Logic’s wires. Make sure your wire set is plugged in the right way— The Black wire is Input 1, and the Gray wire is Ground.
- If you like, you can type your own input names in the text boxes labeled 1 through 8.
- Press the Start button to collect data and display them.
- While data are collecting, you can press stop if you change your mind.

Exploring Your Data



- To pan back and forth:
  - Click and drag the graph with the left mouse button **OR**
  - Use the slider **OR**
  - Flick the graph with the mouse so it slides by. Click again to stop the motion.
- To zoom in and out:
  - Use your mouse's scroll wheel **OR**
  - Single left-click to zoom in, single right-click to zoom out
- To get instant timing information:
  - Hover your mouse pointer over the part of the graph you're interested in
  - You may want to zoom in or out for a clearer picture

Using Timing Markers

- Scrolling Timing Marks
  - These allow you to instantly infer the zoom level and give context to what you're looking at.
  - They are located along the top of the graph.
  - They are aligned to the trigger start point, time 0.
  - These will adjust dynamically based on your zoom setting.

  - Every 10th timing mark (shown higher than the others) indicates the **absolute** time since the trigger.
  - The lower (9 out of 10) timing markers are the **relative** time difference from the last absolute time. These markers have a '+' in front of them.
- User-Positioned Timing Markers
  - User-positioned timing markers are used to get the exact time between two events anywhere in your data.
  - To use these, click the "T1" or "T0" buttons on the meta-data panel (bottom right of the graph). (or use the keyboard shortcuts CTRL-0 and CTRL-1).
  - To cancel placing a timing marker, move your mouse out of the graph area.
  - To remove a timing marker from the graph, click it, or its button, and then move your mouse out of the graph area.
  - To move a timing marker, click it, or its button, and then re-position. Click again to place.
  - To snap to a signal edge, move your mouse near the edge you want. You may need to zoom in for greater accuracy.
  - The time difference between your timing markers T0 and T1 will be displayed as long as both timing markers have been set.

Using the Trigger

- Many times you'll want to capture data only after some event has happened. This makes it much easier to find what you're looking for in the data.
- By default, the trigger is all "don't care's" as indicated by the dashes in the trigger drop-down boxes.
- To set an individual bit of the trigger, use its drop-down box to select '0,''don't care (-),' or '1.'
- There is a pre-trigger buffer that will fill up with all the data preceding the trigger.
  - You can set the size of this buffer from the Options menu.
  - Note that this buffer may or may not fill up all the way depending on how long it takes the trigger to be satisfied.
- For convenience, a "Reset the Trigger" option is available from the Options menu. This will set all the trigger definition bits to 'don't care.'
- **How the trigger works:**
  - The trigger is defined by four 'trigger definition bytes.' Bits in these bytes are 0, 1, or 'don't care.'
  - Logic continuously samples the incoming data, one byte at a time.
  - The trigger starts out comparing its first trigger definition byte with the new incoming data.
    - When the incoming byte matches with the trigger definition byte, the trigger moves to the next trigger definition byte.
    - When the incoming byte doesn't match either the current definition byte or the preceding one, the trigger is reset; i.e., it goes back to the first trigger definition byte.
    - When all the trigger definition bytes have been satisfied, the trigger is satisfied, and data collection starts.

Saving a Session

You can save a session for later. Sessions include all the data you need to start over from where you left off, including:

- Your sampled Data
- Your graph position, zoom level, and timing markers
- Your analyzer settings
- Your window position and size
- Your sample rate, sample depth, trigger settings, and input labels.

To save your session, just click Options->Save Session.

Loading a Session

You can load a session at any time. Note that when you load a session, your previous settings will be lost (unless you saved them to their own session). When a session is loaded, the Logic Software will restart with the new settings. There are three ways to load a session:

- Select Options->Load Session
- Drag a session data file into the Logic Software
- Double-click a session file. (Make sure Logic is not running when you do this).

Emailing a Session

Session files are compressed, so they are convenient way to send your data over email. Anyone with the Logic Software will be able to open them and see exactly what you see. Also, if you need to report an issue with the Logic software, you may want to send us a session file so we can also see the problem.

Exporting Data

Exporting CSV Data

You can export your data in a CSV format from the Options menu:

- Options->Export Data->CSV
- Options->Export Data->CSV (Between T0 and T1)

Note that versions of Excel prior to 2007 have a limit on the number of data they can import: ~65K rows. The number of data you export will exceed this.

The CSV format of the exported data is as follows:

- Byte in hex, bit7, bit6, bit5, bit4, bit3, bit2, bit1, bit0

Note that exported data can not be re-imported. (Use Save Session to do that). Also, note that exported data does not contain the sample rate. We recommend including the sample rate in the file name.

Exporting Binary Data

You can export your data in a binary format from the Options menu:

- Options->Export Data->Binary
- Options->Export Data->Binary (Between T0 and T1)

The binary format is simply one byte for every byte recorded.

Note that exported data can not be re-imported. (Use Save Session to do that). Also, note that exported data does not contain the sample rate. We recommend including the sample rate in the file name.

Exporting Analyzers

You can export your analyzer data. It will save as a CSV text file. The exact format depends on the analyzer.

- Options->Export Data->Analyzer 1
- Options->Export Data->Analyzer 2

Note that exported data can not be re-imported. (Use Save Session to do that).

Saving a Screenshot

You can also save a screenshot of Logic. This is provided in the Options menu:

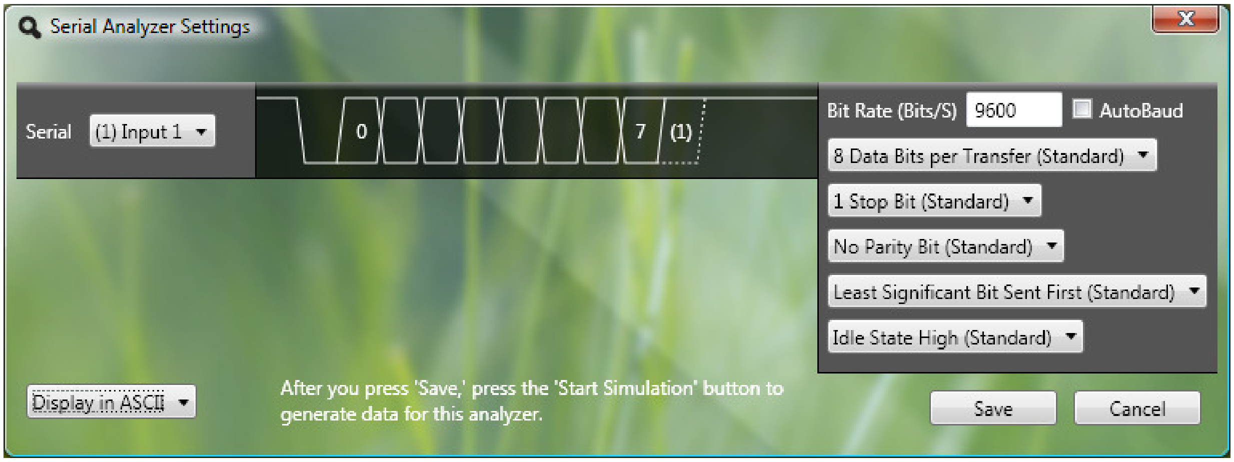
1. Options->Save Screenshot

Protocol Analyzers

Simulating with Protocol Analyzers

To simulate using given protocol analyzer, simply re-run the simulation after setting up that analyzer. For example, after configuring the serial analyzer, click "Start Simulation" to generate simulated serial data with the properties corresponding to the settings you specified.

Serial Analyzer



Settings

1. Input selection
2. Display Selection (Octal, Decimal, Hexadecimal, or ASCII)
3. Bit Rate (integer). Or, use the AutoBaud feature!
    - AutoBaud works by measuring the shortest active-going pulse in the serial data, and using that as the bit-width. (i.e. sets the Bit Rate to 1/bit-width)
    - Note that this assumes you have enough data sampled to have many serial bytes, one of which has at least one 1-bit-with active pulse. If not, don't use AutoBaud.
4. Bits per transfer. This will probably almost always be 8 or 7, but other values are available.
5. Stop Bit. This can be 1, 1.5, 2 or 3.
6. Parity. If you are using parity, you set this to "Even" or "Odd" to have Logic check for parity errors.
7. Bit Order. This is normally Least Significant Bit first, but you have the option.
8. Idle State. This is normally high, but is selectable.
    - Note that this is not the same as inverting your serial signal. Only the idle state, start, and stop bits are effected. Data bits use regular logic (i.e. 0=low, 1=high)

How it Works

The serial analyzer works the same way as a typical bit-banged serial implementation.

- The analyzer waits for the starting edge of the start bit.
- The analyzer waits 1.5 * bit-width, and then samples the first data bit (i.e. the theoretical center of the first data bit)
- The analyzer then repeats this, waiting one bit-width and then sampling (still in the theoretical center of the data bits)
- Lastly the analyzer samples from 1/2 bit width into the stop bit until 1/2bit from the theoretical end of the stop bit. If this isn't as expected it reports a framing error.

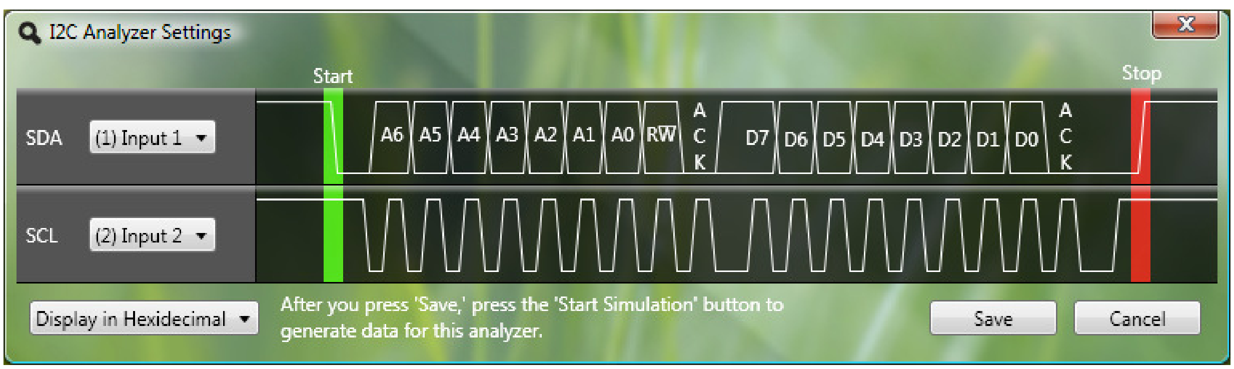
Timing Error Tolerance

By its nature asynchronous serial is tolerant of some degree of timing error. From the above you can compute the exact tolerance, but for typical 8N1 (8 bits of data, no parity, 1 stop bit) the error may be as high as 5% and still decode correctly.

Graphical Display of Sample Timing

If you are sufficiently zoomed in on your serial, the serial analyzer will display diamonds indicating the location where it is sampling – the theoretical center of your data bits. This allows you to immediately gauge how far off your serial is from the bit rate you have specified for the serial analyzer.

I2C Analyzer



Settings

Because I2C is both synchronous and fairly rigidly defined, setup is easy. You just need to specific the inputs providing your clock and data lines.

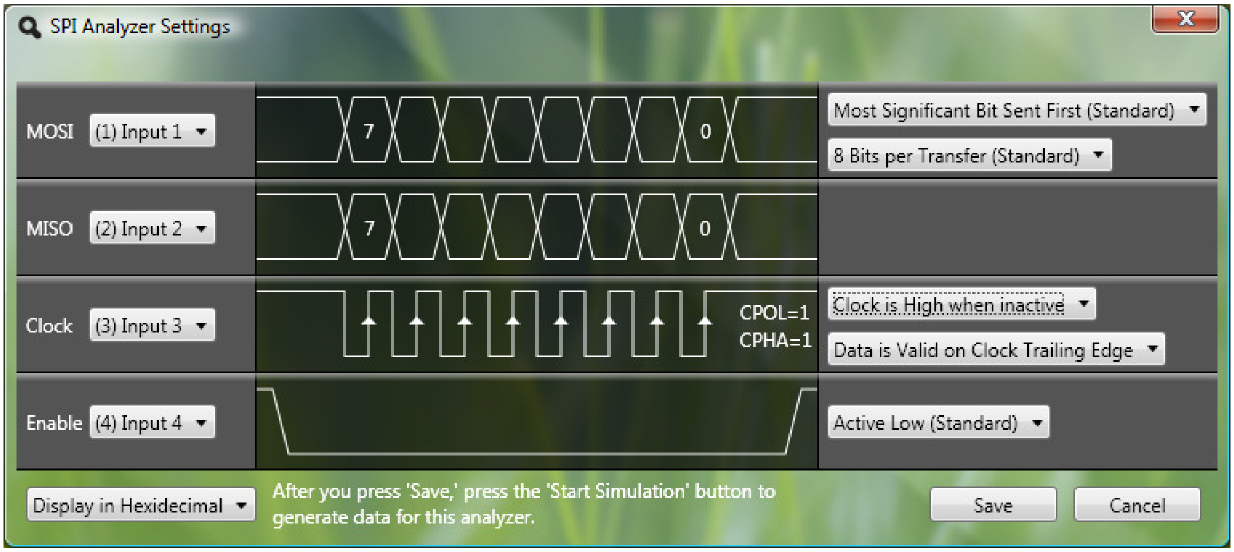
Synchronization

The I2C Analyzer waits for a valid I2C start condition before performing I2C decoding.

I2C Information

For information about I2C, you may want to review the Wikipedia article or the Philips specification here and here: http://en.wikipedia.org/wiki/I%C2%B2C, http://www.nxp.com/acrobat_download/literature/9398/39340011.pdf

SPI Analyzer



Settings

1. Select the inputs for MOSI (Master Out, Slave In), MISO (Master In, Slave Out), Clock (SCK), and Enable (SS, or Slave Select). Note that you can select 'none' for MISO.

2. Select the display format: Octal, Decimal, Hexadecimal, or ASCII.

3. Bit Order: Normally this is most significant bit first, but you can choose.

4. Bits per transfer: Normally this is 8, but you can select a different value.

5. Inactive Clock level. Specifies the polarity of the clock when inactive.

6. Data Valid Clock edge. Specifies the edge of the clock that corresponds to the data being valid (readable)

Synchronization

The SPI Analyzer waits for a valid inactive->active state transition on the Enable line before performing SPI decoding. In addition, SPI information is only decoded when the enable line is active. If the enable line becomes inactive before the current transfer is complete, that transfer is ignored.

SPI Information

For information about the different SPI settings, you may want to review the Wikipedia article here: http://en.wikipedia.org/wiki/Serial_Peripheral_Interface_Bus

Data Integrity

Logic may not be able to achieve a particular sample rate (see the next section) but it will never present incomplete data, or data with samples missing. By design it stops and notifies you as soon as any data is lost.

Maximum Sample Rate

Generally speaking Logic is able to achieve sample rates of 24MHz when no other devices are on the USB host, and your computer is responding sufficiently quickly. However if the USB is servicing other devices, or the computer is performing in a latent manner, a smaller sample rate, such as 16MHz or 12MHz may be the fastest achievable. The reason for this is as follows:

Logic uses USB 2.0, and uses the USB transfer type known as "Bulk". While this transfer type has the largest theoretical average bandwidth (significantly larger even than the 24 MB/s needed by Logic) it also has the lowest priority. This means it may be "bumped" for other USB traffic from other devices. The second issue is that Logic has very limited device-side memory. Specifically it has x4 512 byte buffers. These buffers must be emptied by the USB in such a way that they never all become full at the same time. If this happens, the logic software will report that the sample rate could not be maintained.

This means that even if on average (which would normally be the case) the USB can give Logic at least 24 MB/s, it must not prioritize other devices on the USB for so much time that Logic's small buffer would overflow. For this reason Logic can't guarantee that it will operate at its maximum 24MHz, as this is contingent on a number of factors including computer performance, USB bandwidth availability and latency, other drivers that may be using the USB, etc.

To maximize the sample rate possible on your computer, try the following:

- Make sure that no other applications are consuming significant CPU time.
- Make sure you have enough free RAM so that Windows will not have to use the hard drive to allow Logic to collect data. This will be too slow.
- Try connecting Logic directly to your computer instead of through a USB hub.
- Make sure other devices on the USB are minimally active.
    - To tweak a higher sample rate out of Logic, you might try disconnecting other peripherals from the USB.

Contacting Us

We would love to hear from you regarding your experiences with Logic, or if you need any assistance. Please visit http://www.saleae.com/contact.