TELEDYNE LECROY
Everywhereyoulook™



Instruction Manual

SAS and SATA Decoders

SAS and SATA Decode Software Instruction Manual

© 2024 Teledyne LeCroy, Inc. All rights reserved.

This document does not contain export-controlled information.

Unauthorized duplication or resale of Teledyne LeCroy publications is strictly prohibited. Customers are permitted to duplicate and distribute Teledyne LeCroy, Inc. documentation for internal educational purposes only.

Teledyne LeCroy is a trademark of Teledyne LeCroy, Inc., Inc. 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.

29 December 2024
sas-sata-d-im.pdf

Contents

Introduction 1
- SASbus D Option 1
- SATA Decode Option 1

Serial Decode 2
- Decoding Workflow 3
- Serial Decode Dialog 3
- Decoder Set Up 4
- Decoding Digital Inputs 6
- Failure to Decode 7
- Reading Waveform Annotations 8
- Serial Decode Result Table 11
- Searching Decoded Waveforms 17
- Decoding in Sequence Mode 18
- Improving Decoder Performance 19

ProtoSync 20
- ProtoSync Subdialog 20
- ProtoSync Protocol View 20

Appendix A: Automating the Decoder 21
- Configuring the Decoder 21
- Accessing the Result Table 21
- Reading the Structure of the Result Table 21
- Modifying the Result Table 23

Technical Support 24

About This Manual

This manual explains the basic procedures for using serial data decode software options for Teledyne LeCroy oscilloscopes. It is assumed that you have a basic understanding of the serial data physical layer specifications, and how to use the oscilloscope on which the option is installed. Only features specific to this product are explained in this manual.

While some images may not exactly match what is on your oscilloscope display—or may show an example taken from another standard—be assured that the functionality is identical. Product-specific exceptions will be noted in the text.

Some capabilities described may only be available with the latest version of our MAUI® software. Updates are available from the software download page at teledynelecroy.com under Oscilloscope Downloads > Firmware Upgrades.

Introduction

SASbus D Option

Serial Attached SCSI (SAS) is a communication protocol designed to transfer data between computer storage devices such as hard drives and tape drives. SAS provides backward compatibility from second-generation SATA drives; meaning SATA 3 Gb/s drives may be connected to SAS backplanes, but not vice versa.

The SASbus D option is available for most Teledyne LeCroy oscilloscopes. It permits link and data layer decoding of 1.5, 3, 6, or 12Gb/s SAS physical layer signals.

SASbus D is compatible with the ProtoSync™ option.

SATA Decode Option

Serial Advanced Technology Attachment (SATA) is a computer bus interface used to connect host adapters to mass storage devices like hard disk and optical drives. SATA's evolution brought reduced cable-bulk and cost (reduced from 80 wires to seven), increased speed, more efficient data transfer, and hot swapping.

The SATA Decode option permits link layer decoding of 1.5, 3 and 6 Gb/s SATA physical layer signals.

It is compatible with the ProtoSync™ option.



Note: If you have installed other -DME or -TDME options, the dialogs for Measure/Graph and Eye Diagram creation will appear when the decoder is open. They may or may not appear "grayed out." We do not guarantee the correct operation of the functionality for this protocol.

Serial Decode

The methods described here at a high level are used by all Teledyne LeCroy serial decoders, differing only slightly for signals with an embedded clock and separate clock and data signals.

Bit-level Decoding

The first software algorithm examines the embedded clock based on a default or user- specified vertical threshold level. Once the clock signal is extracted, the algorithm examines the traffic to determine whether a data bit is high or low. The default High and Low levels are automatically determined from a measurement of the amplitude of the signals acquired by the oscilloscope. Alternatively, they can be manually set by the user. The algorithm 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 data bit decoding.

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.

Logical Decoding

After determining individual data bit values, another algorithm performs a decoding of the serial data message after separation of the underlying data bits into logical groups specific to the protocol (Header/ID, Address Labels, Data Length Codes, Data, CRC, Parity Bits, Start Bits, Stop Bits, Delimiters, Idle Segments, etc.).

Message Decoding

Finally, another algorithm applies a color overlay with annotations to the decoded waveform to mark the transitions in the signal. Decoded message data is displayed in tabular form below the grid. Various compaction schemes are utilized to show the data for the duration of the acquisition, from as little as one serial data message acquisition to many thousands. In the case of long acquisitions, only the most important information is highlighted, whereas with the shortest acquisition, all information is displayed with additional highlighting of the complete message frame.

User Interaction

Your interaction with the software in many ways mirrors the order of the algorithms. You will:

- Assign a protocol/encoding scheme and data sources to one of the four decoder panels on the Serial Data and Decode Setup dialogs. Each decoder can utilize different protocols or data sources, or have other variations, giving you maximum flexibility to compare different signals or view the same signal from multiple perspectives.
- Complete the remaining subdialogs required by the protocol/encoding scheme. Once there is an acquisition in buffer, you will see a result table and an annotation overlay on the waveform trace showing the decoded data.
- Work with the annotated waveform, result table and other functionality to analyze the decoding.

Decoding Workflow

We recommend the following workflow for effective decoding:

1. Set up the decoder using the lowest level decoding mode available, but do not yet enable it.
2. Acquire at least one complete transmission reasonably well centered on screen in both directions, with generous idle segments on both sides.

   Note: See Failure to Decode for more information about the required acquisition settings.

3. Stop acquisition, then enable the decoder. It will operate on the acquisition in buffer.
4. Use the various decoder tools to verify that transitions are being correctly decoded. Tune the decoder settings as needed to produce a satisfactory decoding.
5. Once you are correctly decoding in one mode, continue making small acquisitions of five to eight transmissions and run the decoder in higher level modes.
6. Finally, run the decoder on acquisitions of the desired length.

When you are satisfied the decoder is working properly, you can disable/enable the decoder as desired without having to repeat this tuning process, provided the basic signal characteristics do not change.

Serial Decode Dialog

Choose **Analysis > Serial Decode** to access the serial decoder dialogs.

This dialog is best used to turn on/off decoders after they've been fully configured on the Decode Setup dialog.

- To turn on decoders, on the same row as Decode *N*, check **On**. If there is a valid acquisition, a result table and annotated waveform will appear.
- To turn off decoders, deselect the On boxes individually, or touch **Turn All Off**.

Note: If you change the **Protocol** to decode, the last settings configured for that protocol will be resumed.

If the decoder supports data search, use the **Search** button to open the Zoom dialogs. Enter search criteria on the Search subdialog to the right of each Z*n* dialog.

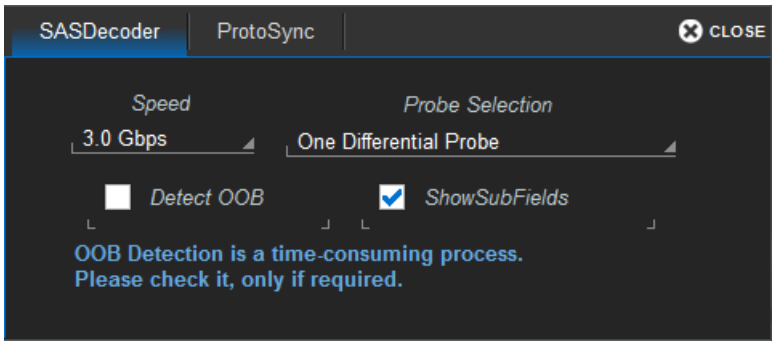
Decoder Set Up

Use Decode Setup and its subdialogs to configure decoders.

1. Choose **Analysis > Serial Decode** from the oscilloscope menu bar.
2. On the Serial Decode dialog, enable the decoder by checking **On** next to the decoder number. This may be done any time, although we recommend having an acquisition in buffer before enabling the decoder.
3. Click the **Setup** button at the end of the row to open the Decode Setup dialog.
4. Select the **Protocol** to be decoded and the inputs (sources). This selection drives the other fields that appear.
5. Define the level of decoding on the subdialogs to the right of the Decode Setup dialog.

SAS Decoder Settings

SASDecoder Subdialog



Select the signal **Speed** from one of the standards.

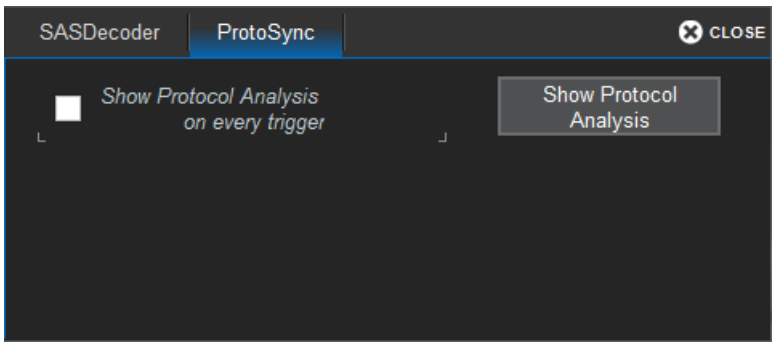
In **Probe Selection**, indicate whether you are using One Differential Probe or Two Single Ended Probes.

Note: The number of Source fields on the Decode Setup dialog changes to accommodate your probe selection. Be sure to select an input channel for each probe.

Optionally, select **Show Subfields** to annotate the subfields on the waveform overlay.

Optionally, select **Detect OOB** to mark Out of Band Signaling on the waveform overlay.

ProtoSync Subdialog

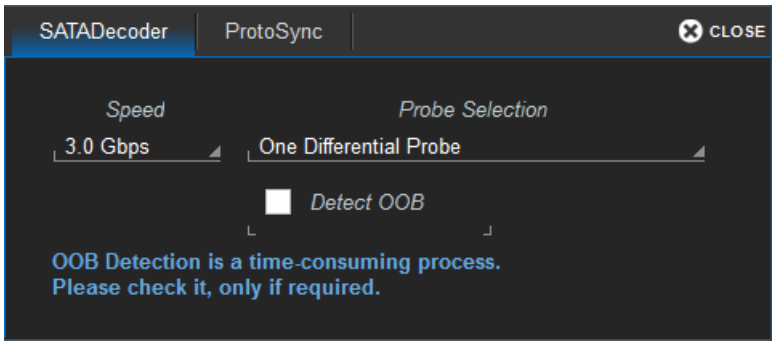


This dialog appears if you have installed the ProtoSync option key *and* the SAS ProtoSync Software on the instrument. Use it to display the decoded data in the format of a protocol analyzer. See ProtoSync.

Note: OOB signals will not appear in the ProtoSync view on the oscilloscope. Selecting to decode OOB data after synching the oscilloscope with the protocol analyzer can cause the ProtoSync link to break. Begin ProtoSync only after your decoding is in place.

SATA Decoder Settings

SATA Decoder Subdialog



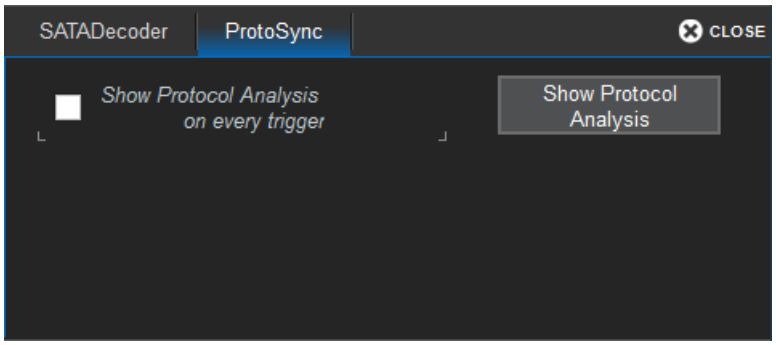
Select the signal Speed from one of the standards.

In Probe Selection, indicate whether you are using One Differential Probe or Two Single Ended Probes.

Note: The number of Source fields on the Decode Setup dialog changes to accommodate your probe selection. Be sure to select an input channel for each source.

Optionally, select Detect OOB to mark Out of Band Signaling on the decoder overlay.

ProtoSync Subdialog



This dialog appears if you have installed the ProtoSync option key *and* the SATA ProtoSync Software on the instrument. Use it to display the decoded data in the format of a protocol analyzer. See ProtoSync.

Note: OOB signals will not appear in the ProtoSync view on the oscilloscope. Selecting to decode OOB data after synching the oscilloscope with the protocol analyzer can cause the ProtoSync link to break. Begin ProtoSync only after your decoding is in place.

Decoding Digital Inputs

While digital lines can be used as sources for most decoders, it is important to make sure that the logic setup for the digital inputs is consistent with the levels set for the decoder and appropriate for the signal.

If you receive a poor decoding when using digital inputs, do the following to troubleshoot.

Assess Signal Levels Using Analog Input

It is good practice to establish the exact signal characteristics using an analog input method prior to using a digital input method.

Connect the signal to an oscilloscope channel (C*n*) and set the amplitude, top and base measurement parameters on it. If the decoder utilizes multiple sources, do this for all signals.

Open the decoder setup dialogs and try decoding the analog signal using the default levels. Adjust Level and Hysteresis according to the measured amplitude until you get a good decoding of the analog signal.

Set Appropriate Logic Levels on Digital Dialogs

When a digital input device is connected to the oscilloscope, setup dialogs for digital groups (Digital*n*) are added to the Vertical menu. Use the Digital*n* dialogs to set the levels used for logic determination by the input device.



Note: Digital1 is turned on by default with all the lines added to the group.

Open the Logic Setup tab and either select a Logic Family that is appropriate to the signal amplitude, or choose User Defined and enter your custom logic Threshold and Hysteresis levels based on what you determined when decoding the analog signal.



Tip: Selecting different Logic Families will display what determination Threshold and Hysteresis values they use on the dialog.

For example, a signal with approximately 1.25 V amplitude, 0 V based, does not match any of the Logic Families but lends itself to a User Defined setup with Threshold at 0.625 V (~50%) and a Hysteresis of 100 mV if the edges are clean.

Continue to adjust your digital group Logic Setup and your decoder Levels in synch until you receive a good digital decoding.



Note: For the MSO Digital Leadset, logic can only be set per digital leadbank, not line by line as on the HDA125. Be sure the lines you use for decoding are all from the same leadbank and set appropriately.

Failure to Decode

Several conditions may cause a decoder to fail, in which case a message will appear in the first row of the summary result table, instead of in the message bar as usual. In these cases, the decoding is turned off to protect you from incorrect data. Adjust your acquisition settings accordingly, then re-enable the decoder.

All decoders will test for the condition Too small amplitude. If the signal's amplitude is too small with respect to the full ADC range, the message "Decrease V/Div" will appear. The required amplitude to allow decoding is usually one vertical division.

If the decoder incorporates a user-defined bit rate (usually these are protocols that do not utilize a dedicated clock/strobe line), the following two conditions are also tested:

- Under sampled. If the sampling rate (SR) is insufficient to resolve the signal adequately based on the bit rate (BR) setup or clock frequency, the message "Under Sampled" will appear. 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.
- Too short acquisition. If the acquisition window is too short to allow any meaningful decoding, the message "Too Short Acquisition" will appear. The minimum number of bits required varies from one protocol to another, but is usually between 5 and 50.

However, there are no tests for the following, so you will want to pay attention to:

- Poor signal quality. Care must be taken when probing high speed serial data signals (typically with a high bandwidth differential probe). Channel loss, reflections and probe loading can degrade the signal. Its best to probe at the termination of a high speed serial link to minimize probe loading effects and reflections. If the signal has significant channel loss, CTLE/DFE equalizers can be used to improve the quality of the signal being decoded. Use the waveform output by the equalizer as the input to the decoder.
- Incorrect Threshold Levels. Configuring an incorrect threshold level can result in no, incomplete or incorrect decoding. With analog traces, the threshold level can be adjusted after the acquisition but this is not possible with digital waveforms. You must make sure that the appropriate logic levels are configured for digital inputs, as described in the Decoding Digital Inputs, before data is acquired. When troubleshooting, it is best to use analog channels instead of digital channels because of the additional information available.



Note: It is possible that several conditions are present, but you will only see the first relevant message in the table. If you continue to experience failures, try adjusting the other settings.

Reading Waveform Annotations

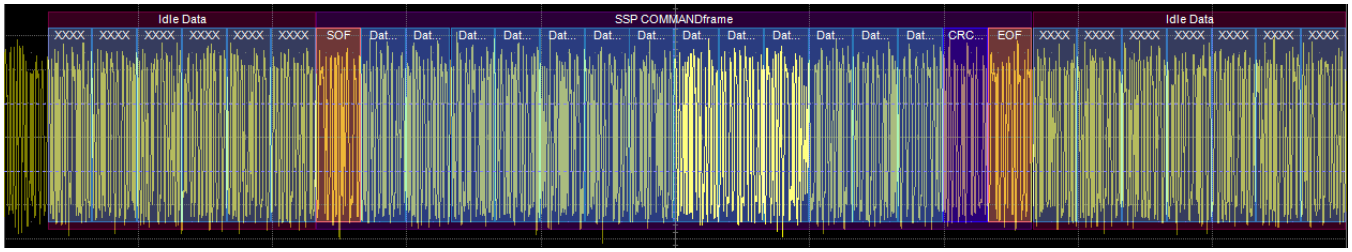
When a decoder has operated successfully on a valid acquisition, an annotated waveform appears on the oscilloscope display, allowing you to quickly see the relationship between the protocol decoding and the physical layer. A colored overlay marks significant bit-sequences in the source signal: Header/ID, Address, Labels, Data Length Codes, Data, CRC, Parity Bits, Start Bits, Stop Bits, Delimiters, Idle segments, etc. Annotations are customized to the protocol or encoding scheme.

The amount of information shown on an annotation is affected by the width of the rectangles in the overlay, which is determined by the magnification (scale) of the trace and the length of the acquisition. Zooming a portion of the decoded trace by clicking a line in the table will reveal the detailed annotations.

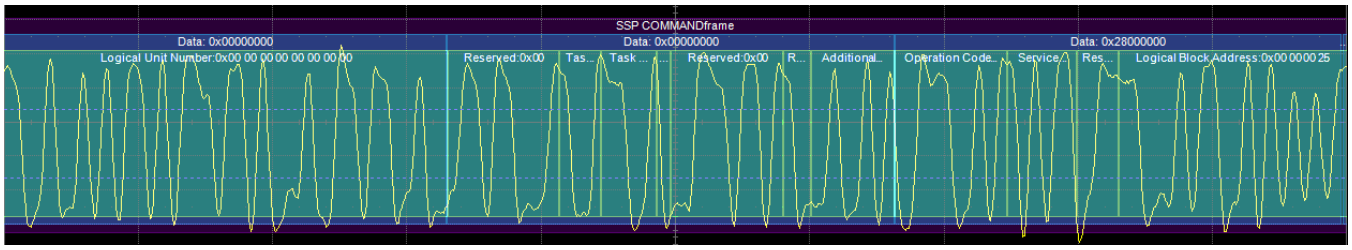
SAS Waveform Annotations

Annotation	Overlay Color (1)	Text (2)
Grouped Primitives	Burgundy (behind primitives)	<primitive> frame (e.g., IDLE frame)
Primitive Data	Blue	<type> Data (e.g., IDLE Data)
Frame	Dark Purple (behind data)	<TYPE> <SUBTYPE> frame (e.g., SSP COMMAND frame, Data frame)
Start/End of Frame	Brick Red	<SOF | EOF>
Data	Blue (behind frame fields)	Data:<word>
Frame Fields: Frame Type Changing Data Point ReTransmit Retry Data Frames Number of Fill Bytes Initiator & Target Port Transfer Tags Data Offset Logical Unit Number Task Attribute & Priority Enable First Burst Additional CBD Length Operation Code Service Action Logical Block Address(es) Transfer Length Control Ignored Data Data	Green (blue-green over Data annotation)	<field>:<bytes>
Cyclic Redundancy Check	Royal Blue	CRC
Out of Band Signaling		OOB
Protocol Error/Unknown	Red	<Protocol Error | Unknown>

1. Combined overlays affect the appearance of colors.
2. Text in brackets < > is variable. The amount of text shown depends on your zoom factors.



Initial decoding of SAS waveform. At this resolution, little information appears on the overlay.

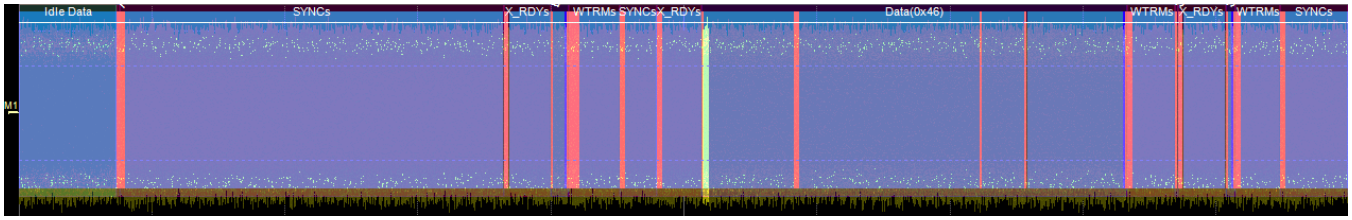


Zoom of decoding showing annotation details.

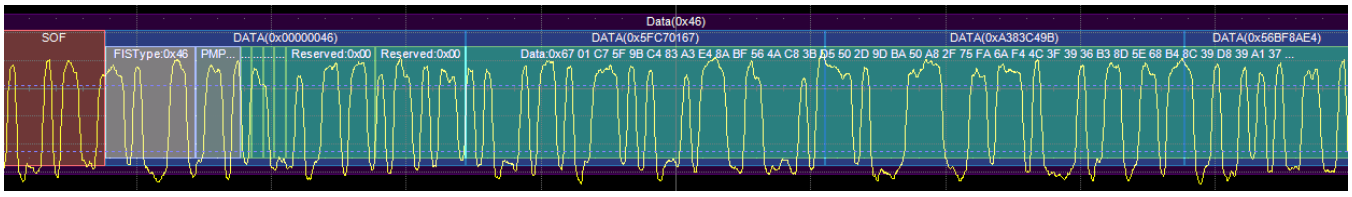
SATA Waveform Annotations

Annotation	Overlay Color (1)	Text (2)
Idle Data	Dark Green (behind Idle data)	Idle Data
Grouped Primitives	Burgundy (behind primitives)	<primitive> (e.g., ALIGN, SYNC, X-RDY, WTRM)
Data	Blue	XXXX<word>
Frame	Dark Purple (behind data)	<TYPE> frame (e.g., DMA_SETUP frame, DATA frame)
Start/End of Frame	Brick Red	<SOF | EOF>
Frame fields: FISType PMPort	Gray	<field>
Frame fields: Data Reserved Bits DMA Buffer Identifier & Offset DMA Transfer Count Status High/Low Error Protocol Specific LBA Sector Count	Green (blue-green over Data annotation)	<field>:<bytes>
Cyclic Redundancy Check	Royal Blue	CRC
Out of Band Signaling		OOB
Protocol Error/Unknown (frame error determined using CRC)	Red	<Protocol Error | Unknown>

1. Combined overlays affect the appearance of colors.
2. Text in brackets < > is variable. The amount of text shown depends on your zoom factors.



Initial decoding of SATA waveform.



Zoom of decoding showing annotation details.

Serial Decode Result Table

When you choose to turn **On** or to **View Decode**, provided there is a valid acquisition to decode using that protocol, a table summarizing the decoder results appears below the grids. This result table shows all data decoded during the most recent acquisition, even when there are too many bursts for the waveform annotation to be legible.

Tip: The result table does not have to be visible in order for the decoder to function. Hiding the table can improve performance when your aim is to use the decoding in downstream processes, like measurements.

Table Rows

Each row of the table represents one index of data found within the acquisition. What exactly this represents depends on the protocol and how you have chosen to "packetize" the data stream when configuring the decoder.

When multiple decoders of different protocols are run at once, the rows are interleaved in a summary table, ordered according to their acquisition time. The Protocol column is colorized to match the input source that resulted in that index of data.

Note: The interleaved summary table will default to the lowest common decoding (e.g., hexadecimal if both support that, but only one supports symbolic).

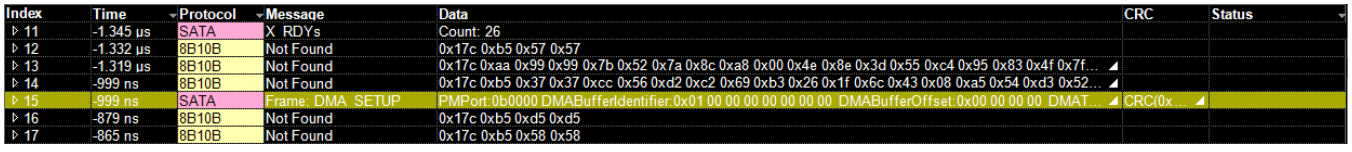
You can change the number of rows displayed on the table at one time. The default is five rows.

Swipe the table up/down or use the scrollbar at the far right to navigate the table. See Using the Result Table for more information about how to interact with the table rows to view the decoding.

Table Columns

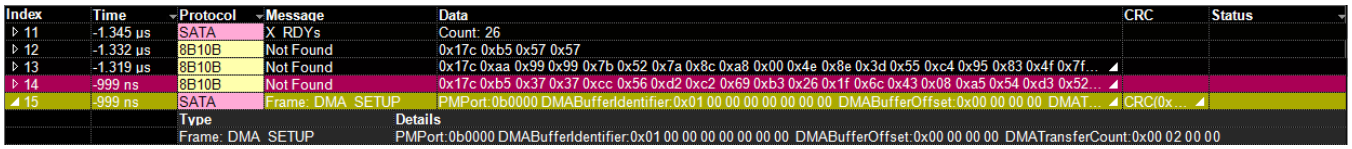
When a single decoder is enabled, the result table shows the protocol-specific details of the decoding. This **detailed result table** may be customized to show only selected columns. When two or more decoders are enabled, a **summary result table** combining results from all decoders shows these columns.

Column	Extracted or Computed Data
Index	Number of the line in the table
Time	Time elapsed from start of acquisition to start of message
Protocol	Protocol being decoded
Message	Message identifier bits
Data	Data payload
CRC	Cyclic Redundancy Check sequence bits
Status	Any decoder messages; content may vary by protocol



Example summary result table, with results from different protocol decoders interleaved on one table.

When you select the Index number from the summary result table, the detailed results for that index drop-in below it.



Exporting Result Table Data

You can manually export the detailed result table data to a .CSV file:

1. Choose **Analysis > Serial Decode** and open the **Decode Setup tab**.
2. Optionally, touch **Browse** and enter a new **File Name** and output folder.
3. Touch the **Export Table** button.

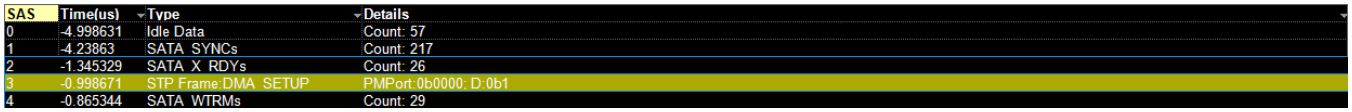
Export files are by default created in the D:\Applications\<protocol> folder, although you can choose any other folder on the oscilloscope or any external drive connected to a host USB port. The data will overwrite the last export file saved, unless you enter a new filename.

Note: Only rows and columns displayed are exported. When a summary table is exported, a combined file is saved in D:\Applications\Serial Decode. Separate files for each decoder are saved in D:\Applications\<protocol>.

The Save Table feature will automatically create tabular data files with each acquisition trigger. The file names are automatically incremented so that data is not lost. From the oscilloscope menu bar, choose **File > Save Table** and select the desired **Decode *N*** as the source.

SAS Detailed Result Table

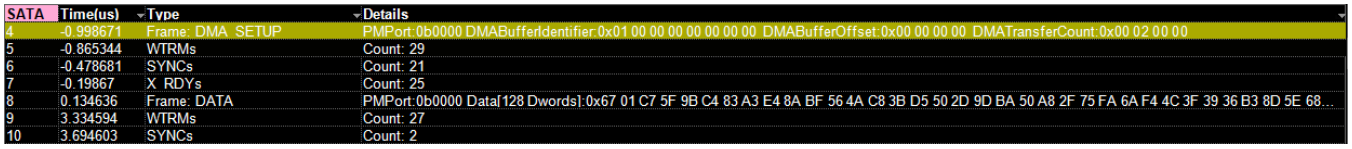
Column	Extracted or Computed Data
Index (always shown)	Number of the line in the table
Time	Time from start of acquisition to start of symbol sequence
Type	Primitive or frame type
Details	For primitives: counts, errors For frames: header info, decoded payload data, errors



Section of typical SAS detailed result table.

SATA Detailed Result Table

Column	Extracted or Computed Data
Index (always shown)	Number of the line in the table
Time	Time from start of acquisition to start of symbol sequence
Type	Primitive or frame type
Details	For primitives: counts, errors For frames: header info, decoded payload data, errors



SATA	Time(us)	Type	Details
4	-0.998671	Frame: DMA_SETUP	PMPort:0b0000 DMABufferIdentifier:0x01 00 00 00 00 00 00 00 DMABufferOffset:0x00 00 00 00 DMATransferCount:0x00 02 00 00
5	-0.865344	WTRMs	Count: 29
6	-0.478681	SYNCs	Count: 21
7	-0.19867	X_RDYs	Count: 25
8	0.134636	Frame: DATA	PMPort:0b0000 Data[128 Dwords]:0x67 01 C7 5F 9B C4 83 A3 E4 8A BF 56 4A C8 3B D5 50 2D 9D BA 50 A8 2F 75 FA 6A F4 4C 3F 39 36 B3 8D 5E 68...
9	3.334594	WTRMs	Count: 27
10	3.694603	SYNCs	Count: 2

Section of typical SATA detailed result table.

Using the Result Table

Besides displaying the decoded serial data, the result table helps you to inspect the acquisition.

Zoom & Search

Touching any cell of the table opens a zoom centered around the part of the waveform corresponding to the index. The Z*n* dialog opens to allow you to rescale the zoom, or to Search the acquisition. This is a quick way to navigate to events of interest in the acquisition.



Tip: When in a summary table, touch any data cell *other than* Index and Protocol to zoom.

The table rows corresponding to the zoomed area are highlighted, as is the zoomed area of the source waveform. The highlight color reflects the zoom that it relates to (Z1 yellow, Z2 pink, etc.). As you adjust the zoom scale, the highlighted area may expand to several rows of the table, or fade to indicate that only a part of that Index is shown in the zoom.

When there are multiple decoders running, each can have its own zooms of the decoding open at once. In this case, multiple rows of the summary table are highlighted to show which indexes are shown in the zooms. These highlights will be different colors to indicate which rows correspond to each decoder.



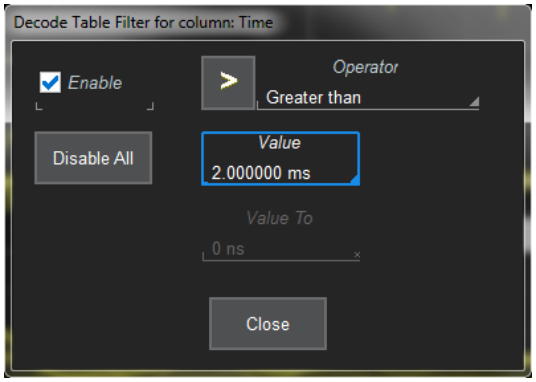
Note: The zoom number is no longer tied to the decoder number. The software tries to match the numbers, but if it cannot it uses the next empty zoom in the sequence.

Index	Time	Protocol	Message	Data	CRC	Status
11	-1.345 µs	SATA	X_RDYs	Count: 26		
12	-1.332 µs	8B10B	Not Found	0x17c 0xb5 0x57 0x57		
13	-1.319 µs	8B10B	Not Found	0x17c 0xaa 0x99 0x99 0x7b 0x52 0x7a 0x8c 0xa8 0x00 0x4e 0x8e 0x3d 0x55 0xc4 0x95 0x83 0x4f 0x7f...		
14	-999 ns	8B10B	Not Found	0x17c 0xb5 0x37 0x37 0xcc 0x56 0xd2 0xc2 0x69 0xb3 0x26 0x1f 0x6c 0x43 0x08 0xa5 0x54 0xd3 0x52...		
15	-999 ns	SATA	Frame: DMA_SETUP	PMPort:0b0000 DMABufferIdentifier:0x01 00 00 00 00 00 00 00 DMABufferOffset:0x00 00 00 00 DMAT...	CRC(0x...	
16	-879 ns	8B10B	Not Found	0x17c 0xb5 0xd5 0xd5		
17	-865 ns	8B10B	Not Found	0x17c 0xb5 0x58 0x58		

Filter Results

Columns of data with a drop-down arrow in the header cell can be filtered: 

Touch the header cell to open the Decode Table Filter dialog.



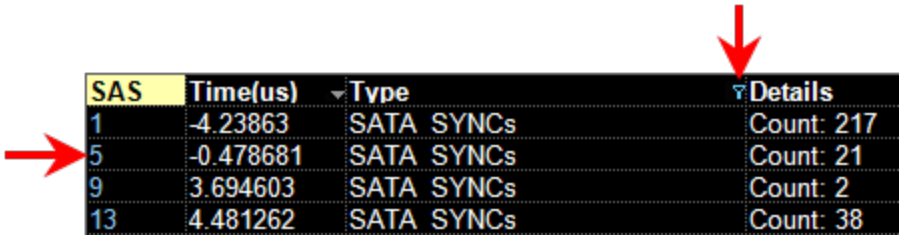
Select a filter Operator and enter a Value that satisfies the filter condition.

Operators	Data Types	Returns
=, ≠	Numeric or Text	Exact matches only
>, ≥, <, ≤	Numeric	All data that satisfies the operator
In Range, Out Range	Numeric	All data within/without range limits
Equals Any (on List), Does Not Equal Any (on List)	Text	All data that is/is not an exact match to any full value on the list. Enter a comma-delimited list of values, no spaces before or after the comma, although there may be spaces within the strings.
Contains, Does Not Contain	Text	All data that contains or does not contain the string

Note: Once the Operator is selected, the dialog shows the Value format that may be entered. Numeric values must be within .01% tolerance to be considered a match. Text values are case-sensitive, including spaces within the string.

Select **Enable** to turn on the column filter; deselect it to turn off the filter. Use the **Disable All** button to quickly turn off multiple filters. The filter settings remain in place and can be re-enabled on subsequent decodings.

Those columns of data that have been filtered will have a funnel icon (similar to Excel) in the header cell, and the index numbers will be colorized.



SAS	Time(us)	Type	Details
1	-4.23863	SATA SYNCs	Count: 217
5	-0.478681	SATA SYNCs	Count: 21
9	3.694603	SATA SYNCs	Count: 2
13	4.481262	SATA SYNCs	Count: 38

Example filtered decoder table.

On summary tables, only the Time, Protocol, and Status columns can be filtered.

If you apply filters to a single decoder table, the annotation is applied to only that portion of the waveform corresponding to the filtered results, so you can quickly see where those results occurred. Annotations are not affected when a summary table is filtered.

View Details

When viewing a summary table, touch the **Index number** in the first column to drop-in the detailed decoding of that record. Touch the Index cell again to hide the details.

If there is more data than can be displayed in a cell, the cell is marked with a white triangle in the lower-right corner. Touch this to open a pop-up showing the full decoding.



Navigate

In a single decoder table, touch the **Index column header** (top, left-most cell of the table) to open the Decode Setup dialog. This is especially helpful for adjusting the decoder during initial tuning.

When in a summary table, the Index column header cell opens the Serial Decode dialog, where you can enable/disable all the decoders. Touch the **Protocol** cell to open the Decode Setup dialog for the decoder that produced that index of data.

Customizing the Result Table

You may customize the size of the result table by changing the Table # Rows setting on the Decode Setup dialog. Keep in mind that the deeper the table, the more compressed the waveform display on the grid, especially if there are also measurements turned on.

Performance may be enhanced if you reduce the number of columns in the result table to only those you need to see. It is also especially helpful if you plan to export the data.

1. On the Decode Setup tab, touch the Configure Table button.
2. On the View Columns pop-up dialog, mark the columns you want to appear and clear those you wish to remove. Only those columns selected will appear on the oscilloscope display.

   Note: If a column is not relevant to the decoder as configured, it will not appear.

   To return to the preset display, touch Default.
3. Touch the Close button when finished.

You may also use the View Columns pop-up to set a Bit Rate Tolerance percentage. When implemented, the tolerance is used to flag out-of-tolerance messages (messages outside the user-defined bitrate +- tolerance) by colorizing in red the Bitrate shown in the table.

Searching Decoded Waveforms

Touching the Action toolbar Search button button on the Decode Setup dialog creates a 10:1 zoom of the center of the decoder source trace and opens the Search subdialog.

Touching the any cell of the result table similarly creates a zoom and opens Search, but of only that part of the waveform corresponding to the index (plus any padding).

Tip: In summary table mode, touch any cell *other than* Index and Protocol to create the zoom.

Basic Search

On the Search subdialog, select what type of data element to Search for. These basic criteria vary by protocol, but generally correspond to the columns of data displayed on the detailed decoder result table.

Optionally:

- Check Use Value and enter the Value to find in that column. If you do not enter a Value, Search goes to the beginning of the next data element of that type found in the acquisition.
- Enter a Left/Right Pad, the percentage of horizontal division around matching data to display on the zoom.
- Check Show Frame to mark on the overlay the frame in which the event was found.

After entering the Search criteria, use the Prev and Next buttons to navigate to the matching data in the table, simultaneously shifting the zoom to the portion of the waveform that corresponds to the match.

The touch screen message bar shows details about the table row and column where the matching data was found.



Advanced Search

Advanced Search allows you to create complex criteria by using Boolean AND/OR logic to combine up-to-three different searches. On the Advanced dialog, choose the Col(umns) to Search 1 - 3 and the Value to find just as you would a basic search, then choose the Operator(s) that represent the relationship between them.

Decoding in Sequence Mode

Decoders sometimes can be applied to Sequence Mode acquisitions. In this case, the index numbers on the result table are followed by the segment in which the index was found and the number of the sample within that segment: *index* (*segment-sample*).

CAN Std	Time	Format	ID	IDE	RTR	DLC	Data
2 (2-1)	9.72882 ms	Std	0x400	0	0	2	6a 6b
3 (3-1)	19.7527 ms	Std	0x400	0	0	2	6a 6b
4 (4-1)	30.2558 ms	Std	0x400	0	0	2	6a 6b
5 (5-1)	40.1663 ms	Std	0x400	0	0	2	6a 6b
6 (6-1)	49.8284 ms	Std	0x400	0	0	2	6a 6b
7 (7-1)	59.8595 ms	Std	0x400	0	0	2	6a 6b
8 (8-1)	69.8913 ms	Std	0x400	0	0	2	6a 6b
9 (9-1)	80.4032 ms	Std	0x400	0	0	2	6a 6b
10 (10-1)	89.9384 ms	Std	0x400	0	0	2	6a 6b
11 (11-1)	99.9688 ms	Std	0x400	0	0	2	6a 6b

Example filtered result table for a sequence mode acquisition.

In the example above, each segment was triggered on the occurrence of ID 0x400, which occurred only once per segment, so there is only one sample per segment. The Time shown for each index in a Sequence acquisition is absolute time from the first segment trigger to the beginning of the sample segment.

Otherwise, the results are the same as for other types of acquisitions and can be zoomed, filtered, searched, or used to navigate. When a Sequence Mode table is filtered, the waveform annotation appears on only those segments and samples corresponding to the filtered results.

Note: Waveform annotations can only be shown when the Sequence Display Mode is Adjacent. Annotations are not adjusted when a Sequence Mode summary table is filtered, only the result table data.

Multiple decoders can be run on Sequence Mode acquisitions, but in a summary table, each decoder will have a first segment, second segment, etc., and there may be any number of samples in each. As in any summary table, the samples will be interleaved and indexed according to their actual acquisition time. So, you may find (3-2) of one decoder before (1-1) of another. Filter on the Protocol column to see the sequential results for only one decoder.

Improving Decoder Performance

Digital oscilloscopes repeatedly capture "windows in time". Between captures, the oscilloscope is processing the previous acquisition. The following suggestions can improve decoder performance and enable you to better exploit the long memories of Teledyne LeCroy oscilloscopes.

Where possible, **decode Sequence Mode acquisitions.** By using Sequence mode, you can take many shorter acquisitions over a longer period of time, so that memory is targeted on events of interest.

Note: For some protocols, the Serial Trigger does not support Sequence Mode acquisitions, although you could still decode Sequence Mode acquisitions made using a different trigger type.

Parallel test using multiple oscilloscope channels. Up-to-four decoders can run simultaneously, each using different data or clock input sources. This approach is statistically interesting because multi-channel acquisitions occur in parallel. The processing is serialized, but the decoding of each input only requires 20% additional time, which can lessen overall time for production validation testing, etc.

Avoid oversampling. Too many samples slow the processing chain.

Optimize for analysis, not display. The oscilloscope has a preference setting (Utilities > Preference Setup > Preferences) to control how CPU time is allocated. If you are primarily concerned with quickly processing data for export to other systems (such as Automated Test Equipment) rather than viewing it personally, it can help to switch the Optimize For: setting to Analysis.

If your goal is downstream processing, turn off tables, annotations, and waveform traces. As long as downstream processes such as measurements or Pass/Fail tests reference a decoder, the decoder can function without actually displaying results. If you do not need to see the results but only need the exported data, you can deselect View Decode, or minimize the number of lines in a table. Closing input traces also helps.

Decrease the number of rows and columns in tables. Only the result table rows and columns shown are exported. It is best to reduce tables to only the essential columns if the data is to be exported, as export time is proportional to the amount of data exchanged.

ProtoSync

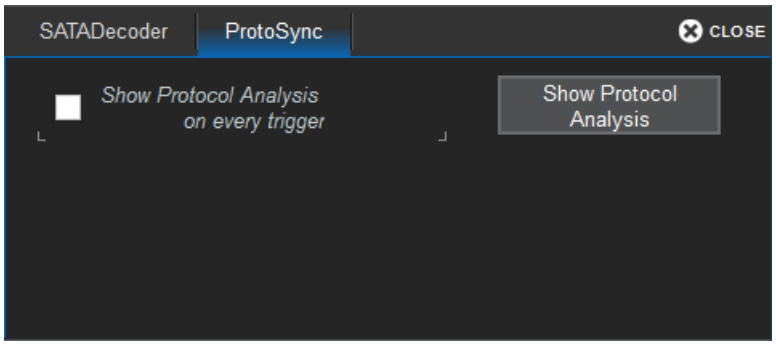
ProtoSync™ adds many of the same algorithms used by the Teledyne LeCroy hardware protocol analyzers to your decoder, enabling you to view the logic layer display of a protocol analyzer in sync with the physical layer display of an oscilloscope. There is no need to export the decoded data to another device, and the integrated graphical user interface makes it easy to switch between the views, or to arrange them side by side on the screen. Changing the decoder on the oscilloscope updates ProtoSync in real time for comprehensive physical and protocol layer debug and analysis.

Note: ProtoSync requires the installation of both the option key and the SAS or SATA Protocol Suite Software. Protocol Suite is available free from teledynelecroy.com. After installing the software, go to Start/Programs/LeCroy/<SAS|SATA> Protocol Suite for documentation explaining the Protocol Analyzer software interface.

ProtoSync Subdialog

When the ProtoSync option is installed, the ProtoSync subdialog appears behind the protocol-specific decoder subdialogs. Either:

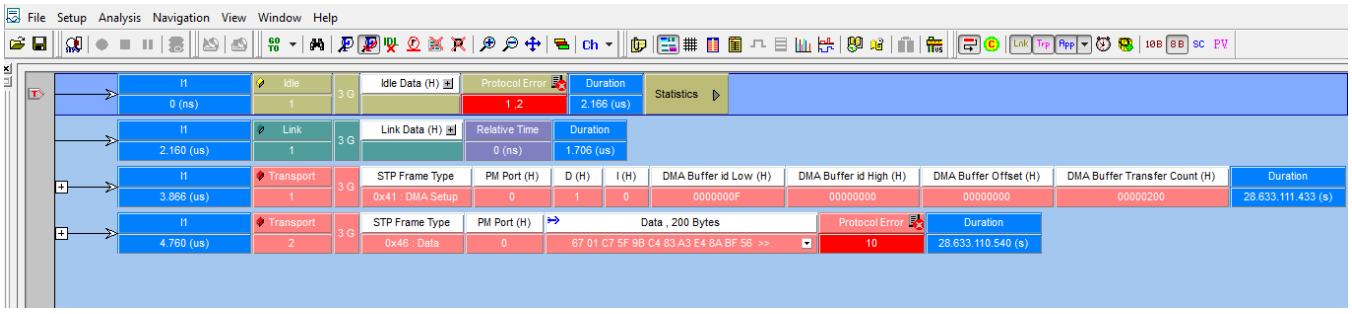
- Touch **Show Protocol Analysis** to manually launch ProtoSync using data from last acquisition.
- Select **Show Protocol Analysis on every trigger**. This option repeatedly sends data to ProtoSync.



ProtoSync Protocol View

The Protocol View displays data packets in a format consistent with hardware protocol analyzers. Each row represents the decoded bits from one transaction/packet, with each cell color-coded to indicate different data types. Transactions are shown as either upstream or downstream.

Selecting any row of the table will simultaneously zoom to that portion of the waveform in the decoder window.



Decoding shown in Protocol View.

Appendix A: Automating the Decoder

As with all other oscilloscope settings, decoder features such as result table configuration and export can be configured remotely using COM Automation.

Note: The examples shown here were taken from a CAN FD decoding, but all decoder result tables share the same Automation structure.

Configuring the Decoder

The object path to the decoder Control Variables (CVARs) is:

app.SerialDecode.Decode*n*

Where *n* is the decoder number, 1 to 4. All relevant decoder objects will be nested under this. Use the MAUI Browser utility (installed on the oscilloscope desktop) to view the entire object hierarchy.

Accessing the Result Table

The decoder Result Table is a complex matrix with secondary tables nested within some of its cells. The table data can be accessed using the Automation object:

app.SerialDecode.Decode*n*.out.Result.cellvalue(*RowA*, *ColA*)(*RowB*, *ColB*)

Where:

n:= 1 to 4

RowA:= 0 to K (0=Row Index Number)

ColA:= 0 to L (0=Column Header)

RowB:= 0=MeasuredValue, 1=StartTime, 2=StopTime

ColB:= 0 to M

Complicating the matter of accessing the table is that there are two types of cell that may appear in the Result table, Simple Cell and Table Cell, which are accessed in slightly different ways, and that some columns are always hidden from view, yet they are still counted among the columns when querying.

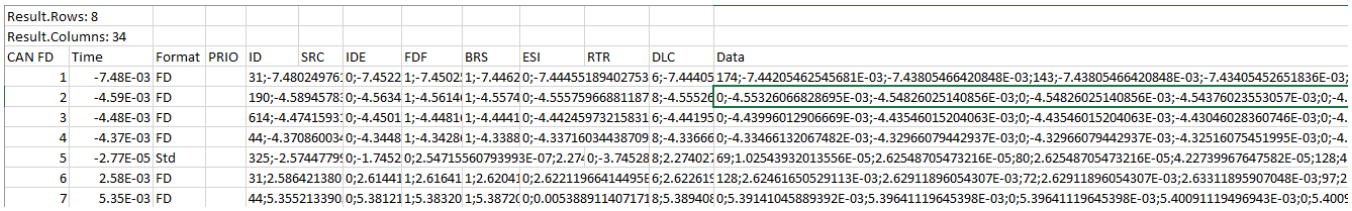
Reading the Structure of the Result Table

In order to successfully access the data, it is necessary to first ascertain how many rows and columns are actually in your decoder result table, and what cell type is used for the column of data you wish to read.

To do this, we have provided the script, ExampleTableSerialDecode.vbs, which by default installs into oscilloscope: C:\LeCroy\XStream\Scripts\Automation\ExampleTableSerialDecode.vbs.

Tip: This script may also be used as a basis for your own remote control programs, or used as is to read decoder table data.

With the decoder table populated, run the script from the oscilloscope (or a PC if you have a remote connection to the oscilloscope). The script will generate the comma-delimited file, ExampleTableSerialDecode.txt, which may be imported into Excel or other spreadsheet software to show the table structure.



Result.Rows: 8												
Result.Columns: 34												
CAN FD	Time	Format	PRIO	ID	SRC	IDE	FDF	BRS	ESI	RTR	DLC	Data
1	-7.48E-03	FD		31;-7.48024976	0;-7.4522	1;-7.4502	1;-7.4462	0;-7.444551894027536			6;-7.44405	174;-7.44205462545681E-03;-7.43805466420848E-03;143;-7.43805466420848E-03;-7.43405452651836E-03;
2	-4.59E-03	FD		190;-4.5894578	0;-4.5634	1;-4.5614	1;-4.5574	0;-4.555759668811876			8;-4.55526	0;-4.553260668286955E-03;-4.54826025140856E-03;0;-4.54826025140856E-03;-4.54376023553057E-03;0;-4.
3	-4.48E-03	FD		614;-4.4741593	0;-4.4501	1;-4.4481	1;-4.4441	0;-4.442459732158316			6;-4.44195	0;-4.43996012906669E-03;-4.43546015204063E-03;0;-4.43546015204063E-03;-4.43046028360746E-03;0;-4.
4	-4.37E-03	FD		44;-4.37086003	0;-4.3448	1;-4.3428	1;-4.3388	0;-4.337160344387098			8;-4.33666	0;-4.33466132067482E-03;-4.32966079442937E-03;0;-4.32966079442937E-03;-4.32516075451995E-03;0;-4.
5	-2.77E-05	Std		325;-2.5744779	0;-1.7452	0;2.54715560793993	E-07;2.274	0;-3.74528			8;2.274027	69;1.02543932013556E-05;2.62548705473216E-05;80;2.62548705473216E-05;4.22739967647582E-05;128;4
6	2.58E-03	FD		31;2.586421380	0;2.61441	1;2.61641	1;2.62041	0;2.622119664144956			6;2.622615	128;2.62461650529113E-03;2.62911896054307E-03;72;2.62911896054307E-03;2.63311895907048E-03;97;2
7	5.35E-03	FD		44;5.355213390	0;5.38121	1;5.38320	1;5.3872(	0;0.005388911407171			8;5.389408	0;5.39141045889392E-03;5.39641119645398E-03;0;5.39641119645398E-03;5.40091119496943E-03;0;5.400

Example spreadsheet after importing ExampleTableSerialDecode.txt.

The first two rows of the imported file will show the total number of rows and columns in the table, in this example 8 rows and 34 columns. This indicates the range of your *RowA* and *ColA* keys.

The third row of the imported file will replicate the column headers of the Result Table (0), with individual records (frames, messages, etc., depending on how you have "packetized" the decoding) appearing in subsequent rows (1-*n*).

Counting from 0 at the far left (Row Index Number), find the column of the data you wish to access. That will be the *ColA* key in your script.

Note: Do not confuse the number/letter of the cells in the imported file with the rows/columns of the Result Table.

Hidden columns (whether hidden by you or the software) must still be counted, so, in the example above, PRIO is column 3, making ID column 4, and so forth. So, if you wished to access the ID of record 6, the first argument of your query would be: (6,4)

Within each column, Simple Cells contain a single value that appears at the specified location in the table. In the above example, columns 0 through 2 are Simple Cells. Simple Cell VBS access syntax is:

vbs? 'return=app.SerialDecode.Decode1.out.Result.cellvalue(*RowA*,*ColA*)'

However, many cells of the Result Table are the Table Cell type, nested tables that may contain multiple "B" columns and always three "B" rows that, when coupled with the column key, each return a different component of the measurement: (0,*ColB*) = MeasuredValue, (1,*ColB*)= StartTime, (2,*ColB*) = StopTime. These cells can be identified by the list of semi-colon delimited values within them. The first three values in the list are Col0, the second three values are Col1, and so forth.

To access Table Cells, the (*RowB*,*ColB*) argument is sent in a second parenthesis, following the A "locators":

vbs? 'return=app.SerialDecode.Decode1.out.Result.cellvalue(*RowA*,*ColA*)(*RowB*, *ColB*)'

Although the image above does now show it, the ID and IDE columns each contain a single-column, three-row nested table. To read the *values* from such columns, you would add the argument (0,0) following your "locators": (*RowA*,4),(0,0) and (*RowA*,6),(0,0) respectively.

Reading the Data column (*RowA*,12) is more complicated, because it contains a *multi-column,* three-row nested table, as indicated by the longer list of values. To access the full Data column value for each record, all *ColB*s must be called by your script.

For example, if these were your decoder results:

CAN FD	Time	Format	ID	IDE	FDF	BRS	ESI	RTR	DLC	Data
1	-7.4822 ms	FD	0x01f	0	1	1	0		6	ae 8f a0 a3 00 06
2	-4.5915 ms	FD	0x0be	0	1	1	0		8	00 00 00 00 00 00 00 00
3	-4.4762 ms	FD	0x266	0	1	1	0		6	00 00 00 00 00 00
4	-4.3729 ms	FD	0x02c	0	1	1	0		8	00 00 00 00 00 00 00 00
5	-27.74 µs	Std	0x145	0	0			0	8	45 50 80 00 00 00 00 00
6	2.58442 ms	FD	0x01f	0	1	1	0		6	80 48 61 44 00 06
7	5.35321 ms	FD	0x02c	0	1	1	0		8	00 00 00 00 00 00 00 00

The following table shows example VBS queries you might add to a remote control program to read data from the decoder result table.

Remote Queries	Returned Value(s)	What Is Read by Query
vbs? 'return=app.SerialDecode.Decode1.out.Result.rows'	8	Number of table rows (incl. header Row 0)
vbs? 'return=app.SerialDecode.Decode1.out.Result.cellvalue(6,0)' vbs? 'return=app.SerialDecode.Decode1.out.Result.cellvalue(6,1)' vbs? 'return=app.SerialDecode.Decode1.out.Result.cellvalue(6,2)'	6 2.58442...E-03 FD	Value in first 3 columns of Row 6, including: Index # in Row 6 Col 0 Time in Row 6 Col 1 Format in Row 6 Col 2
vbs? 'return=app.SerialDecode.Decode1.out.Result.cellvalue(6,12)(0,0)'	128	Data value in ColB0 of Row 6 Col 12
vbs? 'return=app.SerialDecode.Decode1.out.Result.cellvalue(6,12)(1,0)'	2.62461...E-03	StartTime of Data in ColB0 of Row 6 Col 12 (hidden)
vbs? 'return=app.SerialDecode.Decode1.out.Result.cellvalue(6,12)(2,0)'	2.62911...E-03	StopTime of Data in ColB0 of Row 6 Col 12
vbs? 'return=app.SerialDecode.Decode1.out.Result.cellvalue(6,12)(0,1)'	72	Data value in ColB1 of Row 6 Col 12
vbs? 'return=app.SerialDecode.Decode1.out.Result.cellvalue(6,12)(1,1)'	2.62911...E-03	StartTime of Data in ColB1 of Row 6 Col 12 (hidden)

Modifying the Result Table

The CVAR app.SerialDecode.Decode*n*.Decode.ColumnState contains a pipe-delimited list of all the table columns and their current state (visible=on, hidden=off). For example:

app.SerialDecode.Decode1.Decode.ColumnState = "Idx=On|Time=On|Data=On|..."

If you wish to hide or display table columns, send the full string with the state changed from "on" to "off", or vice versa, rather than remove any column from the list.

Technical Support

Live Support

Registered users can contact their local Teledyne LeCroy service center at the number listed on our website.

You can also submit Technical Support requests via the website at:

teledynelecroy.com/support/techhelp

Resources

Teledyne LeCroy publishes a free Technical Library on its website. Manuals, tutorials, application notes, white papers, and videos are available to help you get the most out of your Teledyne LeCroy products. Visit:

teledynelecroy.com/support/techlib

The Datasheet published on the product page contains the detailed product specifications.

Service Centers

For a complete list of offices by country, including our sales & distribution partners, visit:

teledynelecroy.com/support/contact

Teledyne LeCroy
700 Chestnut Ridge Road
Chestnut Ridge, NY, 10977, USA
teledynelecroy.com

Sales and Service:
Ph: 800-553-2769 / 845-425-2000
FAX: 845-578-5985
contact.corp@teledynelecroy.com

Support:
Ph: 800-553-2769
customersupport@teledynelecroy.com



700 Chestnut Ridge Road
Chestnut Ridge, NY 10977
USA

teledynelecroy.com