

USER GUIDE
CAN OPTION
Windows Based Analyzers
FOR THE ALD-232A
BUS ANALYSIS PROBE
(Addendum to ALD-232A Manual)



advanced
logical design, inc.

Limited Warranty

This Advanced Logical Design product has a warranty against defects in material and workmanship for a period of 1 year from date of shipment. During the warranty period, Advanced Logical Design, Incorporated will, at its option, either repair or replace products that prove to be defective.

For warranty service or repair, this product must be returned to Advanced Logical Design, Inc.

The Buyer shall prepay shipping charges to Advanced Logical Design and Advanced Logical Design shall pay shipping charges to return the product to the Buyer. However, the Buyer shall pay all shipping charges, duties, and taxes for products returned to Advanced Logical Design from another country.

Advanced Logical Design does not warrant that the operation of the instrument, software, or firmware will be uninterrupted or error-free.

Table of Contents

1. Introduction	4
1.1. Specifications	4
1.2. Logic Analyzers Supported	4
1.3. Installing the Software	4
1.4. Equipment Supplied	5
1.5. Minimum Equipment Required	5
2. Installation	5
2.1. Initial Inspection	5
2.2. Connection to the Logic Analyzer	5
2.3. Connection to the CAN Bus	6
2.4. POD Configuration	6
3. Operational Overview	7
4. Analyzing CAN	8
4.1. Run Time Display	8
4.2. Logic Analyzer Connections	9
4.3. State Analysis	10
4.3.1. Data collection	10
4.3.2. State Analyzer Configuration	10
4.3.3. 'No Filter' Configuration	12
4.3.4. Filtered Configuration	14
4.3.5. Triggering	15
4.4. Timing Analysis	18
5. Theory of Operation	19

1. Introduction

The CAN option for the ALD-232A-CAN analysis probe allows users to examine the operation of a CAN bus without having to manually decipher the serial protocol for the bus. The ALD-232A-CAN analysis probe will display the data on the bus in an easy to read byte format.

1.1. Specifications

- Bus Loading: Standard CAN bus transceiver load.
- Probe Cable length: 24 in
- Connectors: 1 Male and 1 Female DB9 connector
- Power Required: ~ 325 mA supplied by the logic analyzer
- Mechanical: 4.0" x 7.5" x 2.0"
- Signals supported: CAN_H and CAN_L.
- Bus Speed: 10, 20, 50, 100, 125, 250, 500, 800 kBits/s and 1 Mbits/s

1.2. Logic Analyzers Supported

The CAN bus analysis probe may be used with all currently available 165XX, 167XX- and 169XX system state analyzer modules, or with the 1660, 1670, 1680 or 1690 portable analyzers. Two 17-channel pods are required for proper operation.

Because these different families of analyzers behave differently, there are different versions of the CAN software to support these products. This manual contains the information for the Windows based logic analyzers: 1680, 1690 and 16900.

1.3. Installing the Software

The CAN software for the Windows based analyzers is distributed on a CDROM. Insert this CD into the CD drive on the appropriate analyzer. From the start menu, choose RUN and navigate to **Installation_Files\CAN\setup.exe** file on the CD. Select OK. The install is reasonably fast and does not require any user input.

NOTE: Revision 3.x or later must be installed from Agilent before opening the configurations

This process will install the CAN data formatter, 9 configuration files and 2 sample trigger files on the analyzer's C-drive:

C:\Documents and Settings*current_user*\My Documents\Agilent Technologies\Logic Analyzer\Config Files\CAN

Note: .xml Configurations with "DATA" in the title contain captured data and can be used for demonstration purposes. They must be viewed "OFFLINE".

CANfilter.xml	This configuration provides the most compact CAN data display
CANfilter_DATA.xml	CAN configuration with DATA
CANfilter_DATA01of01.mfb	CAN DATA file
CANnofilter.xml	This provides the CAN data serially as each byte arrives and tags the data type and time.
CANnofilter_DATA.xml	CAN configuration with DATA
CANnofilter_DATA01of01.mfb	CAN DATA file
CANtiming.xml	This configuration provides the CAN signals waveforms
CANtiming_DATA.xml	CAN configuration with DATA
CANtiming_DATA01of01.mfb	CAN DATA file
EIDtrigger.xml	This example trigger shows how to trigger on an extended ID.
SIDtrigger.xml	This trigger example permits triggering on a standard ID.

1.4. Equipment Supplied

- RS232D, RS449, IEEE 1284, and CAN Bus Analysis Probe
- This User Guide
- CDROM configuration files and screen data formatter
- 2 flat ribbon 'Y' Cable for connecting to the target bus for RS232, RS449 and IEEE1284
- 1 CAN POD with two 9-pin D-Sub connectors, one male and one female, and an attachment cable to connect the POD to the ALD-232A box.

1.5. Minimum Equipment Required

In addition to the equipment supplied above, an Agilent analyzer is required.

2. Installation

The ALD-232A-CAN is easy to install. The unit is simply connected between the logic analyzer and the CAN target and the supplied configuration is loaded into the logic analyzer.

2.1. Initial Inspection

Inspect the shipping container for damage. If the shipping container or cushioning material is damaged, it should be kept until the contents of the shipment have been inspected mechanically and electrically. If the contents are incomplete, if there is mechanical damage or defect, or if the instrument does not operate, notify Advanced Logical Design, Inc. If the shipping container is damaged, or the cushioning materials show signs of stress, notify the carrier as well as the ALD office. Keep the shipping materials for inspection by the carrier.

2.2. Connection to the Logic Analyzer

The ALD-232A-CAN and the CAN POD*^T derives its power from the logic analyzer and provides it data through the pod connection cables. Both PODs 1 and 2 should be

* The Can Pod may be optionally powered from the CAN bus.

connected. The pod cables from the logic analyzer are plugged directly into the right side of the ALD-232A. No additional terminators or adapters are required.

Note: Pod 2 plugs into the ALD-232A where Pod 3 is marked.

2.3. Connection to the CAN Bus

The ALD-232A-CAN has several connectors on the top of the unit that allow it to be attached to several busses. The connector used for the CAN bus is the one identified as 'J1'. A short cable is supplied with a mating 26 pin IDC connector on one end and the other end has the CAN POD. The CAN POD has two 9-pin D-Sub connectors, one male and one female, to connect to the CAN bus.

Both 9-pin D-Sub connectors are wired the same and form a straight-through connection for the CAN bus on all 9 pins. The pin-out of the connectors is shown in the table below, and conforms to the CiA Draft Standard 102 Version 2.0 (See Table 1).

Pin	Signal
1	Reserved
2	CAN_L (dominant low)
3	CAN_GND (Ground reference)
4	Reserved
5	CAN_SHIELD (optional CAN shield)
6	GND (optional CAN ground)
7	CAN_H (dominant high)
8	Reserved
9	CAN_V+ (optional CAN external power supply)

Table 1 – Signals of the CAN 9-Pin D connector

The CAN POD has a small recessed slide switch on the side of the POD labeled 'COMMON' and 'ISOLATE'. In the 'COMMON' position the GNDs for both sides of the opto-couplers in the POD are connected together, and power for the CAN transceiver is derived from the ALD-232A-CAN unit. This is the shipping default.

In the 'ISOLATE' position, the GNDs are not connected, and power for the bus side of the opto-couplers and the transceiver must be supplied by the CAN bus. The power must be supplied between pin 9 (CAN_V+) and pin 3 (CAN_GND). In the 'ISOLATE' mode there is a 1M ohm impedance between the two GNDs.

2.4. POD Configuration

The POD configuration for all the supplied configurations sets up the logic analyzer pods 1 and 2 as follows (See Figure 1). **NOTE: Pod 2 plugs into the ALD232A Pod 3 port.**












Bus/Signal Name	Channels Assigned	Width
 AldData	Pod A1[15:0]	16
 AldTag	Pod A2[10:8]	3
 CAN	Pod A2[15]	1
 CLKCAN	Clks[C1], Pod A2[14]	2
 DATASTB	Clks[C2]	1
 EID1	Pod A1[14:4,1:0]	13
 EID2	Pod A1[15:0]	16
 FRMCOUNT	Pod A2[7:0]	8
 IDE	Pod A1[2]	1
 INFRAME	Pod A2[13]	1
 SID	Pod A1[14:4]	11

Figure 1: CAN Pod Configuration

When looking at CAN in timing mode, the internally derived clock ‘CLKCAN’ can be used as a guide to help with bit delineation. The CAN signal presented to the logic analyzer is an internally clocked version of CAN-L. It is clocked by CLKCAN, whose frequency is set by the operator and is synchronized to the incoming transitions of CAN-L.

3. Operational Overview

Immediately after power is applied or **RESET** is depressed, the ALD-232A-CAN will enter self-test mode. After the self-test is completed, the ALD-232A-CAN will enter the RUN mode, with the interface connection (J1, J2 or J3) that was last used. [The first time the ALD-232A-CAN is used, it will come up running in the CAN mode with the bit rate set to 1 Mb/sec]. The LED opposite the appropriate connector will be lit.

All operations of the ALD-232A are controlled by the keyboard. When the logic analyzer cables are plugged into the ALD-232A, the unit will power up in the mode it was last in. Navigation through the options is very simple. An arrow (→) will point to the item that can be changed. A new value may be selected for that item by scrolling through the choices with the ◀ and the ▶ keys (See Figure 2). To move to a different item, use the ▲ and ▼ keys. The item at the top of the screen is the type of analysis desired [RS232 ASYNC, RS232 SYNC, RS449 ASYNC, RS449 SYNC, IEEE 1284 and CAN]. The last screen in the sequence is an informational screen.

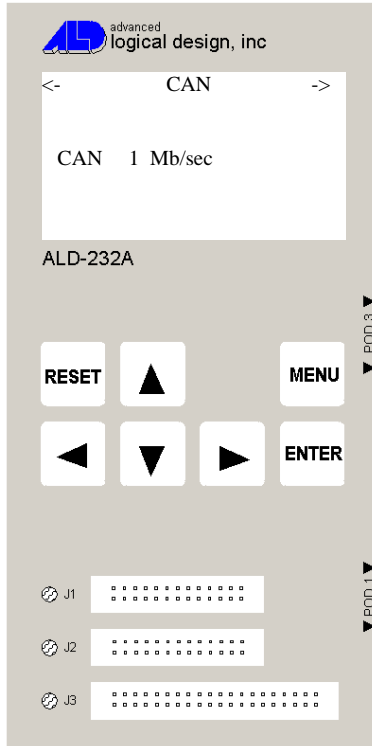


Figure 2: ALD-232A-CAN Top Panel

The operation of the ALD-232A-CAN in modes other than CAN is explained in the basic ALD-232A manual, also enclosed. This addendum covers only the CAN operation.

4. Analyzing CAN

The CAN Bus has been adopted for many automotive and industrial applications. It is a two-wire differential serial bus with a non-collision priority arbitration mechanism. It is used for many low speed applications (up to 1 Mbs). The ALD-232A-CAN conforms to the protocol described in the CAN Specification Version 2.0, by Robert Bosch GmbH. The ALD-232A-CAN supports all bit rates suggested in the CiA Draft Standard 102 version 2.0 (10, 20, 50, 100, 125, 250, 500, 800 kBits/s and 1 Mbits/s).

4.1. Run Time Display

When the analysis probe is running a CAN measurement, the unit's display will show the following information:

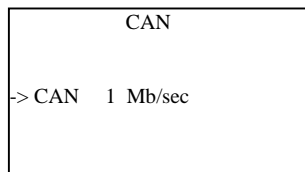


Figure 3 – Changing CAN transfer rate

This indicates that the CAN measurement is being done at 1 Mb/sec. To change the transfer rate use the ▼ to move the selection arrow (→) down to the second line where

- DATASTB** This signal indicates to the logic analyzer, that the shift register contains information of interest to the data formatter. The AldTag tells the data formatter, what kind of information is being presented. DATASTB is used to clock qualify the logic analyzer's capture of AldData.
- SID** The part of the shift register holding the Standard ID, when the correct AldData is presented and the IDE bit is zero.
- EID1** The part of the shift register holding the first 11 bits of the EXTENDED ID, when the correct AldData is presented and the IDE bit is one.
- EID2** The part of the shift register holding the second 16 bits of the EXTENDED ID, when the correct AldData is presented.

The values for the AldData field are shown in the following table.

ALDtag	Data presented	Symbol
0	End of frame	EOF
1	First part of ID	SID/EID1
2	Second part of Extended ID	EID2
3	DLC	DLC
4	Data	DATA
5	CRC	CRC
6	Reserved	RSVD
7	Error Frame	ERR

Table 2: AldTag meanings

4.3. State Analysis

4.3.1. Data collection

The ALD-232A-CAN is primarily intended for state analysis of the CAN bus. Both POD 1 and POD 2 are used for the measurement. The analysis probe accumulates serial data into an internal 16-bit shift register and presents the data to the analyzer at key times of the frame, along with AldTag to indicate the meaning of the data.

4.3.2. State Analyzer Configuration

The overview of the configuration is shown in Figure 5.

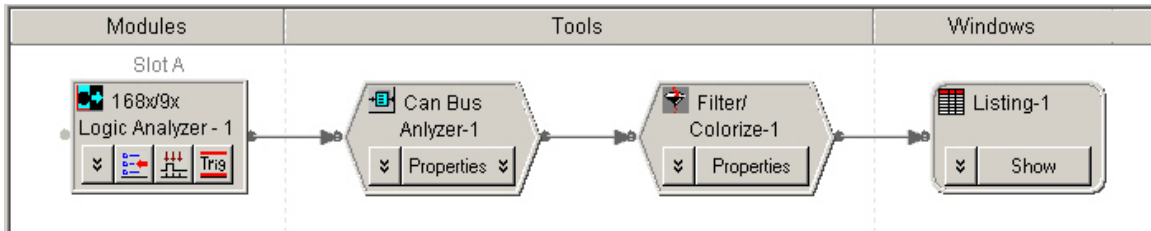


Figure 5: State Analyzer Overview

The data is acquired by the Logic Analyzer tool from the CAN shift register in the ALD-CAN pod. The Can Bus Analyzer tool analyzes the data from the logic analyzer and produces 4 additional fields based upon that analysis:

- ID
- (DLC) Message
- Type
- Data

The Listing tool may display any of the logic analyzer fields and/or the Can Bus Analyzer fields. The Filter/Colorize tool allows the display to do just that, filter in or out or colorize certain records based upon some user-supplied criteria. The ALD-232A-CAN option includes 2 ‘canned’ [sorry about that] configurations that illustrate different ways to look at the data.

The ALD supplied configuration files are installed on the analyzer’s C drive: C:\Documents and Settings\current_user\My Documents\Agilent Technologies\Logic Analyzer\Config Files\CAN. The supplied files also include some example trigger setups. The files supplied are:

Name ▲	Size	Type	Modified
CANfilter.xml	9 KB	XML Document	2/21/2006 11:12 AM
CANfilter_DATA.xml	17 KB	XML Document	3/6/2006 1:31 PM
CANfilter_DATA01of01.mfb	9 KB	MFB File	3/6/2006 1:31 PM
CANnoFilter.xml	9 KB	XML Document	2/21/2006 12:08 PM
CANnoFilter_DATA.xml	17 KB	XML Document	3/6/2006 1:32 PM
CANnoFilter_DATA01of01.mfb	9 KB	MFB File	3/6/2006 1:32 PM
CANTiming.xml	5 KB	XML Document	2/21/2006 12:23 PM
CANTiming_DATA.xml	6 KB	XML Document	3/6/2006 1:30 PM
CANTiming_DATA01of01.mfb	68 KB	MFB File	3/6/2006 1:30 PM
EIDtrigger.xml	2 KB	XML Document	4/22/2004 3:11 PM
SIDtrigger.xml	1 KB	XML Document	4/23/2004 7:49 AM

Figure 6 - Configurations for CAN Analysis

The files CANfilter.xml, CANnoFilter.xml, and CANTiming.xml are the configuration files used to capture data. The configuration files that contain DATA in the title have some sample data saved to show the display of the CAN bus information. These files must be loaded “OFFLINE” to display the saved data. The trigger setup in the configuration files is open [will trigger on anything]. There are also two trigger files

included to show how to trigger on an Extended ID (EID) or Standard ID (SID). The procedure to load these files is covered in section 4.3.5.

4.3.3. 'No Filter' Configuration

In the CANnoFilter.xml configuration, the Filter/Colorize tool is disabled. The Listing tool only displays 2 fields, both from the Can Bus Analyzer tool, *Type* and *Data* together with the time field.

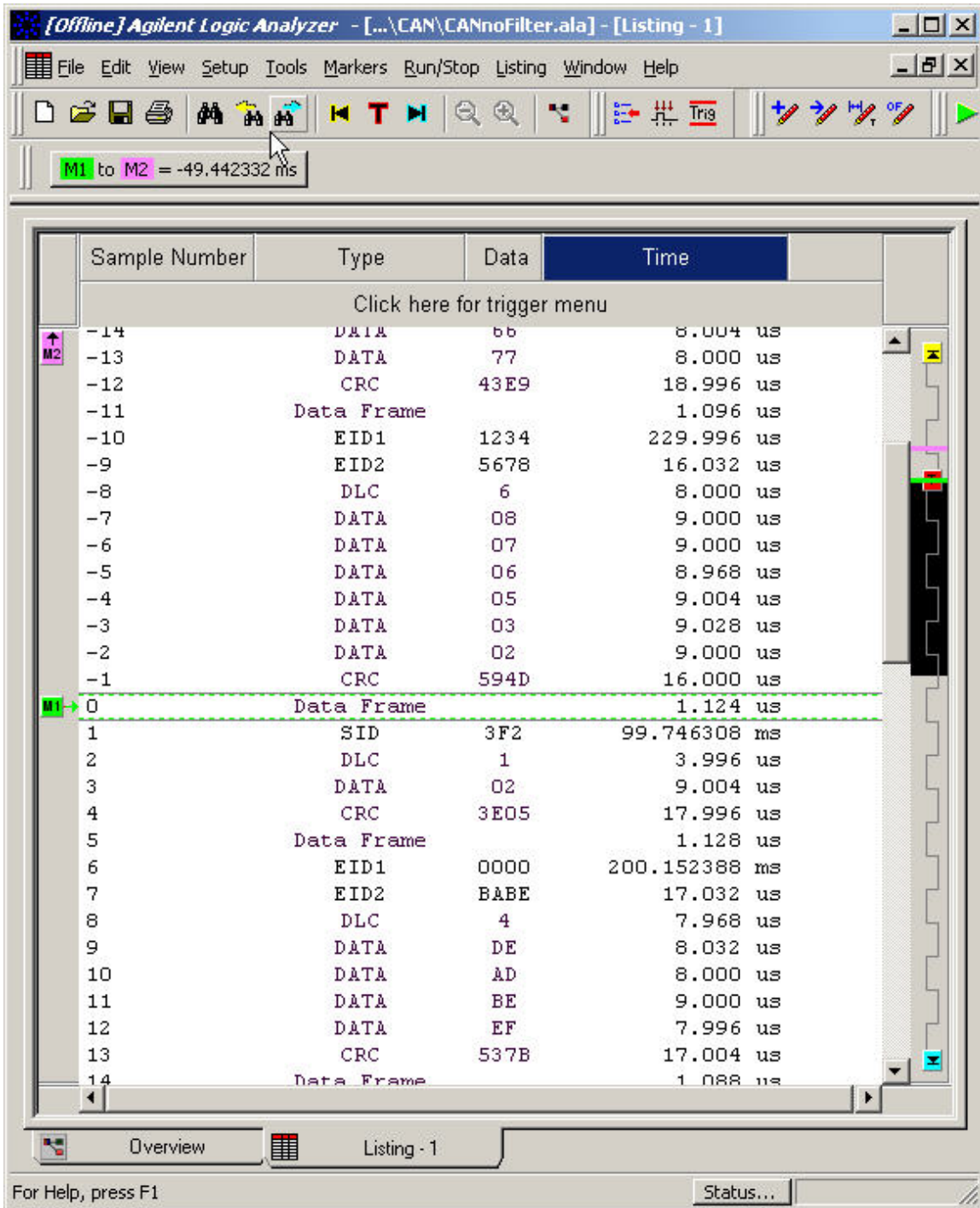


Figure 7: No-Filter Listing

Each CAN byte is displayed on a separate line and the Type field identifies what it is. At the end of the received frame, the type EOF [End of Frame] is detected and 'Data Frame' is displayed. Note that none of the fields from the actual logic analyzer are displayed, only those from the data formatter tool, Can Bus Analyzer.

This type of display is useful in dissecting a message frame. It provides the most detail, while it does take considerable space.

4.3.4. Filtered Configuration

In this configuration, the Filter/Colorize tool is enabled and the Listing tool displays 3 fields from the data formatter, Can Bus Analyzer: Type, ID and (DLC) Message. The filter settings are shown in Figure 8.

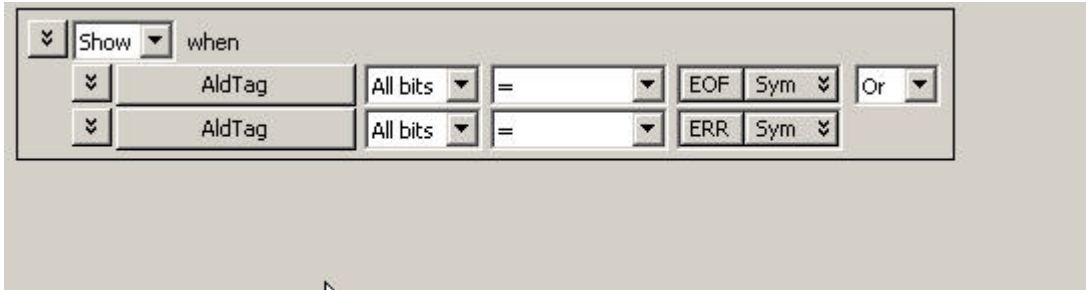


Figure 8: Filter Settings

This tells the analyzer to only SHOW data lines that have the AldTag= EOF or ERR [End of Frame or Error Frame]. At the end of either an error or regular frame, the Can Bus Analyzer tool presents the entire message in the (DLC) Message field. This field contains the DLC for the message in () and all of the data in the message field. The ID field displays the 11 or 29 bit of the message ID. The standard ID (11 bit) value is shown as 3 hex digits whereas the 29 bit Extended ID is shown as 8 hex digits. An example is shown below in Figure 9.

This type of display is more compact and useful for high-level understanding of the message traffic.

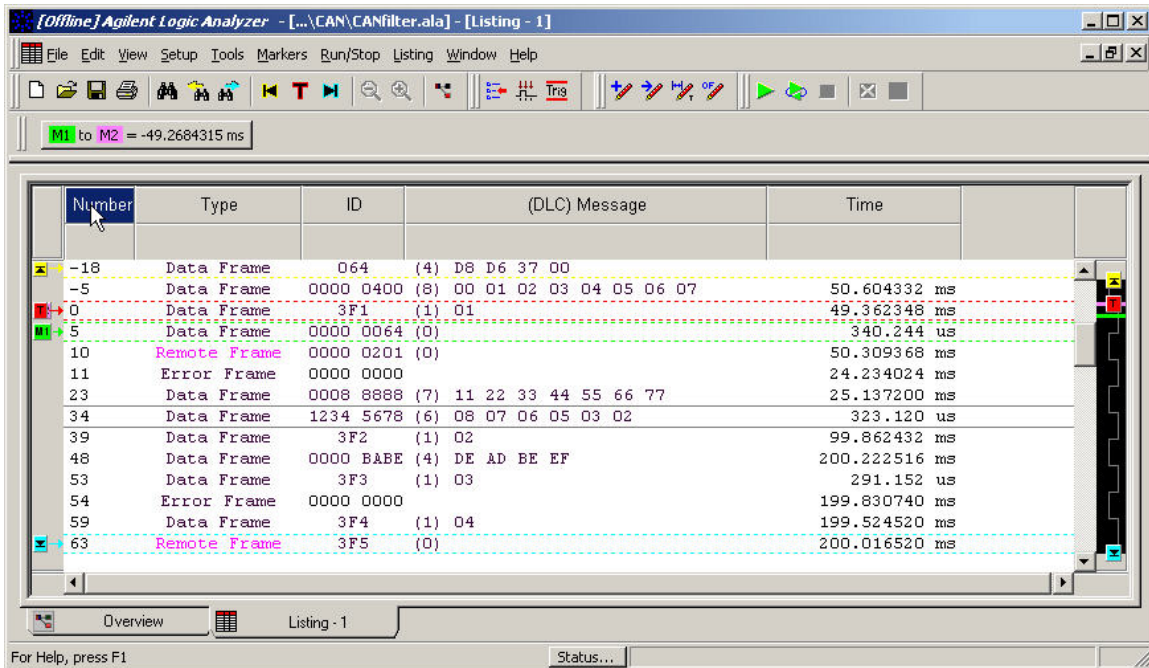


Figure 9: Filtered Data

4.3.5. Triggering

The logic analyzer has very sophisticated triggering capabilities. Developing a skill in using these capabilities is key to discovering any DUT problems in general. The ALD-CAN provides 2 examples of trigger scenarios: a trigger on a simple 11 bit ID and one for a 29 bit, extended ID.

At first this may seem trivial, but because the ID fields in the CAN message are embedded in the data stream and not available as parallel data, the problem is a little more complicated.

To enter a trigger event for CAN analysis, you must use the advanced trigger feature. With the CANnoFilter.xml configuration loaded select the Setup menu and then **Advanced Trigger** or use the icon on the tool bar as shown in Figure 10 below. That will take you to the screen shown in like the one shown in Figure 11. At the bottom of this screen, chose **Recall** then **Open**. From here, navigate to C:\Documents and Settings\current_user\My Documents\Agilent Technologies\Logic Analyzer\Config Files\CAN and select SIDtrigger.xml, click on Open. This trigger looks at the data bits in the shift register that comprise the SID at the time when the SID is present and compares it to the desired SID. If they match, it then goes to the next step where it waits for the End of Frame (EOF). After finding EOF, the analyzer triggers and fills memory.

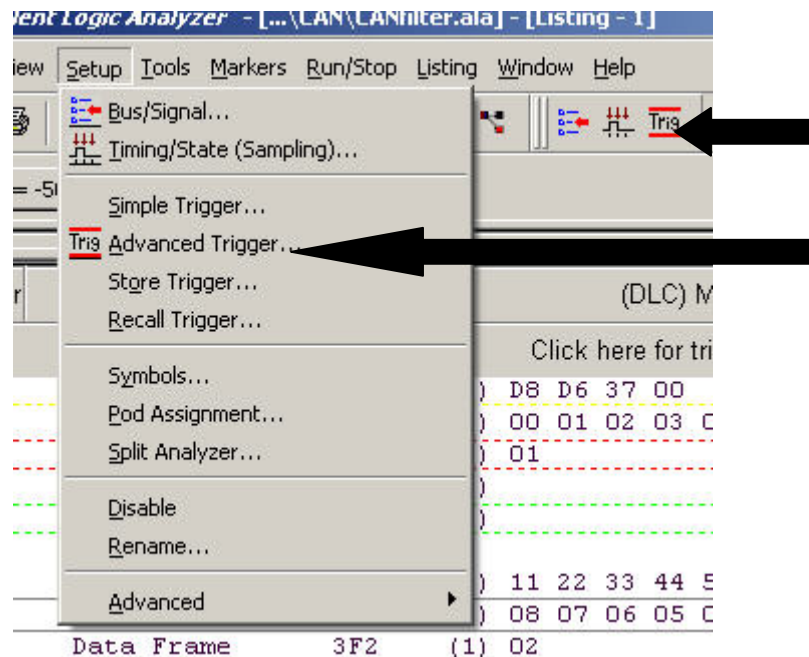


Figure 10: Advanced Trigger Entry

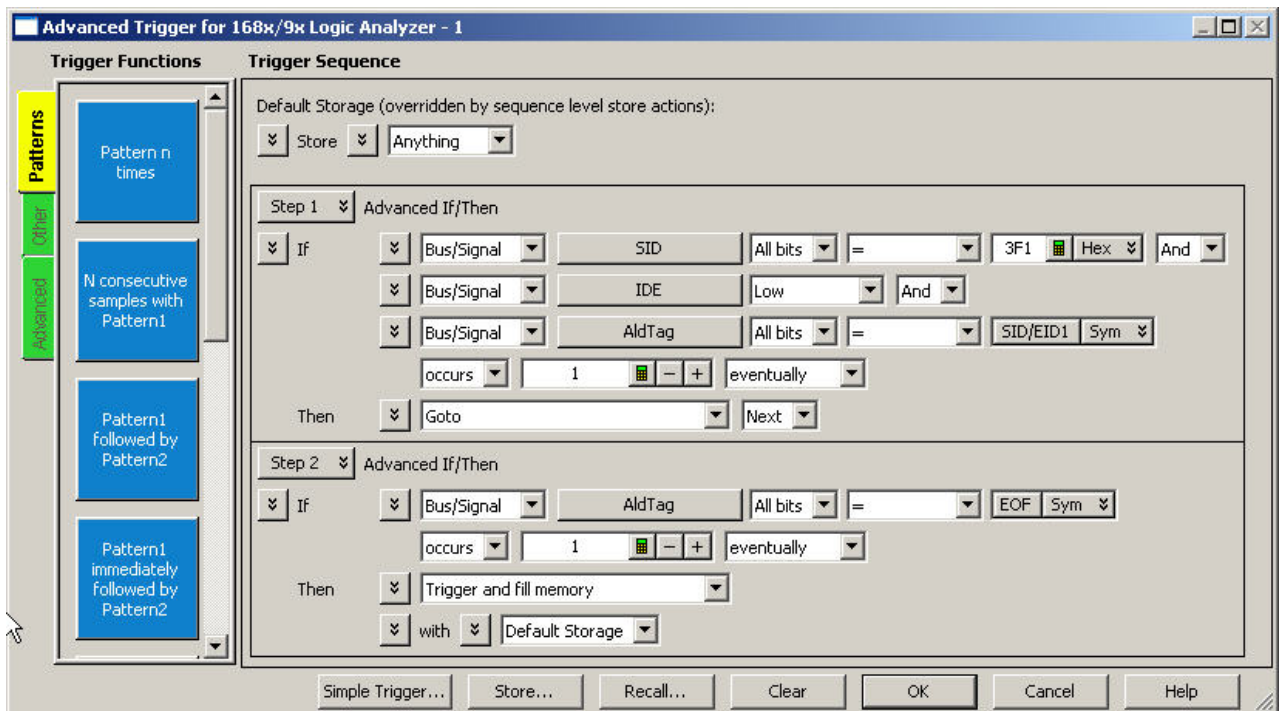


Figure 11: SID Trigger

Let's look at this trigger in more detail. In Step 1, the following conditions must be met:

- 1) The SID must match the desired SID (in this example, 3F1)
- 2) The IDE bit (signifying a SID or EID) must be low (SID) and

3) The AldTag must identify the current state as valid in the SID/EID1 position. This determines the condition that we are looking for. Step 2 delays the actual trigger until the EOF condition is true by waiting for AldTag=EOF. This is not strictly necessary, but if you use the filtered configuration, it only displays data at this point and thus the trigger must wait until EOF.

Now let us examine a slightly more complicated trigger, the EID trigger shown in Figure 12.

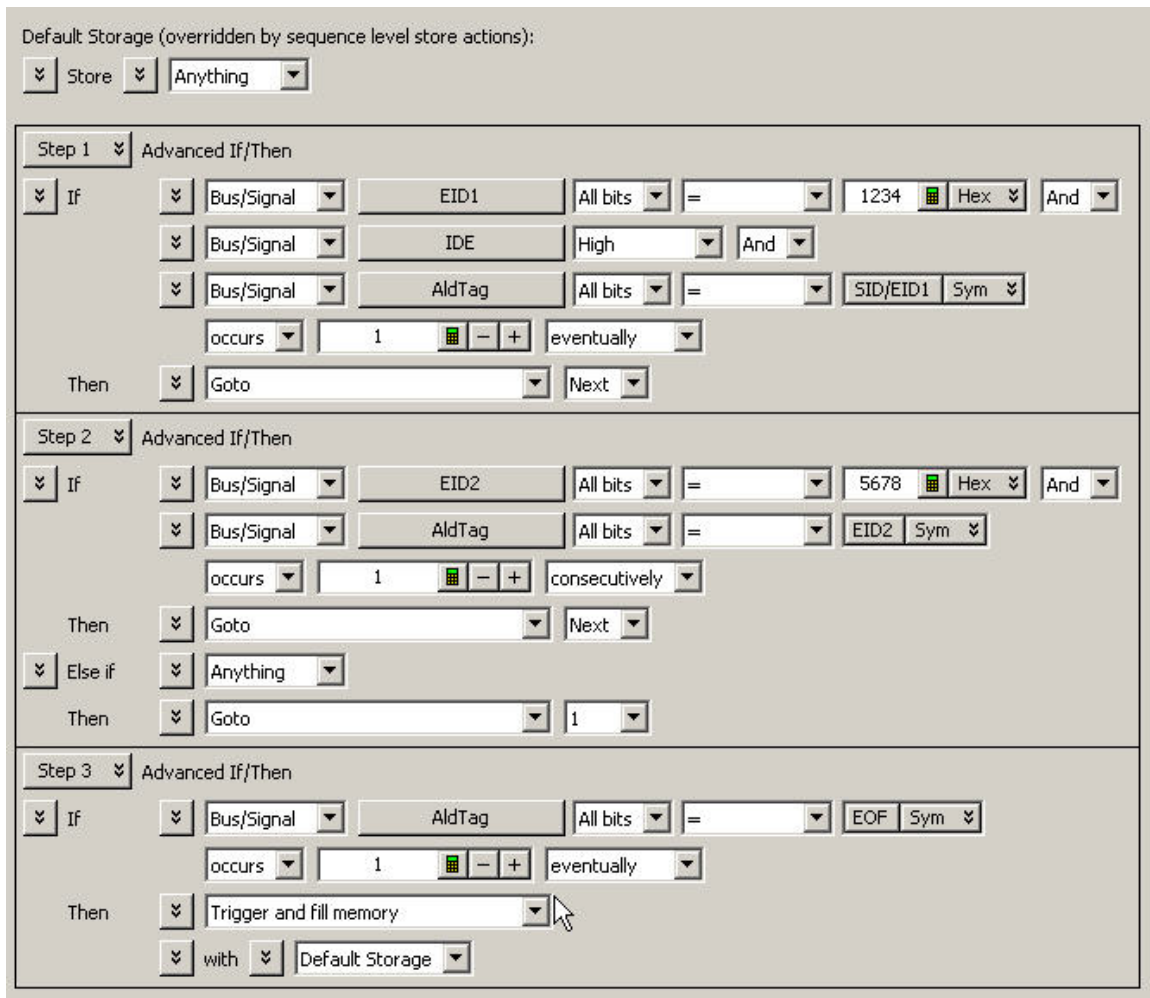


Figure 12: EID Trigger

In the EID case, both components of the extended ID must be found. Step 1 locates the first component (1234) of the EID as well as the EID valid identifier (IDE) and the correct position of the shift register (AldTag=SID/EID1). Step 2 then looks for the second part of the EID (5678 in this example). Note that if it does not find it, it returns to step 1. Finally the logic analyzer triggers when EOF occurs.

from an internal clock generator, running at the user requested frequency and synced to *CAN* input transitions.

The ALD FPGA generates a signal *INFRAME*, to enclose a CAN message frame. In the above figure, it can be seen that this signal forms a simple trigger. *DATASTB* is used in state analysis to qualify the *CLKCAN* signal and capture an event. When used in timing, it will mark the *CLKCAN* edge with a valid *AldTag*.

FRMCOUNT is a simple bit counter that is used to locate a bit in the serial stream. It counts from zero at the detection of the first *CAN* bit. In the above example, you will note that counts 04 and 0E appear to be stretched one clock. The counter only counts valid data bits and skips stuffed bit.

5. Theory of Operation

The block diagram of the ALD-232A-CAN analysis probe is shown below (See Figure 14):

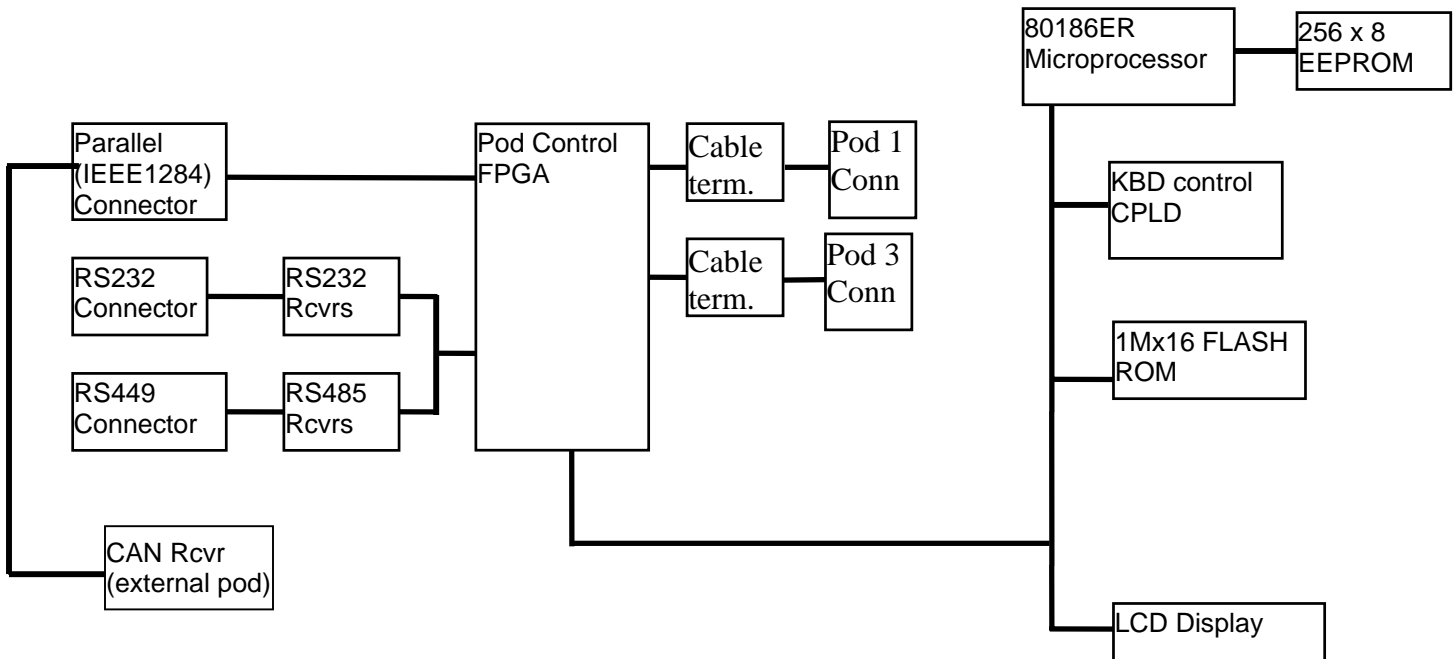


Figure 14 – Operation of 232A-CAN

The ALD-232A-CAN is a complete small processor system, with RAM, ROM, keyboard and display and a data capture system. The microprocessor system downloads the Pod Control FPGA with data appropriate to the operation selected (CAN in this case), and receives parametric data (frequency) from the user to program the FPGA.

The 80186ER microprocessor system is conventional in design with the processor running at 32 MHz. Each time that the user selects a new parameter or option, that selection is recorded in the EEPROM. Data is also stored in the EEPROM after any

mode change persists for more than 5 secs. This allows that selection to survive power loss. The Flash PROM contains not only the operational data, but also binary images of the CAN analysis code for the FPGA. The Pod Control FPGA is downloaded under program control with FPGA binary image for RS232, RS449, IEEE1284, I²C or CAN or any option that may be included.

The Pod Control FPGA collects data from the appropriate interface, RS232, RS449, IEEE1284 or and presents it to the pins of the logic analyzer. The FPGA acts as a dual UART or USRT.

In CAN mode, the CAN signal from the CAN interface pod is connected via the parallel (J1) port connector to the FPGA for processing. This FPGA contains all of the necessary logic to process the CAN signal, decoding the frame and presented it to the logic analyzer.

The FPGA directly generates the signals that are sent to the logic analyzer's Pod 1 and Pod 2.

Rev C
03/06

Advanced Logical Design, Inc.
12280 Saratoga-Sunnyvale Road. Suite 203
Saratoga, CA 95070
Voice: (408)446-1004
Fax: (408)446-1079
email: techsup@ald.com
URL: <http://www.ald.com>