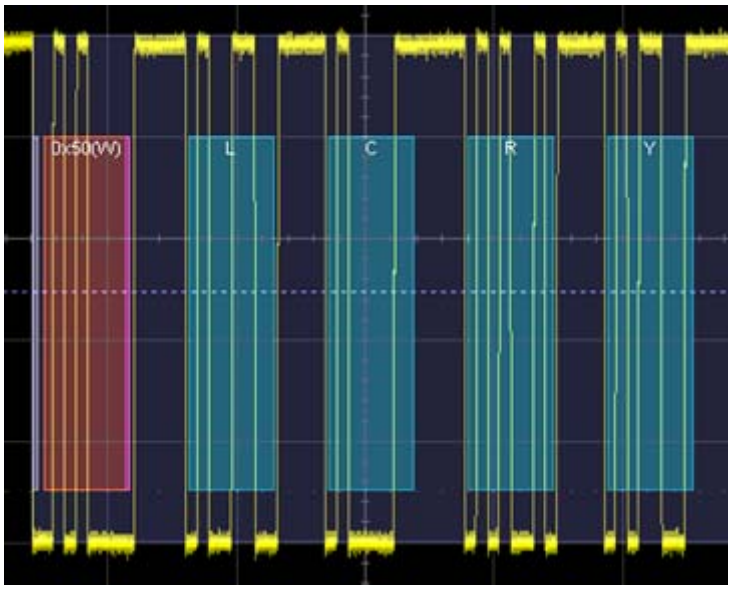
LeCroy

Low-Speed Serial Data Trigger and Decode

Software Options



Operator's Manual

November, 2008

LeCroy Corporation
700 Chestnut Ridge Road
Chestnut Ridge, NY, 10977-6499
Tel: (845) 578-6020, Fax: (845) 578 5985

Internet: www.lecroy.com

© 2008 by LeCroy Corporation. All rights reserved.

LeCroy and other product or brand names are trademarks or requested trademarks of their respective holders. Information in this publication supersedes all earlier versions. Specifications are subject to change without notice.

916855 RevA

TABLE OF CONTENTS

LSSD Introduction 6
The TD Series Software 7
The D Series Software 8
Technical Explanation of D and TD Options 9
Overview 9
Serial Trigger 9
Serial Decode 9
Table Display 10
Accessing the TD Toolsets 10
Overview 10
Accessing Serial Trigger Dialogs 10
Serial Decode and Decode Setup 10
Serial Decode (Summary) Dialog Box 11
Decode Setup Dialog Box 13
Protocol Results Table 15
Searching for Messages 16
Using the I^2Cbus Options 17
Overview 17
Accessing Serial Triggers 18
Creating an I^2Cbus Trigger Condition 19
I^2C Trigger Setup Detail 19
1. Sources Setup 19
2. Trigger Type 20
3. Setup Format 20
4. Address Setup 20
5. Data Pattern Setup 21
6. Ack Setup 23
I^2C Decode Setup Detail 24
Using the SPIbus Options 25
Overview 25
Accessing Serial Triggers 25
Creating a SPIbus Trigger Condition 26
SPIbus Trigger Setup Detail 26
1. SPI Type Selection 26
2. Source Setup 27
3. SPI Format Setup 27
4. Setup Format 28

5. Data Pattern Setup 28
6. InterFrame Setup 28
SPIbus Decode Setup Detail 29
Using the UART-RS232bus Options 30
Overview 30
Accessing Serial Triggers 31
Creating a UART-RS232bus Trigger Condition 31
UART-RS232bus Trigger Setup Detail 32
1. Sources Setup 32
2. UART Setup 32
3. Trigger Type 33
4. Setup Format 34
5. Data Pattern Setup 34
6. InterFrame Setup 34
UART-RS232bus Decode Setup Detail 35
UARTbus Decode Setup Detail - The Basic Tab 35
UARTbus Decode Setup Detail - The Basic Tab 36
Using the CANbus Options 37
Overview 37
Accessing Serial Triggers 37
Creating a CANbus Trigger Condition 38
CANbus Trigger Setup Detail 38
1. Source Setup 38
2. CAN Setup 38
3. Trigger Type 39
4. Setup Format 39
5. Frame ID Setup 39
6. Data Pattern Setup 40
CANbus Decode Setup Detail 41
Using the LINbus Options 42
Overview 42
Accessing Serial Triggers 42
Creating a LINbus Trigger Condition 43
LINbus Trigger Setup Detail 43
1. Sources Setup 43
2. LIN Setup 44
3. Trigger Type 44
4. Setup Format 44

5. Frame ID Setup 45
ID + Data Trigger Setup Detail 45
Error Trigger Setup Detail 46
LINbus Decode Setup Detail 47
Using the FlexRaybus Options 48
Overview 48
Accessing Serial Triggers 48
Creating a FlexRaybus Trigger Condition 49
FlexRaybus Trigger Setup Detail 49
1. Sources Setup 49
2. FlexRay Setup 50
3. Trigger Type 50
TSS (Start) or Frame Trigger Setup Detail 51
Frame Trigger Setup Detail 51
Symbol Trigger Setup Detail 53
Errors Trigger Setup Detail 53
FlexRaybus Decode Setup Detail 54
FlexRaybus Decode Setup Detail - The Basic Tab 54
FlexRaybus Decode Setup Detail - The Levels Tab 54
Using the TD Packages: Characterizing Embedded Controller Performance 55
Overview 55
Characterization Using Cursors 56
Characterization Using Measurement Parameters 56
Gating with Measurement Parameters 57
Statistics and Graphing with Measurement Parameters 58
Pass/Fail Analysis with Measurement Parameters 58
Isolating and Analyzing Serial Bus Activity 59
Overview 59
Capturing Long Pre-Trigger Times 59
Repeatedly Triggering and Saving the Data to a Hard Drive 60
Repeatedly Triggering and Storing All Triggers (Sequence Mode) 61
Serial Trigger Setup for Specific Events 62
I^2Cbus Specifications 66
SPIbus Specifications 68
UART - RS232bus Specifications 70
CANbus Specifications 72
LINbus Specifications 74
FlexRaybus Specifications 76

LSSD Introduction

A variety of Serial Data standards, such as Inter-IC (I^2C), Serial Peripheral Interface (SPI), UART (Universal Asynchronous Receive Transmit), RS-232, CAN (Controller Area Network), LIN (Local Interconnect Network), and FlexRay are used to communicate from microprocessors to peripherals or between embedded controllers.

I^2C is a standardized protocol created by Philips with a documented technical specification. SPI was popularized by Motorola but is not standardized, per se – there are a variety of variants with the differences characterized by how data is clocked, whether data is MSB or LSB format, and whether it is multi-slave or single-slave. UART is a generic "backbone" for many proprietary serial data protocols too numerous to mention, each with different physical layers. RS-232 is a special case of UART, with a more defined protocol and specific physical layer. LIN is a low cost master/slave system designed for low cost implementation in vehicles, typically in what is commonly referred to as "body electronics". FlexRay is a time-triggered automotive communications bus designed for higher speeds and fault tolerance. A summary of the differences is in the table below:

Standard	Definition	Number of Lines	Data rate	Synchronous or Asynchronous
I^2C	Inter IC	2	Up to 3.4 Mb/s	Synchronous
SPI	Serial Peripheral Interface	3	Up to ~50 Mb/s	Synchronous
UART	Universal Asynchronous Receiver/ Transmitter	1	Up to 1 Mb/s (typical)	Asynchronous
RS-232		1	Up to 57.6 kb/s (typical)	Asynchronous
CAN	Controller Area Network	1 (differential)	Up to 1 Mb/s	Asynchronous
LIN	Local Interconnect Network	1	Up to 19.2 kb/s	Asynchronous
FlexRay		1 (differential)	2.5, 5 or 10 Mb/s	Asynchronous

The TD and D Serial Data options are unique oscilloscope tools from LeCroy that will greatly increase your ability to debug and analyze embedded controllers that use serial bus communications. The D products are Serial Decode only, and the TD products include Serial Trigger and Decode. The serial triggers are integrated into the oscilloscope – no external hardware is used – and is selected through the normal oscilloscope trigger menus. Serial data signals are input to the oscilloscope through normal passive or active probes, such as LeCroy's ZS Series of high impedance active probes. Decoding is accessed from the Analysis pull-down menu. The decoding is overlaid on top of the appropriate channel, and is intuitively presented and color-coded for quick understanding. All packages contain Search capability for specific messages, and a Table to display protocol data in summary form underneath the oscilloscope grid.

The various TD options are available for WaveRunner Xi and WaveSurfer Xs oscilloscopes.

The various D options are available with WaveRunner 6000, WavePro 7000 and WaveMaster 8000 oscilloscopes.

The I²C-BUS Specification published by Philips Semiconductors fully describes the I²C standard. As of the date of printing of this manual, Version 2.1, January 2000 is the most recent version (though this does not include the recent FM+ update).

There is no formal SPI standard. Descriptions of SPI and its variants are usually included in the technical documentation for the microprocessor that offers support for SPI.

UARTs evolved from mechanical rotating teletypewriter devices, and formats were formalized with the advent of the first electronic computers. There is no formal UART standard.

RS-232 has a defined physical layer defined in the Electronic Industries Association (EIA) EIA-RS-232-C and the Telecommunications Industry Association (TIA) TIA-232-F. This standard defines the RS-232 physical layer and does not specify use of a UART protocol layer; however, UART is commonly implemented.

The CAN specifications are described in a number of public domain documents, these documents include ISO11519, ISO11898, CAN Specification 2.0A, and CAN Specification 2.0B.

The LIN specification is published by the LIN Consortium. As of the date of the printing of this manual, Version 2.1, November 24, 2006 is the most recent version. Preceding versions are 1.3 and 2.0. SAE published a variation of LIN 2.0 called J2602. This specification shares many attributes of LIN 2.1.

The FlexRay specification is published by the FlexRay Consortium. As of the date of the printing of this manual the FlexRay protocol specification is currently at V2.1 Rev A, accompanying this specification is the V2.1 Rev A Errata V1 document. Separate specifications exist for the physical layer and data link layer.

This manual will assume that you have a basic understanding of the various serial data standard physical layer and protocol layer specifications, and knowledge of how these standards are used in embedded controllers. It also assumes that you have a basic understanding of oscilloscope operation, specifically the LeCroy oscilloscope that the serial trigger and decode option will be used with. Wherever practical or necessary, details on specific oscilloscope features have been included in this manual.

Note: LeCroy has a policy of frequently updating software. While screen images in this manual may not exactly match what is seen on your oscilloscope display, be assured that the functionality is nearly identical.

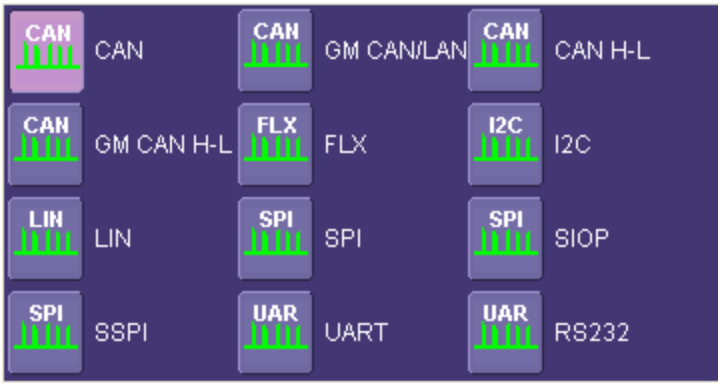
The TD Series Software

The TD option adds the following capability to the LeCroy oscilloscope software user interface dialogs:

1. **Serial Trigger Selection** - If this is the first serial trigger option you have installed on your oscilloscope, an additional icon is shown on your trigger dialog box. It allows a serial trigger condition to be set from within the oscilloscope using an easy-to-understand interface.

2. **Serial Decode** - If this is the first serial decode option you have installed on your scope, an additional set of Serial Decode and Decode Setup dialog boxes are provided for setup of protocol format (as necessary) and decoding. These can be accessed from the Analysis menu.

3. **Decode Protocol Selections** - As serial decode options are added to your oscilloscope, additional protocol selections are available in a pop-up dialog box within the Serial Decode dialog boxes.



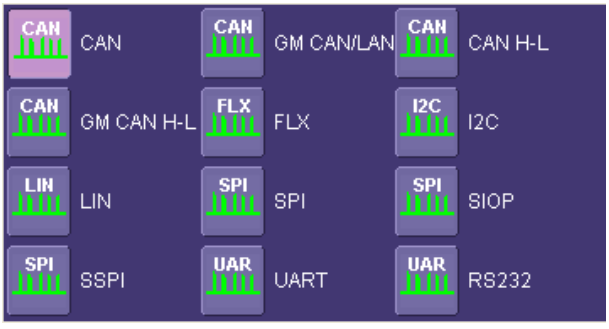
Note: SIOP and SSPI are part of the SPIbus TD package. RS232 is part of the UART-RS232bus TD package.

The D Series Software

The D option adds the following capability to the LeCroy oscilloscope software user interface dialogs:

1. **Serial Decode** - If this is the first serial decode option you have installed on your scope, an additional set of Serial Decode and Decode Setup dialog boxes are provided for setup of protocol format (as necessary) and decoding.

2. **Decode Protocol Selections** - As serial decode options are added to your oscilloscope, additional protocol selections are available in a pop-up dialog box within the Serial Decode dialog boxes.



PLEASE NOTE THE FOLLOWING:

- SIOP and SSPI are part of the SPIbus TD package.
- RS232 is part of the UART-RS232bus TD package.
- GM CAN/LAN, CAN H-L and GM CAN H-L are part of the CANbus TD package.

Technical Explanation of D and TD Options

Overview

LeCroy's offering of serial trigger and decode options utilize advanced trigger circuitry and advanced software algorithms to provide powerful capability for serial data triggering and decoding.

Serial Trigger

TD options contain advanced serial data triggering. This serial data triggering is implemented directly within the hardware of the oscilloscope acquisition system, and contains advanced algorithms to protocol decode, recognize, and trigger on user-defined serial data patterns. This allows a recognized serial data pattern to be used to trigger the oscilloscope at a pre-determined time, and other signals coincident with the desired serial data pattern can be captured simultaneously.

Serial Decode

Both the D and TD options contain powerful protocol decoding and annotation software algorithms. This algorithm is used in all LeCroy serial decoders sold with oscilloscopes, and differs slightly for serial data signals that have a clock embedded in data or a clock separate from data.

The software algorithm examines the embedded clock (UART, RS-232, CAN, LIN, FlexRay) or separate clock line (I^2C, SPI, SSPI, SIOP) for each message based on a default (or user set) vertical level. The algorithm is intelligently applies a hysteresis to the rising and falling edge of the serial data signal to minimize the chance of perturbations or ringing on the edge affecting the decoding. The default level is usually set to 50% and is determined from a measurement of peak amplitude of the signals acquired by the oscilloscope. It can also be set to an (absolute) voltage level, if desired. For serial data signals with embedded clocks (UART, RS-232, CAN, LIN, or FlexRay), the algorithm then performs an analysis of the serial data message to determine the nominal bit width. The clock rate/bit width is measured directly from the clock line for other serial data formats (I^2C, SPI, SSPI, SIOP). Once the clock signal is acquired and the decoding is completed for a serial data message with separate clock and data lines, the oscilloscope channel can be turned OFF to reduce screen clutter.

After determining bit width, a different algorithm performs a decoding of the serial data message into binary format after separation of the underlying data into logical groups (Header/ID, Data Length Codes, Data, CRC, Start Bits, Stop Bits, etc.). Finally, another algorithm provides the appropriate color coding of the message, and displays the protocol message data on the screen, as desired, overlaid on the source trace. Various compaction schemes are utilized to show the data during a long acquisition (many hundreds or thousands of serial data messages) or a short acquisition (one serial data message acquisition). In the case of the longest acquisition, only the most important information is highlighted. In the case of the shortest acquisition, all information is displayed (Header/ID, Data Length Codes, Data, CRC, Start Bits, Stop Bits, etc.) with additional highlighting of the complete message frame.

Note: Although the decoding algorithm is based on a clock extraction software algorithm using a vertical level, the results returned are the same as those from a traditional protocol analyzer using sampling point-based decode. In addition, the clock extraction technique allows partial decoding of messages in the event of physical layer noise, in many cases, whereas a protocol analyzer usually cannot. This is a significant advantage for the LeCroy software algorithm.

If the sampling rate (SR) is insufficient to resolve the signal adequately based on the bit rate (BR) setup or clock frequency, the protocol decoding is turned OFF to protect the operator from incorrect data. The minimum SR:BR ratio required is 4:1. It is suggested that you use a slightly higher SR:BR ratio if possible, and use significantly higher SR:BR ratios if you want to also view perturbations or other anomalies on your serial data analog signal.

Table Display

The tabular display of serial decoded data is a powerful feature allowing you to view all of your protocol messages even if the compaction of serial data messages on the oscilloscope grid means annotation is impractical. The table uses the decoded data (extracted as previously described) as its source. So, if View Decode is not checked ON, the table is not shown.

Accessing the TD Toolsets

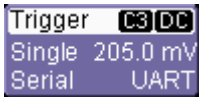
Overview

TD trigger and decoding tools are easily accessible in a variety of ways. The TD options provide an additional Serial selection to the Trigger Type in the Trigger dialog, and a new set of dialogs for Decode setup. These dialogs are shared by all of LeCroy's low-speed serial protocol offerings, so all serial trigger and serial decoding selections are grouped in a common section with nearly identical selection and setup. These dialogs are conveniently accessed with just one or two touches of the screen.

Accessing Serial Trigger Dialogs

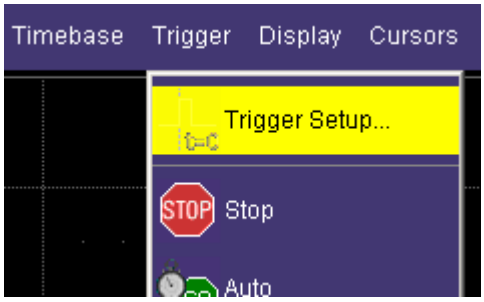
Serial Trigger dialogs can be accessed using one of the following methods:

- Touch the Trigger descriptor button. Then select the Serial trigger type in the dialog box.



OR

- Touch **Trigger → Trigger Setup** from the menu bar. Then select the Serial trigger button and choose the appropriate serial trigger.



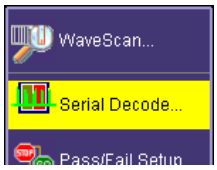
Since each serial protocol is quite different, each serial trigger is also different. Details on setting up serial triggers are covered in corresponding sections of this manual.

Serial Decode and Decode Setup

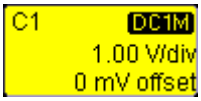
These dialogs set the oscilloscope for protocol decoding of serial data messages with display of the protocol data overlaid on the signal. They also allow quick and easy access to oscilloscope zooming, searching, table display, and table export.

The serial decode and decode setup dialogs are accessed in the following ways:

1. Touch Analysis in the menu bar, and select Serial Decode. This will open and display the Serial Decode Summary dialog, and provide access to Decode Setup.



2. Touch the Channel or Memory Descriptor Box to open the respective dialog box, and touch the Decode button in the bottom toolbar.

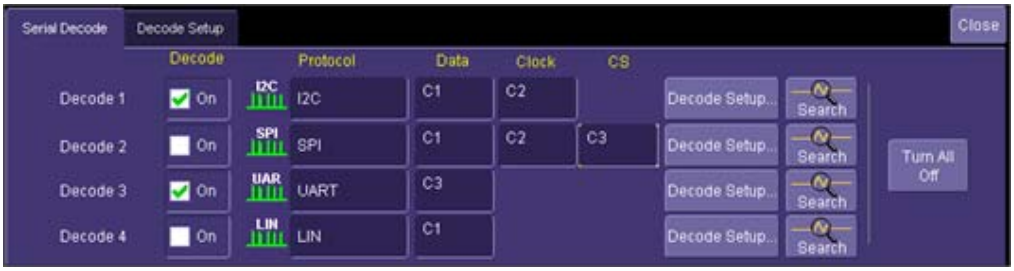


3. Touch a Channel, Memory, or Math trace to open a pop-up dialog that displays a shortcut to the Decode Setup dialog box.

Note: There is a great deal of commonality in decode setup among the various serial data standards. Common areas are discussed in the following topic. Specifics about triggering and decoding each protocol are discussed in later topics.

Serial Decode (Summary) Dialog Box

The Serial Decode dialog box is a summary page showing which decoders are ON and how they are configured. Corresponding shortcuts are also provided for Decode Setup and Search. A sample dialog box is shown here.



Note: This (previous) dialog box reflects an oscilloscope with the I^2C, SPI, UART and LIN options.

There are four independent decoders. A user can operate up to four at a single time, although limitations may occur with regard to how the numbers of channels are accommodated at one time. Practically speaking, if a user decodes signals with a clock and data line (and perhaps also a chip select or other third line), then two simultaneous decodes is the maximum number using the LeCroy oscilloscope analog channels. The addition of the MS-250 or MS-500 Mixed Signal Oscilloscope options allow usage of digital lines for trigger and decoding, which preserves analog channels for other uses. Contact your local LeCroy sales office for more information about this option.

The detail of the Serial Decode dialog box is described as follows:

1. **Decoder #** - There are four independent decoders, each can be set up a different way for a different protocol.

2. **Decode ON Checkbox** - If checked, it means the decoder is turned ON and decodes (provided the protocol is correctly setup in the Decode Setup dialog box).

3. **Protocol** - This pop-up dialog allows selection of a specific serial protocol. In some cases where the protocol is not completely standardized, or where there are higher-level definitions of the protocol, multiple selections may be provided. For example, SPI has variants with no chip select. Examples are Simplified SPI (SSPI), Simple Synchronous Serial I/O Port (SIOP), and RS-232. Each of these has a selection in this pop-up dialog.

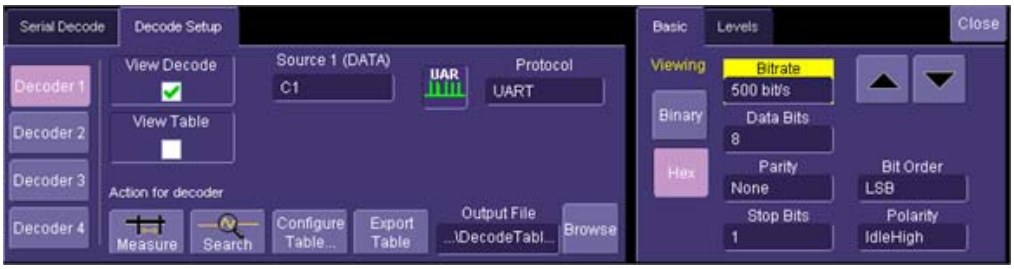
4. **Data and Clock Selection** - This pop-up dialog allows you select a channel or other source for decoding. Some protocols may require a third selection (for instance, SPI also requires a Chip or Slave Selection). Asynchronous protocols, such as UART, RS-232, CAN, LIN, and FlexRay, only require a single source.

5. **Decode Setup Shortcut Button** - This provides quick access to the second tab (Decode Setup) where there are quick buttons for Search, and Table (Configure Table and Export Table). If you have already defined the trigger, then the trigger setup settings will copy over into the decode setup.

6. **Search** - Push this button to open a Zoom (Zx trace) with the corresponding Channel (Cx trace) as its source. In addition, the right-hand dialog in the Zoom trace has Search options specific to the serial protocol to which the Source is assigned.

Decode Setup Dialog Box

The Decode Setup dialog box is where the details of a specific protocol decode is entered. It appears as follows (the UART Decode Setup dialog is shown as an example):



This is a single tab with an indicator on the left side describing to which of the four decoders the setup information pertains.

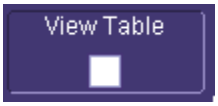
The left side of this dialog box is described here (the right side is explained in the protocol specific topics):

1. **Decoder # Buttons** - Indicates which of the four decoders to which the current information pertains.

2. **View Decode Checkbox** - Use this checkbox to turn decoding turned ON or OFF for the particular decoder. Decoding ON provides a highlight of each message frame with color-coded highlighting and decoding of the various protocol message portions.

> **Note**: Decoding of an entire acquisition with very long acquisitions including hundreds or thousands of messages takes longer.

3. **View Table Checkbox** - Checking this box turns the Table ON. Un-checking it turns the Table OFF.

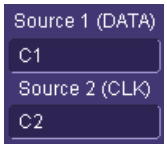


Idx	Time	Addr Length	Address	R/W	Length	Data
8	240.494 ms	7	0x21	1	2	0xff 00 00
9	360.555 ms	7	0x21	0	1	0x08
10	360.698 ms	7	0x21	1	2	0x49 00 00
11	481.865 ms	7	0x21	0	1	0x0a
12	482.007 ms	7	0x21	1	2	0x00 00 00
13	606.294 ms	7	0x20	0	3	0x01 36 00
14	721.235 ms	7	0x20	0	1	0x00
15	721.377 ms	7	0x20	1	2	0x12 36 00
16	841.266 ms	7	0x20	0	1	0x02

PLEASE NOTE THE FOLLOWING:

- If the View Decode checkbox is not checked (meaning decode is not turned ON), then the View Table checkbox is grayed out. The Table cannot display unless decode is occurring.
- When the Table is displayed, it appears similar to that shown previous (the example shown is for I^2C).

4. **Source Selection** - Touch these selections to open a pop-up dialog box which allows for selection of sources for Clock, Data, and (for some protocols) a third line (Chip Select for SPI).



PLEASE NOTE THE FOLLOWING:

- Source selection is dynamically linked to the Protocol selection, so the appearance and number of sources to choose changes based on your selected Protocol.
- Source can be a Channel (C1 – C4), a Memory Trace (M1 – M4), or a Math Function (F1 – F4).
- Use a Channel for a new, real-time acquisition.
- Use a Memory for recalling saved data from a previous acquisition for further analysis. Refer to your oscilloscope's Save and Recall Waveforms topic for more details.
- Use a Math Function to view decoded data on Sequence mode acquisitions. Sequence Mode is a unique capability where you can utilize oscilloscope memory to capture events widely spaced in time and then view them sequentially. Reference the chapter on Isolating and Analyzing Serial Bus Activity for more information on setting the oscilloscope up in this mode.

5. **Protocol Selection** – Touch this selection to open a pop-up dialog box and choose a protocol decoder. Depending on the decoder selected, the correct inputs (Clock, Data, and a third line, if required) are shown to the left.

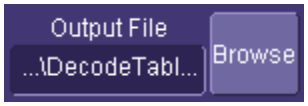
6. **Action for Decoder Toolbar** - Various buttons on this toolbar provide context-sensitive shortcuts for decoding.
   - **Search** allows quick creation of a zoom trace and changes the dialog box to the zoom/search dialog box.

     Acquire long records of message data, and use Search to look through the record for a specific message. When the message meeting the search criteria is found, the complete message is then shown with the Zoom Trace. Use the arrow buttons to navigate forward and backward through the messages. Unsuccessful searches are noted with a line of text.

   - **Configure Table** displays a pop-up dialog box specific to a particular protocol. The dialog contains checkboxes for various columns in the table. Check or uncheck the checkboxes to show or hide specific columns on the table.

   - **Export Table** exports the complete protocol table data to a user-defined file.

   - The **Output File** name and directory can be selected by the user using the controls to the right. Click the **Browse** button to select a file.



Protocol Results Table

The protocol results table provides a quick and easy way to understand all of your protocol data as decoded by the oscilloscope, even when messages are too compact to allow annotation on the display. In addition, the table provides a quick and easy method to view decode results and quickly zoom to a specific message.

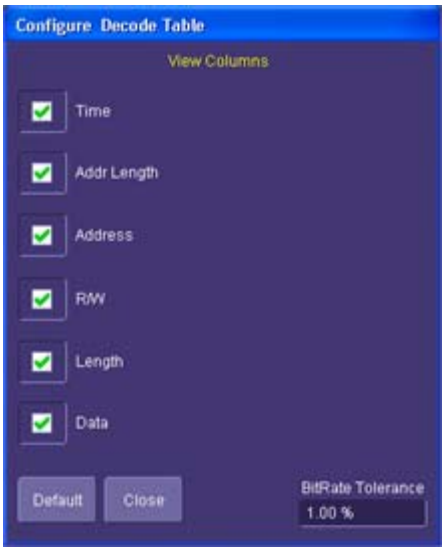
When displayed, the protocol results table appears under the waveform grid. It looks like the following (this example is for I^2C – each protocol's table looks different):

Idx	Time	Addr Length	Address	R/W	Length	Data
8	240.494 ms	7	0x21	1	2	0xff 00 00
9	360.555 ms	7	0x21	0	1	0x08
10	360.698 ms	7	0x21	1	2	0x49 00 00
11	481.865 ms	7	0x21	0	1	0x0a
12	482.007 ms	7	0x21	1	2	0x00 00 00
13	606.294 ms	7	0x20	0	3	0x01 36 00
14	721.235 ms	7	0x20	0	1	0x00
15	721.377 ms	7	0x20	1	2	0x12 36 00
16	841.266 ms	7	0x20	0	1	0x02

Use the vertical scroll bar on the right to navigate the protocol table. If your vertical scroll bar is yellow, the **Adjust** knob on your oscilloscope's front panel can also be used to navigate the table.

If you touch a row, a decoded zoom trace is created that displays the message trace in a zoom.

The table only displays if the **View Table** checkbox is checked and decoding has occurred on the trace. Only one protocol table can be viewed at a time. As described in the previous section, the protocol table can be configured or exported. If you press the **Configure Table** button in the Decode Setup dialog box, a pop-up dialog similar to the following is shown:



- **Default** - Press the Default button to reapply standard settings for a particular protocol.

- **Checkboxes** - Touch items to check the box and include them as table columns for a particular protocol.

- **BitRate Tolerance** - Some protocols have a Bit Rate Tolerance setting. This can be set to any value from 0.01% to 10%. If the bit rate is outside the tolerance range set, then the calculated bit rate appears in red text on the table.

Protocols with a wide variance of bit rates, such as I^2C (which often has clock stretching) do not have this feature.

Searching for Messages

There are several ways to search for specific messages. The following are all valid ways to search messages.

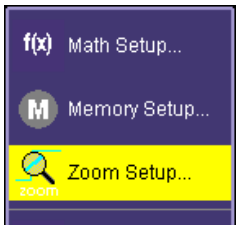
- Touch the decoded waveform. A pop-up dialog is shown where you can select **Decode Search** as follows:

OR

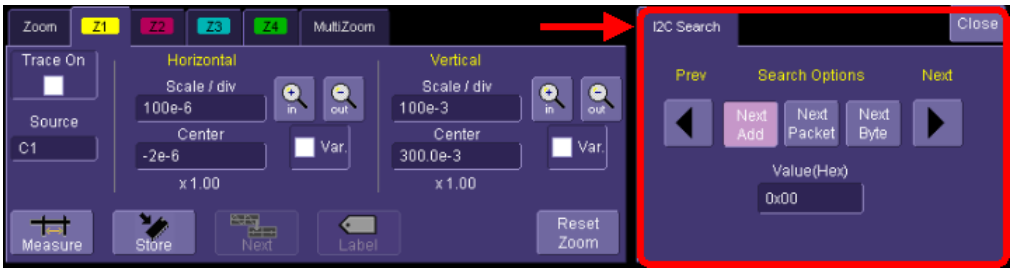
- Touch the **Search** button in the Serial Decode Summary dialog box or the Decode Setup dialog box.

OR

- Go to **Math → Zoom Setup** from the Menu Bar to turn a Zoom ON, define its source, and search directly.



Any of the aforementioned methods show the Zoom dialog box and a corresponding Search dialog box on the right side.



PLEASE NOTE THE FOLLOWING:

- Search capabilities differ by protocol. For instance, SPI has no Address, so there is no capability to Search by Address in SPI, while there is when searching under the I^2C protocol.
- Use the Search Options buttons to define the type of Search you want, enter a value in Hexadecimal format, and use the left and right arrows to move your way from one message to the next.

Using the I²Cbus Options

Overview

Both I²Cbus TD and D options contain powerful software algorithms to extract serial data information from physical layer waveforms measured on your oscilloscope. The extracted information is overlaid (annotated) on the actual physical layer waveforms, and color-coded to provide fast, intuitive understanding.

The I2Cbus TD option contains a very powerful and flexible trigger, but it is also very easy to set up for basic triggering. The I²Cbus TD option contains a conditional I²C DATA trigger to select a range of DATA values to trigger on, not just a single DATA value. Oftentimes, I²C utilizes DATA bytes to specify sub-addresses for accessing memory locations in EEPROMs. Conditional DATA trigger allows triggering on a range of DATA bytes corresponding with reads or writes to specific sub-address memory blocks in the EEPROM. It can also aid in monitoring DATA outputs from I²C- based sensors, such as analog-to-digital converters, and triggering when DATA is outside a safe operating range. In both cases, verifying proper operation becomes a simple task. Other powerful and user-friendly features included in I²Cbus TD trigger include:

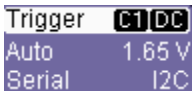
- Ability to define and ADDR or DATA condition in either binary or hexadecimal formats.
- Ability to define an ADDR condition in binary with the DATA condition defined in hexadecimal so as to trigger on a range of ADDR values using Don't Care bits.
- FRAME LENGTH trigger setups.
- EEPROM trigger setups to trigger on up to 96 bits (12 bytes) of DATA at any location within an I²C frame or at a user-defined location in a 2048 byte window.
- All permutations of Read, Write, or R/W Don't Care conditional setup for 7 and 10-bit addresses.
- For any I²C message trigger, select whether an ACK condition should be ACK, NO ACK, or DON'T CARE. You can choose to trigger on a NO ACK condition by itself, or as part of a more complex ADDR/DATA trigger.

If you are not familiar with or are just learning about I²C, start by using the simplest trigger conditions (Start, Stop, ReStart, NoAck) to gain confidence, then set up simple ADDR only conditions. When you are confident with understanding I²C operation, set up an ADDR+DATA condition with a condition of "DATA =". Then, try different setups using other DATA conditions (>, <, INRANGE, etc.). Lastly, experiment with the EEPROM trigger setup, which provides the most flexibility by allowing location of data, with conditions, within specific bytes of a long sequence of DATA bytes.

Accessing Serial Triggers

The I²Cbus serial trigger is accessed from the oscilloscope trigger dialog. Use one of the following methods:

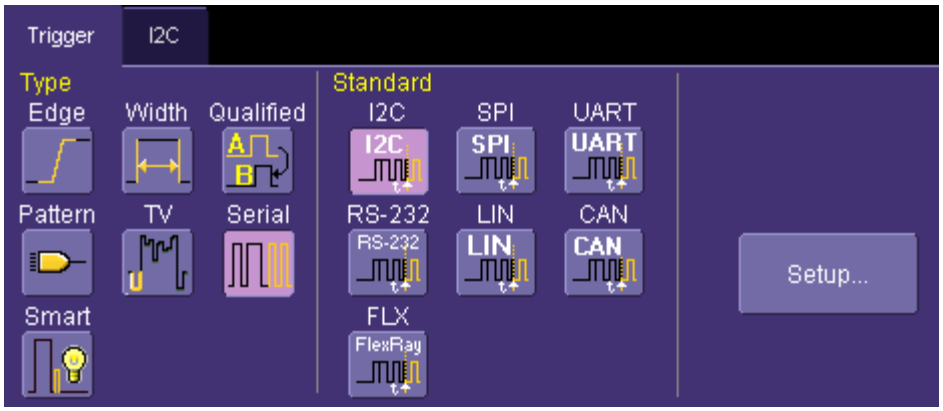
- Touch the Trigger Descriptor Box in the lower right hand corner of the oscilloscope display.



OR

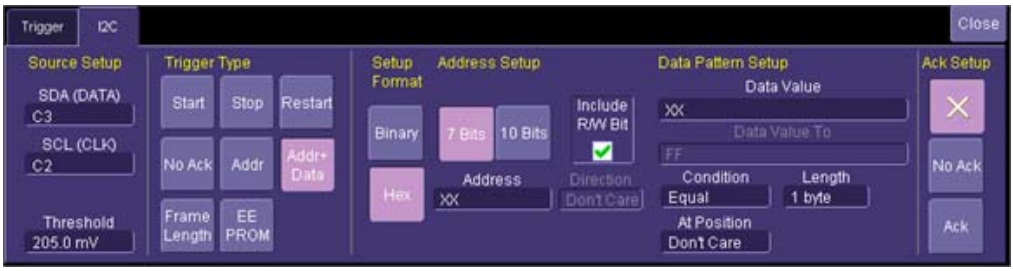
- Touch **Trigger → Trigger Setup** from the Menu Bar. On the Trigger dialog, touch **Serial** on the Type section.

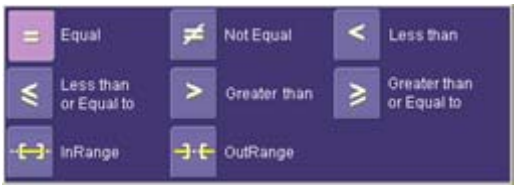
  Select the appropriate serial trigger. The menu automatically changes to a different tab in the Trigger dialog reflecting the selected standard.



Creating an I²Cbus Trigger Condition

The following trigger setup detail topics show the dialog selections for an I²Cbus Trigger with detail on some of the setup conditions.



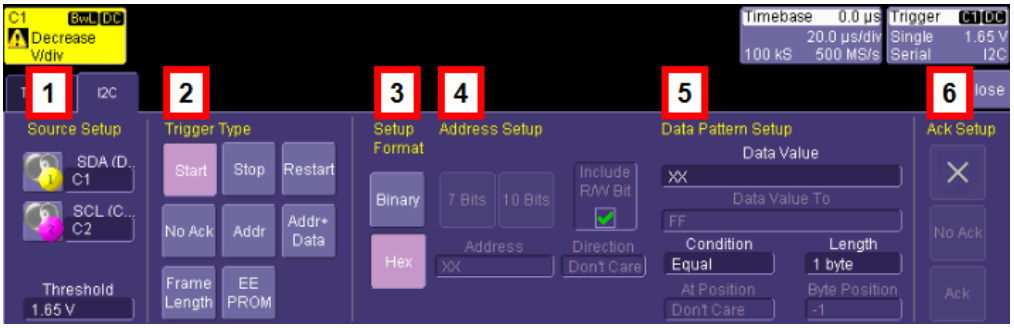


Selection of Trigger Type results in dynamic changes to the I²Cbus Trigger dialog. Simple I²C triggers, such as Start, Stop, ReStart, and NoAck, require no additional setup, while frame-based triggers, such as ADDR, ADDR+DATA, FRAME LENGTH, and EEPROM require addition user-defined setup information.

Select condition values by touching fields (using your finger, or use a mouse pointer). A pop-up dialog box is shown where selections can be made.

I²C Trigger Setup Detail

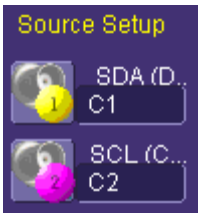
The following topics show the dialog selections for an I²Cbus Trigger.



The previously numbered I²Cbus trigger sections correspond with the following explanations.

1. Sources Setup

- **DATA** and **CLOCK** - The pop-up dialog is used to select the appropriate channel or EXT inputs for each. Set these fields up with caution or your trigger may not function correctly.



- **Threshold (Trigger) -** Adjust the vertical level for the trigger. Much like an Edge trigger, a user must specify the level used in order to process the incoming signals and determine whether the desired serial data pattern is meeting the set trigger condition. This value is used for both DATA and CLOCK signals.

2. Trigger Type

The I^2C trigger can be configured to trigger on simple conditions (i.e. the presence of a START, STOP, RESTART bit, or the absence of an ACK bit (NO ACK). In addition, more complex trigger conditions can be created using ADDR, ADDR+DATA, FRAME LENGTH, or EEPROM setups.

If one of the more complex trigger conditions is selected, then reference the sections below for information on Address and Data Pattern Setup.

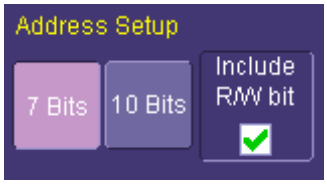
3. Setup Format

Select either Binary or Hexadecimal (Hex) setup mode. The format propagates through the entire I^2C trigger setup.

A user can select Binary mode, and set up the address in binary format, then reselect Hex mode and set up the data in hexadecimal format. Toggling back and forth between the modes does not result in lost information (binary is used internally as the core format for all triggering and decoding operations), though use of don't care bits in a binary setup results in the display of an X (for a full nibble don't care) or a $ (for a partial nibble don't care).

4. Address Setup

These following setup choices demonstrate ADDR, ADDR+DATA, FRAME LENGTH, or EEPROM Trigger Selections.



- **Address Length** - I^2C utilizes either 7 or 10-bit formats for the address, depending on the device. Make the appropriate selection so as to be able to enter the correct address value.
- **Include R/W bit** - If 7-bit address length is selected, another selection will appear for whether the Read/Write bit should be included as part of the address value entered. For instance, some engineers think of the address pattern as including the R/W bit (i.e. 8-bits) and others think of the address pattern as not including the R/W bit (i.e. 7-bits). "Check" the checkbox if you want to include the R/W bit in your entered Address value. If this is done, then the Direction value will auto select either Read or Write (as appropriate) and gray out as not-selectable by the user.

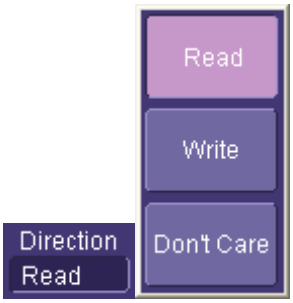
  Note: There is an identical checkbox selection in the I^2C decode setup dialog. These two setups are dynamically linked, so selecting it one way in trigger will result in an identical selection in decode. This ensures that the trigger address format matches decoded information on the display.

- **Address Value Setup** - Enter the Address Value in binary or hex (depending on what was selected in the Setup Mode). The pattern condition for the Address is always "equal".

  Binary addresses allow use of don't care conditions in any bit position (entered as X). Hexadecimal addresses allow use of don't care conditions in any nibble position) also entered as an X. If an address is set up in Binary, then converted to Hex with a Setup Mode change, then any non-nibble length don't care values will be shown as $.

  Note: Address values are always MSB format. Therefore, conversion of address values from binary to hex when don't care values are used will be on that basis.

- **Direction** - Enter a Direction (Read, Write, or Don't Care) for the Address value. If you have selected to use 7-bit addresses with the R/W bit included in the address value, then this selection will be grayed out and not selectable.



5. Data Pattern Setup

This step is explained using demonstrations based on **ADDR+DATA or EEPROM (Data Setup)** and **FRAME LENGTH (Frame Length)** trigger type selections.

- **Data Setup** - These setup selections are displayed if the Trigger Selection is ADDR+DATA or EEPROM.

- **Data Pattern Value** - The pattern value is entered in either Binary or Hexadecimal mode depending on the previous selection of Setup Mode. There are two selections for pattern value - "Data Value" and "Data Value To." The second selection is exposed for entry if the Condition is set to INRANGE or OUT(of)RANGE. Otherwise, it is grayed out. Up to 12 bytes of data can be entered as a pattern value.

  If less than 12 bytes of data is entered for the pattern value, the data is assumed to begin at the 0 (i.e. first) data byte in the I^2C message. If this is not desired, then add preceding or trailing don't care (X) nibbles to the pattern value.

  PLEASE NOTE THE FOLLOWING:

  - When more than one byte of data is entered as a data pattern value, the data is treated as "Most Significant Byte (MSB) First." This is especially important to remember when setting up <=, <, >, >=, INRANGE and OUTRANGE comparisons.
  - In Hexadecimal format, data must be entered as full bytes even though the minimum required acceptable entry is a nibble. If less than a full byte is entered, then a don't care X will precede the pattern values entered.
- **Condition** - The DATA condition can be set many different ways. Possible conditions are <=, <, =, >, >=, not =, in a range, out of a range, or don't care.

  Oftentimes, I^2C utilizes DATA bytes to specify sub-addresses for accessing memory locations in EEPROMs. Conditional DATA trigger allows triggering on a range of DATA bytes that correspond to reads or writes to specific sub-address memory blocks in the EEPROM. It can also aid in monitoring DATA outputs from I^2C-based sensors, such as analog-to-digital converters, and triggering when DATA is outside a safe operating range. In both cases, verifying proper operation becomes a simple task.

- **Length** - The pattern length value defaults to the length, in bytes, of the pattern set in the Data Value selection. If the length is changed to a lesser value, it truncates the beginning of the value. If the length is increased, it would add don't care XX byte values to the beginning of the value.

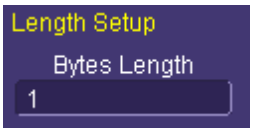
- **At Position, Position** - These selections are present only when the Trigger Selection is EEPROM or ADDR+DATA At Position can be either VALUE or DON'T CARE. When At Position = VALUE, you must also enter a data byte number for Position (0 = the first data byte). For EEPROM triggering, use this to specify a specific location of data, such as a sub-address memory block, that the Pattern Value must occupy in order for triggering to occur. For ADDR+DATA triggering, use this to specify a specific location where the data values should be located without using don't care (X) values in the pattern value. In both cases, you can select a Position in up to a 2048 byte data pattern, starting with Byte 0.

  Note: The first byte is counted as Byte 0, not Byte 1.

- **Frame Length Setup** - This setup selection is displayed if the Trigger Selection is FRAME LENGTH. It is used to trigger on a specific Address value with a defined length of data bytes.

- **Bytes Length** - Specify a data length value between 0 and 2047. 1 is the default value.

  If the Data Length Condition (as follows) is selected to be either INRANGE or OUT(of)RANGE, then this selection will be for the minimum data length value (i.e. the lower value of the range you wish to include or exclude).



Note: All values entered in this field are always in decimal format.

- **Bytes Length Max** - If the Data Length Condition is selected to be either INRANGE or OUT(of)RANGE, then you also need to specify a maximum data length value (i.e. the upper value of the range you wish to include or exclude).

- **Length Condition** - The Data Length Condition can be set to many different values, such as <=, <, =, >, >=, not =, INRANGE, or OUT(of)RANGE. Select the correct condition for your needs.

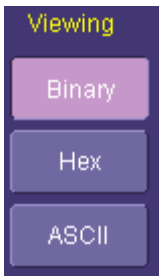
6. Ack Setup

Use this setup to choose whether you want to add an Acknowledge bit condition to your ADDR, ADDR+DATA, FRAME LENGTH, or EEPROM trigger condition. X (Don't Care) would be the most common setup, although ACK or NO ACK might be a useful condition to add for an unusual or hard to find I^2C problem. An example of this would be triggering on an EEPROM write (selected by an ADDR trigger) where the EEPROM failed to acknowledge a byte written.

I²C Decode Setup Detail

I²Cbus Decode Setup Right-Hand Dialogs are shown when I²C is selected as the decode protocol. It provides detailed fields and setup conditions as follows:

- **Viewing** - Select to view the protocol data in Binary, Hexadecimal, or ASCII modes.



Note: If the trigger is set up first, the trigger settings copy into the decode settings.

- **Include R/W Bit** - Some engineers think of the 7-bit address pattern as including the R/W bit (i.e. 8-bits) and others think of the address pattern as not including the R/W bit (i.e. 7-bits). If you decoded I²C messages include 7-bit addresses, mark the checkbox if you want to include the R/W bit in the decoded Address value.

Note: There is an identical checkbox selection in the I²C trigger setup dialog. These two setups are dynamically linked, so selections here in decode results in an identical selection in trigger. This ensures that the decode address format matches trigger setup information.

- **Level Type** and **Level** - The message decoding algorithm setup is performed here. The level is normally set up in %, and defaults to 50%. To adjust the level, touch inside the number area to highlight the box title in yellow, and then use the oscilloscope front panel Adjust knob to adjust. Or touch inside the number area twice and select a value using the pop-up numeric keypad.

PLEASE NOTE THE FOLLOWING:

- The set Level appears as a dotted horizontal line across the oscilloscope grid.
- If initial decoding indicates that there are a number of error frames, make sure that the level is set to a reasonable value.
- DATA and CLOCK can have different level settings, but they are typically the same level.

Using the SPIbus Options

Overview

Both SPIbus TD and D options contain powerful software algorithms to extract serial data information from physical layer waveforms measured on your oscilloscope. The extracted information is overlaid (annotated) on the actual physical layer waveforms, and color-coded to provide fast, intuitive understanding.

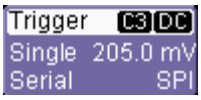
The SPIbus TD option contains a data trigger that can be configured for the many variants of SPI, such as SSPI (single master and slave with predetermined format settings) and SIOP. The basic SPI Type is all-inclusive and the SSPI and SIOP types are just pre-selected settings in the basic SPI trigger.

The SPI trigger does not require use of a Chip Select line. In its place is the ability to set a minimum Interframe Time corresponding with a time that (in AUTO mode) is (typically) 4x a single bit time and less than the interframe time between different message packets. By eliminating the Chip Select line presence requirement, an additional oscilloscope channel is preserved for use with other analog signals. This is a significant feature. It also allows a user to trigger on simplified SPI (SSPI, SIOP, etc.) protocols with a single Master and Slave and no Chip Select line.

Accessing Serial Triggers

The SPIbus serial trigger is accessed from the oscilloscope trigger dialog. Use one of the following methods:

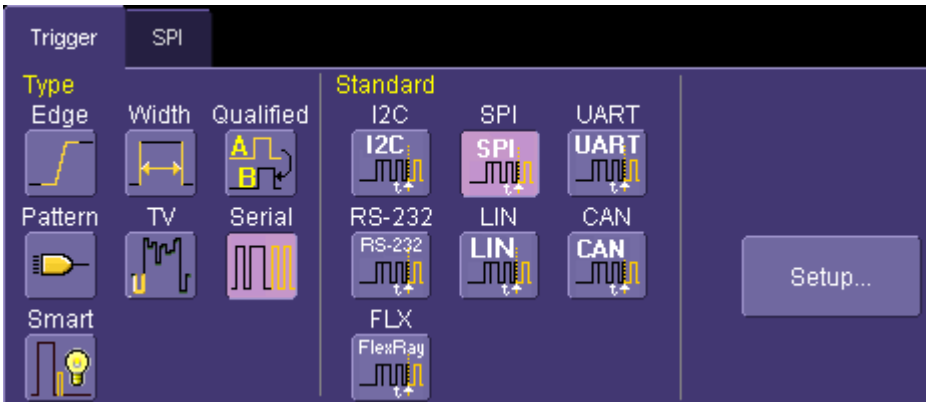
- Touch the Trigger Descriptor Box in the lower right hand corner of the oscilloscope display.



OR

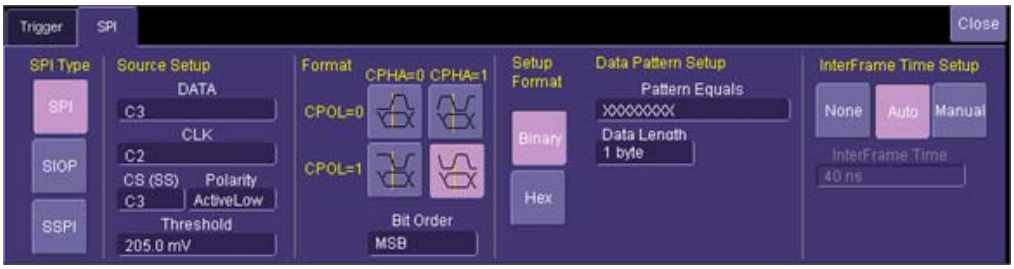
- Touch **Trigger → Trigger Setup** from the Menu Bar. On the Trigger dialog, touch **Serial** on the Type section.

  Select the appropriate serial trigger. The menu automatically changes to a different tab in the Trigger dialog reflecting the selected standard.



Creating a SPIbus Trigger Condition

The SPIbus Trigger dialog, with detail on some of the setup conditions, is shown in the following topics.

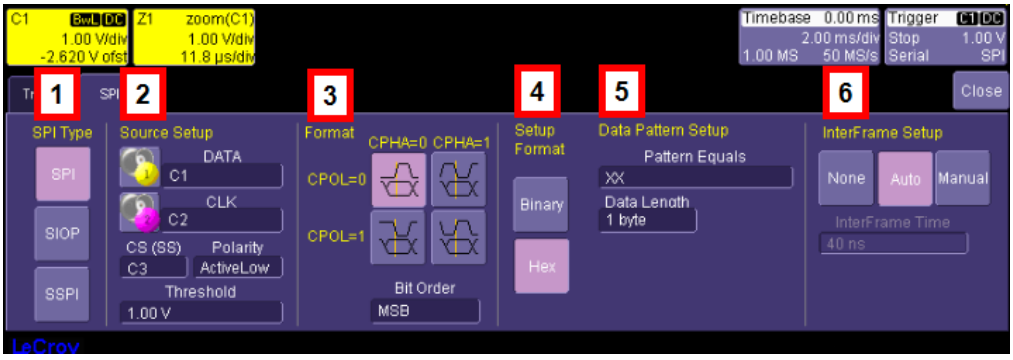


The SPIbus trigger dialog is very flat, meaning there are few dynamic changes to the dialog based on selections within. The one exception is the SPI Type on the far left. When selecting between SPI, SIOP, and SSPI types, the dialog to the right changes to reflect a specific setup type.

Select condition values by touching fields (using your finger, or use a mouse pointer). A pop-up dialog box is shown where selections can be made.

SPIbus Trigger Setup Detail

The following topics show the dialog selections for a SPIbus Trigger.



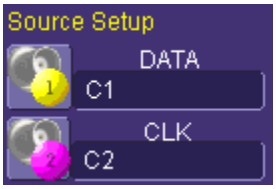
The previously numbered SPIbus trigger sections correspond with the following explanations.

1. SPI Type Selection

- Unlike some other serial data standards (such as I^2C), SPI is not defined by a single standard; rather, there are several implementations of SPI based on fixed clock polarities, phase, and whether Chip Select is present or absent. The basic SPI Type is all-inclusive and the SSPI (Simplified SPI) and SIOP (Synchronous Serial I/O Port) types are just pre-selected settings in the basic SPI trigger and provided for operator convenience. SSPI and SIOP do not use a Chip Select line, but are single Master and single Slave implementations of SPI.

2. Source Setup

- **DATA** and **CLK (CLOCK) -** The pop-up dialog is used to select the appropriate channel or EXT inputs for each. Set these fields up with caution or your trigger may not function correctly.



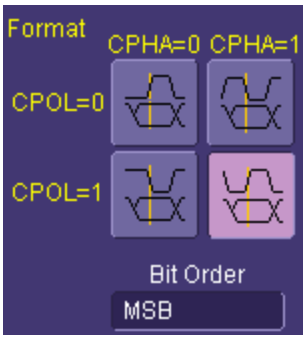
- **CS (Chip Select)** and **Polarity** - These fields are enabled (SPI) or disabled (SSPI, SIOP) based on the SPI Type selected.

If enabled, choose a Channel or EXT, as appropriate, and make a Polarity selection.

- **Threshold (Trigger)** - Adjust the vertical level for the trigger. Much like an Edge trigger, a user must specify the level used in order to process the incoming signals and determine whether the desired serial data pattern is meeting the set trigger condition. This value is used for DATA, CLOCK, and Chip Select signals.

3. SPI Format Setup

- **Clock Polarity and Phase** - SPI requires selections made for the clock polarity and phasing of the data to the clock. SPI microcontrollers and peripherals have settings for CPOL (Clock Polarity) and CPHA (Clock Phase) that are published in the technical datasheets for those products. Selections are made based on the SPI Type chosen previously.



Note: When the basic SPI Type is chosen, you can make selections by clicking on the button containing the graphic that corresponds with your needs as follows:
SPI Mode 0 = CPOL 0 and CPHA 0. SPI Mode 1 = CPOL 0 and CPHA 1. SPI Mode 2 = CPOL 1 and CPHA 0. SPI Mode 3 = CPOL 1 and CPHA 1.

- **Bit Order** - Select either MSB or LSB format, as appropriate.

4. Setup Format

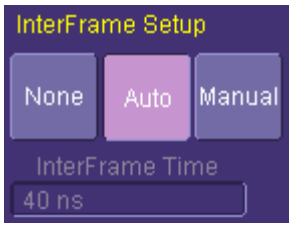
Select either **Binary** or **Hexadecimal (Hex)** setup mode. The mode selected affects the format of the following **Data Pattern Equals** field.

5. Data Pattern Setup

- **Pattern Equals** - Provide an appropriate value based on your Binary or Hexadecimal format selection.
- **Data Length** - Specify a size in bits for your pattern.

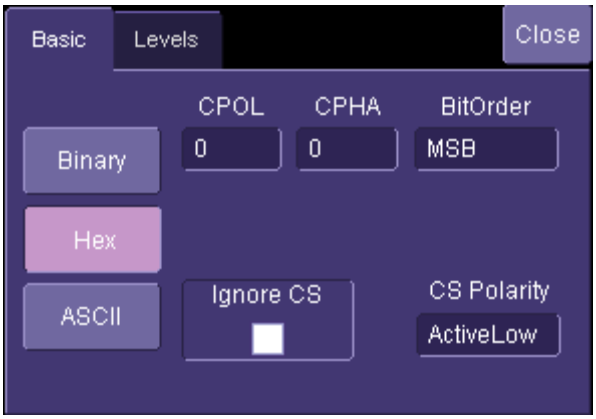
6. InterFrame Setup

Click the appropriate button to select **None**, **Auto**, or **Manual**. The **Manual** button enables the **InterFrame Time** field where you can provide a specific value.



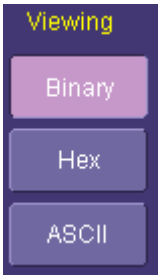
SPIbus Decode Setup Detail

SPIbus Decode Setup Right-Hand Dialogs are shown when SPIbus is selected as the decode protocol. It provides detailed fields and setup conditions as follows:



Note: A similar dialog is shown when SSPI or SIOP are selected; however, these protocols do not use a Chip Select, so the Chip Select selections are omitted.

- **Viewing** - Select to view the protocol data in Binary, Hexadecimal, or ASCII modes.



Note: If the trigger is set up first, the trigger settings copy into the decode settings.

- **Clock Polarity** and **Phase** - SPI requires that selections be made for the clock polarity and "phasing" of the data to the clock. SPI microcontrollers and peripherals have settings for CPOL (Clock Polarity) and CPHA (Clock Phase) that are published in the technical datasheets for those products. These values need to be entered in this section.

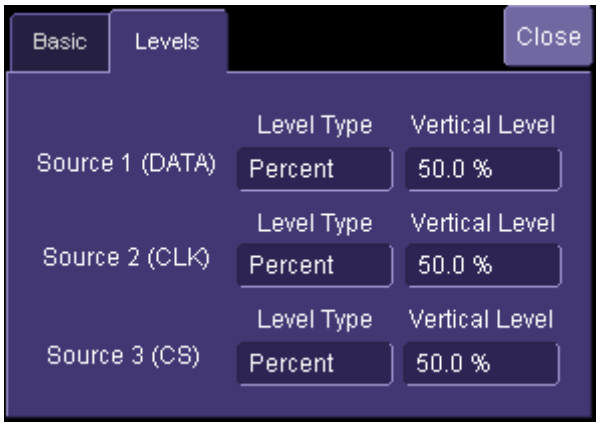
Note: SPI Mode 0 = CPOL 0 and CPHA 0. SPI Mode 1 = CPOL 0 and CPHA 1. SPI Mode 2 = CPOL 1 and CPHA 0. SPI Mode 3 = CPOL 1 and CPHA 1.

- **Bit Order** - Select either MSB or LSB format, as appropriate.

Note: Identical selections for Clock Polarity, Clock Phase, and Data are located in the SPI trigger setup dialog. If you have a single SPI decoder set up, these settings are linked dynamically and copy over from the trigger setup, and vice versa. If you have multiple SPI decoders setup, these settings are also dynamically linked and copy over to the lowest numbered SPI Decoder from the trigger setup, and vice versa. This ensures that the decode address format matches trigger setup information.

- **Ignore CS (Chip Select) and CS Polarity** - Set the Chip Select Polarity to either Active Low or Active High. Also, check the **Ignore CS** box if you want to decode all SPI bytes instead of those active during the Chip Select.

- **Level Type** and **Level** - The message decoding algorithm setup is performed here. The level is normally set up in %, and defaults to 50%. Adjust the level by touching inside the number area to highlight the box title in yellow, and then use the oscilloscope front panel **Adjust** knob to make your change. Alternatively, touch inside the number area twice and select a value using the pop-up numeric keypad.



PLEASE NOTE THE FOLLOWING:

- The set Level appears as a dotted horizontal line across the oscilloscope grid.
- If initial decoding indicates that there are a number of error frames, make sure that the level is set to a reasonable value.

Using the UART-RS232bus Options

Overview

Both UART-RS232bus TD and D options contain powerful software algorithms to extract serial data information from physical layer waveforms measured on your oscilloscope. The extracted information is overlaid (annotated) on the actual physical layer waveforms, and color-coded to provide fast, intuitive understanding.

The UART-RS232bus TD option allows triggering on both DATA conditions and Parity ERRORS. DATA triggering can be set conditionally to select a range of DATA values to trigger on, not just a single DATA value. Other powerful and user-friendly features included in UART-RS232bus TD trigger include:

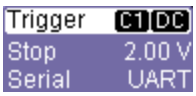
- Ability to define the UART byte with 9-bit DATA, with the 9th DATA bit functioning as an alert bit with a value settable to 0, 1, or X.
- Ability to define as few as 5 bits of DATA in the UART byte.
- Polarity settable to either IdleLow or IdleHigh.
- Decoding in Binary, Hexadecimal, or ASCII formats.
- Triggering on up to 12 bytes of DATA in a data string up to 2048 bytes long.
- Ability to define the frame the UART byte messages into a single long message packet for purposes of triggering.
- Shortcut setup for RS-232 triggering and decoding.

If you are not familiar with or are just learning about UART or RS-232, start by using the simplest trigger conditions (single data byte, any position). Then, experiment with the Interframe Time Setup to "frame" the UART messages into message packets, and trigger on a specific byte value at a known location. Lastly, try triggering on multiple bytes conditionally (INRANGE, or GREATER THAN) in a known location.

Accessing Serial Triggers

The UART-RS232bus serial trigger is accessed from the oscilloscope trigger dialog. Use one of the following methods:

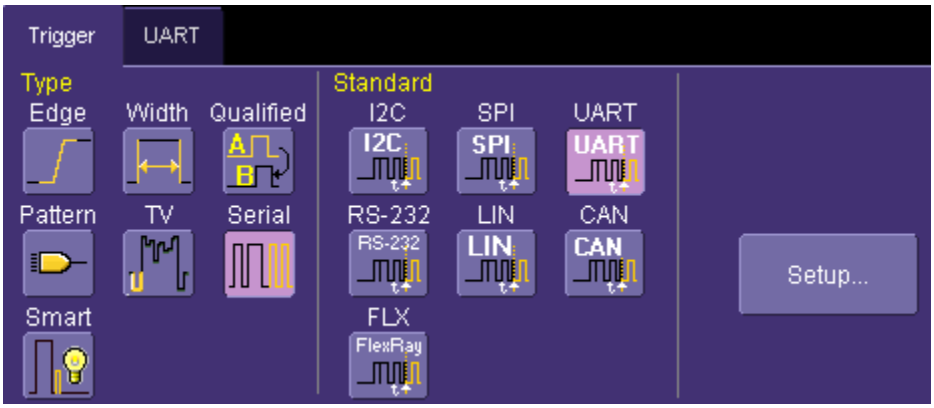
- Touch the Trigger Descriptor Box in the lower right hand corner of the oscilloscope display.



OR

- Touch **Trigger → Trigger Setup** from the Menu Bar. On the Trigger dialog, touch **Serial** on the Type section.

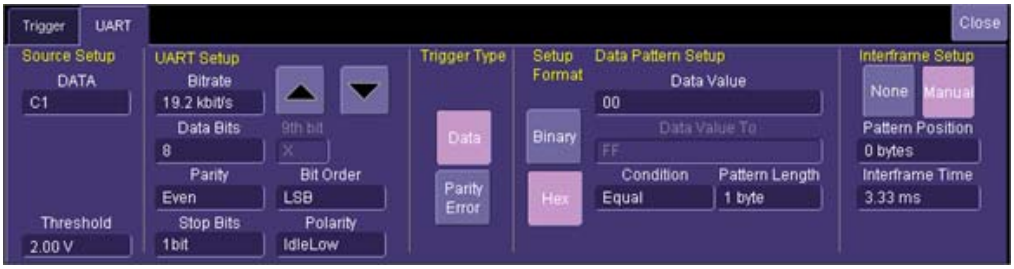
  Select the appropriate serial trigger. The menu automatically changes to a different tab in the Trigger dialog reflecting the selected standard.

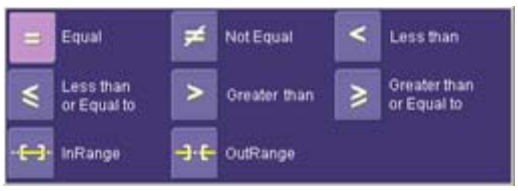


Creating a UART-RS232bus Trigger Condition

The UARTbus Trigger dialog, with detail on some of the setup conditions, is shown in the following topics.

Note: The RS-232 Trigger dialog is nearly the same, but contains less flexibility. Therefore, only the UART Trigger dialog is described here.





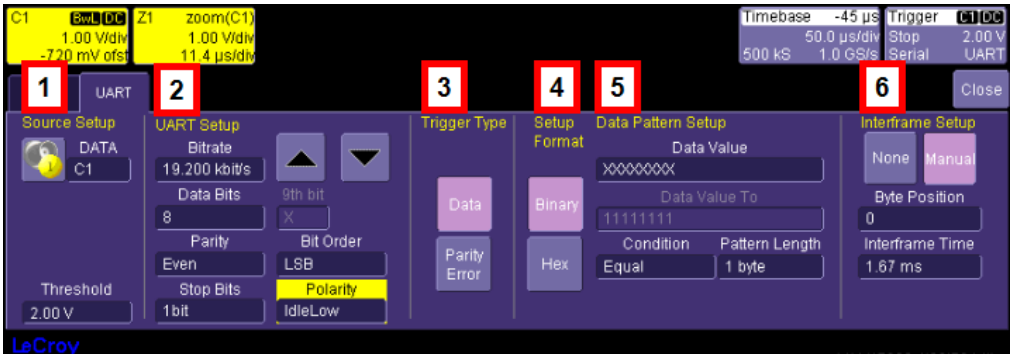
The Source and UARTbus Setup information must be defined. The datasheet for your part should contain the information you need to properly setup the UART Trigger.

Selection of Trigger Type results in dynamic changes to the UART Trigger dialog. Simple Parity ERROR triggering requires no additional setup, while DATA triggers require defining of the Data Pattern, selection of Condition, etc. Also, if you are looking for the exact Position of DATA, then the Interframe Time must be defined.

Select condition values by touching fields (using your finger, or use a mouse pointer). A pop-up dialog box is shown where selections can be made.

UART-RS232bus Trigger Setup Detail

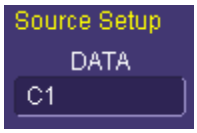
The following topics show the dialog selections for a UARTbus Trigger (again, since they're almost identical, but UART contains a bit more detail than RS-232).



The previously numbered UARTbus trigger sections correspond with the following explanations.

1. Sources Setup

- **DATA** - The pop-up dialog is used to select the appropriate channel or EXT inputs for each. Set these fields up with caution or your trigger may not function correctly.

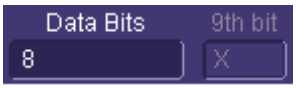


- **Threshold (Trigger) -** Adjust the vertical level for the trigger. Much like an Edge trigger, you must specify the level used to process the incoming signals and determine whether the desired serial data pattern is meeting the set trigger condition.

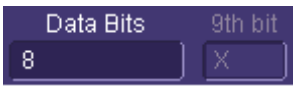
2. UART Setup

- **Bitrate** - Use the Bitrate field to adjust the value and match the bus to which you are connected. This bitrate selection is dynamically linked to the decoding bitrate (they are always the same value). Use the arrows to move through standard bit rates (300 b/s, 1.2, 2.4, 4.8, 9.6, 19.2, 28.8, 38.4, 57.6, 76.8, 115.2, 230.4, 460.8, 921.6, kb/s, 1.3824 1.8432, 2.7648 Mb/s) and make a selection. Or, touch the number twice (with a finger, or using a mouse) and open a pop-up keypad and enter the value directly.

- **Data Bits** - Select the number of data bits per byte (not including the START, STOP, or PARITY bits). Trigger on UART with a 9th DATA bit used as an "Alert" bit by entering Data Bits = 9, and then define the 9th Alert bit as a 0, 1, or X (don't care) as needed.

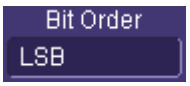


- **Parity** - Choose from **Odd**, **Even**, or **None** in the Parity field. Only when Odd or Even values are made in this field is the **Parity Error** Trigger Type enabled.



- **Stop Bits** - Choose **1**, **1.5**, or **2** Stop Bits in this field.

- **Bit Order** - Choose either Most Significant Bit (**MSB**) or Least Significant Bit (**LSB**) bit order in this field.



Note: This field defaults to LSB and cannot be changed on an RS-232 trigger.

- **Polarity** - Choose the Polarity of the UART signal as either **IdleLow** (Data 1 = High) or **IdleHigh** (Data 1 = Low).

Note: This field defaults to IdleLow and cannot be changed on an RS-232 trigger.

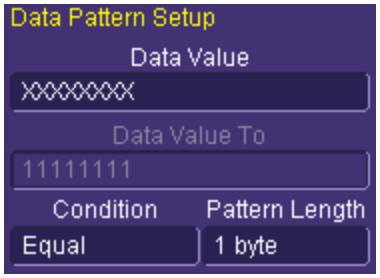
3. Trigger Type

The Data button is selected by default unless **Odd** or **Even Parity** is selected on the **Parity** field. Then, the **Parity Error** Trigger Type button is enabled for use.

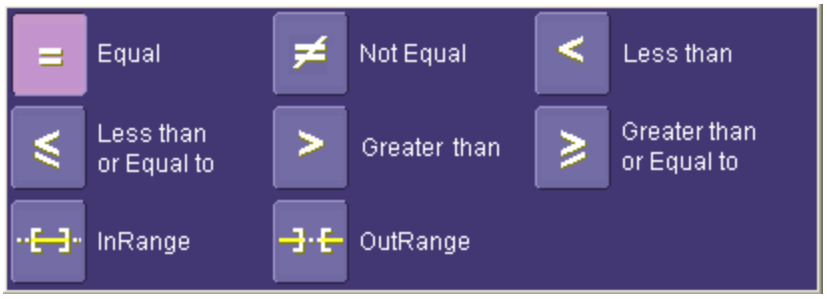
4. Setup Format

Select either **Binary** or **Hexadecimal (Hex)** setup mode. The mode selected affects the format of the following **Data Value** and **Data Value To** fields.

5. Data Pattern Setup



- **Data Value** - Provide an appropriate value based on your Binary or Hexadecimal format selection.
- **Data Value To** - Specify a size in bits for your pattern.
- **Condition** - Possible entries for this field include <=, <, =, >, >=, not =, in a range, out of a range, or don't care.



- **Pattern Length** - The pattern length value defaults to the length, in bytes, of the pattern set in the Data Value selection. If the length is changed to a lesser value, it truncates the beginning of the value. If the length is increased, it would add don't care XX byte values to the beginning of the value.

6. InterFrame Setup

- Click the appropriate button to select **None** or **Manual**. The **Manual** button enables the **Byte Position** and **Interframe Time** fields where you can provide specific values.

UART-RS232bus Decode Setup Detail

The following topic shows the Right-Hand Dialog selections for the **Basic** and **Levels** tabs when UARTbus is selected as the decode protocol (again, since they're almost identical, but UART contains a bit more detail than RS-232). Detailed fields and setup conditions are as follows:

UARTbus Decode Setup Detail - The Basic Tab

- **Viewing** - Select to view the protocol data in either Binary, Hexadecimal (Hex), or ASCII modes.

Note: If the trigger is set up first, the trigger settings copy into the decode settings.

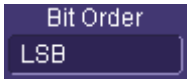
- **Bitrate** - Adjust this bitrate value to match the one on the bus to which you are connected. This bit rate selection is dynamically linked to the decoding bitrate (they are always the same value). Use the arrows to move through standard bitrates (300 b/s, 1.2, 2.4, 4.8, 9.6, 19.2, 28.8, 38.4, 57.6, 76.8, 115.2, 230.4, 460.8, 921.6, kb/s, 1.3824 1.8432, 2.7648 Mb/s) and make a selection. Or, touch the number twice (with a finger, or using a mouse) and open the pop-up keypad to enter the value directly.

- **Data Bits** - Select the number of data bits per byte (not including the START, STOP, or PARITY bits). If you wish to decode on UART with a 9th DATA bit used as an “Alert” bit, select Data Bits = 9.

- **Parity** - Choose from Odd, Even, or None in the Parity field selection box.

- **Stop Bits** - Choose 1, 1.5, or 2 Stop Bits in the Parity field selection box.

- **Bit Order** - Choose either Most Significant Bit (MSB) or Least Significant Bit (LSB) bit order in this selection box.

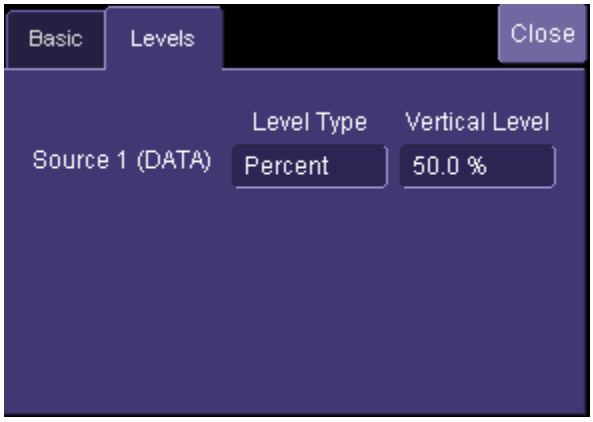


Note: For RS-232 decode, the selection defaults to LSB and cannot be changed.

- **Polarity** - Choose Polarity of the UART signal as either IdleLow (Data 1 = High) or IdleHigh (Data 1 = Low).

Note: For RS-232 decode, the selection defaults to IdleLow and cannot be changed.

UARTbus Decode Setup Detail - The Basic Tab



Source 1 (DATA) **Level Type** and **Vertical Level** - The message decoding algorithm setup is performed here. The level is normally set up in %, and defaults to 50%. Adjust the level by touching inside the field and highlight the box title in yellow, then use the oscilloscope front panel **Adjust** knob to make the change. Alternatively, touch inside the field twice and select a value using the pop-up numeric keypad.

Using the CANbus Options

Overview

The CANbus TD option contains powerful software algorithms to extract serial data information from physical layer waveforms measured on your oscilloscope. The extracted information is overlaid (annotated) on the actual physical layer waveforms, and color-coded to provide fast, intuitive understanding.

The CANbus TD option allows triggering on CAN Frames and Errors. Frame triggering can be set to trigger on any frame, one specific Frame ID, a range of Frame IDs, Remote Frames and Errors. Frame triggering and data triggering can be done for a single ID or message or a range of IDs and data by using the conditional trigger capabilities. Other powerful and user-friendly features included in CANbus TD include:

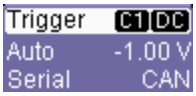
- The ability to trigger and decode CAN at bit rates from 10 kb/s to 1 Mb/s.
- The ability to create powerful, conditional Frame ID and Data triggers.
- Triggering on CAN protocol errors and remote frames.

If you are unfamiliar or are just learning about CAN, start by using the simplest trigger conditions (All Frames or Frame ID). Next, experiment with an ID and Data to trigger on a specific value. Then, try a conditional ID + Data trigger (ID Greater Than or In Range).

Accessing Serial Triggers

The CANbus serial trigger is accessed from the oscilloscope trigger dialog. Use one of the following methods:

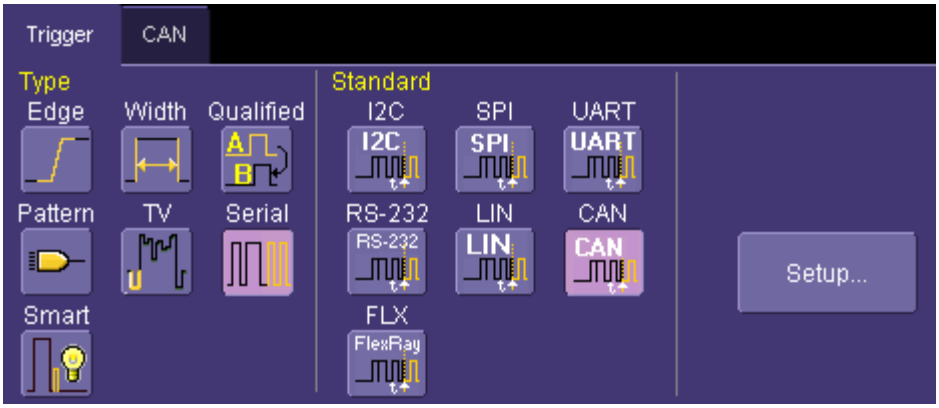
- Touch the Trigger Descriptor Box in the lower right hand corner of the oscilloscope display.

  OR

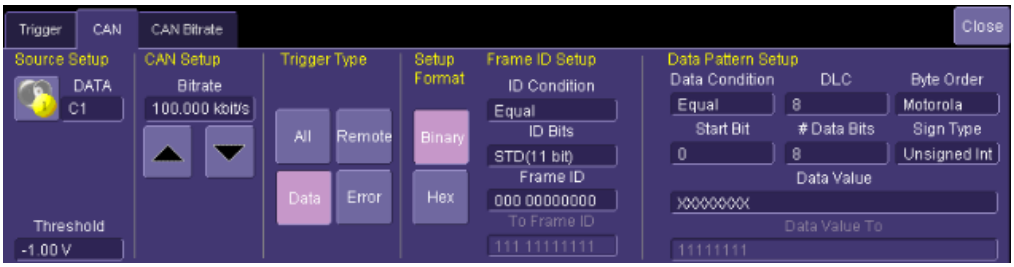
- Touch **Trigger → Trigger Setup** from the Menu Bar. On the Trigger dialog, touch **Serial** on the Type section.

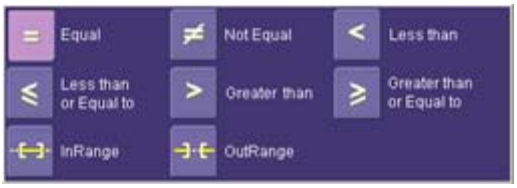
  Select the appropriate serial trigger. The menu automatically changes to a different tab in the Trigger dialog reflecting the selected standard.



Creating a CANbus Trigger Condition

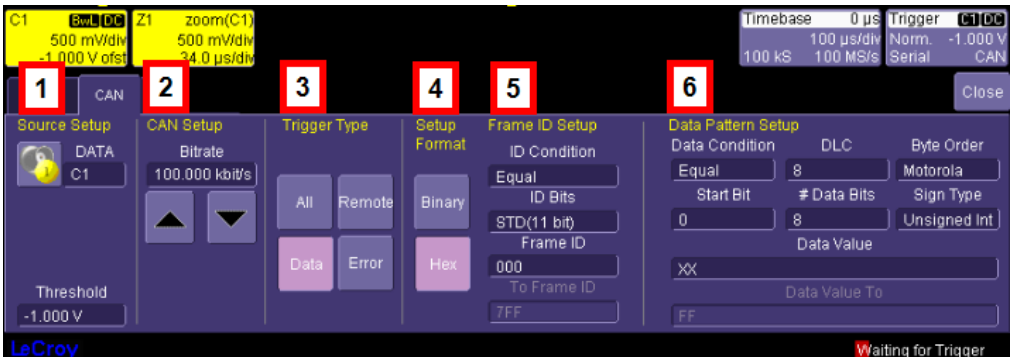
The CANbus Trigger dialog, with detail on some of the setup conditions, is shown in the following topics.





CANbus Trigger Setup Detail

The following topics show the dialog selections for a CANbus Trigger.



The previously numbered CANbus trigger sections correspond with the following explanations.

1. Source Setup

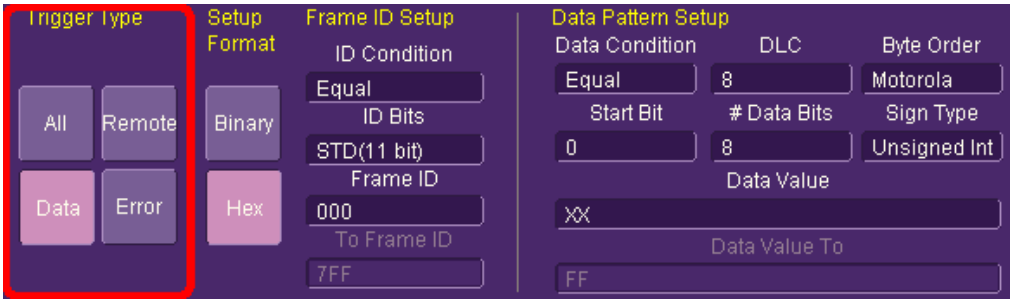
- **DATA** - The **DATA** field's pop-up dialog is used to select the appropriate channel or EXT input for each. Set this field up with caution or your trigger may not function correctly. Use the **Threshold** field to adjust the vertical level for the trigger. Much like an Edge trigger, a user must specify the level used in order to process the incoming signals and determine whether the desired serial data pattern is meeting the set trigger condition.

2. CAN Setup

- **Bitrate** - Use the Bitrate field to adjust the value and match the bus to which you are connected. This bitrate selection is dynamically linked to the decoding bitrate (they are always the same value). Use the arrows to move through standard bit rates (10, 25, 33.333, 50, 83.333, 100, 125, 250, 500, and 1000 kb/s) and make a selection. Or, touch the number twice (with a finger, or using a mouse) and open a pop-up keypad and enter the value directly.

3. Trigger Type

Trigger Type - Depending on your Trigger Type selection, certain Frame ID and Data Pattern Setup fields are enabled or disabled as follows:



- **All** - Triggers on all signals. No Frame ID and Data Pattern ID Setup fields are enabled.
- **Remote** - Only Frame ID Setup fields are enabled.
- **Data** - Both Frame ID and Data Pattern ID Setup fields are enabled.
- **Error** - Triggers only when an error signal occurs. No Frame ID and Data Pattern ID Setup fields are enabled.

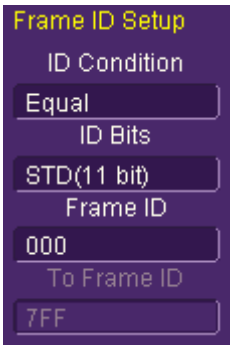
4. Setup Format

Select either **Binary** or **Hexadecimal (Hex)** setup mode. The mode selected propagates through the entire CANbus trigger setup.

Try selecting Binary mode, and set up the Frame ID in binary format, then re-select HEX mode and set up the data in hexadecimal format. Toggling back and forth between the modes does not result in loss of information.

5. Frame ID Setup

Frame ID Setup is used to trigger on a specific Frame ID value with either 11 or 29 Bits.



When CANbus trigger selections are either Remote or Data, use the Frame ID Setup fields as follows:

- **ID Condition** - The ID condition can be set to many different values. If the ID condition is set to "=", then a data definition can also be set. Any other ID condition precludes setting up a Data condition.

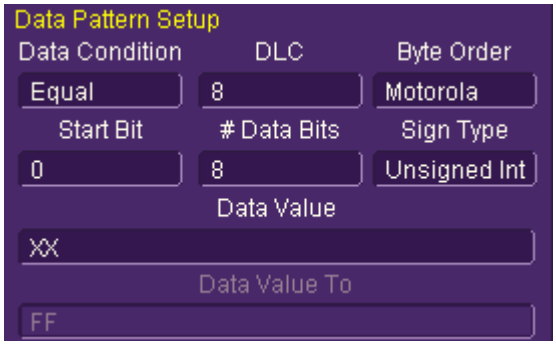
  The ID condition can be set to <=, <, =, >, >=, not =, in a range, out of a range, or don't care.

- **ID Bits -** The trigger can be set to trigger on CAN messages with either 11-bits (Standard CAN) or 29-bits (Extended CAN). You can also set the trigger so that it triggers on a message that meets a condition for either the 11-bit or 29-bit ID. For instance, there might be an 11-bit ID value that is present in both an 11-bit and a 29-bit ID, and by choosing ALL, you could trigger when that ID is present on either of those messages.
- **Frame ID** - Specify the desired frame ID for triggering here.

- **To Frame ID** - When using an in range or out of range ID Condition (previous), specify a To Frame ID value for triggering.

6. Data Pattern Setup

Fields on this section of the dialog are only enabled when using the Data trigger type.



- **Data Condition** - The Data Condition can be set to many different values. The Data condition can be set to <=, <, =, >, >=, not =, in a range, out of a range, or don't care.
- **DLC -** The DLC (data length code) can be set to any integer value from 0 to 8. It should match the DLC of the CAN message you want to trigger on. If you set it to a value less than 0, it will default to 0. If you set it to a value greater than 8, it will default to 8.
- **Byte Order** - Choose from either Motorola (default) or Intel byte orders.
- **Start Bit** and **# Data Bits** - The CANbus trigger allows you to trigger on up to 64 contiguous data bits (8 data bytes). This maximum 64-bit string can start at any location in the CAN message data field - it is not limited to the start of a full byte or a nibble.

  The Start Bit can be any value from 0 to 63. If you enter a value less than 0, it will default to 0. If you enter a value more than 63, it will default to 63. The Start Bit value is always in LSB format (i.e., the bit number as shown on the decoded waveform, with bit 0 being at the far left and bit 63 being at the far right of the data string). Remember that the 1st data byte is bits 0-7, the 2nd data byte is bits 8-15, etc. Also, make sure that your Start Bit value makes sense in relation to the DLC Value. For instance, a Start Bit value of 32 with a DLC Value of 4 is not going to result in a successful trigger.

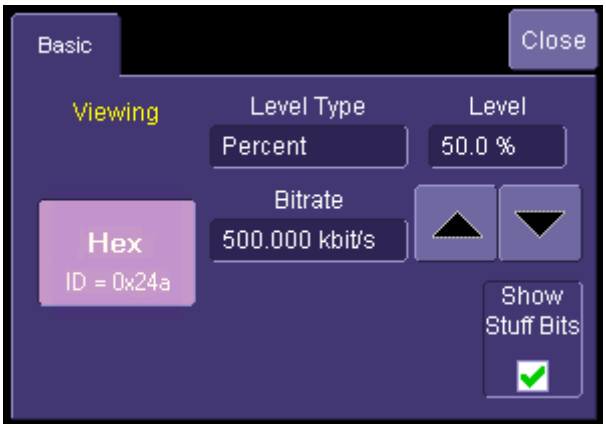
  The # Bits can be any value from 1 to 64. If you enter a value less than 1, it will default to 1. If you enter a value more than 64, it will default to 64.

- **Sign Type** - Choose between signed and unsigned integer format.
- **Data Value** and **Data Value To** - The Data Value is set in Binary or Hexadecimal format. For Hexadecimal, if desired, you can precede the ID value with "0x", but this is not necessary. Be sure to enter a Data Value that matches the DLC Value.

  When using an in range or out of range Data Condition (previous), specify a **Data Value To** value for triggering.

CANbus Decode Setup Detail

CANbus Decode Setup Right-Hand Dialogs are shown when CAN is selected as the decode protocol. It provides detailed fields and setup conditions as follows:



- **Viewing** - The decode format is displayed here as Hexadecimal for CANbus.
- **Bitrate** - Adjust the bit rate value here to match the bit rate on the bus you are connected to. This bit rate selection is dynamically linked to the decoding bit rate (they are always the same value). Use the arrows to move through standard bit rates (10, 25, 33.333, 50, 83.333, 100, 125, 250, 500, and 1000 kb/s) and make a selection. Or, touch the number twice (with a finger, or using a mouse) and open the pop-up keypad to enter the value directly. Any value from 10-1000 kb/s may be entered in this way.

- **Show Stuff Bits** – Use this checkbox to indicate whether you want stuff bits highlighted on each CAN message frame.

- **Level Type** and **Level** - The message decoding algorithm setup is performed here. The level is normally set up in %, and defaults to 50%. To adjust the level, touch inside the number area to highlight the box title in yellow, then use the oscilloscope front panel Adjust knob to adjust. Or touch inside the number area twice and select a value using the pop-up numeric keypad.

  The set Level appears as a dotted horizontal line across the oscilloscope grid.

  If your initial decoding indicates that there are a number of error frames, make sure that the level is set to a reasonable value.

Using the LINbus Options

Overview

Both LINbus TD and D options contain powerful software algorithms to extract serial data information from physical layer waveforms measured on your oscilloscope. The extracted information is overlaid (annotated) on the actual physical layer waveforms, and color-coded to provide fast, intuitive understanding.

The LINbus TD option allows triggering on both Sync Breaks (Start of Frame), Frame ID, Frame ID+DATA, and some ERROR condition. Set DATA triggering conditionally and select a range of DATA values (instead of a single DATA value) on which to trigger. Other powerful and user-friendly features of the LINbus TD trigger include:

- Ability to trigger and decode LIN Version 1.3, 2.x, and SAE J2602 formats, even when LINbus traffic contains mixed versions.
- Ability to decode LINbus in either Binary or Hexadecimal formats.
- Triggering on Checksum, Header Parity, and Sync Byte Errors

If you are not unfamiliar with or are just learning about LIN, start by using the simplest trigger conditions (Break, or Frame ID). Then, experiment with an ID+DATA condition with DATA Equal to a specific value. Then, try a conditional ID+DATA trigger (DATA set to Greater Than or In Range).

Accessing Serial Triggers

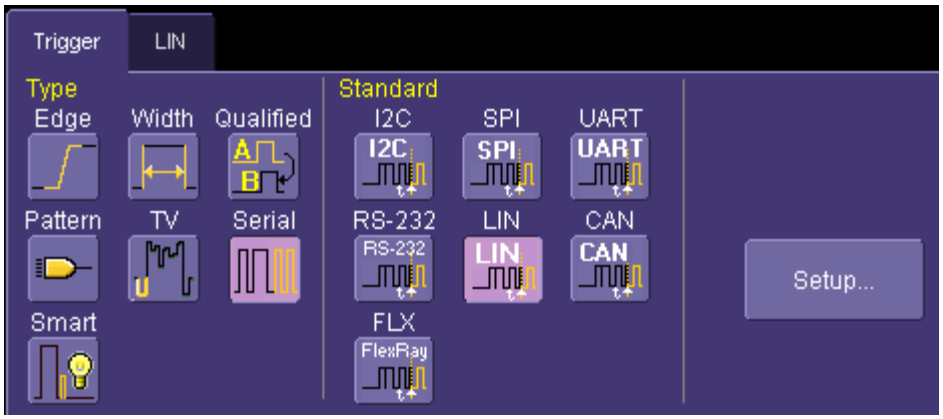
The I^2Cbus serial trigger is accessed from the oscilloscope trigger dialog. Use one of the following methods:

- Touch the Trigger Descriptor Box in the lower right hand corner of the oscilloscope display.

OR

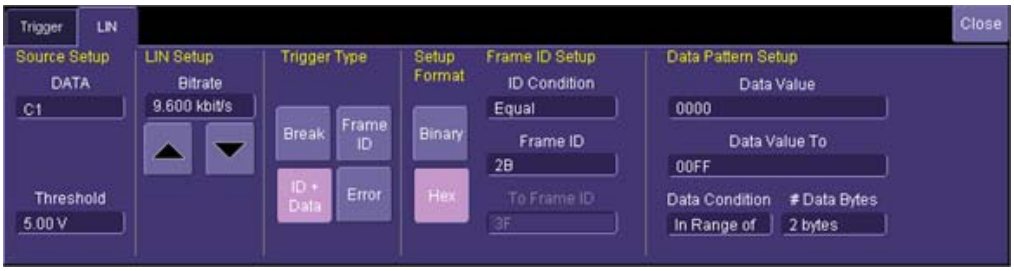
- Touch **Trigger → Trigger Setup** from the Menu Bar. On the Trigger dialog, touch **Serial** on the Type section.

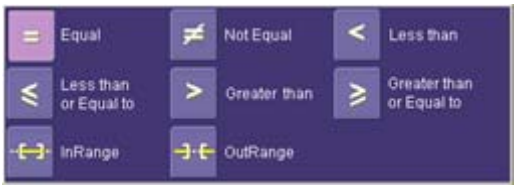
  Select the appropriate serial trigger. The menu automatically changes to a different tab in the Trigger dialog reflecting the selected standard.



Creating a LINbus Trigger Condition

The following trigger setup detail topics show the dialog selections for a LINbus Trigger with detail on some of the setup conditions.





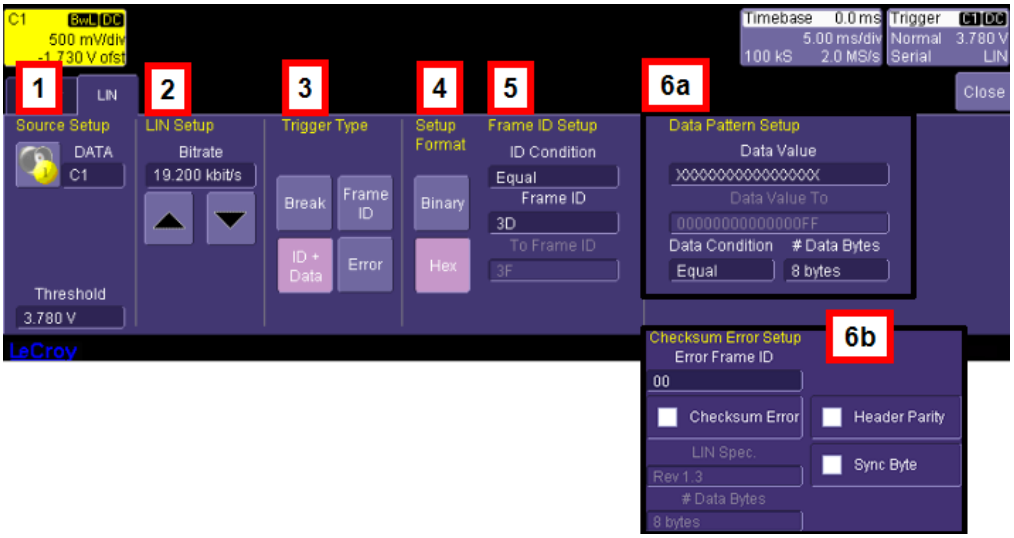
The Source Setup information must be defined. The datasheet for your part should contain the information you need to properly setup the LINbus Trigger.

Previous Trigger Type selections result in dynamic changes to the LINbus Trigger dialog.

Select condition values by touching fields (using your finger, or use a mouse pointer). A pop-up dialog box is shown where selections can be made.

LINbus Trigger Setup Detail

The following topics show the dialog selections for a LINbus Trigger.



The previously numbered LINbus trigger sections correspond with the following explanations.

1. Sources Setup

- **DATA** - The pop-up dialog is used to select the appropriate channel or EXT input for each. Set this field up with caution or your trigger may not function correctly.
- **Threshold (Trigger) -** Adjust the vertical level for the trigger. Much like an Edge trigger, a user must specify the level used in order to process the incoming signals and determine whether the desired serial data pattern is meeting the set trigger condition.

2. LIN Setup

- **Bitrate** - The LIN trigger can be configured to trigger on LIN busses at several different bitrates including 1.2 kb/s, 2.4kb/s, 4.8kb/s, 9.6 kb/s, 10.417kb/s, and 19.2kb/s.

3. Trigger Type

The LIN trigger can be configured to trigger on simple Start of Frame (Break) conditions, ID only, or complete ID+DATA conditions with DATA conditions other than equals. Some Error Frame triggering is also supported.

Choose a desired Trigger Type and the trigger dialog changes based on the selection made.

For example, the following trigger selections disable or enable fields on the **Setup Format**, **Frame ID Setup**, **Data Pattern Setup**, and **Checksum Error Setup** (Checksum error only for Error trigger type) sections of the trigger dialog:

- **Break** - When selected, the Setup Format, Frame ID Setup, and Data Pattern Setup fields are disabled.
- **Frame ID** - When selected, Setup Format and Frame ID Setup fields are enabled, and the Data Pattern Setup fields are disabled.
- **ID + Data** - When selected, Setup Format, Frame ID Setup, and the Data Pattern Setup fields are enabled.
- **Error** - When selected, Setup Format fields are enabled and the Frame ID Setup fields are disabled. The Pattern Setup fields (step 6a, as follows) aren't shown (and are therefore disabled). Instead, the Checksum Error Setup fields (step 6b, as follows) are shown and enabled.

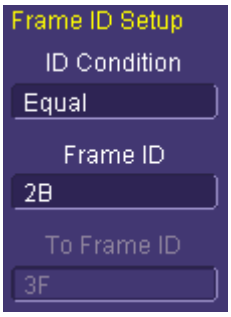
4. Setup Format

With the Frame ID, ID + Data, or Error trigger types chosen, select either the Binary or Hexadecimal setup format. The format propagates through the entire LIN trigger setup.

Toggling back and forth between the formats does not result in lost information (binary is used internally as the core format for all triggering and decoding operations), though use of don't care bits in a binary setup results in the display of an X (for a full nibble don't care) or a $ (for a partial nibble don't care).

5. Frame ID Setup

Frame ID Setup is used to trigger on a specific Frame ID value with either 11 or 29 Bits.



When LINbus trigger selections are either Frame ID or ID + Data, use the Frame ID Setup fields as follows:

- **ID Condition** - Select from the <=, <, =, >, >=, not =, In Range of, or Out of Range conditions available.
- **Frame ID** - Provide a value in either Binary or Hexadecimal mode based on the Setup Format selection made in the previous step.
- **To Frame ID** - If the Frame ID condition (previous) is In Range of or Out of Range, provide a value here to specify the full ID trigger range.

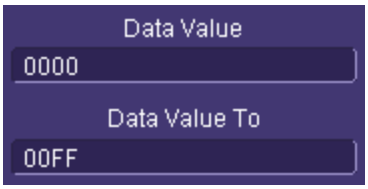
> **Note**: If the Frame ID is equal to 3C or 3D, the **# Data Bytes** field in the following Data Pattern Setup step defaults to 8.

ID + Data Trigger Setup Detail

6a. Data Pattern Setup

When the LINbus trigger selection is **ID + Data**, use the Data Pattern Setup fields as follows:

- **Data Value** - Provide a value in either Binary or Hexadecimal mode based on the selection made in the previous Setup Format step.

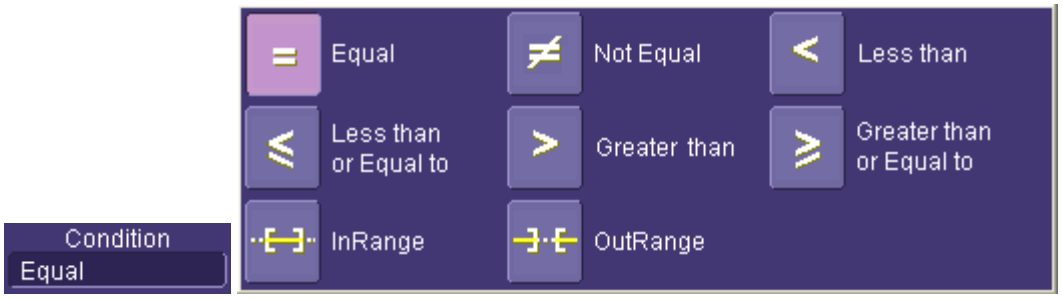


- **Data Value To** - This field is only enabled when the following **Condition** field contains an **In Range of** or **Out of Range** value.

  PLEASE NOTE THE FOLLOWING:

  - Up to 8 bytes of data can be entered as a pattern value.
  - If less than 8 bytes of data is entered for the pattern value, the data is assumed to begin at Data Byte 1 in the LIN message. If this is not desired, then add preceding or trailing don't care (X) nibbles to the pattern value.
  - In Hexadecimal format, data must be entered as full bytes even though the minimum required acceptable entry is a nibble. If less than a full byte is entered, then a don't care X will precede the pattern values entered.

- **Condition** - Select from the <=, <, =, >, >=, not =, In Range of, or Out of Range conditions available.

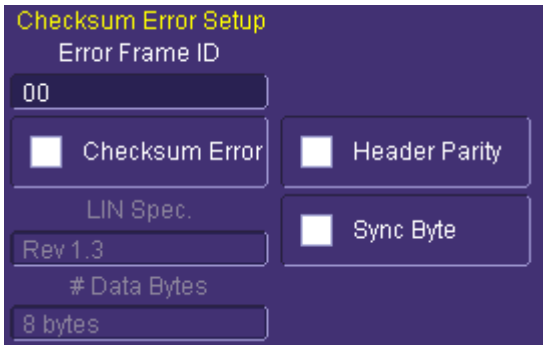


- **# Data Bytes** - This field value defaults to the length, in bytes, of the pattern set in the Pattern Value selection. If you were to change the length to be less than this value, it would truncate the beginning of the pattern value. If you were to increase the pattern length, it would add don't care XX byte values to the beginning of the pattern value. The maximum number of data bytes is 8, per the LIN standard.

Error Trigger Setup Detail

6b. Checksum Error Setup

When the LINbus trigger selection is **Error**, use the Checksum Error Setup fields as follows:



- **Error Frame ID** - Provide a value in either Binary or Hexadecimal mode based on the selection made in the previous Setup Format step.
- Use the **Checksum Error**, **Header Parity**, and **Sync Byte** checkboxes to include or exclude the specific Error Frame Trigger's trigger type.

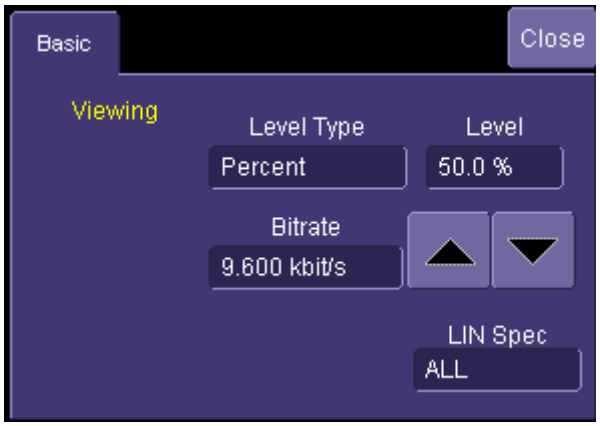
> **Note**: When the Checksum Error checkbox is selected, the **LIN Spec.** and **# Data Bytes** fields are enabled.

- **LIN Spec.** - Select a LINbus specification from the available choices.
- **# Data Bytes** - Provide a value using the pop-up keypad.

> **Note**: The value entered in this field is dynamically copied to the Data Pattern Setup entry for # Data Bytes (ID + Data Trigger Type selection).

LINbus Decode Setup Detail

LINbus Decode Setup Right-Hand Dialogs are shown when LIN is selected as the decode protocol. It provides detailed fields and setup conditions as follows:



- **Viewing** - Select to view the protocol data in either Binary or Hexadecimal formats.

Note: If the trigger is set up first, the setup format (Binary or Hex) made from the trigger dialog is displayed here.

- **Level Type** and **Level** - The message decoding algorithm setup is performed here. The level is normally set up in %, and defaults to 50%. Adjust the level by touching inside the field to highlight the box title in yellow, then use the oscilloscope front panel **Adjust** knob to make the change. Alternatively, touch inside the number area twice and select a value using the pop-up keypad.

PLEASE NOTE THE FOLLOWING:

  - The set Level appears as a dotted horizontal line across the oscilloscope grid.
  - If your initial decode indicates there are a number of error frames, verify your level is set to a reasonable value.

- **Bitrate** - Adjust this bitrate value to match the one on the bus to which you are connected. This bit rate selection is dynamically linked to the decoding bitrate (they are always the same value). Use the arrows to move through standard bit rates (1.2, 2.4, 4.8, 9.6, 10.417, or 19.2 kb/s) and make a selection. Or, touch the number twice (with a finger, or using a mouse) and open a pop-up keypad and enter the value directly. Any value from 1-20 kb/s may be entered this way.

- **Show Stuff Bits** - Use this checkbox to indicate whether you want stuff bits highlighted on each CAN message frame.

Using the FlexRaybus Options

Overview

Both FlexRaybus TD and D options contain powerful software algorithms to extract serial data information from physical layer waveforms measured on your oscilloscope. The extracted information is overlaid (annotated) on the actual physical layer waveforms, and color-coded to provide fast, intuitive understanding. This is especially helpful for FlexRay, an emerging standard that many engineers are just starting to use.

The FlexRaybus TD option allows triggering on TSS (Start), Frame, Symbol or Errors. Conditionally set frame triggers to select a range of Frame ID values on which to trigger, instead of just a single ID. Other powerful and user-friendly features included in FlexRaybus trigger include:

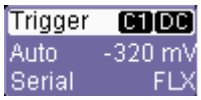
- Ability to trigger and decode FlexRay protocol version 2.1 at 10 Mb/s, 5 Mb/s or 2.5 Mb/s.
- Ability to create powerful Frame triggers including Cycle Count and Frame Qualifiers.
- Triggering on FSS, BSS, FES, Header CRC and Payload CRC errors as well as CID, CAS/MTS and Wakeup Patter Symbols.

If you are unfamiliar with or are just learning about FlexRay, start by using the simplest trigger conditions (TSS, or Frame ID). Next, experiment with an ID + Count Equal to a specific value. Finally, try a conditional ID + Cycle Count trigger (ID Greater Than or In Range).

Accessing Serial Triggers

The FlexRaybus serial trigger is accessed from the oscilloscope trigger dialog. Use one of the following methods:

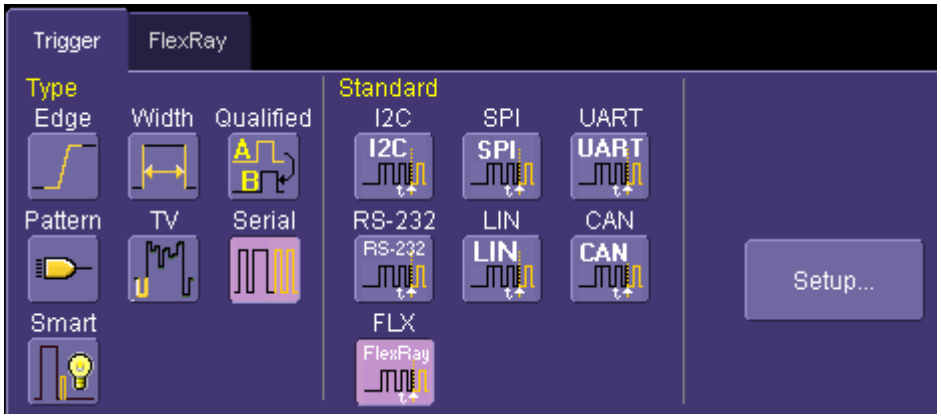
- Touch the Trigger Descriptor Box in the lower right hand corner of the oscilloscope display.



OR

- Touch **Trigger → Trigger Setup** from the Menu Bar. On the Trigger dialog, touch **Serial** on the Type section.

  Select the appropriate serial trigger. The menu automatically changes to a different tab in the Trigger dialog reflecting the selected standard.



Creating a FlexRaybus Trigger Condition

The following trigger setup detail topics show the dialog selections for a FlexRaybus Trigger with detail on some of the setup conditions.

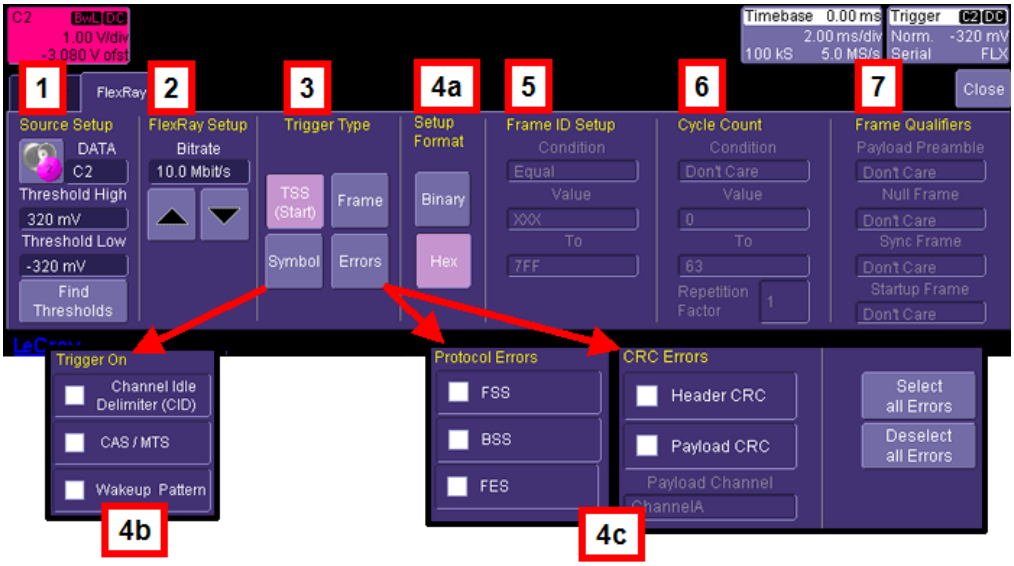
The Source Setup information must be defined. The datasheet for your part should contain the information you need to properly setup the FlexRay Trigger.

Selection of Trigger Type results in dynamic changes to the FlexRay Trigger dialog.

Select condition values by touching fields (using your finger, or use a mouse pointer). A pop-up dialog box is shown where selections can be made.

FlexRaybus Trigger Setup Detail

The following topics show the dialog selections for a FlexRaybus Trigger.



All Trigger Setups use **Source Setup**, **FlexRay Setup**, and **Trigger Type** fields.

Other fields vary based on the Trigger Type Selections as follows:

- **TSS (Start)** and **Frame Triggers** have **Setup Format** fields.
- **Frame Trigger** has **Setup Format**, **Frame ID Setup**, **Cycle Count**, and **Frame Qualifier** fields.
- **Symbol Trigger** has **Trigger On** and **System Parameters** fields.
- **Errors Trigger** has **Protocol and CRC Errors** fields.

Previously numbered FlexRaybus trigger sections correspond with the following explanations.

1. Sources Setup

- **DATA** - The pop-up dialog is used to select the appropriate channel or EXT input for each. Set this field up with caution or your trigger may not function correctly.

- **Threshold (Trigger) High**, **Low**, and **Find** - Adjust the vertical level thresholds for the trigger. FlexRay is a tri-level signal and requires 2 voltage threshold settings which enable the oscilloscope to distinguish between 1 and 0.

  Like an Edge trigger, the level must be specified to process the incoming signals and determine if the desired serial data pattern meets the set trigger condition.

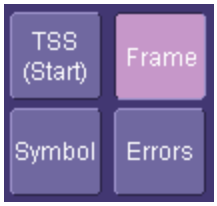
  If desired, use the automated **Find Threshold** button to detect and set appropriate thresholds.

2. FlexRay Setup

- **Bitrate** - The FlexRay trigger can be configured to trigger on FlexRay signals at 2.5 Mb/s, 5 Mb/s and 10 Mb/s as defined in the FlexRay specification.

3. Trigger Type

The FlexRay trigger can be configured to trigger on simple TSS (Start), FlexRay Frame (ID, Cycle Count, Frame Qualifiers) or FlexRay Symbols (CID, CAS/MTS and Wakeup Pattern) In addition Error Frame triggering is supported for FSS, BSS and FES, Header CRC and Payload CRC errors.



Select the Trigger Type desired. The trigger dialog dynamically changes based on the selection made in the following manner:

- **TSS (Start)** - When selected, the Setup Format fields are shown and enabled. However, the Frame ID Setup, Cycle Count, and Frame Qualifiers fields, while shown, are disabled.
- **Frame** - When selected, the Setup Format, Frame ID Setup, Cycle Count, and Frame Qualifiers fields are shown and enabled.

  Note: Some Frame ID Setup and Cycle Count fields are enabled based on selected condition values indicated in respective detail sections).

- **Symbol** - When selected, the Setup Format, Frame ID Setup, Cycle Count, and Frame Qualifiers fields aren't shown (and are therefore disabled). Instead, the Trigger On and System Parameters fields (step 4b and 5b, as follows) are shown and enabled.
- **Errors** - When selected, the Setup Format, Frame ID Setup, Cycle Count, and Frame Qualifiers fields aren't shown (and are therefore disabled). Instead, the Protocol Errors, CRC Errors, Select all Errors, and Deselect all Errors fields (step 4c and 5c, as follows) are shown and enabled.

  Note: When the Payload CRC checkbox on the CRC Errors section is selected, the Payload Channel field is enabled. Select a channel as desired.

TSS (Start) or Frame Trigger Setup Detail

4a. Setup Format - TSS (Start) or Frame Trigger Setup Only

When the FlexRaybus trigger selection is **TSS (Start)** or **Frame**, select either Binary or Hexadecimal (Hex) setup mode. The format propagates through the entire FlexRaybus trigger setup.

Note: Completely different fields are shown (instead of Setup Format, Frame ID Setup, Cycle Count, and Frame Qualifiers sections) when Symbol and Error FlexRaybus triggers are used.

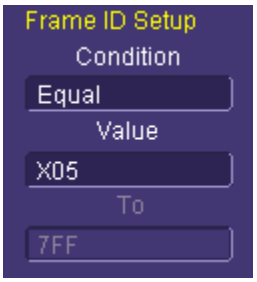
A user can select Binary mode, and set up the address in binary format, then reselect Hex mode and set up the data in hexadecimal format. Toggling back and forth between the modes does not result in lost information (binary is used internally as the core format for all triggering and decoding operations), though use of don't care bits in a binary setup results in the display of an X (for a full nibble don't care) or a $ (for a partial nibble don't care).

Frame Trigger Setup Detail

5. Frame ID Setup - Frame Trigger Setup Only

When **Frame** is selected as the FlexRaybus trigger, use the Frame ID Setup fields as follows:

Note: The Frame ID setup fields are shown, but disabled when **TSS (Start)** is selected as the FlexRaybus trigger. Completely different fields are shown in these sections when Symbol and Error FlexRaybus triggers are used.



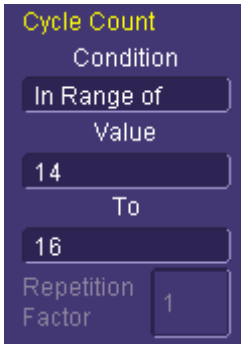
- **Condition** - Select from the <=, <, =, >, >=, not =, In Range of, or Out of Range conditions available. The default setting is Equal.
- **Value** - Use this field's keypad to enter the desired Frame ID. Entering X signifies Don't Care and the default ID is XXX.
- **To** - When the condition is set to **In Range of** or **Out of Range**, select a To value specifying the full ID range for the trigger.

6. Cycle Count - Frame Trigger Setup Only

Cycle Count combines with Frame ID enabling powerful FlexRay triggering. The Cycle Count is a decimal value between 0 and 63 correlating to the FlexRay Cycle Count numbering system. The default Value is Cycle Count 0 (Value).

When **Frame** is selected as the FlexRaybus trigger, use the Cycle Count fields as follows:

Note: The Cycle Count fields are shown, but disabled when **TSS (Start)** is selected as the FlexRaybus trigger.

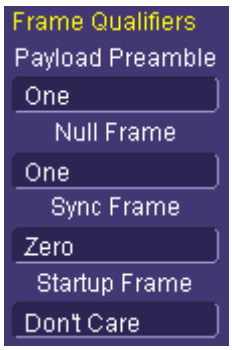


- **Condition** - Select from the <=, <, =, >, >=, not =, In Range of, or Out of Range conditions available. The default setting is Equal.
- **Value** - Use this field's keypad to enter the desired Frame ID. Entering X signifies Don't Care and the default ID is XXX.
- **To** - When the condition is set to **In Range of** or **Out of Range**, select a To value specifying the full ID range for the trigger.
- **Repetition Factor** - When the condition is set to **Equal**, this field can be set to a value of 1, 2, 4, 8, 16, 32 or 64 for triggering when Cycle multiplexing is used.

7. Frame Qualifiers - Frame Trigger Setup Only

Defined in the FlexRay specification, these fields allow an additional level of complexity in creating a very powerful FlexRay trigger. The default Qualifier setting is Don't Care, each field can be set to One, Zero or Don't Care as independent variables in the trigger setup.

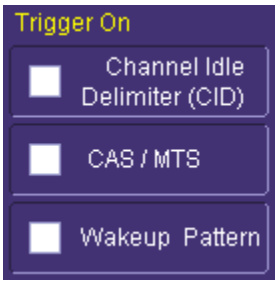
Note: The Frame Qualifiers fields are shown, but disabled when **TSS (Start)** is selected as the FlexRaybus trigger.



Symbol Trigger Setup Detail

4b. Trigger On - Symbol Trigger Setup Only

When **Symbol** is selected as the FlexRaybus trigger, use the Trigger On fields as follows:

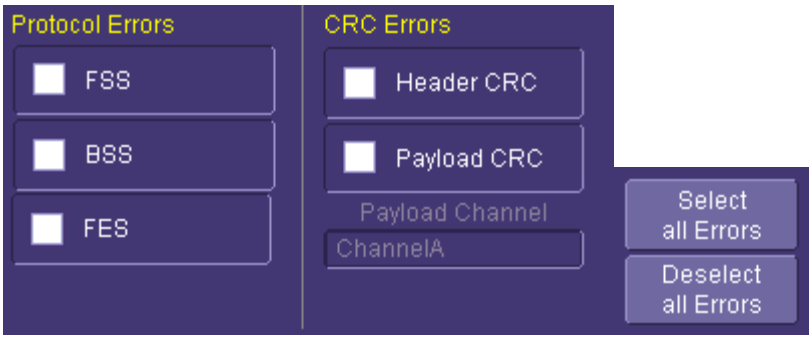


Include or exclude **Channel Idle Delimiter (CID)**, **CAS/MTS**, or **Wakeup Pattern** from your trigger by checking our un-checking as desired. Multiple values may be selected and included in your Symbol trigger.

Errors Trigger Setup Detail

4c. Protocol and CRC Errors - Error Trigger Only

When **Errors** is selected as the FlexRaybus trigger, use the Protocol and CRC Errors fields in the following manner:



- **Protocol Errors** - Include or exclude **FSS**, **BSS**, or **FES** errors from your trigger by checking our un-checking as desired.
- **CRC Errors** - Include or exclude **Header CRC** or **Payload CRC** errors from your trigger by checking our un-checking as desired.
- **Payload Channel** - If you include Payload CRC errors in your Error trigger, use this field to select the correct channel.
- **Select all Errors** and **Deselect all Errors** - Use these buttons to conveniently select or deselect all **Protocol** and **CRC Errors** with a single click.

FlexRaybus Decode Setup Detail

The following topic shows the Right-Hand Dialog selections for the **Basic** and **Levels** tabs when FlexRaybus is selected as the decode protocol. Detailed fields and setup conditions are as follows:

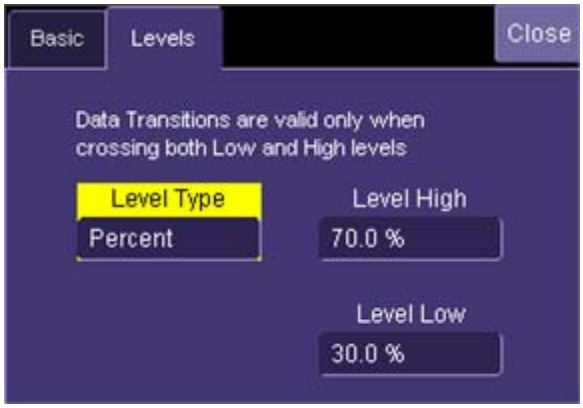
FlexRaybus Decode Setup Detail - The Basic Tab

- **Bitrate** - Adjust this bitrate value to match the one on the bus to which you are connected. This bit rate selection is dynamically linked to the decoding bitrate (they are always the same value). Use the arrows to move through standard bitrates (2.5, 5.0 or 10.0 Mb/s) and make a selection. Or, touch the number twice (with a finger, or using a mouse) and open the pop-up keypad to enter the value directly.

- **Channel** - Select the appropriate channel for decoding (based on whether it's coming from Channel A or Channel B of the FlexRay bus). The channel selection drives the CRC computation.

> **Note**: Decode still works when the wrong channel is selected. It results in CRC errors being shown on the decode. Fix it by switching the channel selection.

FlexRaybus Decode Setup Detail - The Levels Tab



The message decoding algorithm setup is based on these field values. FlexRay is a tri-level signal and requires 2 levels for the oscilloscope to distinguish between 1 and 0. The **Level Type** is normally set up as Percent, and defaults to 70% and 30% **Level High** and **Low** values, respectively.

Selecting the **Absolute** Level Type allows entry of voltage levels (on the Level High and Low fields) instead of percentages.

Change the **Level High** and **Low** values by selecting the field (so it's highlighted) and using the oscilloscope front panel **Adjust** knob to provide a new amount. Alternatively, select the field twice and provide a value using the pop-up keypad.

Change **Level High** and **Low** values by selecting the field (so it's highlighted) and use the oscilloscope front panel **Adjust** knob to provide a new amount. Alternatively, select the field twice and provide a value using the pop-up keypad.

The Level set is then shown as a dotted horizontal line on the oscilloscope grid.

Note: If your initial decoding indicates a number of error frames, ensure your level is set to a reasonable value.

Using the TD Packages: Characterizing Embedded Controller Performance

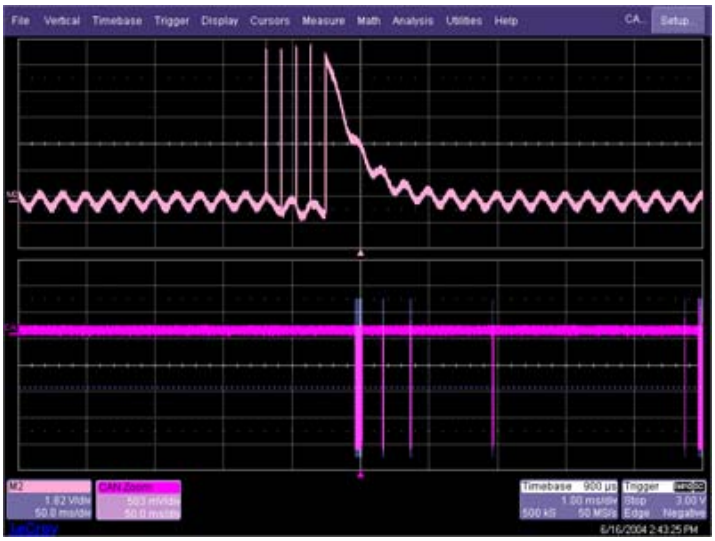
Overview

The standard oscilloscope contains a number of built-in tools, such as cursors, measurement parameters, and statistical analyzers. They can be used to characterize performance for serial data signals (just as they are also used to characterize performance on other signals). You may want to use cursors to make single-shot timing measurements and measurement parameters when you need to accumulate statistical data over many different acquisitions. Measurement parameters are also helpful to determine the underlying integrity of the serial data physical signals.

All TD packages provide basic tools to characterize embedded controller performance. The tools can be used on the decoded channels, memories, zooms, functions, etc. just like they are used on any un-decoded channels, memories, zooms, functions, etc. You also can use normal Edge or SMART Triggers on an analog channel input to trigger the oscilloscope when a certain analog signal occurs, and then measure to a particular serial data message using the decoded info as your guide.

Note: The following examples use CANbus messages; however, similar needs exist for I^2C and SPI, and the included oscilloscope tools described in the following sections can be applied in the same way.

Take the following example of an analog signal creating a burst of CAN messages:



This data was acquired over a 500 ms duration. It is likely you want to understand whether the analog signal input to your electronic control unit (ECU) is creating the desired CAN message output from the ECU. There are a few ways this can be done as the following topics explain.

Characterization Using Cursors

Use horizontal cursors to mark locations on the waveform where the time measurement should be done, and then read the cursor values to establish the measurement. Adjust the timebase or create zooms of the decoded trace(s) as needed in order to view the signal with enough detail. This is a good method for single-shot / single measurements.

Characterization Using Measurement Parameters

Measurement parameters can be used to make signal integrity or timing measurements of serial data signals.

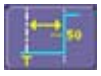
Basic parameters, such as Amplitude, Rise, Fall, Overshoot, etc. are ideal for signal integrity checks. Timing parameters, such as Delay, Delta Delay, Delta Time @ Level, etc., are ideal for measuring timing from trigger to other signals (such as from an I^2C or SPI Trigger to an analog signal). Delta Trig Time is ideal for measuring the time between segments of a Sequence Mode acquisition.

Please see Isolating and Analyzing Serial Bus Activity topics for more information on Sequence mode.

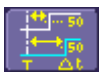
- **Amplitude** - Noise and overshoot resistant measurement of the amplitude of the signal (measurement of amplitude from Top to Base).

- **Base** - Value of the lowermost state in a bi-modal waveform, such as an I^2C, SPI, or CAN Message.

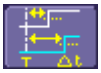
- **Delay** - Time from the trigger to the first transition at the 50% amplitude crossing.



- **Delta Delay** - Time between the 50% crossing of the first transition of two waveforms.

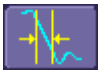


- **Delta Time @ Level** - Time between selectable levels of two waveforms. (Not available on WaveSurfer Series)



- **Delta Trig Time** - The time from last trigger to this trigger (usually used in Sequence mode).

- **Fall (90-10), Fall 80-20, Fall@Level** - Transition time on the falling edge. Three selections are available for the user to determine at which vertical level the measurement is made. (Fall@Level is not available on WaveSurfer Series)



- **Maximum** - Highest value in the input waveform.

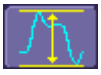
- **Mean** - Average of all data values.

- **Minimum** - Lowest value in the input waveform.

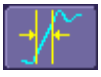
- **Overshoot Negative** - Overshoot following a falling edge.

- **Overshoot Positive** - Overshoot following a rising edge.

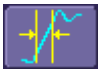
- **Peak to Peak** - Difference between the Maximum and Minimum data values.



- **Rise (10-90), Rise (20-80), Rise@Level** - Transition time on the rising edge. Three selections are available for the user to determine at which vertical level the measurement is made. (Rise@Level is not available on WaveSurfer Series).



- **Top** - Value of the uppermost state in a bi-modal waveform, such as an I^2C, SPI, or CAN Message.

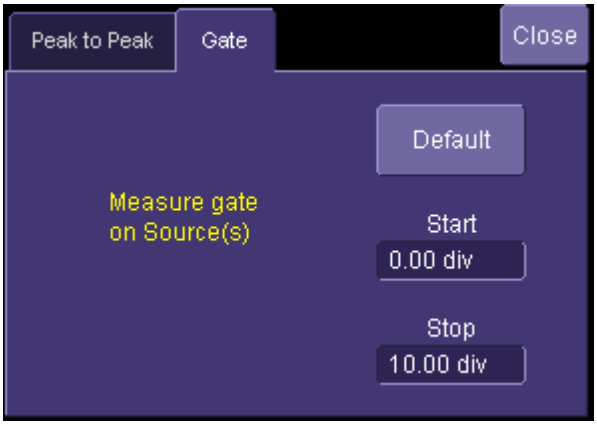


Gating with Measurement Parameters

Gating is available on each standard parameter. This allows you to set a measurement window in which the parameter should be made active. This also allows you to eliminate unwanted portions of the acquisition from your measurement.

Select gating from the Measure dialog by selecting the tab for the appropriate measurement (P1, P2, etc.), and then setting start and stop values for the gate.

Please refer to your oscilloscope's online help for more information about Gating.

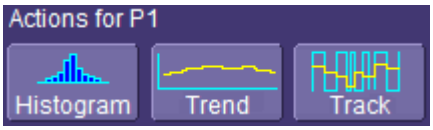


Statistics and Graphing with Measurement Parameters

Statistics and Histicons are included with nearly every LeCroy oscilloscope (Histicons are not available with WaveSurfer Series). They allow you to gather numerical and visual information on the distribution of your various measurements.

Turn on Statistics and Histicons separately in the Measure dialog. Just touch the checkbox to turn it ON and touch again to turn it OFF.

In addition, some optional LeCroy programs (such as JTA2) add the ability to produce larger histograms, trends, and tracks of your measurement parameters. If you have this capability, access it through the Measurement Parameter setup dialog (the Px tab).



Pass/Fail Analysis with Measurement Parameters

Set up Pass/Fail conditions by touching **Analysis → Pass/Fail Setup** on the menu bar.

Refer to your oscilloscope's online help for more pass/fail setup detail.

Pass/Fail analysis using measurement parameters is simple to set up and powerful. For instance, you can define a timing measurement, define the limits for the timing measurement, and then run the oscilloscope in a Normal trigger mode, capturing thousands of measurement events.

What's more is Pass/Fail can then be used to save the Waveform in the event of a Fail, or send an email in the event of a fail.

Isolating and Analyzing Serial Bus Activity

Overview

The combination of Serial Data Triggering, Decoding, and normal oscilloscope features is a powerful combination of tools that can make it very easy to find latent Serial Data HW or SW problems in your circuit. No longer is the oscilloscope a tool just for the hardware engineer. Now, software engineers can also easily visualize the Serial Data signals and relate them to programming code and operation. The TD options can enable the HW Engineer and SW Engineer to "speak the same language" when it comes to system debugging and performance checking.

Some common Serial Data analysis needs and methods are discussed in the following topics.

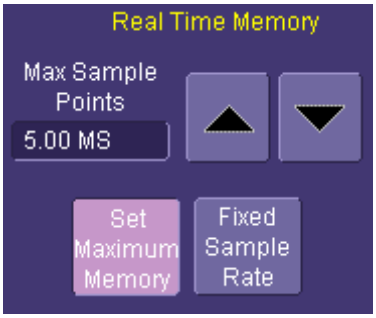
Capturing Long Pre-Trigger Times

LeCroy oscilloscopes are available with optional, very long acquisition memory. For instance, the WaveRunner Xi Series oscilloscopes can capture up to 12.5 Mpts on 4 channels, or 25 Mpts on 2 channels. If your Serial Data signals are 1 Mb/s, and you sample at the minimum required and available sample rate (5 MS/s) you can capture 5 seconds of Serial Data traffic.

If necessary the capture can be 100% pre-trigger, 100% post-trigger, or something in between.

1. First, adjust Pre-Trigger and Post-Trigger time using the **Delay** knob on the oscilloscope's front panel.

2. Now, optimize your Sample Rate or Memory Length by accessing the Horizontal Dialog on your oscilloscope and selecting either **Set Maximum Memory** mode or **Fixed Sample Rate** mode.



- If you choose to **Set Maximum Memory**, you can decrease the memory usage so you do not sample at too high a sample rate (too high a sample rate slows down the decoding algorithm). Then, adjust your timebase setting to a length sufficient to capture the event.

> **Note**: Make sure your timebase setting and memory length combined do not result in too low a sample rate. Otherwise, adequate capture and decode is not performed.

- More commonly, you will probably choose to **fix the sample rate** to a specific value providing the necessary oversampling required to capture your Serial Data messages (at least 4X the bit rate). Also, it affords a high enough sample rate to capture transients you may want to see on Serial Data and analog signals (at least 2X the frequency of any expected transients, preferably 10X).

Reference your oscilloscope's online help for more information about the core oscilloscope settings aforementioned in this topic.

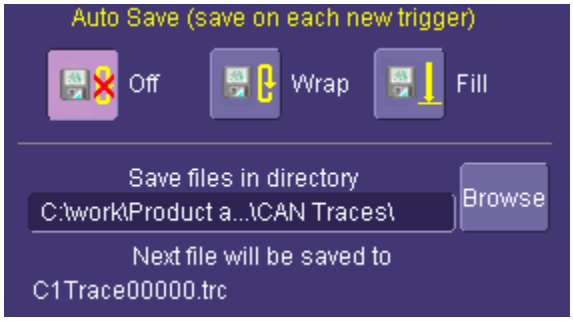
Repeatedly Triggering and Saving the Data to a Hard Drive

You may wish to set up your oscilloscope to capture a short or long memory acquisition for a certain trigger condition, and then save data to a hard drive or memory stick whenever the trigger condition is met.

Note: While this can be easily done in most LeCroy oscilloscopes, realize there is significant trigger "dead time" while using this method. Minimize dead time by using the method described in the following Repeated Triggering and Storing all Triggers (Sequence Mode) topic.

Repeatedly trigger and save the data to a hard drive by doing the following:

1. First, set up your desired serial data (or other) trigger condition (see corresponding protocol sections previously covered in this documentation for details).
2. Now, choose **File → Save Waveform** from the menu bar. The **Auto Save** dialog is shown where you can set up the Save Waveform conditions.



As the previous screen-shot shows, you can disable the Auto Save function by clicking the OFF button. Or, select WRAP where Auto Save occurs until the hard drive is filled, and then discards the oldest data in order to write the newest data. Lastly, select FILL which Auto Saves until the hard drive is filled and then stops.

PLEASE NOTE THE FOLLOWING:

- Be sure to choose a Binary file format if you wish to recall the traces into a LeCroy oscilloscope for later analysis.
- Even though the LeCroy oscilloscope hard drives are very large, it is a good idea to make sure your trigger condition is set correctly before running your acquisitions.

Repeatedly Triggering and Storing All Triggers (Sequence Mode)

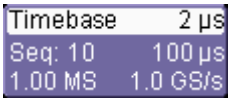
LeCroy oscilloscope's have a powerful Sequence Mode function which stores all triggered events by minimizing the dead time between triggers to < 800 nanoseconds. It's ideal for finding repetitive problem causes on your serial data buses or associated signals. (Sequence Mode is not available on WaveSurfer Series oscilloscopes.)

Sequence Mode uses long acquisition memory divided into "segments." As triggered events are acquired, they are stored in acquisition segments for recalling at a later date. The length of each sequence mode acquisition segment and the total number of segments allowed is roughly determined by the total acquisition memory in the oscilloscope. For instance, on a WaveRunner Xi you can acquire 10,000 segments each a maximum of 625 samples long, or 10 segments each a maximum of 1.25 megasamples long, or something in between. Different acquisition memory lengths have different ranges of segments and segment lengths. You can define any number of segments from 2 to the maximum for that memory length (refer to your oscilloscope's specifications for details) and any length of segment (provided there is sufficient acquisition memory). After acquisition of all segments is complete, you can recall them one-by-one and view them in decoded format on the oscilloscope screen.

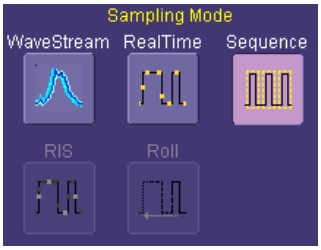
Acquisition dead time is kept to a minimum because there are no operations performed during the acquisition. All data for each triggered event is written only into high-speed acquisition memory. Until the entire sequence is completed, there is no updating of the oscilloscope display, or other operations causing unnecessary dead time. It's ideal for situations where you cannot take a chance on losing data.

In the following example, we have only acquired Channel 1 in sequence mode. Keep in mind, additional analog or other signals can also be acquired (if desired or necessary) in order to perform a more proper analysis.

1. Touch the **Timebase** trace descriptor label to open the Timebase dialog.



2. In the Sampling Mode area, select Sequence Mode.



   Touch the tab labeled **Sequence** that is now shown next to the Timebase tab.

3. On the Sequence tab, select the **Display Mode** and select the **Number of Segments Displayed** at one time.

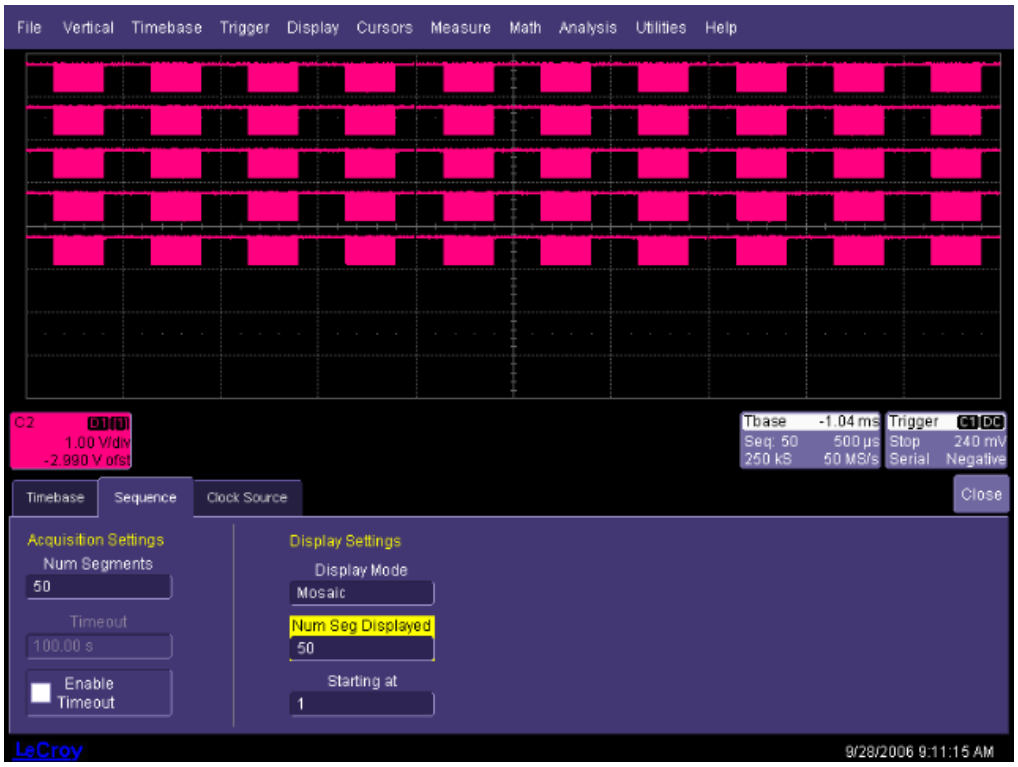
   Note: If you have acquired more segments than you can display at one time, use the **Starting at** field to specify a segment at which to begin the display.

Serial Trigger Setup for Specific Events

At this point, the Serial Trigger is now set up to capture the desired event. The following sections provide event-based setup explanations.

- You can trigger on a specific address or data value and capture long pre-trigger time to determine what precedes the message.

  Our example here uses an I^2C Start trigger. Start the sequence mode acquisition by pressing the front panel SINGLE trigger button. Each time the trigger condition is met, the TRIG'D light on the front panel flashes. When you've acquired the set number of segments, the trigger STOPS and a display similar to the following is shown (this is a 50 segment acquisition in Mosaic display mode).

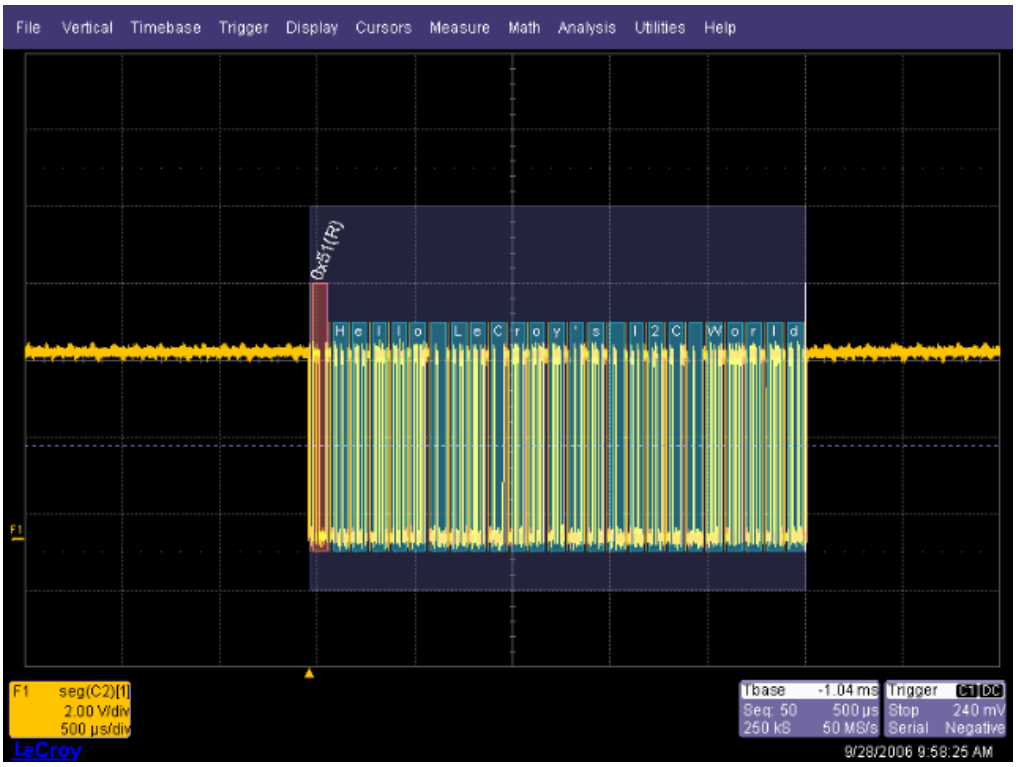


- You can display an individual segment separately from the main channel display by selecting **Math → Math Setup** from the menu bar. Choose a math trace to define as a Segment (in this case, F1 is defined as a Segment of C2). Use the channel that your serial data was acquired on (in this case Channel 2) as a Source.

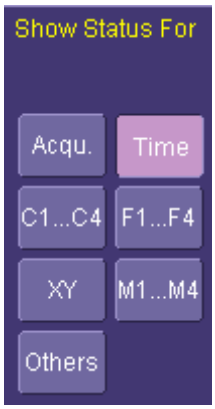
Display the trace by checking the TRACE ON checkbox. Select the segment for viewing by touching the Select tab and selecting a segment using the pop-up keypad or the front panel Adjust knob.

- You can view decoded data on the individual segment by setting up the Decode to use the Math trace as the source for Data (in this case F4 is the Source). If you wish to change the segment that is decoded, just select a new segment from the Math trace dialog (as shown in the previous step).

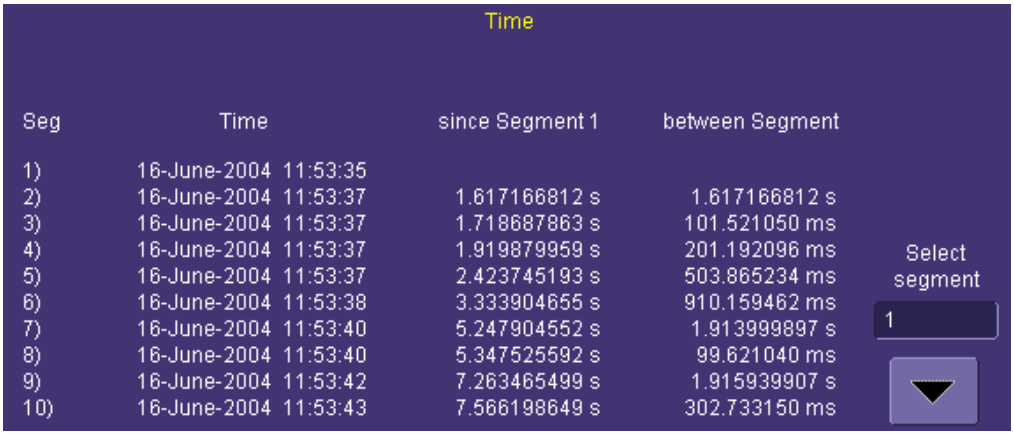
Note: Conserve display space by turning off the Channel and only select the segment you wish to view as the following screen-shot shows.



- You can view the timestamps for each segment by selecting **Vertical → Channels Status** from the menu bar and selecting **Time** on the **Show Status For** section of the dialog as shown.



A display of timestamp information for each segment in the sequence acquisition is then shown (as follows).



Note: Ten timestamps fit on the display at one time. Choose which segments to display by using the **Select Segment** control. You can also page through the segments one at a time by using the **Adjust** knob on the front panel.

I²Cbus Specifications

Note: Specifications are subject to change without notice.

Definition	I²Cbus
Protocol Setup	Not Applicable

Decode Capability	I²Cbus
Format	Hexadecimal, Binary, ASCII
Decode Setup	Threshold definition required. Default is to Percent amplitude. Choose to Decode address values including/not including the R/W bit in address value.
Decode Input	Any analog Channel, Memory or Math trace. Clock channel may be turned OFF and data still decodes (reduces screen clutter).
Number of Decoded Waveforms	**Up to 4 buses may be decoded at one time.** In addition, zooms can be displayed (with decoded information).
Location	Overlayed over DATA waveform, on Grid. (Use multi-grid if there is more than one decoder ON)
Visual Aid	Color Coding for FRAME, START/ReSTART bit, ADDR, R/W, DATA, ACK, and STOP bit. Decode information is intelligently annotated based on timebase setting.

Trigger Capability	I²Cbus
Format	Hexadecimal or Binary ADDRESS and DATA can be set up with different formats.
Trigger Setup	Trigger on START, ReSTART, STOP, Missing ACK, ADDR, DATA, ADDR+DATA, ADDR+DATA Length, EEPROM.
Address (ID) Condition Setup	Specify One 7 or 10 bit ADDRESS supported with full Read, Write, or R/W="Don't Care" Choose to Trigger on address values that include/don't include R/W bit in address value.
Conditional Trigger Setup	Conditional Data triggering available. Choose from <=, <, =, >, >=, <>, in range, out of range or don’t care conditions.
Data Setup	**Hexadecimal:** # Data Bytes = 0 to 12. Data can be defined by nibble. **Binary:** Any combination of 0,1, or X for 1-96 bits. Data pattern can be set to start on any byte in a 2048 byte window (EEPROM mode).
Ack Condition	For any ADDR, ADDR+DATA, ADDR+DATA LENGTH, or EEPROM frame setup, select an ACK Condition of ACK, NO ACK, and DON'T CARE.
Bit Rates	Full range over I²C specification for Standard, Fast, Fast-Mode Plus, and High-Speed modes. Auto-detected.
Trigger Input	Any analog Channel or the EXT input. Clock may be input to EXT to conserve available analog Channels.

Trigger Capability	I^2Cbus
Trigger Design	Internal to oscilloscope, settable like any other oscilloscope trigger.

Search Capability	I^2Cbus
Search Options	Search by any Packet, ADDRESS, or DATA in Hexadecimal formats.

Compatibility	I^2Cbus
Compatible with...	TD (Trigger & Decode) Option fully compatible with WPZi, WRXi, and WSXs D (Decode) Option fully compatible with WR6000, WP7000, and WM8000 Series.

SPIbus Specifications

Note: Specifications are subject to change without notice.

Definition	SPIbus
Protocol Setup	Select CPOL (Clock Polarity 0 or 1), CPHA (DATA Polarity 0 or 1), DATA = MSB or LSB. Also, may select SIOP or SSPI defaults.

Decode Capability	SPIbus
Format	Hexadecimal, Binary, ASCII
Decode Setup	Threshold definition required. Default is to Percent amplitude. Select CPOL, CPHA, DATA = MSB, or LSB.
Decode Input	Any analog Channel, Memory or Math trace. Clock and/or Slave Select channel may be turned OFF and data still decodes (reduces screen clutter).
Number of Decoded Waveforms	**Up to 4 buses may be decoded at one time.** In addition, zooms can be displayed (with decoded information).
Location	Overlayed over DATA waveform, on Grid. (Use multi-grid if there is more than one decoder ON)
Visual Aid	Color Coding for FRAME and DATA. Decode information is intelligently annotated based on timebase setting.

Trigger Capability	SPIbus
Format	Hexadecimal or Binary
Trigger Setup	Trigger on DATA for any of the four SPI Modes with either MSB/LSB or (with or without) Slave Select.
Address (ID) Condition Setup	Not Applicable
Conditional Trigger Setup	Trigger on Data values with = condition.
Data Setup	**Hexadecimal:** # Data Bytes = 0 to 8. Data can be defined by nibble. Triggers on that data pattern regardless of position. **Binary:** Any combination of 0,1, or X for 1-64 bits. Triggers on that data pattern regardless of position.
Ack Condition	Not Applicable
Bit Rates	Any. Auto-detected.
Trigger Input	Any analog Channel or the EXT input. Clock may be input to EXT to conserve available analog Channels.
Trigger Design	Internal to oscilloscope, settable like any other oscilloscope trigger.

Search Capability	SPIbus
Search Options	Search by DATA in Hexadecimal formats.

Compatibility	SPIbus
Compatible with...	TD (Trigger & Decode) Option fully compatible with WPZi, WRXi, and WSXs D (Decode) Option fully compatible with WR6000, WP7000, and WM8000 Series.

UART - RS232bus Specifications

Note: Specifications are subject to change without notice.

Definition	UART - RS232bus
Protocol Setup	**For UART:** • Select BitRate Select # Data Bits (5-9) • Select # Stop Bits (1, 1.5, 2) • Select Bit Order (MSB or LSB) • Select Polarity (IdleLow or IdleHigh) **For RS-232:** • Select BitRate Select # Data Bits (5-8) • Select Parity (Odd, Even, None) • Select # Stop Bits (1, 1.5, 2)

Decode Capability	UART - RS232bus
Format	Hexadecimal, Binary, ASCII
Decode Setup	Threshold definition required. Default is to Percent amplitude. Select BitRate, # Data Bits, Parity, # Stop Bits, Bit Order, and Polarity (for RS-232, no Bit Order or Polarity setup).
Decode Input	Any analog Channel, Memory, or Math trace.
Number of Decoded Waveforms	**Up to 4 buses may be decoded at one time.** In addition, zooms can be displayed (with decoded information).
Location	Overlayed over DATA waveform, on Grid. (Use multi-grid if there is more than one decoder ON)
Visual Aid	Color Coding for Start Bit, Stop Bit, Parity Bit, and DATA. Decode information is intelligently annotated based on timebase setting.

Trigger Capability	UART - RS232bus
Format	Hexadecimal or Binary
Trigger Setup	Trigger on DATA or Parity ERROR.
Address (ID) Condition Setup	Not Applicable
Conditional Trigger Setup	Conditional Data triggering available. Choose from <=, <, =, >, >=, <>, in range, out of range, or don't care conditions.
Data Setup	**Hexadecimal:** # Data Bytes = 0 to 12. Data can be defined by nibble. **Binary:** Any combination of 0,1, or X for 1-96 bits. May specify particular data position anywhere in a 2048 byte sequence.
Ack Condition	Not Applicable
Bit Rates	Any from 300 b/s to 10 Mb/s

Trigger Capability	UART - RS232bus
	(User settable)
Trigger Input	Any analog Channel or the EXT input.
Trigger Design	Internal to oscilloscope, settable like any other oscilloscope trigger.

Search Capability	UART - RS232bus
Search Options	Search by DATA in Hexadecimal format, or search for Next ERRORs.

Compatibility	UART - RS232bus
Compatible with...	TD (Trigger & Decode) Option fully compatible with WPZi, WRXi, and WSXs D (Decode) Option fully compatible with WR6000, WP7000, and WM8000 Series.

CANbus Specifications

Note: Specifications are subject to change without notice.

Definition	CANbus
Protocol Setup	Select Bitrate (10, 25, 33.333, 50, 83.333, 100, 125, 250, 500, 1000 kb/s, or user-defined between 10-1000 kb/s).

Decode Capability	CANbus
Format	Hexadecimal
Decode Setup	Threshold definition required. Default is to Percent amplitude. Select BitRate.
Decode Input	Any analog Channel, Memory, or Math trace.
Number of Decoded Waveforms	**Up to 4 buses may be decoded at one time.** In addition, zooms can be displayed (with decoded information).
Location	Overlayed over DATA waveform, on Grid. (Use multi-grid if there is more than one decoder ON)
Visual Aid	Color Coding for Frame, ID, DLC, DATA, CRC, Ack, Stuff Bits and Errors. Decode information is intelligently annotated based on timebase setting.

Trigger Capability	CANbus
Format	Hexadecimal or Binary
Trigger Setup	Trigger on All Frames, Frame ID, ID with Data, Remote Frames, or Error Frames
Address (ID) Condition Setup	Specify one Frame ID or a range of Frame IDs. Frame ID trigger can be combined with Data.
Conditional Trigger Setup	Conditional Frame ID and Conditional Data triggering available. Choose from <=, <, =, >, >=, <>, in range, out of range or don't care conditions.
Data Setup	**Hexadecimal:** # Data Bytes = 0 to 8. Data can be defined by nibble. Triggers on that data pattern regardless of position. **Binary:** Any combination of 0,1, or X for 1-64 bits. Triggers on that data pattern regardless of position.
Ack Condition	Not Applicable
Bit Rates	10, 25, 33.333, 50, 83.333, 100, 125, 250, 500, 1000 kb/s or user-defined between 10-1000 kb/s.
Trigger Input	Any analog Channel or the EXT input.
Trigger Design	Internal to oscilloscope, settable like any other oscilloscope trigger.

Search Capability	CANbus
Search Options	Search for Any Frame, Any Error, or Frame ID in Hexadecimal format.

Compatibility	CANbus
Compatible with...	TD (Trigger & Decode) Option fully compatible with WPZi, WRXi, and WSXs No D (Decode) option available.

LINbus Specifications

Note: Specifications are subject to change without notice.

Definition	LINbus
Protocol Setup	Select LIN Version 1.3, 2.x, or SAE J2602. Select BitRate (1.2, 2.4, 4.8, 9.6, 10.417, 19.2 kb/s, or user-defined 1-20 kb/s). Decodes LIN messages on buses with mixed LIN Versions.

Decode Capability	LINbus
Format	Hexadecimal, Binary
Decode Setup	Threshold definition required. Default is to Percent amplitude. Select BitRate.
Decode Input	Any analog Channel, Memory, or Math trace.
Number of Decoded Waveforms	**Up to 4 buses may be decoded at one time.** In addition, zooms can be displayed (with decoded information).
Location	Overlayed over DATA waveform, on Grid. (Use multi-grid if there is more than one decoder ON)
Visual Aid	Color Coding for Frame, Break, Synch, ID, ID Parity, Data, CRC. Decode information is intelligently annotated based on timebase setting.

Trigger Capability	LINbus
Format	Hexadecimal or Binary
Trigger Setup	Trigger on (Sync) Break (Start of Message), Frame ID, Frame ID+DATA, Error Frame (Any combination of Checksum, Header Parity, or Sync Byte error frames).
Address (ID) Condition Setup	Specify one Frame ID or a range of Frame IDs. Frame ID trigger can be combined with Data.
Conditional Trigger Setup	Conditional Frame ID and Conditional Data triggering available. Choose from <=, <, =, >, >=, <>, in range, out of range, or don't care conditions.
Data Setup	**Hexadecimal:** # Data Bytes = 0 to 8. Data can be defined by nibble. Triggers on that data pattern regardless of position or in user settable location. **Binary:** Any combination of 0,1, or X for 1-64 bits. Triggers on that data pattern regardless of position or in user settable location.
Ack Condition	Not Applicable
Bit Rates	1.2, 2.4, 4.8, 9.6, 10.417, 19.2 kb/s selectable. Any value from 1 kb/s to 20 kb/s may be entered.
Trigger Input	Any analog Channel or the EXT input.
Trigger Design	Internal to oscilloscope, settable like any other oscilloscope trigger.

Search Capability	LINbus
Search Options	Search for Any Frame, Any Error, or Frame ID in Hexadecimal format.

Compatibility	LINbus
Compatible with...	TD (Trigger & Decode) Option fully compatible with WPZi, WRXi, and WSXs D (Decode) Option fully compatible with WR6000, WP7000, and WM8000 Series.

FlexRaybus Specifications

Note: Specifications are subject to change without notice.

Definition	FlexRaybus
Protocol Setup	Select BitRate (2.5, 5 or 10 Mb/s) Select FlexRay Channel A or Channel B.

Decode Capability	FlexRaybus
Format	All decoding is hexadecimal except for Cycle Count which is decoded using a decimal format.
Decode Setup	Two threshold definitions required. Default is to Percent amplitude. Select BitRate. Select FlexRay Chanel A or Channel B.
Decode Input	Any analog Channel, Memory, or Math trace.
Number of Decoded Waveforms	**Up to 4 buses may be decoded at one time.** In addition, zooms can be displayed (with decoded information).
Location	Overlayed over DATA waveform, on Grid. (Use multi-grid if there is more than one decoder ON)
Visual Aid	Color Coding for TSS, FSS, Frame Qualifiers, Slot ID, Payload Length, Header CRC, Cycle Count, Data, BSS, Payload CRC, and FES. Decode information is intelligently annotated based on timebase setting.

Trigger Capability	FlexRaybus
Format	Hexadecimal or Binary for Frame ID. Decimal for Cycle Count.
Trigger Setup	***Trigger on TSS (Start), Frame ID, Cycle Count, Symbols and Errors*** **Symbols:** • Channel Idle Delimiter (CID) Symbol • Collision Avoidance Symbol (CAS) • Media Access Test Symbol (MTS) • Wakeup Pattern (WUP) **Errors:** • Frame Start Sequence (FSS) Error – triggers when the logic high time between the TSS and the first byte is too long • Byte Start Sequence (BSS) Error – triggers anytime the BSS pattern is not seen between bytes where expected • Frame End Sequence (FES) Error – triggers when the FS is not seen after the last byte
Address (ID) Condition Setup	Specify one Frame ID or a range of Frame IDs. Frame ID trigger can be combined with Cycle Count and Frame Qualifiers. Cycle count values range form 0 to 63 with Repetition factor of 1, 2, 4, 8, 16, 32, 64 Frame Qualifier values are 1, 0 or Don't Care for Payload Preamble, Null Frame, Sync Frame,

Trigger Capability	FlexRaybus
	and Startup Frame Qualifiers.
Conditional Trigger Setup	Conditional Frame ID and Conditional Cycle Count triggering available. Choose from <=, <, =, >, >=, <>, in range, out of range, or don't care conditions.
Data Setup	Not Applicable
Ack Condition	Not Applicable
Bit Rates	2.5, 5.0, or 10 Mb/s selectable.
Trigger Input	Any analog Channel or the EXT input.
Trigger Design	Internal to oscilloscope, settable like any other oscilloscope trigger.

Search Capability	FlexRaybus
Search Options	Search for Any Frame, Any Error, or Frame ID in Hexadecimal format.

Compatibility	FlexRaybus
Compatible with...	TD (Trigger & Decode) Option fully compatible with WPZi, WRXi, and WSXs D (Decode) Option fully compatible with WR6000, WP7000, and WM8000 Series.