

Saleae Logic 4 USB Logic Analyzer



Key Features

Feature	Description
Channels	Four digital inputs, one analog input (shared with digital channel 0)
Sample Rates	Digital: 12 MSPS, 6 MSPS, 3 MSPS, 1 MSPS Analog: 6 MSPS, 125 KSPS, 5 KSPS, 1 KSPS, 100 SPS, 10 SPS
Bandwidth	Digital: 3 MHz square wave when sampling at 12 MSPS Analog: 600 kHz when sampling at 6 MSPS
Trigger	Edge or pulse width trigger
Software	Cross Platform Windows, Mac and Linux support
Supported Protocols	Serial, SPI, I2C, Atmel SWI, Biss-C, CAN, DMX-512, HD44780, HDLC, I2S, JTAG, LIN, MDIO, MIDI, Manchester, Modbus, 1-Wire, PS/2 Keyboard & Mouse, SMBus, SWD, Synchronous Parallel, USB, UNI/O
Capture Buffer Length	Limited by installed memory and digital data density. When recording analog at 6 MSPS, captures 30-60 seconds usually possible. Without analog, buffer length is dependent on digital activity density. 1 MHz SPI: up to 10 Minutes. I2C – several hours – 9600 baud serial – over 1 day.
Analog Resolution	8 bits, 19.5mV per LSB
Analog Input Range	0V to 5V
Supported digital IO standards	2.5V, 3.3V, 5.0V, RS-232, +12V TTL

What is a Logic Analyzer?

Logic analyzers are great for debugging embedded applications.

They operate by sampling a digital input connected to a device under test (DUT) and then displaying the recording on your computer.

This is great for debugging a wide range of embedded problems. For instance, If an I2C device isn't responding, you can record SDA and SCL to see if the start condition, device address, and ACK/NAK bit frame look right. It's a huge help for any project with a microcontroller, FPGA or ARM chip.

Easy-to-use Software

The Saleae Logic devices connect to your PC over USB. Just download the software at www.saleae.com.



Navigate your data easily and intuitively with Logic's fluid and fully animated mouse-driven interface. Drag the display with the mouse, zoom with the scroll wheel, even toss the display to find nearby events.

Protocol Decoding

The Saleae products support decoding for over 20 different protocols. This means when you record those SPI messages, or a CAN bus, you won't need to stare at transitions and clock edges. Instead, Just add the protocol decoder and read the data entire bytes – or even transactions at a time.

