EMX-1434
204.8kSamples/s 4 Channel Arbitrary Waveform Generator with Tach and DIO

User's Manual

P/N: 82-0145-100

Released Feb 12th, 2014
Revised Sep 10, 2015

VTI Instruments Corp.

2031 Main Street
Irvine, CA 92614-6509
(949) 955-1894

TABLE OF CONTENTS

CERTIFICATION

VTI Instruments Corp. (VTI) certifies that this product met its published specifications at the time of shipment from the factory. VTI further certifies that its calibration measurements are traceable to the United States National Institute of Standards and Technology (formerly National Bureau of Standards), to the extent allowed by that organization's calibration facility, and to the calibration facilities of other International Standards Organization members. Note that the contents of this document are subject to change without notice.

WARRANTY

The product referred to herein is warranted against defects in material and workmanship for a period of one year from the receipt date of the product at customer's facility. The sole and exclusive remedy for breach of any warranty concerning these goods shall be repair or replacement of defective parts, or a refund of the purchase price, to be determined at the option of VTI. Note that specifications are subject to change without notice.

For warranty service or repair, this product must be returned to a VTI Instruments authorized service center. The product shall be shipped prepaid to VTI and VTI shall prepay all returns of the product to the buyer. However, the buyer shall pay all shipping charges, duties, and taxes for products returned to VTI from another country.

VTI warrants that its software and firmware designated by VTI for use with a product will execute its programming when properly installed on that product. VTI does not however warrant that the operation of the product, or software, or firmware will be uninterrupted or error free.

LIMITATION OF WARRANTY

The warranty shall not apply to defects resulting from improper or inadequate maintenance by the buyer, buyer-supplied products or interfacing, unauthorized modification or misuse, operation outside the environmental specifications for the product, or improper site preparation or maintenance.

VTI Instruments Corp. shall not be liable for injury to property other than the goods themselves. Other than the limited warranty stated above, VTI Instruments Corp. makes no other warranties, express or implied, with respect to the quality of product beyond the description of the goods on the face of the contract. VTI specifically disclaims the implied warranties of merchantability and fitness for a particular purpose.

TRADEMARKS

Java Runtime Environment™ are trademarks or registered trademarks of Sun Microsystems, Inc. or its subsidiaries in the United States and other countries. LabVIEW™ and LabWindows/CVI™ are trademarks of National Instruments Corporation. Visual Basic®, Windows®, and Internet Explorer® are registered trademarks of the Microsoft Corporation or its subsidiaries. Linux® is a registered trademark of the Linux Foundation. IVI™ is a trademark of the IVI Foundation. Bonjour™ is a trademark of Apple, Inc.

RESTRICTED RIGHTS LEGEND

Use, duplication, or disclosure by the Government is subject to restrictions as set forth in subdivision (b)(3)(ii) of the Rights in Technical Data and Computer Software clause in DFARS 252.227-7013.

VTI Instruments Corp.
2031 Main Street
Irvine, CA 92614-6509 U.S.A.

DECLARATION OF CONFORMITY

Declaration of Conformity According to ISO/IEC Guide 22 and EN 45014

MANUFACTURER'S NAME	VTI Instruments Corporation
MANUFACTURER'S ADDRESS	2031 Main Street Irvine, California 92614-6509
PRODUCT NAME	Arbitrary Waveform Source
MODEL NUMBER(S)	EMX-1434 DAC Modules
PRODUCT OPTIONS	All
PRODUCT CONFIGURATIONS	All

VTI Instruments. declares that the aforementioned product conforms to the requirements of the Low Voltage Directive 73/23/EEC *and the* EMC Directive 89/366/EEC *(inclusive* 93/68/EEC*) and carries the "CE" mark accordingly. The product has been designed and manufactured according to the following specifications:*

SAFETY	IEC 61010-1:2001 (2nd Ed.); EN 61010-1:2001 (2nd Ed.)
EMC	IEC 61326-1:2006 EN55011 Class A Group IEC 61000-4-2 IEC 61000-4-3 IEC 61000-4-4 IEC 61000-4-5 IEC 61000-4-6 IEC 61000-4-8 IEC 61000-4-11 CISPR 11 (2004) Class A ICES-001 (Issue 4) AS/NZS CISPR 11 (2nd ED) Class A FCC Part 15 Subpart B Class A

This product was tested in a typical configuration.

I hereby declare that the aforementioned product has been designed to be in compliance with the relevant sections of the specifications listed above as well as complying with all essential requirements of the Low Voltage Directive.

February 2010

Steve Mauga, QA Manager

GENERAL SAFETY INSTRUCTIONS

Review the following safety precautions to avoid bodily injury and/or damage to the product. These precautions must be observed during all phases of operation or service of this product. Failure to comply with these precautions, or with specific warnings elsewhere in this manual, violates safety standards of design, manufacture, and intended use of the product. Note that this product contains no user serviceable parts or spare parts.

Service should only be performed by qualified personnel. Disconnect all power before servicing.

TERMS AND SYMBOLS

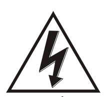
These terms may appear in this manual:

WARNING Indicates that a procedure or condition may cause bodily injury or death.

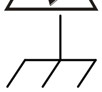
CAUTION Indicates that a procedure or condition could possibly cause damage to equipment or loss of data.

These symbols may appear on the product or in the manual:

ATTENTION - Important safety instructions



Indicates hazardous voltage.



Frame or chassis ground

Indicates that the product was manufactured after August 13, 2005. This mark is placed in accordance with *EN 50419, Marking of electrical and electronic equipment in accordance with Article 11(2) of Directive 2002/96/EC (WEEE)*. End-of-life product can be returned to VTI by obtaining an RMA number. Fees for take-back and recycling will apply if not prohibited by national law.

WARNINGS

Follow these precautions to avoid injury or damage to the product:

Use Proper Power Cord To avoid hazard, only use the power cord specified for this product.

Use Proper Power Source To avoid electrical overload, electric shock, or fire hazard, do not use a power source that applies other than the specified voltage.

The mains outlet that is used to power the equipment must be within 3 meters of the device and shall be easily accessible.

User Proper Fuse To avoid fire hazard, only use the type and rating fuse specified for this product.

Power Consumption Prior to using EMX series plug-in modules, it is imperative that the power consumption of all modules that will be installed in the mainframe be calculated for all power supply rails. The required information can be found in *Appendix B* of the *EMX Series User's Manual* (P/N: 82-0142-100). *Failure to do so may result in damaging the instrument and the chassis.*

WARNINGS (CONT.)

Avoid Electric Shock

To avoid electric shock or fire hazard, do not operate this product with the covers removed. Do not connect or disconnect any cable, probes, test leads, etc. while they are connected to a voltage source. Remove all power and unplug unit before performing any service. ***Service should only be performed by qualified personnel.***

Ground the Product

This product is grounded through the grounding conductor of the power cord. To avoid electric shock, the grounding conductor must be connected to earth ground.

Operating Conditions

To avoid injury, electric shock or fire hazard:

- Do not operate in wet or damp conditions.
- Do not operate in an explosive atmosphere.
- Operate or store only in specified temperature range.
- Provide proper clearance for product ventilation to prevent overheating.
- When selecting the installation location, be certain that there is enough space around the power plug and the outlet so that they are readily accessible. Do not insert the power cord into an outlet where accessibility to the plug cord is poor.
- All unused slots should be closed with the dummy filler panels to ensure a proper air circulation. This is critical to avoid overheating of the cards.
- DO NOT operate if any damage to this product is suspected. ***Product should be inspected or serviced only by qualified personnel.***

Improper Use

The operator of this instrument is advised that if the equipment is used in a manner not specified in this manual, the protection provided by the equipment may be impaired.
Conformity is checked by inspection.

SUPPORT RESOURCES

Support resources for this product are available on the Internet and at VTI Instruments customer support centers.

VTI Instruments Corp.
World Headquarters

VTI Instruments Corp.
2031 Main Street
Irvine, CA 92614-6509

Phone: (949) 955-1894
Fax: (949) 955-3041

VTI Instruments
Cleveland Instrument Division

5425 Warner Road
Suite 13
Valley View, OH 44125

Phone: (216) 447-8950
Fax: (216) 447-8951

VTI Instruments, Pvt. Ltd.
Bangalore Instrument Division

Mallika
#75/76, Millers Road,
Bangalore – 560 052
India

Phone: +91 80 4040 7900
Fax: +91 80 4170 0200

Technical Support

Phone: (949) 955-1894
Fax: (949) 955-3041
E-mail: support@vtiinstruments.com

Visit http://www.vtiinstruments.com for worldwide support sites and service plan information.

SECTION 1

INTRODUCTION

OVERVIEW

The EMX-1434 Smart PXI Express 4-Channel 204.8kSa/s Arbitrary Waveform Generator incorporates best-in-class analog design methodology to source high quality analog waveforms. Along with tachometer inputs and DIO channels, the instrument is ideal for a wide range of applications including noise vibration, and harshness (NVH), machine condition monitoring, rotational analysis, acoustic test, modal test, as well as shaker tables control.

FEATURES

- 4-Channel analog output, 24-Bit DAC, 204.8 kSa/s/channel
- Analog output range of 0 to ±10V
- 2-Channel tachometer input, ±25V and ±250V
- 4-Channel Digital I/O
- External emergency shutdown input
- Self-calibration
- Sine, Random, Chirp, Arbitrary waveform generation
- Continuous, Burst and Sequenced output mode
- Synchronized operation with EMX-4250/4350/4380 digitizer
- Distributed architecture using IEEE1588 precision time protocol

Analog Output

Signal quality is the highest priority. The audio DAC and the amplifiers used in the source were chosen for their ability to generate clean signals with the lowest noise and distortion. Only thin-film resistors and COG capacitors are used in the signal path to ensure low distortion.

A six pole anti-imaging filter follows the audio DAC to ensure no aliased signals in the output.

A programmable gain amplifier is incorporated in the DAC output to increase the dynamic range, lower the noise, and lower the output distortion.

Various output modes such as Sine, Chirp, continuous and burst Random, as well as Arbitrary waveform generation.

Tachometer Input

Differential input signals.

AC/DC coupling.

Programmable level and hysteresis thresholds.

High accuracy RPM estimation algorithm.

IEEE1588 timestamps for resampling EMX-4250/4350/4380 data.

Self-Calibration

Runtime self-calibration ensures that instruments deliver the most accurate results possible by compensating for ambient temperature fluctuations, without the need to disconnect field wiring. This maximizes measurement accuracy across the entire measurement path using precision internal voltage sources to validate and adjust coefficients. This eliminates inaccuracies generated by internal circuitry temperature gradients. Embedded NIST traceable calibration eliminates lengthy test system down-time, simplifies calibration processes, and reduces spare equipment requirements, maximizing facility up-time and utilization with this completely automated embedded process. All internal calibration can be performed in-place without removing instrumentation.

NOTE	**This does not eliminate the need for annual factory calibration using traceable external standards.**

Flexible Application Programming Options

EMX-1434 module is delivered with an application programming interface (API) that conforms to the industry standard IVI™ specifications for its class. The IVI drivers port seamlessly into the most commonly used application development environments such as LabVIEW™, LabWindows/CVI™, Matlab®, and Visual Studio®, among others. The intuitive APIs simplify programming and expose all available instrument functionality, eliminating the need for low-level coding. The EMX series driver is designed to support advanced functionality such as:

- Sophisticated arming and triggering options.
- Multiple FIFO models.
- Streaming API for high speed data transfer.
- LXISync API and IEEE1588 to synchronize with other LXI devices over Ethernet.

While IVI is intended for Windows®-based operating systems, VTI's innovative approach to driver development allows users to develop their applications using an IVI-like interface that can be imported into Linux® and other operating systems. This flexibility provides system developers with true OS independence without the need to sacrifice the convenience that instrument drivers deliver.

PREPARATION FOR USE

UNPACKING

When an EMX-1434 is unpacked from its shipping carton, the contents should include the following items:

- An EMX-1434
- VTI Instruments Distribution CD

All components should be immediately inspected for damage upon receipt of the unit. ESD precautions should be observed while unpacking and installing the instrument into a PXI Express chassis. The part number, the model number, and the serial number can be found on the side cover of the card as shown in Figure 1-1.

FIGURE 1-1: EMX-1434 SIDE VIEW

INSPECTING EMX-1434

The EMX-1434 module was carefully inspected both mechanically and electrically before shipment. They should be free of marks or scratches and they should meet their published specifications upon receipt.
If the module was damaged in transit, do the following:

- Save all packing materials.
- File a claim with the carrier.
- Call a VTI Instruments sales and service office.

INSTALLATION

INSTALLING EMX-1434

1. Setup the PXI Express chassis. See the chassis installation guide of your chassis model for assistance.
2. Make sure the PXI Express chassis is powered off.
3. Select a PXIe slot or a hybrid slot in the PXI Express chassis and carefully insert the EMX-1434.
4. After the EMX-1434 is inserted all the way, secure it with the screws at the top and bottom of the EMX-1434 front panel.

DETERMINE SYSTEM POWER REQUIREMENTS

The power requirements of the PXIe module are provided in the *Specifications* section. As with any backplane-based system where modules share power, such as VXI and PXI, the possibility exists where certain plug-in module combinations can draw too much power from a power supply rail. As such, it is imperative that the chassis provide adequate power for the modules installed.

INSTALLING DRIVER SOFTWARE

After the hardware has been assembled, the next step in installing an EMX-1434 is to install the instrument driver. Refer to the Section 2, *Getting Started with the Instrument*, to continue the installation process.

STORAGE AND SHIPMENT

STORING INSTRUMENTS

Store the module in a clean, dry, and static-free environment. For other requirements, see storage and transport restrictions in the ***Error! Reference source not found.***

TRANSPORTING INSTRUMENTS

Package the module using the original factory packaging or packaging that is identical to the factory packaging. Containers and materials identical to those used in factory packaging are available through VTI Instruments offices.

If returning the module to VTI Instruments for service, contact a VTI Service Center to set up an RMA (return material authorization). The following information will be required:

- Serial number.

- Model number.
- Type of service required.
- Return address.
- If applicable, a description of the problem that is being encountered which provides specific detail relating to the instrument being returned.

In any correspondence, please refer to the Serial number and RMA number. Mark the container "FRAGILE" to ensure careful handling. If it is necessary to package the module in a container other than the original packaging, observe the following (although use of other packaging material is not recommended):

- Wrap the module in heavy paper or anti-static plastic.
- Protect the front panel with cardboard.
- Use a double-wall carton made of at least 350 lbs. test material.
- Cushion the module to prevent damage.

CAUTION	Do not use styrene pellets in any shape as packing material for the module. The pellets do not adequately cushion the module and do not prevent the module from shifting in the carton. In addition, the pellets create static electricity that can damage electronic components.

SPECIFICATIONS

ANALOG OUTPUT SPECIFICATIONS

	EMX-1434	
Output Specifications		
Number of Channels	4	
Output Modes	Sine, burst sine, pseudo random noise, and band translation. Arbitrary waveform with loop or continuous output and burst	
Digital-to-Analog Converter	Independent 24-bit per channel	
Overvoltage Protection	±40 V peak	
Dynamic Range	115 dB, 0-51.2 k Hz spurious free	
THD	-95 dB, to 20 kHz	
Cross Channel Phase Match	±0.01 ° per 1 kHz	
Cross Channel Amplitude Match	±0.01 dB 10 Hz to 20 kHz	
Sampling Rates (FS)	131,072 and 204,800 samples per second	
Flatness	±0.01 dB to 35 kHz ±0.06 dB 35 kHz to 93 kHz	
Phase Linearity	±0.005° DC to 10kHz ±0.02° 10 kHz to 30 kHz ±0.5° 30 kHz to 93 kHz	
Channel-to-Channel Crosstalk	-100 dB to 10 kHz -95 dB 10 kHz to 93 kHz	
Maximum Amplitude	±10 V	
Output Impedance	<0.5 Ω	
Maximum Output Current	±25 mA	
Maximum Capacitive Load	Aberrations begin at 20 nF Overshoot and ringing but no oscillations at 1 μF	
Residual DC Offset	<±1 mV	
Amplitude Control		
Amplitude Range	-20 dB, -10 dB, -6 dB, -4 dB, -2 dB, 0 dB	
Amplitude Scale Factor	0 to 1	
Amplitude Ramp-Down Time When Shutdown Input is Active	5 ms in normal ramp-down mode	10 ms in slow ramp-down mode
Sine Output Mode		
Sine Frequency:		

	EMX-1434		
FS = 204.8 kHz	80 kHz		
FS = 131.072 kHz	51.2 kHz		
Frequency Resolution:			
FS = 204.8 kHz	47.7 µHz		
FS = 131.072 kHz	30.5 µHz		
Amplitude Accuracy	±0.05 dB		
Generated Frequency Accuracy	Clock source dependent; internal clock 50 ppm.		
Noise Output Mode			
Frequency Spans	80 kHz or 51.2 kHz Full Span – with/without decimated by 5 and/or decimated by 2 with max. of 16 times		
Passband Flatness	<1.2 dB p-p		
Crest Factor	4:1 typical		
Percent In-Band Energy	>90 % typical		
Minimum Span	Full span/($5 \bullet 2^{16}$)		
Center Frequency Resolution:			
FS = 204.8 kHz	47.7 µHz		
FS = 131.072 kHz	30.5 µHz		
Frequency and Band Translation	Span Min	Span Max	Center Frequency Max
FS = 204.8 kHz	0.244140625 Hz	80 kHz	80 kHz
FS = 131.072 kHz	0.15625 Hz	51.2 kHz	51.2 kHz
Arbitrary Output Mode			
Maximum Signal Bandwidth	80 kHz or 51.2 kHz		
Buffer Size	20M Samples total, Max 100 Buffers		
Continuous Arb data Rate	User must supply data @ rate = Fs/($5^m \bullet 2^n$) Where Fs=204.8 kHz or 131.072 kHz m = 0 or 1 n = 0,1,2… 16		
Constant Level Output			
Output Level at 1 kHz	±10 V		
Residual DC offset	<±1 mV		
Summing Input			
Maximum Input	±10 V		
Gain, Summing Input to Signal Output	1		
Input Impedance	2 kΩ		
Flatness	±0.01 dB to 35 kHz ±0.06 dB 35 kHz to 93 kHz		

ANALOG INPUT SPECIFICATIONS

Tachometer Input	
Channels	2
Frequency Input Range	1 MHz
Ranges	±25 V ±250 V
Input Type	Differential
Coupling	DC, AC 1.2 Hz
Minimum Pulse Width	600 ns
Threshold	Programmable ±95 % of range
Hysteresis	Programmable ±1 % of range

DIGITAL INPUT / OUTPUT SPECIFICATIONS

Channels	4
V In High V In Low	3.5 V min. 1.5 V max.
V Out High	4.9 V – I Out •100 Ω
V Out Low	0.1 V + I Out •100 Ω
Overvoltage Protection	±15 V peak

Max Slew Rate	50 V/µs
Shutdown Input	A normally open contact between GND and a 38 kΩ resistor pulled up to +3.3 V. *Note: that this is not a safety rated shutdown and that if a safety rated shutdown is required then the user is responsible for such, not VTI Instruments*

SYSTEM TIMING SPECIFICATIONS

IEEE 1588 Clock Specifications:	
Clock Oscillator Accuracy	±50 ppm
Synchronization Accuracy	Reports “synchronized” when < ±100 ns of the 1588 master clock
Timestamp Accuracy Resolution	As good as time synchronization down to 50 ns 25 ns
IEEE 1588 Based Trigger Timing	
Alarm:	
Trigger Time Accuracy	As good as time synchronization down to 50 ns
Time to Trigger Delay	50 ns
Receive LAN [0-7] Event:	
Trigger Time Accuracy	As good as time synchronization down to 50 ns
Time to Trigger Delay:	
Future Timestamp	50 ns typical
Past/zero Timestamp	1 ms max.
Hardware Trigger Timing:	
DIO Bus Time to Trigger Delay	57 ns typical

ENVIRONMENTAL SPECIFICATIONS

Operating Temperature	-20 °C to 50 °C
Storage Temperature	-40 °C to 70 °C
Humidity	5 % to 95 % non-condensing
Altitude Shock and Vibration Random Vibration Sinusoidal Shock	3,000 m Conforms to MIL-PRF-28800F 10 min per Axis, MIL-PRF-28800F Class 3 5 to 55 Hz Resonance Search per MIL-PRF-28800F Class 3 each Axis 30 g/Axis, 11 ms half Sine pulse per MIL-PRF-28800F Class 3

Notes:

1) All specifications are typical unless otherwise stated as a minimum or maximum.
2) All specifications are subject to change without notice.
3) All specifications met within 24 hours and 5 °C of self-calibration temperature unless otherwise specified.

PHYSICAL CHARACTERISTICS

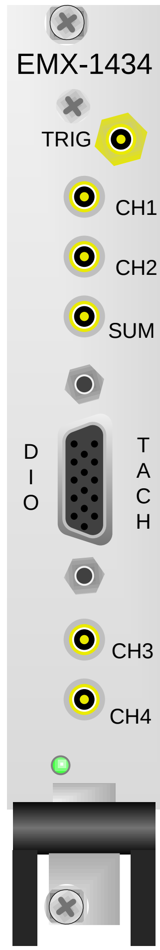


FIGURE 1-2: FRONT VIEW

PHYSICAL SPECIFICATIONS

Size	Single PXI slot – 160 mm × 100 mm (6.3 in. × 3.96 in.)
Weight	306 g (10.8 oz.)
Coaxial Connectors	SMB Amphenol 142184
Tach / Digital I/O	15-pin High Density Female Norcomp Inc. 181-015-213R561

POWER REQUIREMENTS

+12 V	638 mA typical	840 mA maximum
+5 V	0	0
+5 V standby	0	0
+3.3 V	1.203 A typical	1.39 A maximum
-12 V	0	0

Safety and EMC

Safety Compliance:	EN 61010-1, IEC 61010-1
	UL 61010-1, CSA 61010-1
	2006/95/EC; Low-Voltage Directive (safety)
EMC Compliance:	EN 61326 (IEC 61326): Class A emissions, basic immunity
	EN 55011 (CISPR 11): Group 1, Class A emissions
	AS/NZS CISPR 11: Group 1, Class A emissions
	FCC 47 CFR Part 15B: Class A emissions
	ICES-001: Class A emissions

EMX-1434 Tach Digital I/O Connector Pin Assignments

The table below shows the connector pin assignments for the EMX-1434.

Pin locations looking at the D-Subminiature connector on the EMX-1434 front panel.

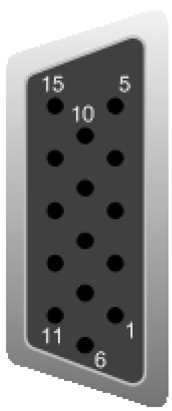


Figure 1-3: Tach Digital I/O Connector

Pin	Description
1	Tachometer input Ch1 +
2	Tachometer input Ch1 -
3	DAC marker output (future use)[1]
4	Digital input/output Ch0
5	External shutdown input
6	Ground
7	Ground
8	Ground
9	Digital input/output Ch1
10	Ground
11	Digital input/output Ch2
12	Digital input/output Ch3
13	Tachometer input Ch2-
14	Tachometer input Ch2+
15	Ground

[1] This feature is currently not supported.

SECTION 2

GETTING STARTED WITH THE INSTRUMENT

INTRODUCTION

This section provides assistance in getting the EMX-1434 instruments running and making simple measurements. It shows how to install the instrument drivers and how to run some of the example programs that are included.

SYSTEM REQUIREMENTS

- A PC with either a supported Windows OS (XP, Vista 32, Windows 7 [32-bit and 64-bit], and Windows 8 [32-bit and 64-bit]) or Linux OS.
- An Ethernet port and a LAN cable to connect to EMX-2500 Ethernet controller.
- For Windows, any programming language that supports IVI-C, or IVI-COM, such as MS VisualStudio, Mathworks Matlab®, NI LabVIEW®. For Linux, a C++ compiler, such as GCC.

DRIVER INSTALLATION

To control the EMX series instruments programmatically (via a user generated program or through tools such as Agilent VEE®, NI LabVIEW®, Mathworks Matlab®, etc.), two additional components must be installed: the IVI Shared Components library (for Windows OS only) and the VTI Instruments driver (provided). For 32-bit Windows OS, install the 32-bit driver. For Windows 7 (64-bit) and Windows 8 (64-bit), the 64-bit driver installer installs both 64-bit and 32-bit compatible drivers. The following sections describe installing the required software.

IVI Shared Components Installation (Windows Only)

If this component was installed during a previous LXI instrument installation, please proceed to *Instrument Driver Installation*. First, close all other open programs, leaving only Windows Explorer open. Navigate to the *<CD-ROM Drive>:\EX Platform Requisites* directory on the CD and run the **IVISharedComponentsX.X.X.exe** program. Next, follow the on-screen instructions. Do not proceed to the next step until this installation completes successfully. If instructed to reboot the PC, it will be necessary to do so at that time.

Alternatively, the latest IVI shared components can be downloaded and installed from IVI Foundation Web page, shown in Figure 2-1.

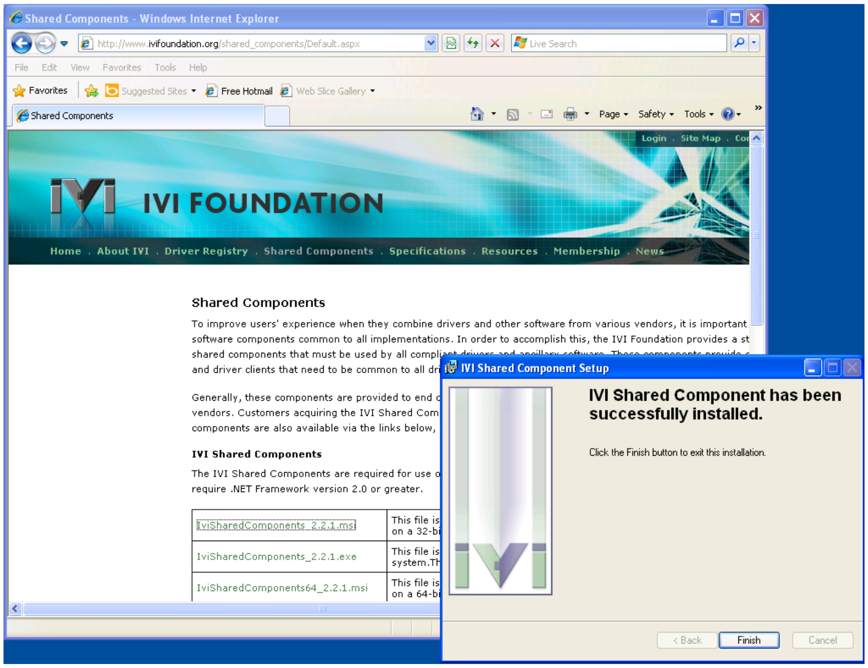


FIGURE 2-1: IVI SHARED COMPONENT INSTALLATION

Instrument Driver Installation

If the VTEXDSA instrument driver was installed previously on the host PC, proceed to *Platform/LXISync Instrument Driver Installation*. To install the VTEXDSA instrument driver, navigate to *<CD-ROM Drive>:\Drivers\LXI Drivers\EMX Series*, on the CD, open the appropriate zip file in this directory, and then run the .msi installer. See Figure 2-2.

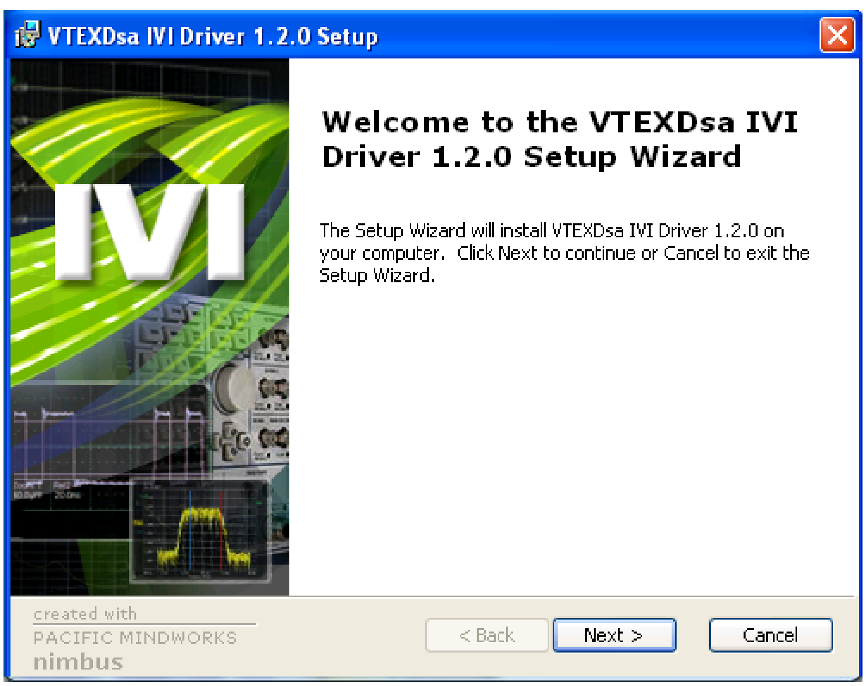


FIGURE 2-2: INSTRUMENT DRIVER INSTALLATION

The Linux drivers (32-bit and 64-bit) are located under *<CD-ROM Drive>:*\Drivers\Linux Drivers\Linux EMX Series. Find out the version of your operating system (32-bit or 64-bit) and unzip the corresponding version into a folder.

Platform/LXISync Instrument Driver Installation

NOTE	Complete this step only if the LXISync capabilities of the EMX platform are required. Also, if this driver was installed previously on the host PC, software installation is now complete.

To install the Platform/LXISync Instrument driver, navigate to *<CD-ROM Drive>:\LXI Drivers\EMX Platform Driver, IVI* on the CD and run the .msi installer located in this directory.

Please refer to *VTEX DSA Driver's online help file* for programming guidelines. Additional information about IVI drivers can be found on the web at http://ivifoundation.org. Information about the LXI standard and LXI technology can be found at http://www.lxistandard.org.

BUILDING AND RUNNING EXAMPLE PROGRAMS

The instrument drivers come with example programs that the user can build and execute. Example programs are in C++ and C# (Windows only) programming language. They are installed in the *Examples* sub-folder under the standard IVI driver installation folder (Figure 2-3 and Figure 2-4).

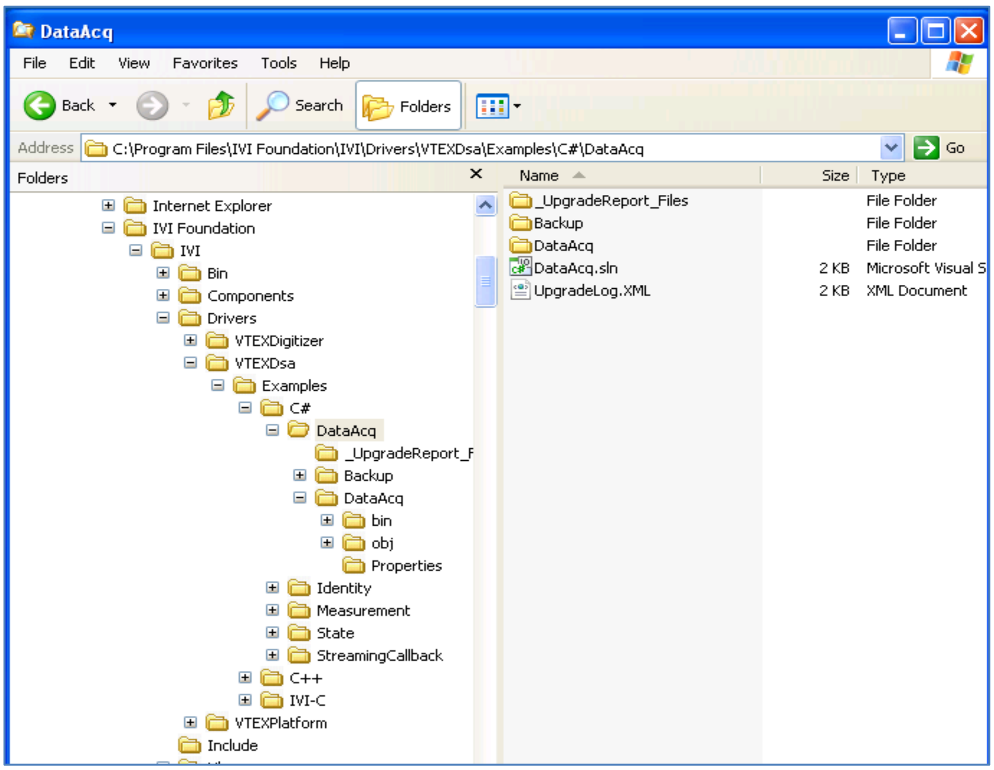


FIGURE 2-3: EXAMPLE PROGRAMS

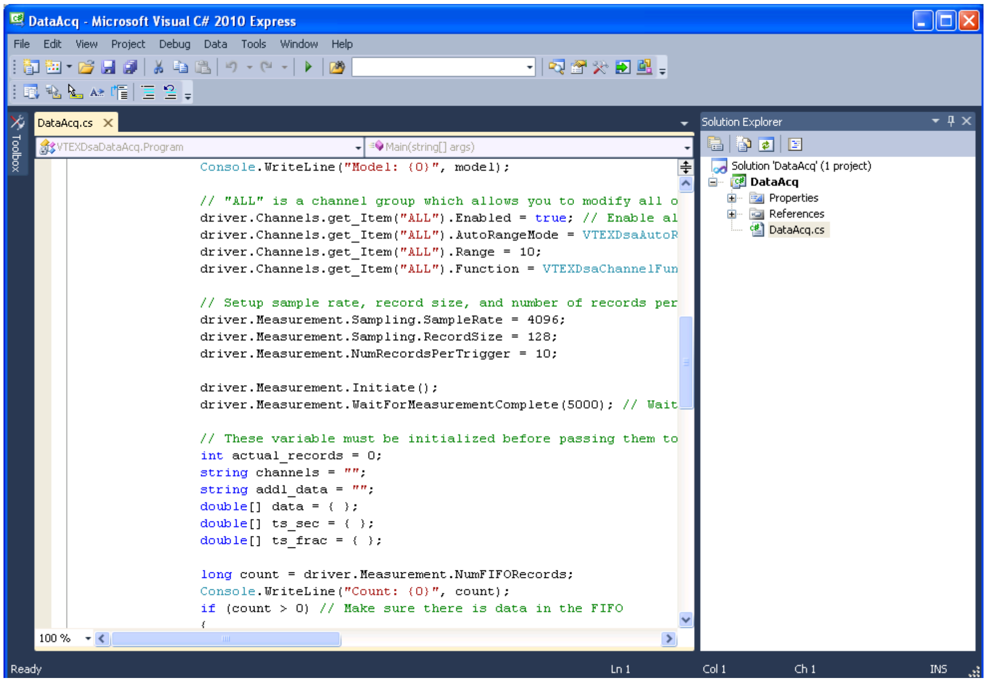


FIGURE 2-4: BUILDING AN EXAMPLE

Linux Examples

Linux examples are stored at */opt/vti/share/doc/dsa/examples/*. To build them, copy that folder to a writable location, change directory to the *examples* folder and run *make*.

```
~$ cd /tmp

/tmp$ cp -r /opt/vti/share/doc/dsa/examples/ .

/tmp$ cd examples/

/tmp/examples$ make

g++ -o Initialization Initialization.cpp -I/opt/vti/include  -
L/opt/vti/lib -Wl,-rpath=/opt/vti/lib -lDsa

/tmp/examples$ ./Initialization 10.20.10.158
```

WEB INTERFACE

From WEB browser, the user can open EMX-2500 Ethernet controller's web page by specifying EMX-2500 URL. The *Cards* page shows the list of instruments plugged into the chassis along with their slot numbers, firmware revisions and serial numbers. See Figure 2-5.

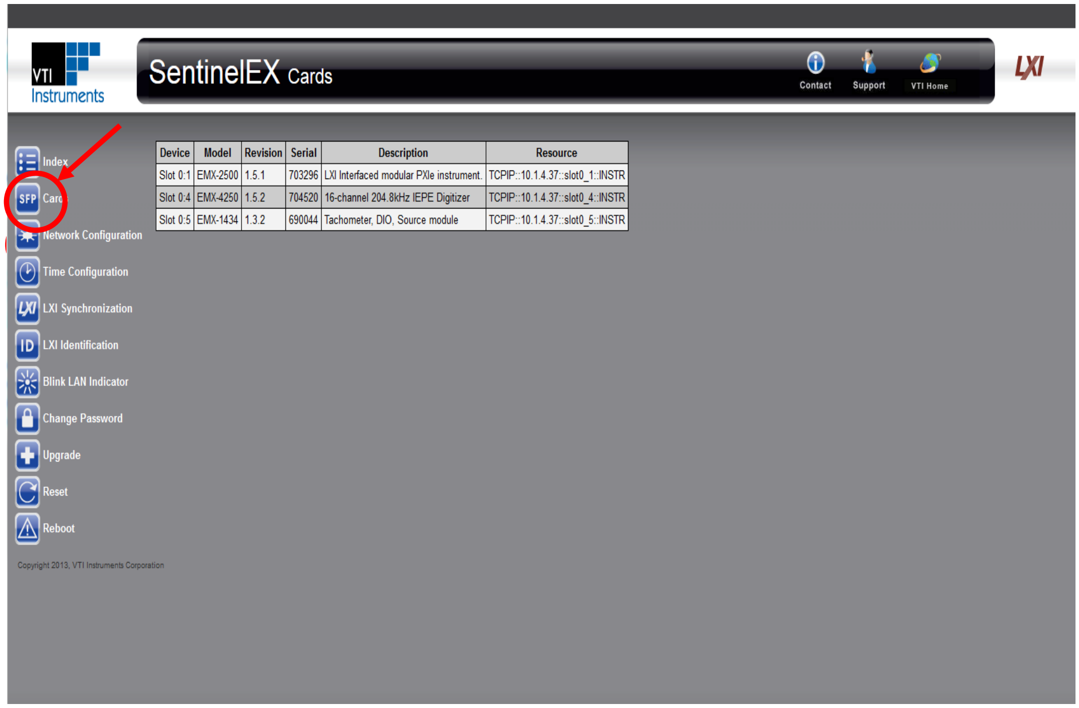


FIGURE 2-5: EMX-2500 WEB INTERFACE

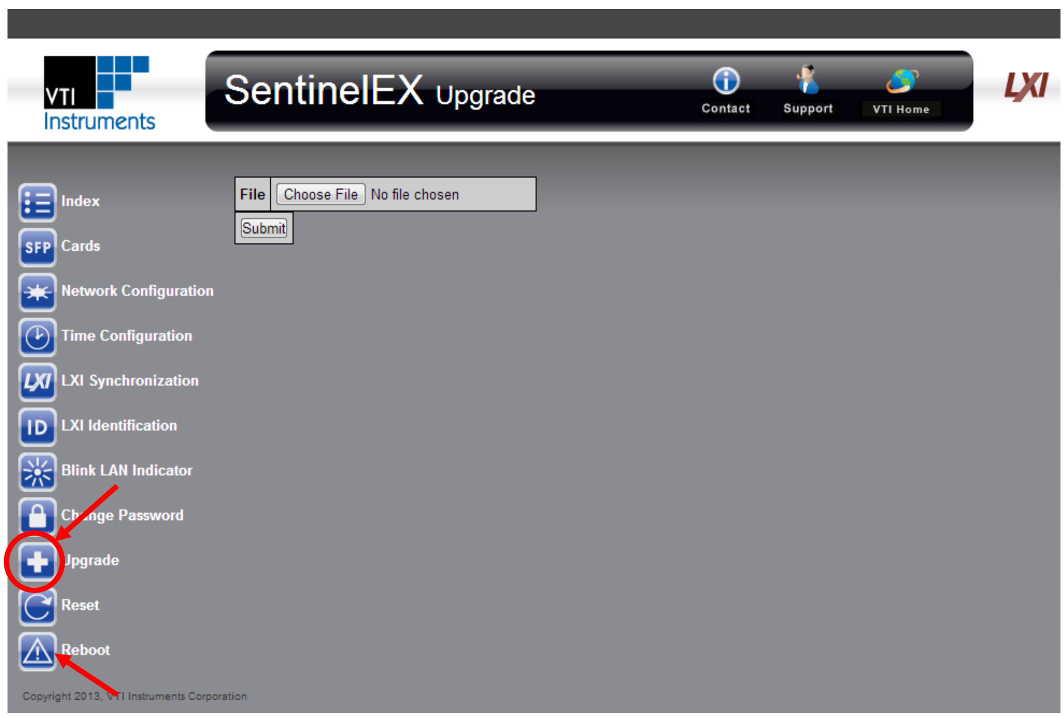


FIGURE 2-6: FIRMWARE UPDATE USING WEB INTERFACE

EMX-1434 firmware can be updated from the *Upgrade* page as shown is Figure 2-6. Simply select new firmware image file. All instruments in the chassis that match to the firmware image file will be automatically upgraded. After call cards are upgraded, the system must be rebooted for the new firmware to take effect.

SECTION 4

USING THE EMX-1434

INTRODUCTION

This section describes how to use EMX-1434 using instrument driver. The instrument driver provides API that allow the user to configure and control EMX-1434 card using high level commands.

Two types of drivers are available. One is an IVI driver that is based on the industry standard IVI driver architecture specification. The IVI driver exposes both COM and C language interface API for Windows OS. The other driver has C++ API for Linux OS. Both Windows and Linux drivers have consistent API design, so that the application software developed for one format can be easily migrated to the other operating system. Our driver is compatible with both 32-bit and 64-bit operating systems.

In general, the API description in this document applies to both Windows and Linux drivers unless otherwise specified.

INSTRUMENT DRIVERS

OVERVIEW

Two drivers, VTEXDSA and VTEXPlatform (or libDSA and libPlatform for Linux), are used to program EMX-1434 instruments along with VTI's digitizer instruments, such as EMX-4250/4350/4380. The DSA (VTEXDSA and libDSA) driver is similar to Digitizer driver, but it adds a capability to configure and control EMX-1434. It is essentially a super set of Digitizer driver. The Platform (VTEXPlatform and libPlatform) driver is used to configure EMX-2500 Ethernet controller card and CMX09/CMX18 PXIe chassis. EMX-2500 Ethernet controller makes EMX-1434 an LXI device so that they can synchronize using IEEE1588 PTP and LAN events over Ethernet.

IVI DRIVERS

The IVI Foundation defines IVI driver specifications. The IVI specification information is available at IVI Foundation web site, http://www.ivifoundation.org. The IVI-3.2 Inherent Capability Specification defines a common set of basic functionality that all IVI drivers must support. This ensures that users can perform basic operations and identify its capability for any IVI drivers using the exact same API functions. The IVI drivers are implemented using a common code provided by the IVI Foundation in order to guarantee this consistent behavior. This common code is called IVI Shared Components. The IVI Shared Components must be installed separately prior to any IVI drivers from VTI. The shared components installer is available for download from the IVI Foundation web site.

IVI Foundation specifies IVI drivers based on Microsoft Component Object Model, called IVI-COM and an IVI driver using standard C language API, called IVI-C. For those who develop applications in Windows .NET languages, such as C#, VB.NET, or other Object Oriented Languages such as C++, IVI-COM that give APIs logically organized by interfaces. IVI-C gives more traditional C language functions.

While VTI's IVI drivers are architected based on IVI-COM, the driver installer also installs a wrapper library that exposes IVI-C functions, so that the user can develop in both types of environments.

Header and Library files

IVI driver specifications define the install directory structure and the software components to be installed.

For 32-bit Windows system, the root of install directory is:
C:\Program Files\IVI Foundation\IVI.
For 64-bit Windows system, the 32-bit driver is installed at:
C:\Program Files (x86)\IVI Foundation\IVI
and the 64-bit driver is installed at C:\Program Files\IVI Foundation\IVI.

Driver header files and library files are installed in several sub directories.

Bin sub directory contains IVI-COM and IVI-C driver DLL files.

Component sub directory contains IVI-COM and IVI-C shared components files.

Drivers sub directory contains the driver specific sub directories. VTEXDSA drivers online help files and example programs are installed here.

Include sub directory contains header files.

Lib sub directory contains 32-bit import library files.

Lib_x64 sub directory contains 64-bit import library files.

DRIVER FOR LINUX OS

In addition to the IVI drivers for Windows OS, C++ libraries are provided for the Linux OS compatible with LSB (Linux Standard Base) 4.0 or later. The Linux drivers are supported on distributions running Linux kernel version 2.6.32 or later. In addition, the Linux drivers require GCC version 4.3 or later and glibc 2.11 or later. The driver for Linux organizes and names each C++ class and members consistent to the IVI-COM driver. The driver description in this manual applies to both IVI driver and the library for Linux OS.

The Linux drivers are supplied as RPM packages which are supported by a wide variety of distributions. In distributions which natively support RPM packages, such as Red Hat Enterprise Linux, Fedora Linux, or CentOS, the driver packages can be installed by running the command:

```
rpm –Uvh packagename.rpm
```

There are many other popular Linux distributions which do not natively support RPM packages, but instead use 3rd party tools to install them. Debian and Ubuntu Linux are both very popular, but do not support RPM packages out of the box. To use these packages on these systems, 'alien' is recommended which should be available in the package repository for these distributions. To install the drivers using alien, run the command:

```
alien –i packagename.rpm
```

Currently, there are both 32-bit and 64-bit driver packages for libDigitizer, libDsa, and libPlatform. There is also a package which installs common libraries and dependencies used by all drivers, libCommon. The package for libCommon is universal for both 32-bit and 64-bit systems, and should be installed before installing any of the other driver packages.

Header and Library Files

/opt/vti/include sub directory contains header files.

/opt/vti/lib sub directory contains driver shared object files.

/opt/vti/share/doc sub directory contains release notes, driver online help and example programs.

COMPATIBILITY

DRIVER, INSTRUMENTS, OS

Driver	Instruments	Operating System
Digitizer	EMX-4250, EMX-4350, EMX-4380	Windows XP, Vista, 7 (32-bit and 64-bit), 8 (32-bit and 64-bit), Linux (32-bit and 64-bit)
DSA	EMX-4250, EMX-4350, EMX-4380, EMX-1434	Windows XP, Vista, 7 (32-bit and 64-bit), 8 (32bit and 64-bit), Linux (32-bit and 64-bit)
Platform	EMX-2500, CMX09, CMX18	Windows XP, Vista, 7 (32-bit and 64-bit), 8 (32-bit and 64-bit), Linux (32-bit and 64-bit)
Switch	SMX Series Switch Cards	Windows XP, Vista, 7 (32-bit and 64-bit), 8 (32-bit and 64-bit), Linux (32-bit and 64-bit)

DRIVER AND FIRMWARE REVISIONS

The instrument driver and the firmware have three revision fields as <Major>, <Minor>, and <Build>. For the firmware installed on the instrument to be compatible with the driver being used, the <Major> version number must be equal and the <Minor> version must be equal or newer than the driver. Otherwise, the firmware needs to be updated. It is recommended to use the same <Major> and <Minor> version pair, if possible.

DRIVER API AND INSTRUMENTS

DSA drivers are designed to work with variety of digitizer and signal generator cards from VTI. Not all API functions defined in the driver apply to every card since each model is unique in the feature set. Calling unsupported API functions will result in an *unsupported* error.

DSA driver APIs	EMX-1434 only	EMX-1434 with EMX-4250/4350/4380

Alarm	**Not supported**	**Not supported**
Arm	Supported	Supported
Calibration	Supported	Supported
Channels	**Not supported**	Supported AutoCal, AutoRange, Filter, Offset, Measurement, Strain, Temperature, Transducer, Weighting are **not supported**
Configuration	Supported	Supported
Events	Supported	Supported
Measurement	Supported DataFormat, FIFOMode, NumAcquiredRecords, NumFIFORecords, Read are **not supported**	Supported
Platform	Supported	Supported
ReferenceOscillator	Supported OutputEnabled is **not supported**	Supported OutputEnabled is **not supported**
Start	Supported	Supported
StreamingData	**Not Supported**	Supported
Sync	Supported	Supported
Temperature	Supported	Supported
Time	Supported	Supported
Trigger	Supported MaxQueueSize, QueueEnabled **not supported**	Supported MaxQueueSize, QueueEnabled **not supported**
Dac	Supported	Supported
Dio	Supported	Supported
Tach	Supported	Supported

DRIVER STRUCTURE

MEASUREMENT

This section provides information related to configuring the basic measurement setup and control. The basic measurement configuration and control can be done through driver’s Measurement interface. The Measurement interface configures parameters that are global to the entire system, rather than individual DAC, Tachometer or ADC channels, or instrument modules when more than one module is included in the driver. Measurement usually involves not only EMX-1434, but also digitizer cards such as EMX-4250/4350/4380.

The parameters which the user can set using Measurement interface are…

- Sampling parameters including DAC, ADC sampling rate, white noise span, data record size, or burst length.

- Number of data acquisition records by digitizer card at each trigger event, such as burst signal output by EMX-1434 DAC channel.

- FIFO mode of operation of digitizer card.
- Data format of digitizer card.

The Measurement interface can be used to query the current measurement state information.

- Measurement state machine state.
- Total number of records available in FIFO of digitizer card.

Methods to control measurement, such as...

- Initiating measurement.
- Aborting measurement.
- Retrieving acquired data from digitizer card.

CHANNELS AND CHANNEL GROUPS

This section provides information related to using channels and channel groups. For more detailed information, see the online help file provided with the DSA driver.

The *Channel*s interface contains both channel objects and channel group objects in the same array. A channel object represents an individual analog input channel. When there is no digitizer card in the driver session, this interface becomes empty. A channel group object represents one or more analog input channels as a group. When a driver is initialized, or reset, an array of all analog channels and one channel group object that represents all analog channels and one or more channel groups that represent all analog channels from each digitizer model. For example, when you have two EMX-4250 and one EMX-4350 in a single driver session, the Channels interface contains an array total of 39 channel objects. They are 32 EMX-4250 individual analog inputs, 4 EMX-4350 individual analog inputs, one channel group object that represents all 36 analog input channels, one channel group object for all 32 EMX-4250 inputs and one channel group for all EMX-4350 inputs. These channel groups are named as “All”, “EMX-4250”, and “EMX-4350”.

The channel array is created in the ascending slot order of the chassis. The first element in the array is the first analog input of the digitizer card that is installed at the lowest slot in a chassis. When there is more than one chassis included in the total system, the order is determined by the resource string used in the driver’s *Initialize* call.

The individual channel objects are used to configure or query individual input channel’s configuration.

The channel group objects can be used to configure multiple channels to the same value. In general, the user can configure multiple channels faster using a channel group than setting channels individually. Querying the current setting through channel group works only when all channels are set to the same value.

The channel name is defined as <slot no>!CH<channel no>. For example, 4!CH2 indicates the 2nd channel of a card installed at slot4 of a chassis. The predefined channel group names are “ALL” or an instrument model number such as “EMX-4350”. Optionally, the user can create new custom channel groups using *AddChannelGroup* method. When more than one chassis is in the session, the channel name of the 2nd chassis adds 100 to the slot number, as 104!CH2. And add 200 for the 3rd, 300 for the 4th chassis.

The *NumChannels* property gives total number of individual input channels, while the *Count* property is the total number of channel or channel group objects in an array.

DAC, TACH AND DIO CHANNELS

The *Channel*s under *DAC, Tach and DIO* interface represent an array of analog output, tachometer input and DIO channels. Individual channel object represents each analog output, or tachometer input or DIO channel. Unlike analog input *Channels* interface, there is no DAC, Tach or DIO channel group objects. For example, when you have two EMX-1434's in a single DSA driver session, the *DAC Channels* interface contains an array total of 8 DAC channel objects. They are 8 individual analog output channels of two EMX-1434 cards.

The DAC, Tach and DIO channel arrays are created in the ascending slot order of the chassis. The first element in the array is the first analog output, tachometer input or DIO channel of the EMX-1434 card that is installed at the lowest slot in a chassis. When there are more than one chassis is included in the total system, the order is determined by the resource string used in the driver's *Initialize* call.

The individual DAC, Tach and DIO channel objects are used to configure or query individual analog output, tachometer input, or DIO channel's configuration.

The channel name is defined as <slot no>!DAC(or TACH, or DIO)<channel no>. For example, 4!DAC2 indicates the 2^{nd} analog output channel of a EMX-1434 card installed at slot4 of a chassis. When more than one chassis are in the session, the channel name of the 2^{nd} chassis adds 100 to the slot number, as 104!TACH2. And add 200 for the 3^{rd}, 300 for the 4^{th} chassis.

The *Count* property is the total number of DAC, Tach or DIO channel in an array.

START, ARM, TRIGGER AND ALARM

The EMX-4250/4350/4380 and EMX-1434 implement sophisticated Arm/Trigger model as shown in Figure 3-1. This trigger model conforms industry standard trigger models defined in IVI Digitizer specification, or LXI Sync specification with some extra features.

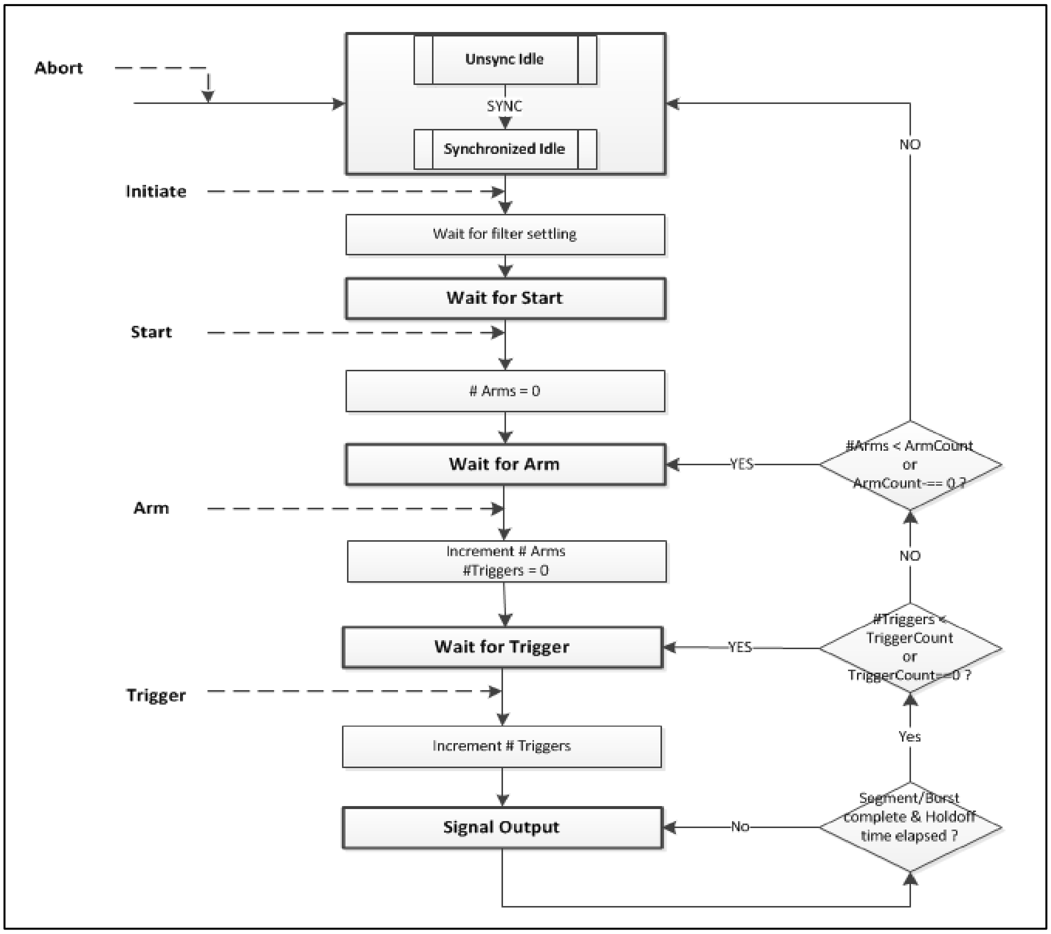


FIGURE 3-1: DAC TRIGGER MODEL

Sync

Sync is an event to synchronize the entire system. The *Sync* interface is used to configure the synchronized between cards. All cards are simultaneously started by SYNC signal and then they synchronize the state machine transition each other by coordination signal. The *Line* and *CoordinationLine* properties define which PXI trigger lines to send these signals. When multiple chassis are synchronized by IEEE1588 clock and LAN events, LAN must be specified as coordination mechanism as *CoordinationLine* = “LAN, PXI0”.

Start

Once the measurement is initiated by the Initiate command, the measurement state machine moves to a *Wait for Start* state, once the instrument completes all preparation and becomes ready to start taking data, or start generating signal from DAC channel immediately. The *Start* interface provides methods and properties to advance to the next state. The amount of time it takes to reach to the *Wait for Start* after measurement *Initiate* varies depending on the measurement configuration. It takes much longer when the measurement sample rate (or measurement span) is low, because the filter settling time is longer. The *Start* interface is useful when the user wants to have instrument complete all the preparation and hold in that state, so that it can start taking data or generating signal from DAC channels immediately without wasting time.

Arm

The instruments must be armed before triggering data acquisition or signal generation. The *Arm* interface is used to configure this arming condition. The *Sources* property in *Arm* interface defines the sources of arming events. The default arming source is *Immediate*, which means automatic arming. The *SourceOperator* property allows the user to define an arming condition by logically combining multiple arming sources with certain limitations. The *Delay* property defines the amount of time the instrument waits before moving out from *Wait for Arm* state after the defined arming conditions are met. The *ArmCount* property defines how many times the measurement repeat arming and triggering before it completes. The default is once. Setting the value of *ArmCount* to 0 forces the measurement to repeat arm and trigger indefinitely until it is aborted by the *Abort* command.

Trigger

The data acquisition by a digitizer card begins when a triggering condition is met. The EMX-1434 DAC channel starts signal output when a card is ready to output signal and a trigger condition is met. The triggering condition is configured using the *Trigger* interface. Similar to Arm interface, the *Sources* property in *Trigger* interface defines the sources of triggering events.. For digitizer card, the *Delay* property defines the amount of time between the trigger event and the beginning of the data acquisition. The *Delay* value can be negative, indicating pre-trigger data acquisition. In this case, the acquired data block starts earlier than the trigger event. This is achieved by buffering the data in the instruments FIFO priory to the trigger event. However, the *Delay* value has no effect to EMX-1434 card. The signal generation from DAC channels occur without waiting for the delay.. The signal output starts immediately at the trigger event regardless of the trigger *Delay* value, if the DAC channel has been ready. The *TriggerCount* property defines how many times the trigger events are accepted and data blocks are acquired from ADC channels. For DAC channels, when the *OperationMode* is *Burst*, burst signal is generated for every trigger event. After this amount of triggers are processed, the measurement waits for the next arming condition, or finishes. The *HoldOff* time specified the minimum amount of time the measurement has to wait before it can be triggered again, after it detected a previous trigger. Any trigger event that occurred during this *HoldOff* time will be ignored.

Figure 3-2. illustrates the relationship between burst signal output from EMX-1434 DAC channel and data acquisition records by a digitizer channel. The shaded area indicates the acquired data records by a digitizer card.

The digitizer card acquires data record after waiting for a time specified by Trigger.*Delay*. The digitizer card acquires the number of records specified by *NumRecordsPerTrigger* property.

If EMX-1434 DAC channel has been ready, when it received a trigger event, it immediately starts to output a burst signal for a duration specified by percent *BurstDutyCycle* of a *RecordSize*. The actual signal output from EMX-1434 is delayed by a certain amount (DAC Delay) due to a group delay of the output filter. When the output signal is arbitrary waveform or band limited random noise, interpolation filter is applied. This introduces additional 31.5 samples group delay. See Appendix D, Random Noise Generation section for more information.

The next burst signal will occur only after both the complete single burst record length and the trigger *Holdoff* time have elapsed.

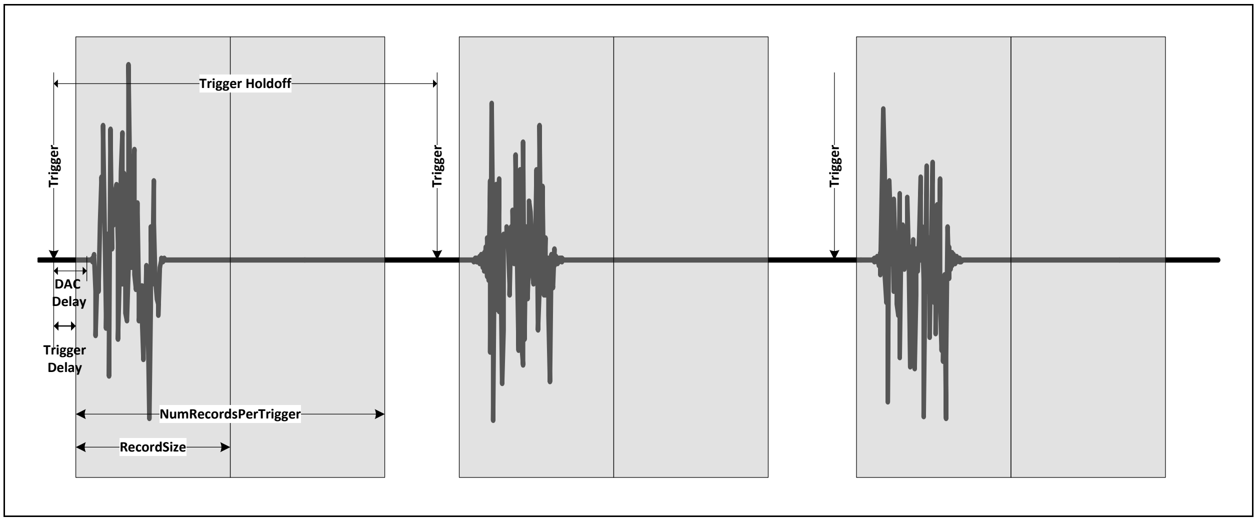


FIGURE 3-2: TRIGGER AND DELAYS IN BURST

Alarm

The Alarm is a mechanism to generate events at fixed time interval. The Alarm starts at the time specified by *TimeSeconds* and *TimeFraction*, and repeats for *RepeatCount* times. The *Period* property defines the interval between the two successive alarm events. The Alarm can be used as an arm or a trigger event source. The Alarm is currently not supported by EMX-1434.

REFERENCE CLOCK AND TIME STAMP

The *Source* property and *TimestampSource* property in ReferenceOscillator interface configures reference oscillator to generate DAC sampling clock and data time stamp clock. PXI_CLK10(10MHz) or PXIe_CLK100(100MHz) reference signal is used by default. When the system is synchronized to the IEEE1588 PTP grand master via Ethernet, the IEEE1588 clock can be used as a reference oscillator.

LXI AND LAN EVENTS

When EMX-1434 is used with EMX-2500 Ethernet controller (and optionally with Platform driver), it can act as an LXI device. These instruments can be synchronized to IEEE1588 PTP grand master clock. They can be armed or triggered by LAN events or they can generate LXI LAN events to synchronize with other LXI devices through LXI Sync interface defined by IVI standard. See Figure 3-3.

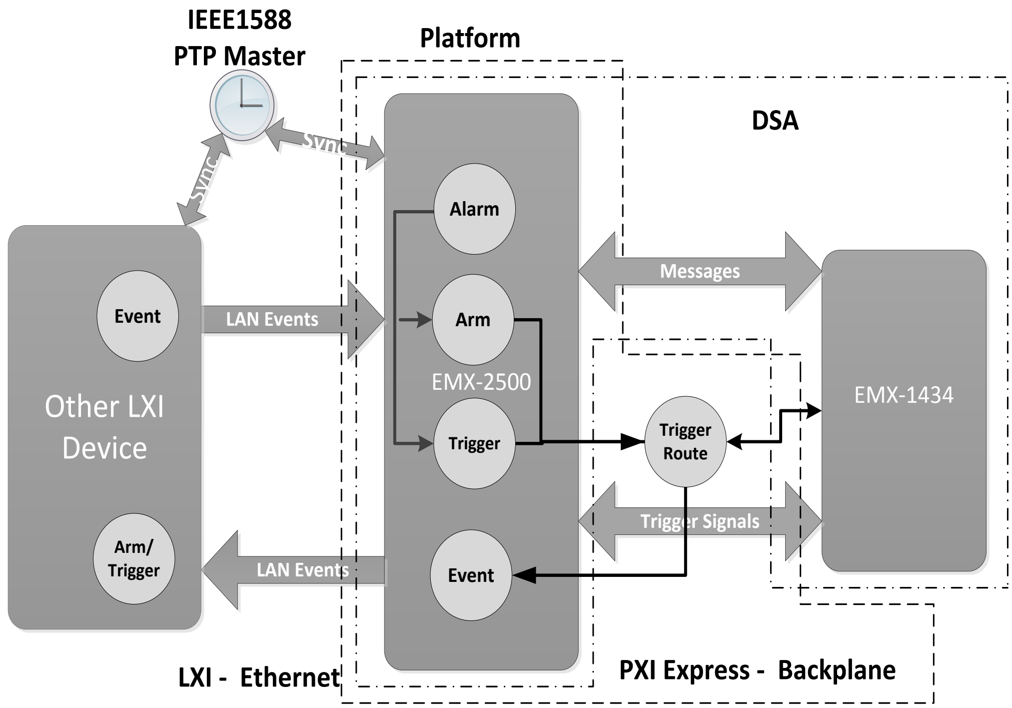


Figure 3-3: LXI and LAN Events

MULTIPLE CARDS, SEGMENTS, CHASSIS

INITIALIZING DRIVER WITH MULTIPLE CARDS

Unlike many typical instrument drivers, both DSA drivers treat multiple EMX-1434 and EMX-4250/4350/4380 cards, collectively as a single instrument. The user can initialize a single driver for one EMX-1434 and two EMX-4250 cards and treat them as a single 32 analog channel and 4 DAC, channels instrument instead of creating three separate driver sessions.

A driver must be initialized first before starting to communicate to instruments. The resource name is passed as an argument to *Initialize* method that specifies which instruments to communicate. The resource name has the following syntax.

<address 1>[::<slot 1>,<slot 2>,..,slot N>] | <address 2> [::<slot 1>,<slot 2>,…,<slot M>] | …

Where

<address x> is the IP address or host name of EMX-2500

<slot x> is the slot number identifier of the instrument in a chassis. The slot number identifier is a string as “slot0_5” indicates 5th slot of the first chassis controlled by an EMX-2500. The slot1_6 indicates the 6th slot of the 2nd chassis extended by a bus extender, the extended chassis is not supported yet.

The slot numbers are optional. When no slot numbers are specified, all supported instruments within the chassis will be used.

Where there are more than one EMX-2500 and chassis are involved, they are concatenated with “|” character.

MULTIPLE CARD MEASUREMENT, COORDINATION LINE

A PXI system can be built with more than one bus segment by using standard PCI-PCI bridge technology. VTI CMX18, 18-slot chassis has three independent trigger bus segments, while VTI CMX09, 9-slot chassis has only one PXI trigger bus segment. When all cards included in a measurement are in a single chassis and a single PXI trigger bus segment, they will be automatically synchronized using one of PXI trigger lines on the chassis backplane. The default trigger line to be used is PXI0. Any other PXI trigger line can be specified by *CoordinationLine* property in the *Sync* interface, if the PXI0 trigger line is used by other instruments in a same trigger bus segment. In multiple card configurations, the card installed at the lowest slot number becomes a trigger master and the other cards synchronize to it using a trigger coordination engine implemented in each instrument. However, in this configuration, any cards in the system can detect an arm or trigger event. For instance, all EMX-1434 or EMX-4250/4350/4380 cards can be triggered by a trigger pulse at any one of the trigger inputs in the system.

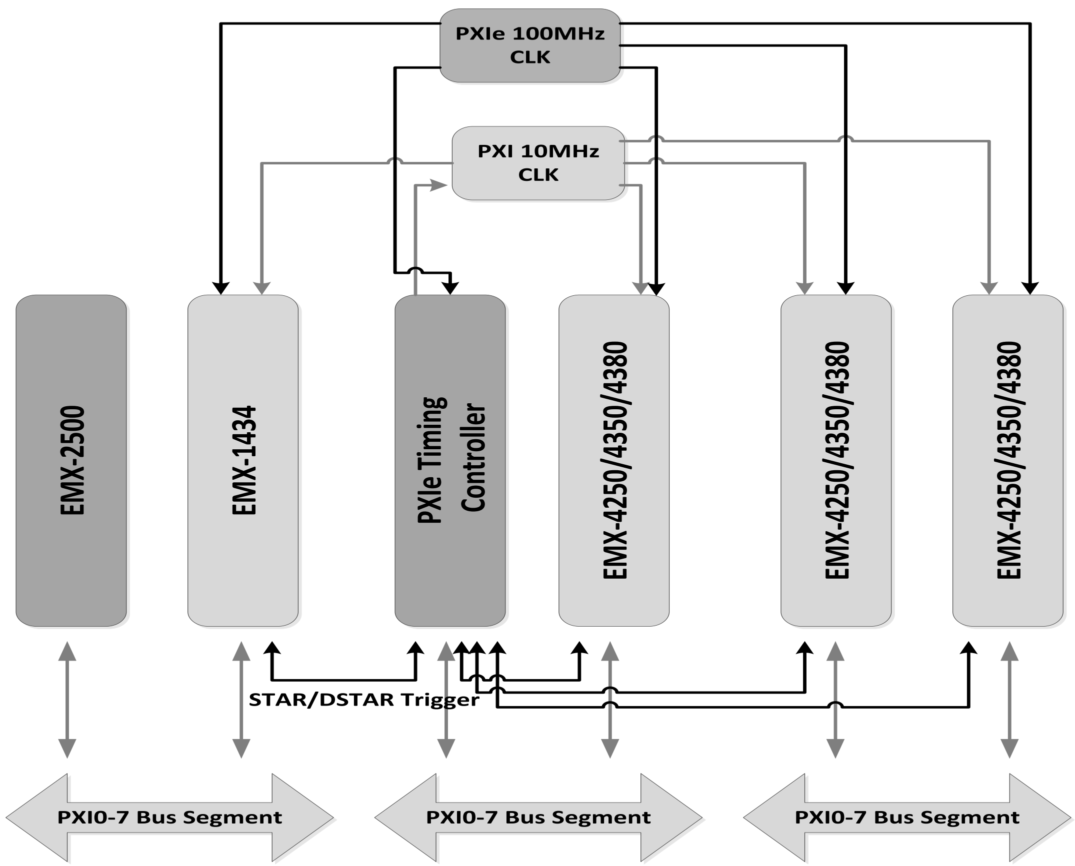


FIGURE 3-4. PXIE CHASSIS AND TRIGGER BUS

MULTIPLE SEGMENT MEASUREMENT

In order to synchronize all cards installed on multiple trigger segments, trigger lines between bus segments must be bridged together using *Routes* interface in Platform driver. This is because PXI lines in each trigger segment are independent trigger buses. See EMX-2500 User's Manual or Platform driver's online help for more detail.

When two trigger lines in different segments are bridged using *Routes* interface, the signal travels only in one direction. Unlike single segment installation, this restricts the arm or trigger source to the cards in the same segment as a trigger master card. The trigger master card is usually the card installed at the lowest slot in a chassis. The arm or trigger events cannot be initiated from any cards in the other segments, unless you establish bi-directional coordination using multiple trigger coordination lines. The bi-directional coordination is not currently supported.

MULTIPLE CHASSIS SYNCHRONIZATION WITH TRIGGER LINE

Multiple chassis synchronization is essentially the same as multiple segments synchronization within a single chassis. A coordination trigger line must be bridged between two chassis instead of two trigger bus segments. This can be achieved by physically connecting PXI trigger lines using external cables, through EMX-2500 trigger connector. The PXI trigger signal can be routed out from the master segment to the EMX-2500 TRIG connector. The signal is then distributed to all other chassis using external cables and routed into slave segments from EMX-2500 trigger connector. The signal routing between PXI trigger line and EMX-2500 TRIG connector is configured using the Platform driver. In this configuration, only cards in the master segment in a chassis can detect an arm or trigger event. All other cards in slave chassis synchronize to it.

For the best sample-to-sample synchronization between chassis, coordinating trigger events may not be good enough. The sampling clock needs to be synchronized together as well. When the VTI's CMX09 or CMX18 chassis are used, the PXI-10MHz reference clock can be synchronized together using BNC connectors on the back of each chassis.

FIGURE 3-5: EMX-2500 TRIGGER AND CMX09 10 MHZ REF

MULTIPLE CHASSIS SYNCHRONIZATION WITH LAN EVENTS

When chassis are geographically separated apart for a long distance, it may not be practical to connect a physical trigger line and a reference clock line between them. EMX-2500 makes EMX-1434 and EMX-4250/4350/4380s in each chassis an LXI device, they can be synchronized together using IEEE 1588 clock and LAN events.

When multiple chassis that contain EMX-1434 and EMX-4250/4350/4380 cards and EMX-2500 Ethernet controllers are initialized in single DSA driver session, they can be automatically synchronized using LAN event specified by LXI specification. This is equivalent to the synchronization by a trigger signal on trigger cable, but by a LAN message over Ethernet cable. In this configuration, sampling and timestamp clocks are synchronized to IEEE 1588 PTP master chosen by the best master clock (BMC) algorithm. The PTP master can be a IEEE 1588 capable GPS unit, or one of EMX-2500s. See Figure 3-6.

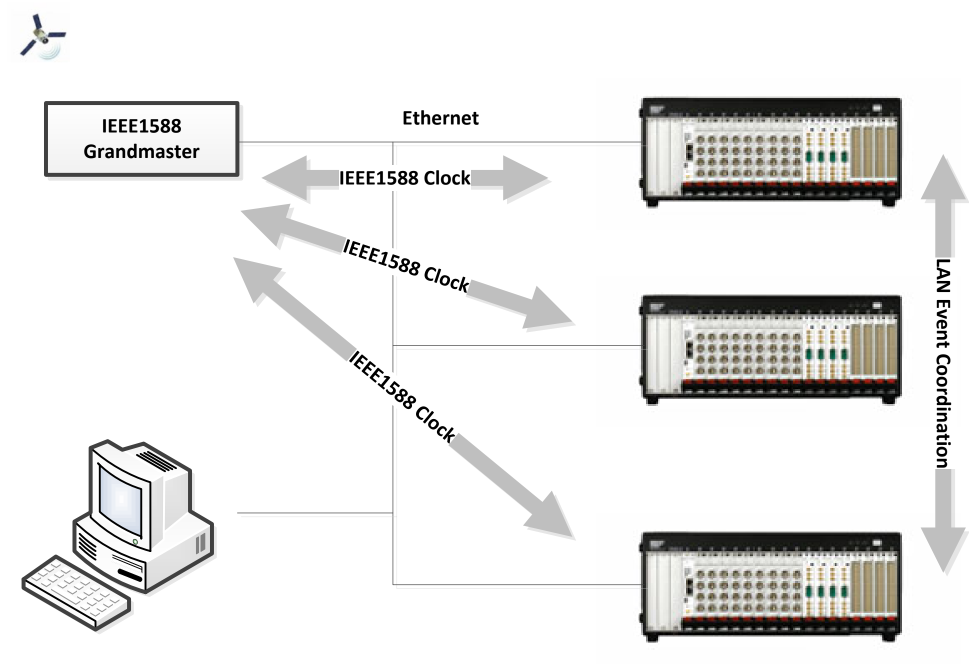


FIGURE 3-6 LAN SYNCHRONIZATION

The key advantage of this method is that the user can achieve almost equivalent synchronization performance between chassis without running extra trigger cables or clock cables. Just as trigger line synchronization, the entire system can be triggered by an external trigger signal connected to one of instruments, by sending trigger command, by sending trigger LAN event, or by analog signal at one of ADC channels crossing trigger threshold level.

When chassis are separated by a very long distance, or the LAN communication is slow, certain restrictions can apply. For example, when one of analog channels detects a trigger, it sends a LAN message to all chassis so that they trigger at the same time. A problem can occur if the other channel in the other chassis also detects a trigger condition before it receives a trigger LAN message from the first chassis that detects a previous trigger condition. In order to avoid this confusion, the user may need to apply restrictions so that only certain channels can detect trigger events.

SIGNAL GENERATION

The EMX-1434 employs several different “modes” of operation when generating waveforms. The top-level modes can be identified:

• *OutputMode*

• *OperationMode*

OutputMode determines the type of waveform(s) that will be generated. *OutputMode* and its settings are further described in Waveform *OutputModes* later in this section. Three settings may be selected for this mode:

- **StandardWaveform**: This mode generates standard waveforms (Functions), such as dc, sine, square, triangle, ramp up, and ramp down. The default for this mode is sine.
- **ArbitraryWaveform**: This mode generates a single arbitrary waveform.

- **ArbitrarySequence**: This mode is a superset of the Arbitrary Waveform mode. Rather than generating a single arbitrary waveform, multiple arbitrary waveforms can be combined into a *sequence*. A sequence consists of a list of arbitrary waveforms and repeat counts.
- **ArbitraryStreaming:** This mode generates arbitrary waveform continuously. For this mode to work, the arbitrary waveform samples must be continuously supplied to the instrument while the signal is generated from a digital-to-analog converter.
- **COLA:** This is a special output mode that generates sine waveform with a constant level that has the same frequency and phase as adjacent DAC channel. This output mode is valid only when the adjacent channel is configured to output standard sine waveform.

OperationMode determines how the waveforms are initiated and repeated. This mode can be set to three different values:

- **Continuous**: When set to continuous, the signal is generated continuously. This is the simplest setting and is the default for this mode. This setting is valid for all *OutputModes*.
- **Burst**: When set to Burst, a waveform/sequence is generated **n** times when a trigger is received, where **n** is equal to the value of the *TriggerCount* property.
- **Sequenced**: This mode is only available for the *ArbitrarySequence OutputMode*. When set to *Sequenced*, a trigger is used to advance through each item in the sequence. The *TriggerCount* parameter does not apply when this setting is used.

WAVEFORM OUTPUT MODE

Standard Waveform (Function)

The EMX-1434 can generate several standard waveforms: dc, sine, ramp up, ramp down, triangle, and square. These waveforms are configured using the following parameters: frequency, duty cycle, initial phase, dc offset, and amplitude. DC offset and amplitude can be changed during standard waveform generation. The new values can take effect immediately at the end of the current clock sample, or later when instructed. This is further discussed in *UpdateMode* property section.

There are two *StandardWaveform* types that require special configuration parameters. They are *Chirp* and *Random*.

Standard Waveform (Function) Chirp

Chirp is a swept sine function in which the frequency increases or decreases with time. With *Chirp* waveform, *MinFrequency* and *MaxFrequency* determines the frequency range of sweep, and *Time* specifies the total duration of sweep.

***Linear* and *Log* Chirp Type**

Linear chirp increases (or decreases) the instantaneous frequency, $f(t)$ at constant rate, while *Log* chirp varies it exponentially.

$f(t) = f_0 + kt$ … *Linear* chirp

$f(t) = f_0 k^t$ … *Log* chirp[2]

[2] The Log chirp is currently not supported by EMX-1434.

Where, $f(t)$ is an instantaneous frequency at time t, f_0 is the initial frequency and k is a constant.

***Single* and *Roundtrip* Chirp Mode**

The *Single* chirp mode changes frequency in one direction specified by *Direction* property, and *Roundtrip* chirp mode sweeps in both directions.

When *Operation* mode is *Continuous*, the chirp starts at the first trigger after *Initiate* and repeats the sweep indefinitely until the measurement is aborted. In *Burst* operation mode, chirp sweeps once at every trigger event. Parameters such as *BurstDutyCycle*, or *RecordSize* do not apply to *Chirp* waveform.

StandardWaveform (Function) Random

EMX-1434 generates 3 types of random signals with different distribution characteristics.

White Normal

White normal random has flat power spectral density along the frequency axis, while normal (Gaussian) amplitude distribution. With white normal random noise, the user can optionally specify the frequency range of signal by *Span* and *CenterFrequency*. This prevents exciting DUT (Device Under Test) by out of band signal. The band limited noise feature is not available in other random types.

White Uniform

White uniform random also has a constant PSD (Power Spectral Density), but the amplitude distribution is also flat. EMX-1434 produces white uniform random always at the full frequency range regardless of the frequency *Span* or *CenterFrequency* settings.

Pink Noise

Similar to white uniform random, *Pink* noise is also full frequency span noise. Unlike white random, pink noise has logarithmic frequency distribution, so that it provides constant energy in each octave frequency bands.

Random Seed

When generating any type of random noise, the user can also specify a random *Seed*. By selecting a different *Seed* value, it is possible to generate uncorrelated random noise from each DAC channel. This is important when they are used as stimulus signal for multiple inputs – multiple outputs system test.

Arbitrary Waveform Generator (AWG)

In either of the AWG modes (ArbitraryWaveform or ArbitrarySequence), channels can be set to output complex waveforms by loading one or more waveforms and linking them together. As the channels are independently isolated, the mode setting of one channel has no effect on other channels. The EMX-1434 supports a large number of waveforms and sequences, as seen below:

• Up to 100 waveforms can be saved on the EMX-1434.

• Up to 100 distinct waveforms can be linked together to form a single sequence.

• Up to 100 independent sequences can be created.

The waveforms can be utilized by multiple sequences, and sequences can be used by multiple channels. Thus, any channel can run any sequence, independent of other channels. A waveform can be used only once, looped up to 65,536 times (in a repeat), or looped continuously. At the end of a burst, the last value in the sequence is maintained as the output for that channel.
Waveform generation among channels can also be configured freely by the user. Channels can operate independently of each other, or may be synchronized using the internal clock, an external

clock provided through the front panel, or by using triggers, such as the CMX series mainframe Trigger Bus lines, software or LAN events.

Arbitrary Waveform and Arbitrary Sequence

The EMX-1434 implements deep on-board memory to store large waveforms. Arbitrary waveforms can be created and stored in memory. These waveforms can then be combined sequentially by "linking and looping." Using this method, complex waveform sequences can be generated in a predetermined order. Using the example in Figure 3-7, the following section will describe how this is accomplished.

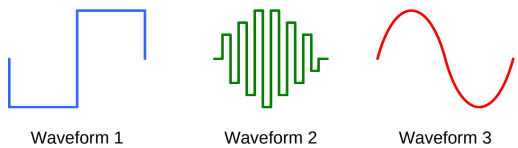


FIGURE 3-7: ARBITRARY WAVEFORMS

Linking and Looping

"Linking and looping" is the process by which waveforms and sequences are combined and repeated. The EMX-1434 provides for this functionality in two modes:

- Arbitrary waveform mode
- Arbitrary sequence mode

In ArbitraryWaveform mode, a single waveform is generated, but can be "looped", or repeated, up to 65,536 times using the *Burst OperationMode* or looped indefinitely by using the *Continuous OperationMode* setting.

In ArbitrarySequence mode, the user can define multiple waveforms and combine, or "link", them in any order. Each entry in the sequence must contain two parameters: waveform name and loop count. See Figure 3-8.

Waveform	Loop Count
2	2x
1	1x
3	2x
1	3x

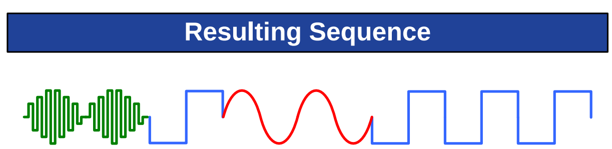


FIGURE 3-8: DEFINING A SEQUENCE WITH NAME AND LOOP COUNT

Shared Waveform and Instruction Memory

EMX-1434 uses the same physical memory for both waveform data and sequencing instructions for all channels. In most function generators, the memory for AWGs is small and separated from that of the waveform data memory, limiting the maximum number of waveforms that can be sequenced by the AWG and its overall flexibility. The EMX-1434's 128 MB of on-board memory stores data and sequencing instructions together, vastly expanding the space available for sequencing instructions. Shared memory provides the user the flexibility to use long sequences with small waveforms, short sequences with large waveforms, or any other combination.

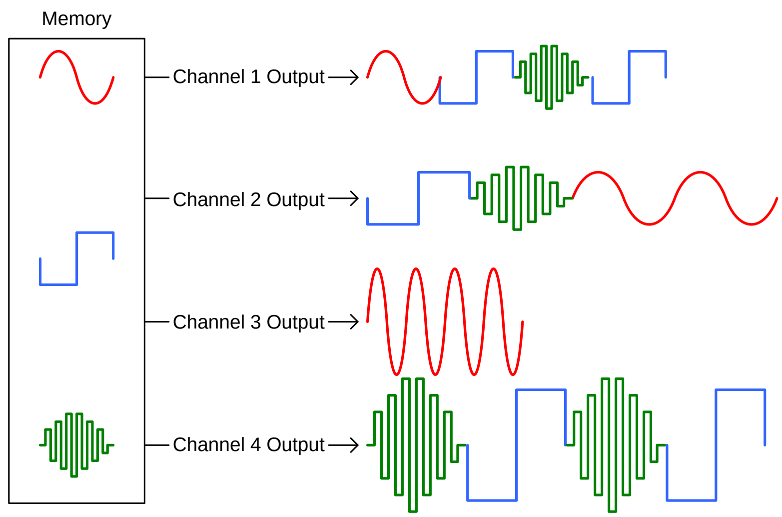


FIGURE 3-9: WAVEFORMS SHARED ACROSS CHANNELS

Note in Figure 3-9 that, although the waveforms are specifically defined in memory, the user can alter the amplitude and frequency on a per channel basis. This is done by adjusting the sample rate, offset, and gain for the channel.

Arbitrary Streaming

In *Arbitrary* or *Sequence* output mode, EMX-1434 outputs arbitrary waveforms preloaded in the instrument's memory repeatedly. The waveform cannot be modified or new waveform cannot be loaded while the signal is generated. On the other hand, in arbitrary *Streaming* output mode, the user must keep feeding new waveform while signal is being generated. This makes it possible for the user to adjust the output waveform based on the response of the device under test. This feature is often used in closed shaker control applications. This output mode is also useful when the entire arbitrary waveform does not fit in the instruments memory. In order to output signal continuously without glitches, the user's application must keep feeding new data at the same rate as (or faster than) signal is generated.

Range, Amplitude and Offset

The magnitude of generated signal is determined by *Amplitude* (for Standard Waveform), or *Gain* (for Arbitrary Waveform). The full scale of the signal is limited by the *Range* of DAC channel. In general, minimum DAC *Range* that can accommodate the largest signal magnitude provides the best Signal-to-Noise ratio and the lowest distortion. In addition to the *Amplitude* (or *Gain*), additional DC offset can be specified by *Offset.* EMX-1434 uses a separate DC offset DAC to control this DC offset, the *Offset* can be any value within +-1.0V regardless of the *Range* settings.

The arbitrary waveform loaded to the instrument is normalized to +-1.0. The actual signal magnitude is determined as the normalized value multiplied by *Gain*. When the result of multiplication exceeds the *Range* value, the signal will be clipped.

Parameter Change on the Fly

Changing the amplitude and/or frequency of the signal could cause undesirable glitches to the output signal when it is done during the signal generation. In order to minimize this glitches, or impact to the test, EMX-1434 provides five different parameter update methods in the Standard Waveform output mode.

Immediate Update

In this mode, the parameter change takes effect immediately whenever the user sets a new parameter value. This is a default mode of operation.

Software Update

In this mode, changes in *Amplitude* or *Frequency* property will not take effect until the user sends *SendSoftwareUpdate* command. The user can change both amplitude and frequency at the same time when the change will have minimum impact to the test device.

Update at Zero Crossing

Parameter changes when the signal is close to 0V. The user can change multiple parameters and then *SendSoftwareUpdate* command. All changes take effect at the next 0V crossing of signal after the command.

Update at Trigger

This is similar to Software update mode. The difference is the parameter changes happen at the next trigger event rather than *SendSoftwareUpdate* command.

Update at Trigger Zero Crossing

Similar to *Zero Crossing* mode, but the change occurs at 0V after a trigger event, instead of *SendSoftwareUpdate* command.

Sample Rate, Span and RecordSize

The *SampleRate* specifies the interval between the arbitrary waveform samples uploaded to the instrument. Increasing, or decreasing the sample rate will change the generated signal frequency. It does not specify the actual Digital-to-Analog converter's conversion rate.

The *Span* parameter determines the signal range of the white Gaussian noise when the signal type is Random. When the *Downconversion* property is enabled, the frequency range of signal is centered around the *CenterFrequency* property. Both *Span* and *CenterFrequency* have no impact on the *Uniform* or *Pink* noise.

The *RecordSize* along with the *SampleRate* and *BurstDutyCycle* determine the burst signal time length when the *OperationMode* is *Burst*. The burst signal's time length is calculated as

Time Length = RecordSize/SampleRate * BurstDutyCycle/100

When the EMX-1434 is used with a digitizer card, such as EMX-4250 or EMX-4350, and the *SampleRate* and *RecordSize* values are the same, *BurstDutyCycle* becomes the actual signal duty cycle within each data acquisition record. It is important to understand that the *BurstDutyCycle* is not the actual signal duty cycle when you simply observe the signal generated from DAC channel.

Trigger and Trigger Delay

When the *OperationMode* is *Continuous*, the signal starts at the first trigger event after the measurement is initiated. The signal stops at the measurement *Abort*, disabling the DAC channel, or at the emergency shutdown.

In *Burst* mode, single burst or sweep (when signal type is Chirp) occurs at each trigger events. Any trigger events will be ignored until the trigger holdoff time is elapsed. The trigger event after the holdoff time will cause next burst or sweep after the current burst or sweep is complete.

In Sequenced mode, the signal generation starts at the first trigger and advances the sequence by a trigger event.

The *TriggerDelay* parameter does not apply to the DAC signal output. However, analog signal from DAC channel has constant 56 samples (of DAC conversion rate) delay with EMX-1434. When EMX-1434 is used with one of digitizer cards and wish to synchronize the start of data acquisition record with the generated signal, the Analog-to-Digitizer converter channels can be delayed by the same amount of time using the *TriggerDelay* property.

COLA

The COLA (Constant Output Level Adapter) is a special output mode that digitally simulates an analog device that was commonly used with analog tracking filters in the past. This output mode is available only when the adjacent DAC channel output mode is set to continuous sine output mode. The COLA channel outputs a constant amplitude sine wave that is the same frequency and phase as the adjacent DAC channel, but the amplitude remains constant until the measurement is aborted. The initial *Amplitude* property of this DAC channel determines the signal level. Changing the *Amplitude* during the signal generation will not take effect until the measurement is aborted and re-initiated for the next time. The Frequency or Phase property of this DAC channel will not affect the output signal.

OPERATION MODE

Operation Mode defines how the EMX-1434 behaves in response to internal and external triggers (for more information on triggers, see ***Error! Reference source not found.***). This mode has three settings to control how waveform generation begins and ends: *Continuous*, *Burst*, and *Sequenced*. The user can use both hardware (such as front panel and backplane) and software and LAN event triggers to initiate waveform generation.

Continuous

As defined by the IVI standard, the *Continuous* setting causes the entire waveform or sequence to be generated continuously. The EMX-1434 begins generation of the signal with the first waveform in the sequence. Once the waveform (or all waveform segments in a sequence) have been generated, the waveform is repeated again. This continues until generation is aborted by the user.

Figure 3-10 provides an example of how this works. A sequence comprised of three waves (Wave 1, Wave 2, and Wave 3) is set to *Continuous* mode. The output begins shortly after the trigger condition is satisfied. Here, rather than generating the sequence only once, the EMX-1434 loops the sequence until the generation session is stopped. Note that, at time t_2, the EMX-1434 loops back to the first waveform defined in the sequence and that the entire sequence repeats.

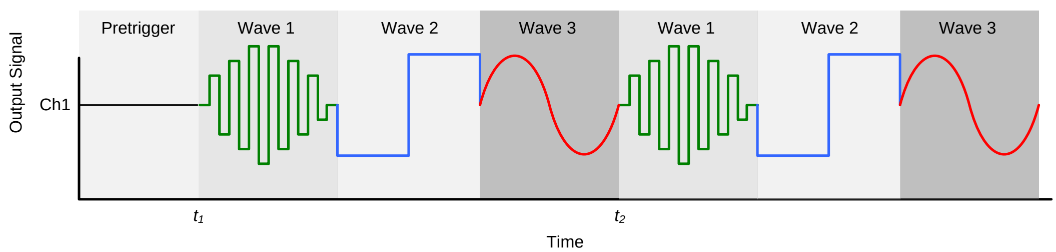


FIGURE 3-10: SEQUENCING USING THE CONTINUOUS OPERATION MODE

Burst

Like *Continuous*, *Burst* is defined by the IVI standard. In the *Burst* setting, the EMX-1434 generates one complete output (whether it's a standard/arbitrary waveform or sequence) ***n*** times (where ***n*** is equal to the *TriggerCount*), stops, and then holds the last output value until acted upon by a trigger. At the end of a burst, the EMX-1434 holds the last value of the generated output.

Sequenced

For the *Sequenced* setting, the EMX-1434 advances through each "sequence item" of the arbitrary sequence upon receipt of a trigger. As described in the ***Error! Reference source not found.*** discussion, a "sequence item" is defined by a waveform handle and a repeat count. If the repeat count of a sequence item is set to "0", that item will be repeated indefinitely until a trigger is received. If a trigger occurs after the waveform has been generated ***n*** times, then the next step in the sequence begins. Figure 3-11 shows a simple example where two waveforms are used in a sequence once and the trigger is received after the first waveform is completely generated. Note that, when Wave 1 and Wave 2 generation is completed, the last sample value of the wave continues to be driven until the next trigger is received.

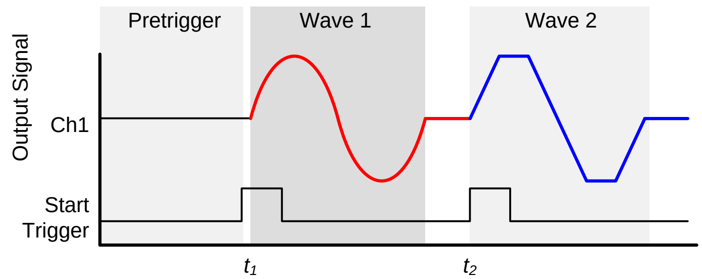


FIGURE 3-11: SEQUENCED MODE WITH TRIGGER SENT AFTER WAVEFORM COMPLETION

If, however, a trigger occurs during waveform generation, the behavior of the instrument depends on the SYNC/ASYNC setting for that channel. Advancement from one waveform to the next can be either SYNChronous or ASYNChronous When set to SYNC, the current waveform completes generation before advancing to the next waveform as shown in Figure 3-12.

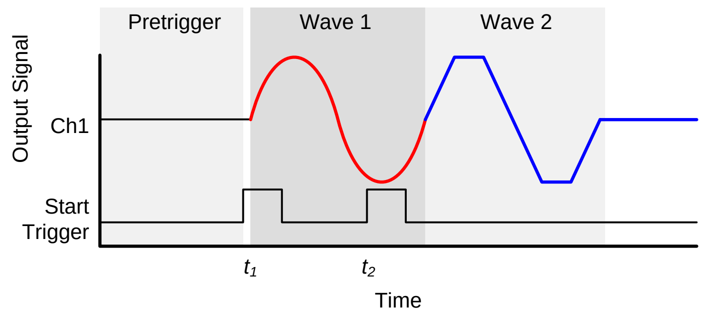


FIGURE 3-12: SYNCHRONOUS SEQUENCE MODE ADVANCEMENT

If the channel is set to ASYNC, generation of the current waveform is terminated and the next waveform in the sequence begins generation immediately. This is illustrated in Figure 3-13.

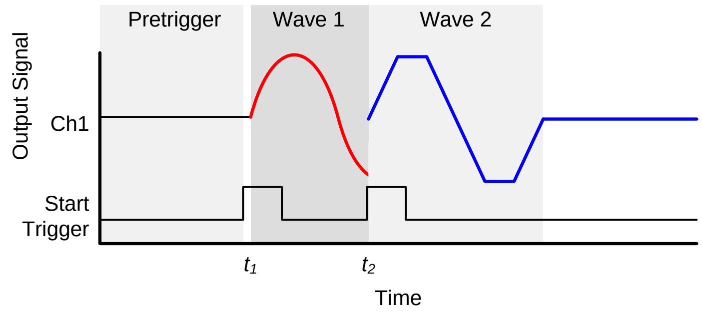


FIGURE 3-13: ASYNCHRONOUS SEQUENCE MODE ADVANCEMENT

As seen in preceding examples, the output starts on the rising edge of the start trigger. Note, however, that there is a slight delay between when the trigger occurs and when the channel 1 output begins. This delay is a function of the sampling rate and interpolation factors and is described in the specifications. Also, note that in this particular example, a sequence of two waveforms has been configured for generation. Once the last waveform in the sequence is generated, the EMX-1434 continues to drive the last sample value and will continue to do so until acted on by another trigger.

EMERGENCY SHUTDOWN

Shorting the EMX-1434's front panel Digital IO pins 5 and 6 causes the all DAC channels output on the card to ramp down and shut off. For this shutdown, the signal down ramp rate is determined by *ShutdownRampRate* property, rather than normal *RampRate*. The signal ramps down in about 5msec when in *Fast* ramp down mode and 10msec in *Slow* ramp down mode. Once EMX-1434 is shutdown, EMX-1434 DAC channel will not output signal until the shutdown signal is removed and *ClearShutdown* command is sent.

NOTE: *ClearShutdown* command must be sent after the signal ramp down is complete.

TACHOMETER INPUTS

TRIGGERING FROM TACHOMETER INPUTS

The tachometer signal at the tachometer channel can be used to trigger a measurement. When a tachometer channel is specified as a trigger source, EMX-1434 generates a trigger event when a pulse edge is detected at the channel. Unlike analog input channels on a digitizer card, the tachometer channels do not have low-pass filter in the signal path. This makes it more accurate to detect the pulse edges.

TACHOMETER PARAMETERS

Level, Hysteresis, Slope

The tachometer channel detects a tachometer pulse edge when a signal crosses the trigger threshold level in the direction specified by *Slope.*

MinFrequency, MaxFrequency

The *MinFrequency* specifies the lowest possible frequency of the tachometer pulses in Hz. The inverse of this value is the maximum time length between tachometer pulses. When no tachometer pulse is detected within this time window, EMX-1434 assumes that no signal is connected or the rotating device is halted.

The inverse of *MaxFrequency* essentially specifies the trigger hold-off time. Any signal crossings detected by a triggering circuit during this time will be considered as false edges caused by noise and ignored.

PulsesPerRevolution

This parameter is used to convert the intervals between tachometer pulses into rotating speed. For example, if the rotating speed of a device is constant[3], it can be easily calculated as

$$\text{RPM} = \frac{60}{\Delta t * PPR}$$

Where
RPM: Number of revolutions per minute.
$\Delta \boldsymbol{t}$: Tachometer pulse interval in seconds.
PPR: Number of pulses per revolution.

PHASE MEASUREMENT

One of the important applications of tachometer inputs is a phase measurement of a rotating device. Synchronously measuring a marker signal on a shaft by one of tachometer channels and vibration signal by digitizer’s analog input channels give the phase relation between them. For more information, see “Using with Digitizer” section and Appendix A, “Phase Measurement and correction”

RPM COMPUTATION AND DIGITAL RATIO SYNTHESIS

When EMX-1434 tachometer channels detect a new pulse edge, it computes a new RPM value. In order to detect a more accurate RPM while the rotating device changes its speed, EMX-1434 dynamically recalculates changes of the shaft speed during the last pulse interval.

Digital Ratio Synthesis is the feature that EMX-1434 mathematically increases or decreases tachometer pulse rate by interpolating the measured pulses. This is useful when the DUT cannot produce the desired tachometer pulse rate due to physical constraints of the tachometer device.

TACHOMETER TIME STAMPS

The user can read time stamp values of every tachometer pulse edges using FIFO.Read() method of the channel. The time stamp values are available any time after the measurement is initiated until it is aborted regardless of the measurement state. Start, Arm, or Trigger events have no effect on the tachometer timestamp measurement.

DATA ACQUISITION

[3] EMX-1434 does not assume constant speed in calculating RPM.

Using DSA driver, EMX-4250/4350/4380 digitizer card can be used for the data acquisition while EMX-1434 synchronously generates stimulus signal from DAC channels. EMX-1434 can be also used in rotating machinery applications using its tachometer channels. The tachometer channels can be used to arm or trigger the EMX-4250/4350/4380's data acquisition, based on the rotating speed or shaft position. See also EMX-4250/4350/4380 User's Manual for more detail.

MEASUREMENT PROCESS

MEASUREMENT SETUP

The measurement with DSA driver controls both the signal generation from EMX-1434 DAC channels and the data acquisition from EMX-4250/4350/4380 analog input channels.

After the instruments are initialized by driver *Initialize* method, or the previous measurement is finished, instruments are in *Idle* state. While in the idle state, user prepares for the next data measurement by configuring setup parameters.

Start, Arm and Trigger

Properties in Start, Arm and Trigger interface configure the measurement gating condition and timing. Start determines when the measurement starts after the instruments has been configured and ready for the next measurement. By the time of Start, all hardware configurations must be completed and filters are settled so that the measurement can be armed or triggered and signal generation starts without delay.

Sampling Parameters

The sampling parameters are specified by properties in *Sampling* interface. The *Sampling* in *Measurement* interface configures parameters that are common to entire system. Optionally, the user can configure different sampling parameters for different DAC channels from *DAC.Channels* interface with some restrictions.

MEASUREMENT INITIATION

After the user completes all the configuration for the measurement condition and DAC channel and analog frontend using driver API, the user can initiate the measurement process by calling *Initiate* method in *Measurement* interface. During the measurement initiation, the instrument starts to prepare for signal generation (EMX-1434) or the data acquisition (EMX-4250/4350/4380). If the instrument has not been synchronized yet, the SYNC signal will be sent and DAC and filters will be reset. When the reference oscillator has been changed, the DAC sampling clock must be re-locked with PLL. The filter settling also happens during this period. Once filters are settled and the instrument becomes ready, the measurement moves to *Wait for Start* state. The source of the *Start* event is *Immediate* by default. In this case the measurement will automatically start, otherwise, it must be started by an event specified by *Source* property of *Start* interface.

MEASUREMENT LOOP

Once the measurement is started, the state machine cycles through arming and triggering for the number of times specified by *ArmCount* and *TriggerCount.* The measurement stops when the specified number of arm and trigger loops are completed, or aborted by an *Abort* command.

ARMING

The arm is a gating condition to trigger the measurement. In order to trigger the measurement, the measurement must be armed first. There are several ways to arm the measurement. The default condition is to arm automatically, or *Immediate* arming.

Self-Arming

When the *Immediate* arm source is enabled, EMX-1434 arms by itself.

Arming by User's Command

EMX-1434 arms by *SendSoftwareArm* method when a *Software* arm source is enabled.

Arming at a Certain Time Interval

EMX-2500 Ethernet controller can be configured to assert one of PXI0-7 trigger line at a certain time interval using Platform driver's Alarm interface. EMX-1434 can be armed with these events.

Arming from Backplane Trigger Line by Other Instruments

EMX-1434 can receive an arm event from the other instruments in the same PXI chassis by configuring one of PXI trigger lines, PXI0-7 as an arm source.

Arming from LAN Events by Other Instruments

The user, or the other LXI device can send LAN event to arm EMX-1434 using DSA driver. Alternatively, the LAN event can be sent to EMX-2500 Ethernet controller using Platform driver. Upon receiving a LAN arm event, EMX-2500 can assert one of PXI0-7 trigger line to arm EMX-1434, or any other devices within a same chassis. The latter method is more generic, but more restricted.

Arming by Tachometer RPM

The EMX-1434 can generate arm events at an predefined rotating speed interval. This feature is useful to acquire data at certain speed interval while a rotating machine is running up, coasting down or running at steady speed with some variation.

EMX-1434 starts computing RPM immediately after the measurement is initiated by *Initiate()* method. If you initiate measurement while the DUT is still not stable, EMX-1434 may compute an erroneous RPM and start arming unexpectedly. On the other hand, if the measurement is initiated after the DUT became stable, the digitizer cannot start acquiring data until filters are settled and enough tachometer pulses are obtained. The user may miss important data when the DUT spins up or down in short period. In order to avoid this problem, the user can configure the driver to Software start mode by specifying *Start.Source* to "Software", then initiate measurement long time before the DUT becomes ready to spin up without causing unexpected arming. The measurement can be armed immediately after the user sends *Start.SendSoftwareStart()* command.

Arming by DIO Pattern

The EMX-1434 can be used to arm a measurement when a certain digital value is detected at DIO channel.

TRIGGERING

Once the measurement is armed, the EMX-1434 becomes ready to receive an trigger event to start signal generation from a DAC channel. When a tachometer channel is enabled, EMX-1434 computes the RPM value at every trigger events.

Self-Triggering

When the *Immediate* trigger source is enabled, EMX-1434 triggers by itself and the DAC channel generates signal or advances arbitrary sequence as soon as it is ready to receive a new trigger event.

Triggering by User's Command

EMX-1434 triggers by *SendSoftwareTrigger* method when a Software trigger source is enabled.

Triggering by Analog Signal

When EMX-1434 is used with other digitizer cards, EMX-4250/4350/4380, it can be triggered when analog signal at one of digitizer card's input channels crosses the trigger threshold level. In order to trigger from the analog signal, the channel must be enabled as a trigger source.

Triggering by External Pulse.

EMX-1434 can receive a trigger from front panel trigger SMB connector by enabling EXT trigger source.

Triggering from Backplane Trigger Line by Other Instruments

EMX-1434 can receive a trigger from the other instruments in the same PXI chassis by configuring one of PXI trigger lines, PXI0-7 as a trigger source.

Triggering from LAN Events by Other Instruments

The user, or the other LXI device can send LAN event to trigger EMX-1434 using DSA driver. Alternatively, the LAN event can be sent to EMX-2500 Ethernet controller using Platform driver. Upon receiving a LAN trigger event, EMX-2500 can assert one of PXI0-7 trigger line to trigger EMX-1434, or any other devices within a same chassis. The latter method is more generic, but more restricted.

Triggering by Stimulus Signal

When one of EMX-1434's DAC channels is specified as a trigger source, EMX-1434 generates a trigger event whenever the trigger source channel generates a new signal. When digitizer cards, EMX-4250/4350/4380s are included in the same DSA driver session, they can be triggered by this event for the data acquisition.

When the DAC *OperationMode* is *Continuous*, the EMX-1434 generates one trigger at the start of signal generation.

When in *Burst* mode, a trigger is generated at the start of every burst.

For the *Sequenced* setting, the EMX-1434 generates a trigger when it advances to the next "sequence item" of the arbitrary sequence. As described in the ***Error! Reference source not found.*** discussion, a "sequence item" is defined by a waveform handle and a repeat count. If the repeat count of a sequence item is set to "0", that item will be repeated indefinitely.

DATA RETRIEVAL

EMX-1434 tachometer channels measure tachometer pulse arrival timestamps. This information becomes available once the measurement is initiated and when any tachometer pulse is detected at one of enabled tachometer channels. This information is kept in the instruments FIFO buffer and then sent to the host computer for the user's application using tachometer channel's *FIFO.Read()* method. Unlike digitizer data acquisition, the tachometer arrival time is available during the measurement regardless of its state. Arming or Trigger events have no effect to this data.

Tachometer channels computes RPM and is available anytime when the tachometer channel is enabled. The information is available from RPM property, and the RPM values at the trigger event can be obtained from Additional Data field of the *Measurement.Read()* method when the EMX-1434 is used with digitizer cards, EMX-4250/4350/4380.

When EMX-1434's DIO channels are enabled and configured as input, the digital value can be read from DIO channel's *State* property. This information is available anytime.

SIGNAL GENERATION

The *Continuous* mode DAC channel starts signal output at the first trigger event and continues until the measurement is aborted or the channel is disabled. In *Burst* mode, a burst signal output at every trigger event. The *Sequenced* mode DAC channel starts the first sequence at the first trigger and advances to the next sequence at every trigger events.

EVENTS

The *Events* is an optional feature that EMX-1434 sends notification to user, or other instruments when a certain event occurred. The notification can be by a LAN message or a signal to one of the PXI trigger lines. The user can specify an event when the measurement state is armed, triggered, or finished. See DSA driver's online help for more information.

USING WITH DIGITIZER EMX-4250/4350/4380

This section describes how the EMX-1434 can be used together with EMX-4250/4350/4380 digitizer instruments. See EMX-4250/4350/4380 User's Manual for more detailed information of these instruments.

Currently, EMX-1434 is configured and controlled by DSA driver. When both EMX-1434 and EMX-4250/4350/4380 are included in a single DSA driver session, they work synchronously. When the user wishes to control EMX-1434 independently from the data acquisition, a separate driver instance needs to be created.

STIMULUS SIGNAL GENERATION

EMX-1434 has four DAC output channels. For stimulus-response measurement, EMX-1434 can be used as a stimulus signal generator. Since EMX-1434 can generate four independent signals at the same time, it can be used for MIMO (Multiple-Input-Multiple-Output) system analysis. Using a single DSA driver session, the data acquisition and stimulus signal generation can be synchronized together by the common trigger, sampling clock, and a state machine.

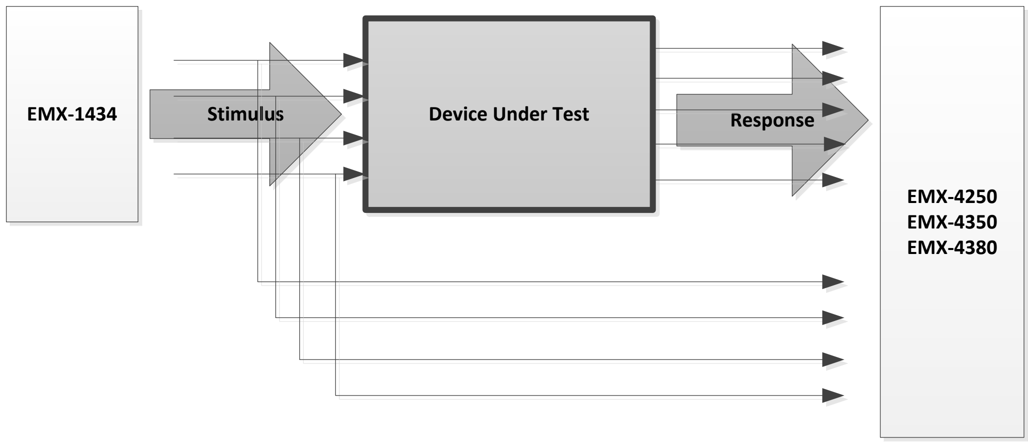


FIGURE: 3-14 STIMULUS AND RESPONSE

TACHOMETER INPUTS

EMX-1434 has two tachometer input channels. The main application for tachometer inputs is for a rotating machinery testing. EMX-1434 can measure instantaneous rotating speed in RPM from tachometer pulses connected to one of tachometer channels. The RPM values can be associated to the EMX-4250/4350/4380 data records returned at the trigger events.

Some applications require acquired measurement data associated to a certain phase reference signal. This signal can also be a single pulse per revolution tachometer signal. For this application, the EMX-1434's tachometer channel can be used as a trigger source of the measurement.

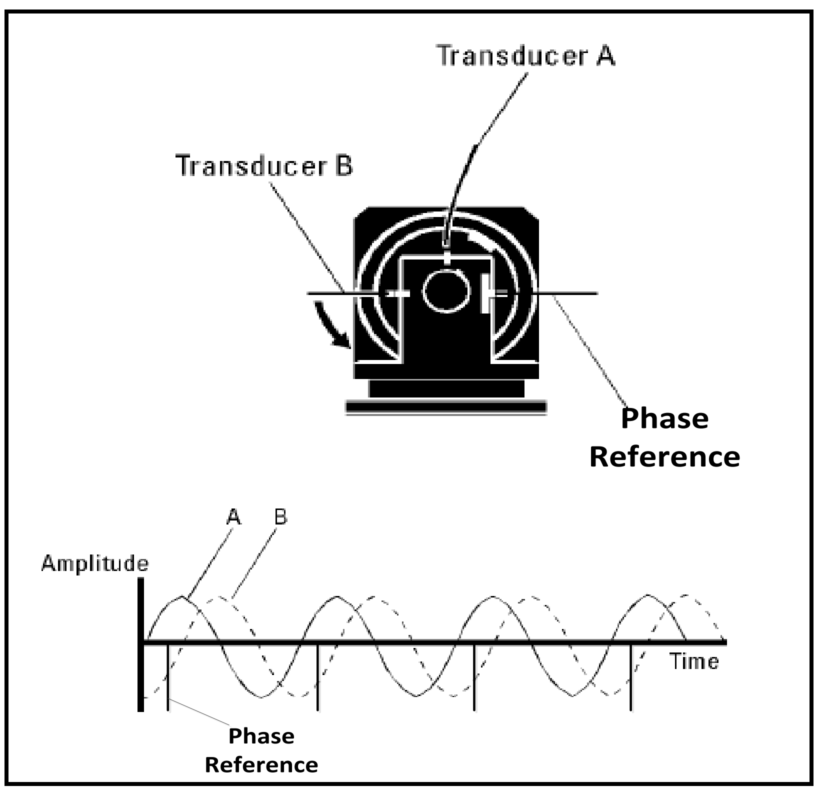


FIGURE: 3-15 TACHOMETERS

EMX-1434 can be configured to continuously return time stamps when the tachometer signal is detected. The array of time stamps can be used to resample [4] the data from EMX-4250/4350/4380's for synchronous measurement.

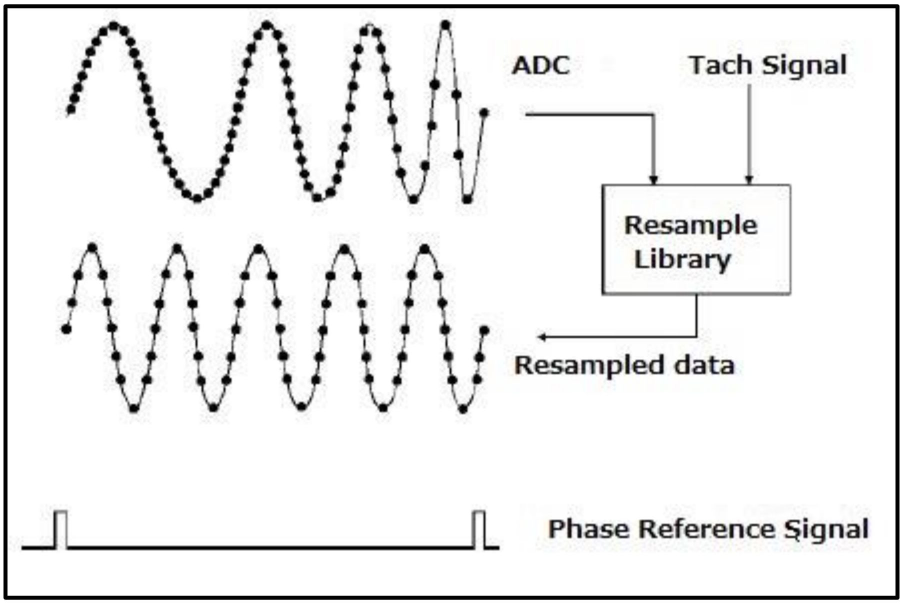


FIGURE 3-16: RESAMPLING ADC DATA BY TACHOMETER PULSES

EMX-1434 tachometer channels can be used to arm measurement based on the rotating speed (RPM). This feature can be used in an application that requires data acquisition at a certain RPM interval.

DIGITAL INPUT AND OUTPUT

EMX-1434 has 4 bit DIO channels. The digital signal can be associated to the EMX-4250/4350/4380 measurement data record. Optionally DIO value can be used as an arming condition of the measurement.

WHERE TO FIND MORE INFORMATION

There is more information available using EMX-1434 card.

DRIVER API REFERENCE

The complete driver's API reference is available as online help. Each driver comes with .chm format help file.

Each driver comes with several useful example programs in C++ and C#.

OTHER MANUALS

Separate User's Manuals are available for EMX-2500, CMX09, CMX18 and EMX-4250/4350/4380.

[4] VTI provides a separate library to resample data acquired by EMX-4250/4350/4380 and other VTI digitizer.

SPECIFICATION INFORMATION

EMX-1434 conforms many industry standards in both hardware and software architecture. Although, products can be used without knowing these standards, some knowledge can be useful to take full advantage of our products.

LXI specification is available from LXI consortium at www.lxistandard.org

PXI specification is available from PXI System Alliance at www.pxisa.org

IVI driver specification is available from IVI Foundation at www.ivifoundation.org

IEEE 1451 Smart Transducer Interface Standard and IEEE 1588 Precision Synchronization Protocol Standard at www.nist.gov/el/isd/ieee

ANSI/VITA 49.0 (VRT) is specified as a part of VITA specification at www.vita.com

SECTION 4

MODULE INFORMATION

OVERVIEW

The Smart Dynamic Signal Analyzers family of products incorporates best-in-class analog design methodology to deliver industry leading measurement accuracy. These instruments are ideal for a wide range of applications including noise, vibration, and harshness (NVH); machine condition monitoring; rotational analysis; acoustic test; modal test; as well as general purpose high speed digitization and signal analysis.

THE FRONT PANEL

Connections

The coaxial connectors are all of type SMB jack with male pin.

CH1, CH2, CH3, and CH4 are analog output connectors.

TRIG is trigger input

SUM is the summing input.

The TACH-DIO connector is a female high-density 15 pin D-Sub.

This also provides connections for the External Shutdown input and the Marker output, as well as numerous grounds.

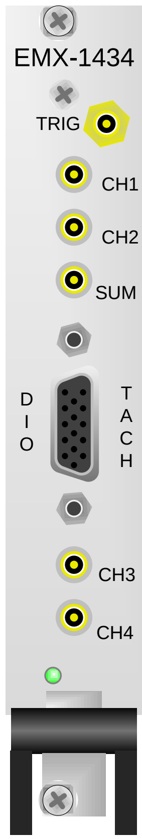


FIGURE 4-1: FRONT VIEW

LED

The LED is a bicolor LED

Green OK and power is applied

Yellow Warning
Signal over range
Loss of PLL lock
Data FIFO overflow/underflow
No tachometer pulse detected if tachometer is enabled
Output is enabled

Red Emergency Shutdown

CIRCUITRY

Analog Output and Summing Input

There are four analog output connectors, one per channel. Each output is the sum of the Signal DAC, an Offset DAC, and an optional summing input.

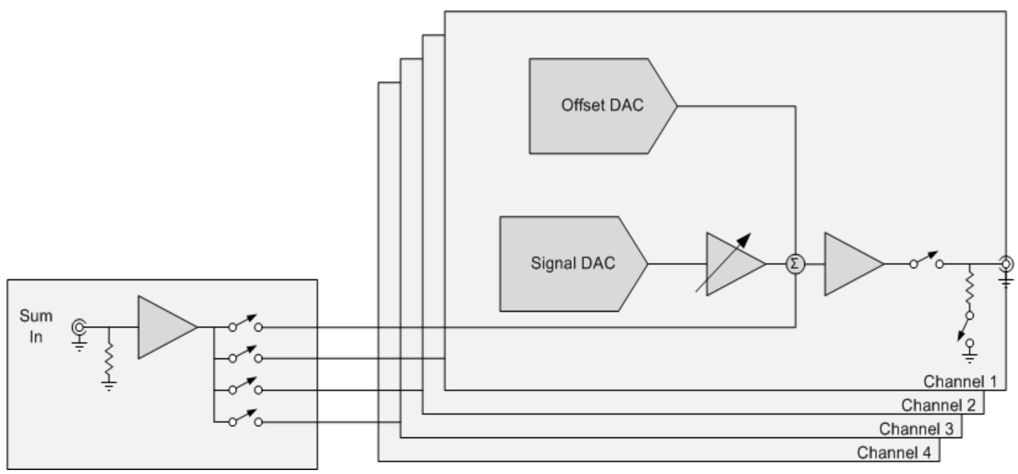


FIGURE 4-2: ANALOG OUTPUT

The Signal DAC generates the various waveforms. The amplitude at the output connector is a function of the amplitude sent to the Signal DAC and the gain setting of the programmable gain amplifier (PGA).

Although not shown in this simplified block diagram, there is a six-pole anti-imaging filter to ensure no out-of-band aliasing is affecting the output signal.

When the output is disabled a relay contact between the amplifier and the output connector opens and the relay contact to ground, with a 100 Ω resistor in series, closes. This ensures that the load does not float, picking up ambient noise.

For low signal levels the PGA should be operated at higher gain; this improves the signal to noise ratio and reduces THD.

The Offset DAC is used in shaker table applications to re-center the table under heavy load applications and generates ±1 V maximum.

The Sum In allows the user to connect an external signal source, or a different EMX-1434 output channel's signal and add it to one or more of the Signal DAC's outputs. This can be useful when complex signals are desired that may be, for example, the sum of two sine waves. An alternative way to add waveforms is to calculate them and generate them in a single channel using the arbitrary mode.

Note that the output protection circuitry trips at a nominal ±10.5 V, and it is possible that the Offset DAC + the Signal DAC + the Summing signal could exceed this voltage. It is the responsibility of the user to ensure that this does not occur.

The response from the Sum Input to the Channel Output is:

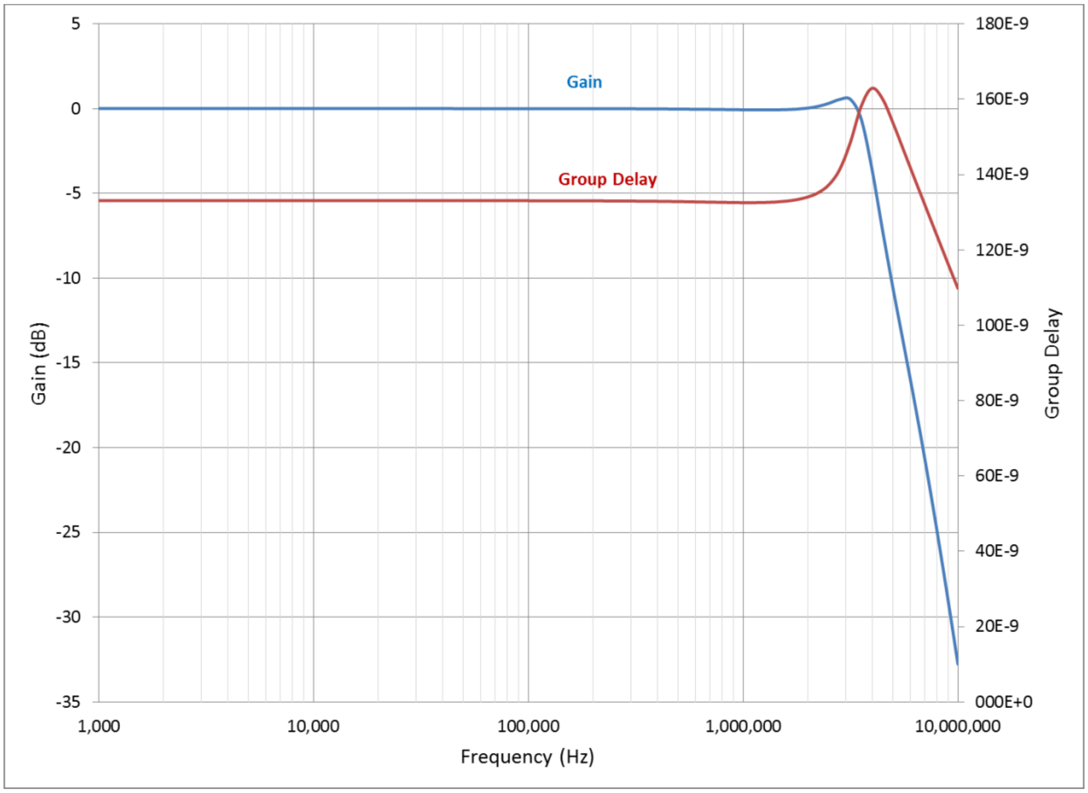


FIGURE 4-3: ANALOG OUTPUT GAIN AND GROUP DELAY

Analog Output and Capacitive Loading

The analog output is quite immune to capacitive loading; there is no peaking or roll-off even with capacitances up to 10 nF. However, when capacitance loading approaches 30-40 nF slew-rate limiting affects the wave shape as shown in Figure 4-4.

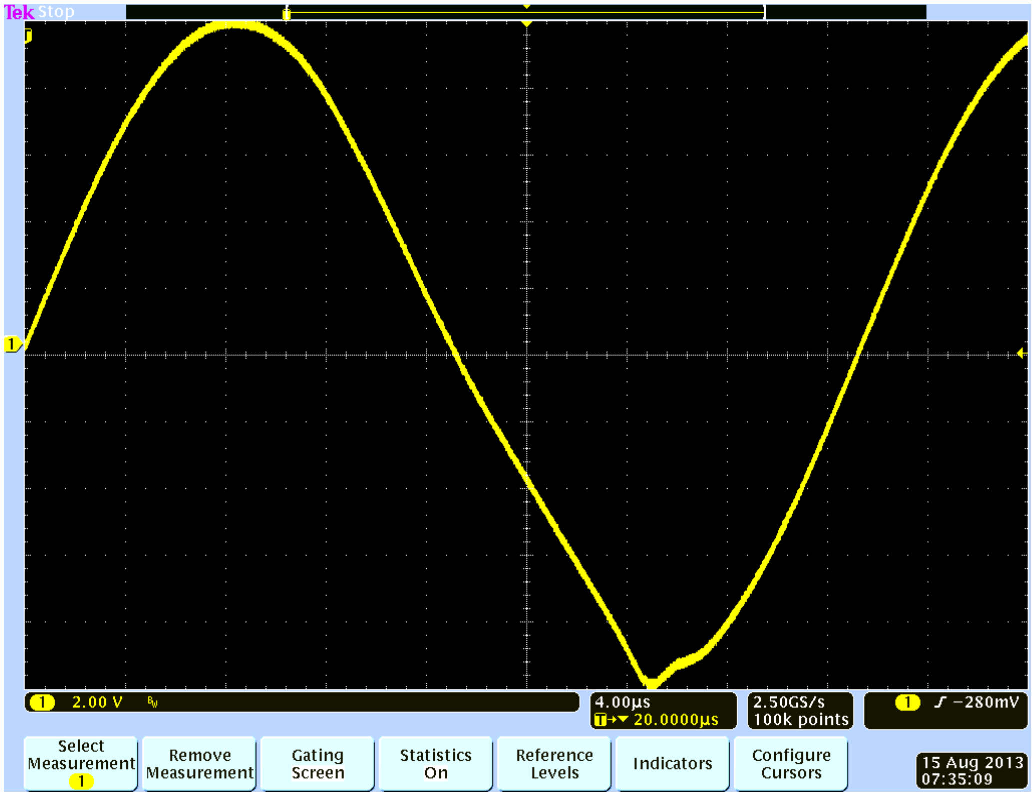


FIGURE 4-4: SLEW RATE LIMITING

This is an output set to 30 kHz, 20 V peak-to-peak sine wave with a 40 nF capacitive load. No slew rate limiting was observed at 10 nF. Slew rate limiting was beginning to be observable at 25 nF.

When the capacitance and rate of change of output voltage are high enough, the output amplifiers begin to current limit, causing the slew rate limiting seen above.

The current limit of the opamps is 60 mA minimum, so, for example, to avoid slew rate limiting the maximum capacitive load for a sine wave would be:

$$C_{max} = \frac{60\ \text{mA}}{V_{peak} \cdot 2 \cdot \pi \cdot freq}$$

The typical source of capacitive loading is cable capacitance. Typical coaxial cable has about 108 pF/m (33 pF/ft), so with a 10 V peak sine wave at our maximum frequency of 80 kHz, the maximum cable length would be 110.5 m (362 ft), which would limit the capacitive load to 12 nF. The calculation for other than sine waves is more complex but the concept is the same:

$$C_{max} = 60\ \text{mA} \cdot \frac{dt}{dV}$$

Analog Output Protection

The output is protected in two ways:

1. Back-to-back 12 volt zener diode clamps.

2. A window comparator monitors the analog output voltage. If it exceeds ±10.5 V a relay disconnects the amplifier output from the SMB connector.

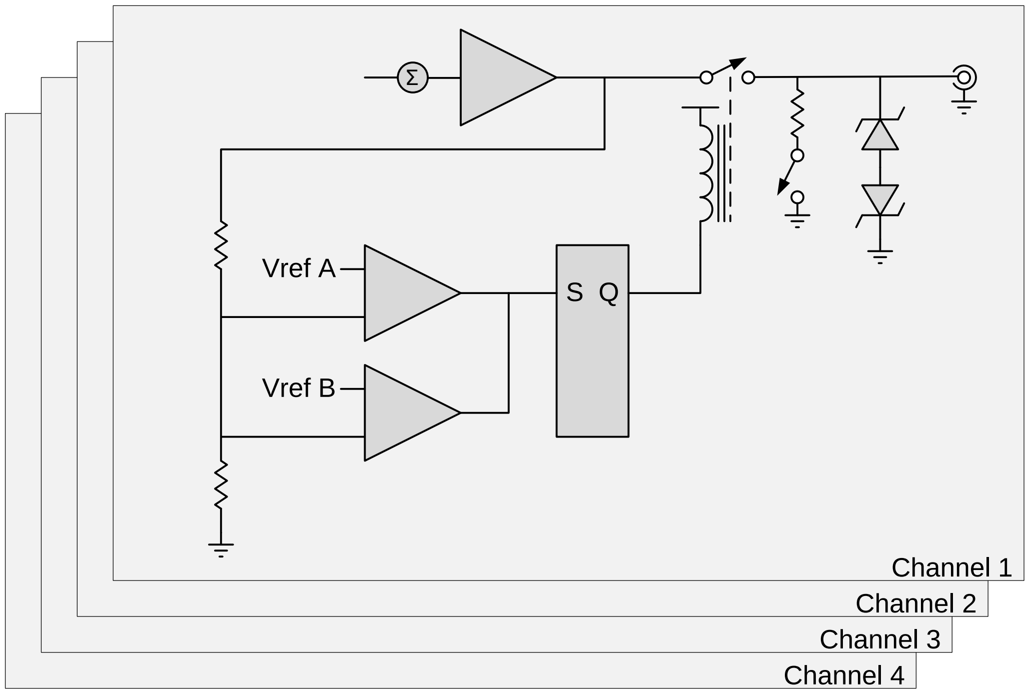


FIGURE 4-5: ANALOG OUTPUT PROTECTION

Tachometer Input

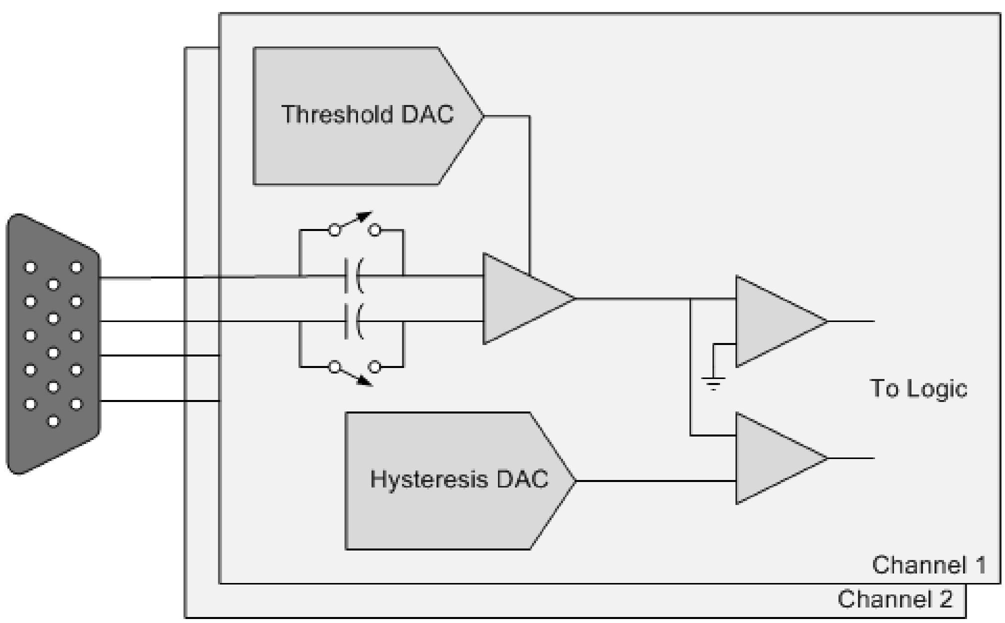


FIGURE 4-6: TACH BLOCK DIAGRAM

There are two tachometer input channels. Each channel has:

- Differential inputs.
- Two user selectable voltage ranges, 0 to ±25 V peak and 0 to ±250 V peak.
- User selectable AC or DC coupling. When AC coupling is enabled, the amplifier has a -3 dB frequency of 1.2 Hz.
- Two comparator modes; level only and level with programmable hysteresis.

In level only mode the Threshold DAC sets the trigger level and an analog comparitor detects the crossing of the input signal at that level. The comparitor has a small amount of electrical hysteresis to ensure minimal noise response; this hysteresis is not adjustable.

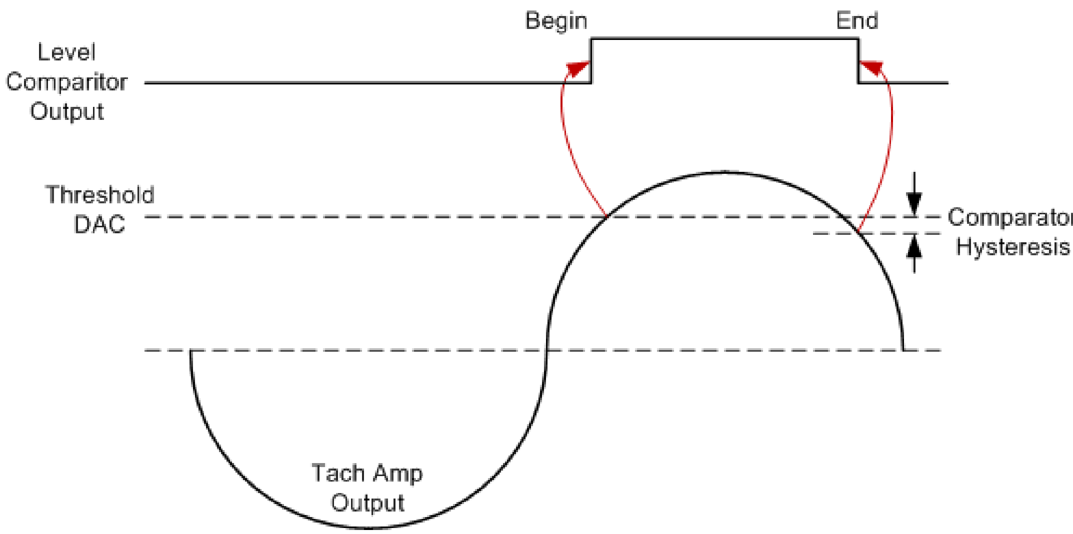


FIGURE 4-7: LEVEL ONLY MODE TRIGGERING

In level with programmable hysteresis mode, the Threshold DAC sets the trigger level and an analog comparator detects the crossing of the input signal at that level; this is considered the start of the tach pulse. A second DAC, the Hysteresis DAC sets the threshold of a second comparator; when the input exceeds that level, in the other direction, the other edge of the tach pulse is detected. This programmability provides the user the ability to reject higher levels of noise.

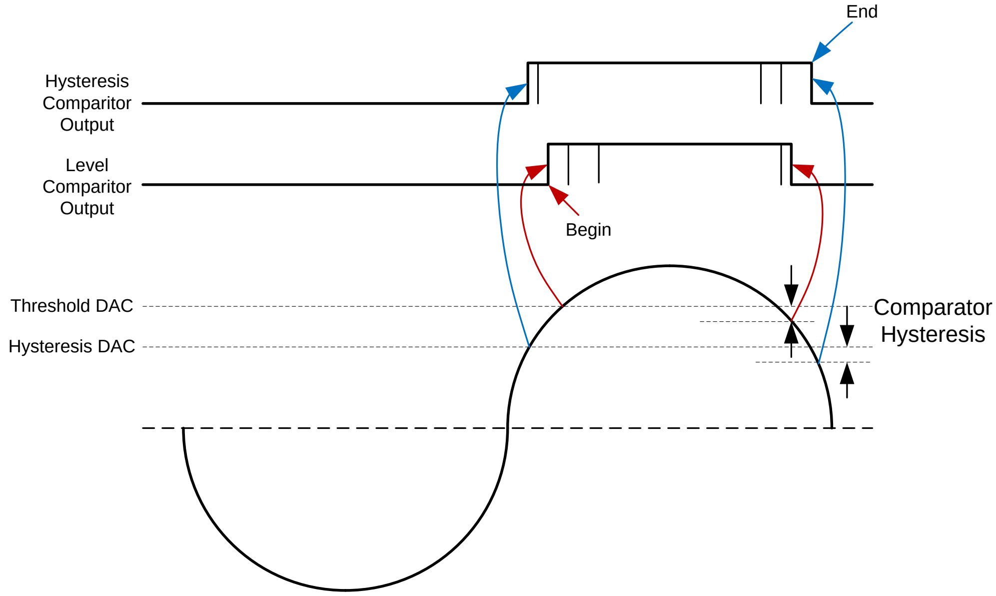


FIGURE 4-8: PROGRAMMABLE HYSTERESIS MODE TRIGGERING

Frequency response and group delay for the tachometer amplifier.

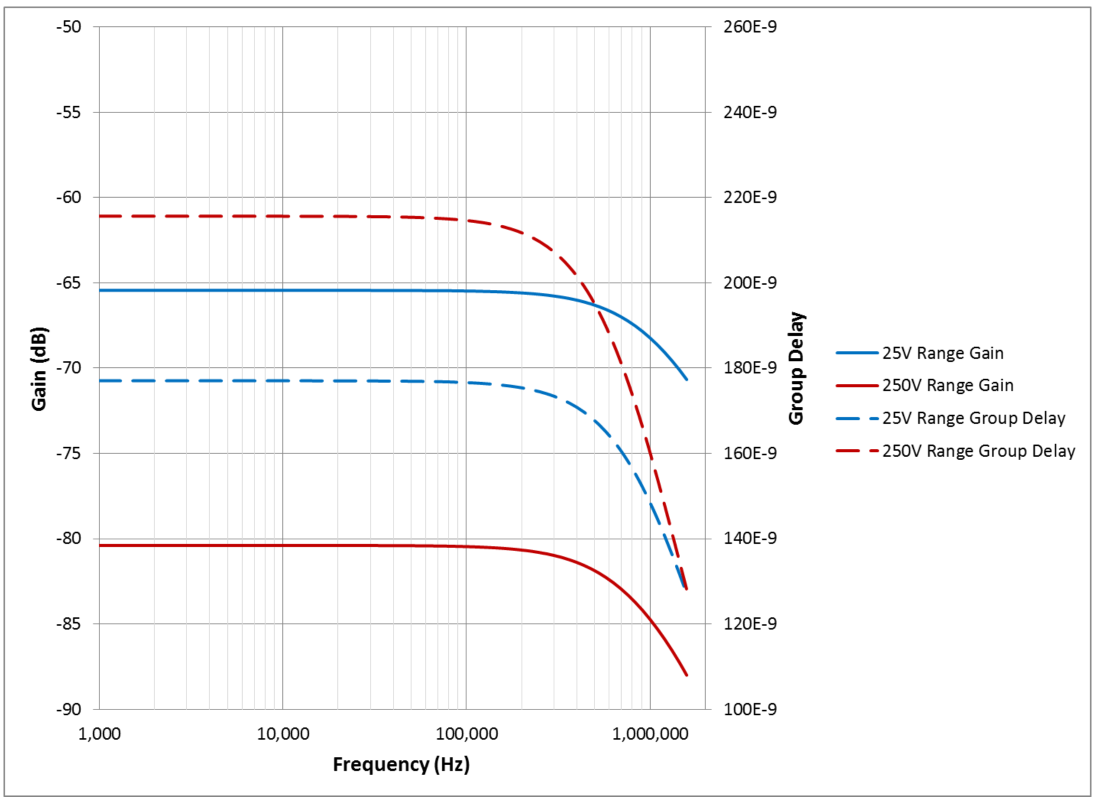


FIGURE 4-9: TACH FREQUENCY RESPONSE AND GROUP DELAY

External Shutdown Input

Shorting pin 5 (input) and ground of the TACH DIO connector causes the source to ramp down and shut off. There is a weak pull-up resistor (38 kΩ) from the input to +3.3 V so an external switch must have contacts that are able to reliably switch a light load.

When a shutdown is detected, the unit holds the analog output at whatever level it was when the shutdown occurred, and then ramps the signal toward zero volts. Two rates of Shutdown are available, normal shutdown which has a 2 ms time constant, and when Slow Shutdown is enabled a 1 sec time constant.

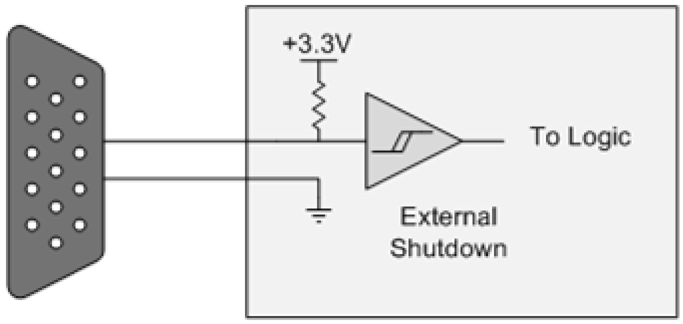


FIGURE 4-10: EXTERNAL SHUTDOWN

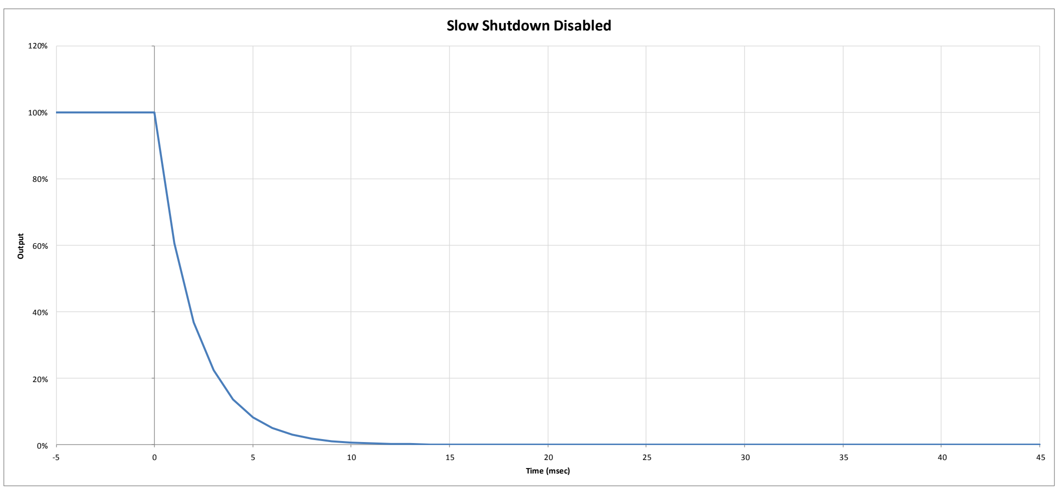


FIGURE 4-11: EXTERNAL SHUTDOWN TIME –
SLOW SHUTDOWN DISABLED

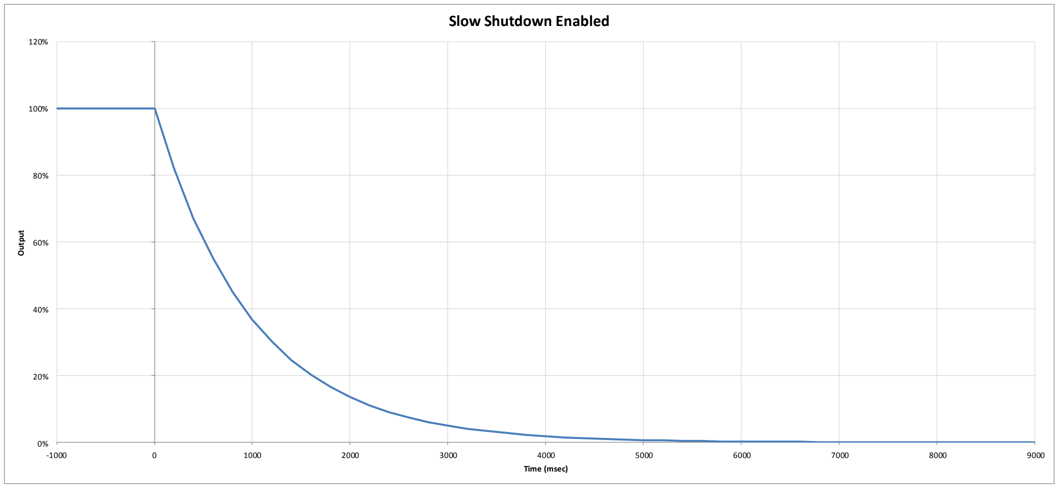


FIGURE 4-12: EXTERNAL SHUTDOWN TIME - SLOW SHUTDOWN ENABLED

Marker Output[5]

The marker is a logic level output on pin 3 of the TACH DIO connector with respect to ground on pin 5. This signal is programmable:

- It may be associated with any output channel.
- Its location within the waveform may be programmed.
- Its slope may be programmed.
- Its width may be programmed.

Digital Input / Output

There are four channels of digital input/output. Each may be independently programmed as either an input or an output.

- A digital input may be used to arm the card.
- Digital outputs may be changed programmatically.
- Digital inputs may be read programmatically.

Trigger Input

Provide a logic level pulse to the trigger input to start the trigger process. A 10 kΩ pull down resistor ensures no false triggering.

A trigger can initiate a sine sweep, for example, or allow hand shaking among cards and other equipment

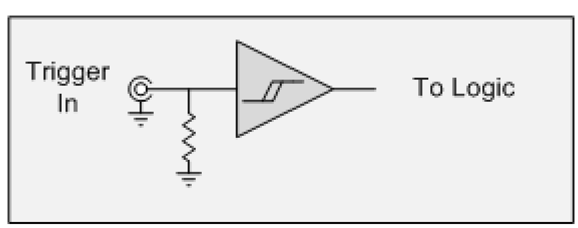


FIGURE 4-13: TRIGGER INPUT

[5] This feature is currently not supported.

APPENDIX A

PHASE MEASUREMENT AND CORRECTION

OVERVIEW

The Fourier transform of the time domain signal acquired by EMX-4250/4350/4380's analog channel gives the amplitude and phase of each frequency components. The amplitude indicates the magnitude of the phenomena and the phase represents the time shift. When the instrument acquires a signal, the analog signal goes through multiple stages of analog and digital filters before the data reaches the user's application. Each filter adds a certain time delay to the signal.

When you are trying to measure the time difference between two signals of the same frequency, if both signals go through the same analog and digital filters, then the amount of time shift can be considered to be equal, since digital filters can achieve the perfect phase match. In this case, the phase difference between two signal does not require any special correction.

On the other hand, if one of signals does not go through these filters, the phase of the other signals must be corrected for the delay added in each filtering stage. This is the case when the user triggers the measurement with a phase reference signal, such as TDC (Top Dead Center) pulse using EMX-1434 tachometer channels, or Trig input SMB at front panel. The trigger signal is directly detected at the trigger detection circuit for the trigger inputs, while all of the analog signals go into ADC channels and they are filtered to the desired frequency span. Thus, delay is introduced. Fortunately, our digital filters are all linear phase FIR filters that add a constant time shift independent of the signal frequency, so they can be corrected mathematically. The timestamps associated to the data samples are already corrected for the delays introduced by analog and digital anti-aliasing filters. However, there are other phase corrections that the users have to consider, dependent upon the test setup and the accuracy requirement.

SUB-SAMPLE TRIGGER DELAY

When the measurement is triggered by a trigger signal sent directly to the front panel trigger connector, or backplane trigger line, the trigger event occurs asynchronous to the ADC sampling clock. This trigger event time is measured by the timestamp clock and recorded, and returned to the user. This information can be obtained by parsing *AdditionalData* string returns at the *Measurement.Read* (or *MemoryRead* for streaming) method. The *Read* method also returns the timestamp of the first sample in the data record. The difference between the trigger timestamp and the data timestamp can be used to correct for more accurate phase measurement.

When one of tachometer channels is used for the trigger detection (phase reference), the trigger event time can be obtained from two different methods. The first and more common method is from an *AdditionalData* string as described above. Alternatively, the tachometer edge time can be also obtained from tachometer channel's *FIFO.Read()* method. There can be a small difference (100-150 nano-second) between two methods. In general, the tachometer channel's *FIFO.Read()* method gives more accurate time stamps and should be used for this purpose.

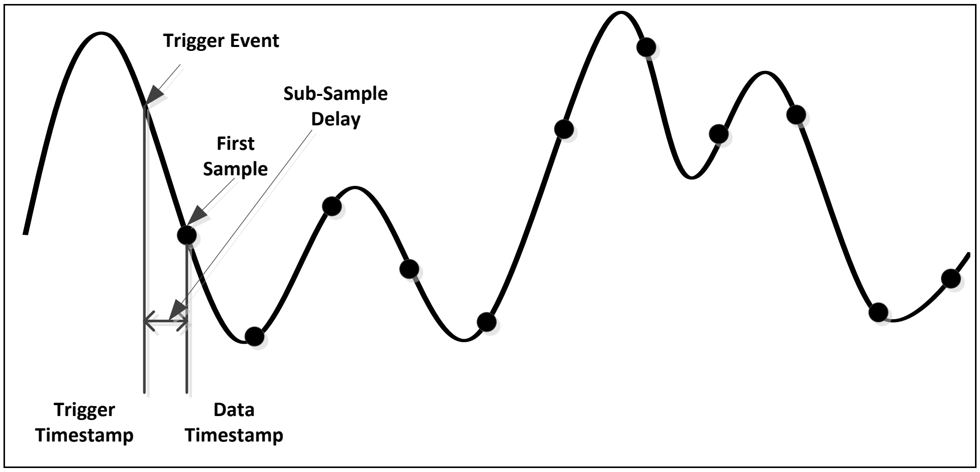


FIGURE A-1: SUB-SAMPLE TRIGGER DELAY

AC COUPLING FILTER

When the analog signal is measured in AC coupling mode, the analog AC coupling, high-pass filter adds non-linear phase. This delay can be significant, and the amount of delay is frequency dependent. See Module Information section for the AC coupling filter phase performance.

TRANSDUCER PHASE DELAY

Additional delay can be introduced by the transducer being used, dependent upon the transducer architecture. This may also need to be considered dependent upon the measurement accuracy requirement. This is beyond the scope of this manual. Contact transducer manufacturer for more information.

APPENDIX B

RPM MEASUREMENT AND ARMING

The tachometer pulses are generated at equal angular intervals around a rotating shaft. Since the angular velocity can constantly change and tachometer pulses have finite angular resolution, the arming RPM usually occurs between two tachometer pulse edges. In order to detect the time when the shaft speed reaches to a certain RPM, interpolation must be performed. When you observe this speed change for a very short period of time, it is reasonable to assume that the rate of speed change (or acceleration) is constant during this short period. Based on this assumption, we can construct a quadratic formula between shaft angle and time from three consecutive tachometer pulse edges. Once the formula is computed, we'll be able to calculate the time when the shaft speed reaches to the arm RPM.

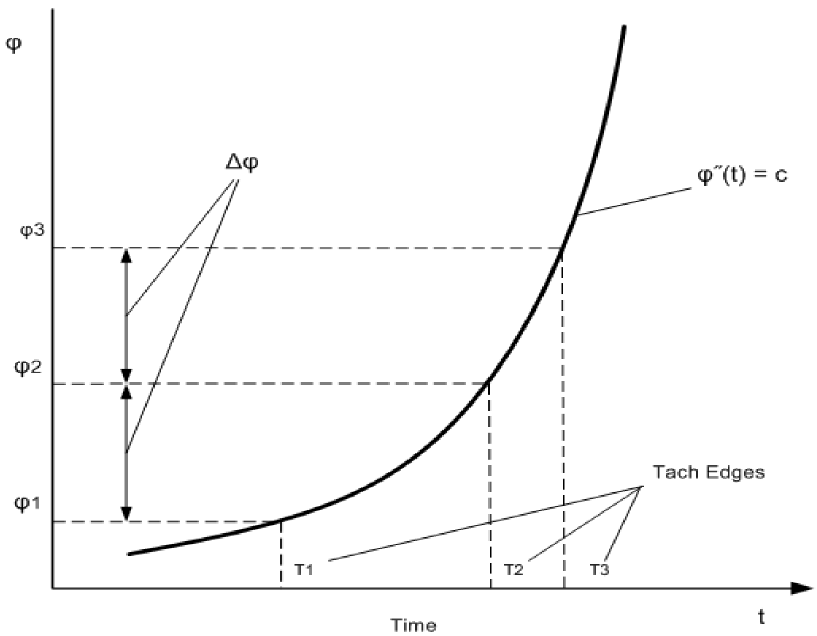


FIGURE B-1:

The following equation describes how quadratic equation can be obtained from three consecutive tach pulse edges, t1, t2 and t3.

$\frac{\partial^2 \theta}{\partial t^2} = a$ …. (1) constant acceleration.

$\frac{\partial \theta}{\partial t} = at + b$ …. (2) rotating speed

$\theta = \frac{1}{2}at^2 + bt + c$ ….(3) shaft angle

You can obtain coefficients from three consecutive tachometer pulses $\boldsymbol{t_1}, \boldsymbol{t_2}, \boldsymbol{t_3}$

$$\boldsymbol{\theta_1} = \frac{1}{2}\boldsymbol{a}\boldsymbol{t_1}^2 + \boldsymbol{b}\boldsymbol{t_1} + \boldsymbol{c}$$

$$\boldsymbol{\theta_1} + \Delta\boldsymbol{\theta} = \frac{1}{2}\boldsymbol{a}\boldsymbol{t_2}^2 + \boldsymbol{b}\boldsymbol{t_2} + \boldsymbol{c}$$

$$\boldsymbol{\theta_1} + 2\Delta\boldsymbol{\theta} = \frac{1}{2}\boldsymbol{a}\boldsymbol{t_3}^2 + \boldsymbol{b}\boldsymbol{t_3} + \boldsymbol{c}$$

Where,
$\Delta\boldsymbol{\theta}$: Angle interval between tach pulses
$\Delta\boldsymbol{t_n}$: The time between $\boldsymbol{t_n}$ *and* $\boldsymbol{t_{n-1}}$
Since initial shaft position and time $\boldsymbol{\theta_1}$ and $\boldsymbol{t_1}$ is arbitrary, there is no need to derive $\boldsymbol{c}$

Once the quadratic equation is obtained, you can calculate the exact time, when the shaft speed reaches target speed, as far as the time falls between $\boldsymbol{t_1}$ and $\boldsymbol{t_3}$.

When the next tachometer pulse edge is detected, the above equation must be re-computed from the last three consecutive tachometer pulse edges.

The equation also allows you to compute the time from any given shaft angle. Using this equation, the user can synthesize tachometer pulses (timestamps) at any shaft angle interval, not limited to the original interval. Using this formula, we'll be able to implement digital ratio synthesizer.

While this method can estimate machine speed more accurately than traditional method which simply computes RPM from interval between two successive tachometer pulses, it can also return unrealistic number when the tachometer signal is very poor or device under test is unstable. In order to avoid arming measurement while device is still unstable, the user can hold measurement in *Start* state, and use *SendSoftwareStart* method to indicate that the machine is ready to start taking data. Unlike *Initiate* method, the measurement can be armed immediately without waiting for filters to be settled.

APPENDIX C

CALIBRATION AND SELF TEST

CALIBRATION AND ADJUSTMENTS

Two types of calibration are performed, factory calibration and self-calibration. The two are almost identical. Both use a precision 24-bit analog to digital converter (ADC) and a precision and stable voltage reference. These create a stable and accurate DC voltmeter that measures the source output at each PGA gain, correcting for offset voltage shifts and gain changes. This meter also ensures that tachometer DAC settings are accurate.

Internal DC Voltmeter

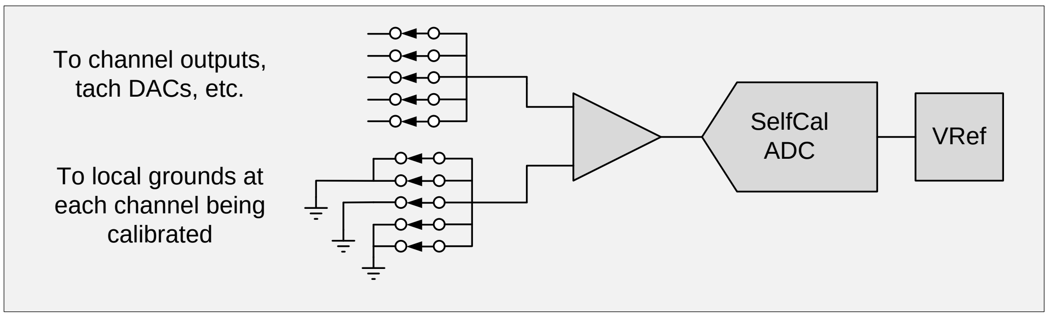


FIGURE C-2: SELF-CALIBRATION

A differential amplifier is used at the input to shift the ±12.5V full scale input to the 0 to 2.5V full scale input that the ADC expects. This differential amplifier also remotely senses ground at the point the measurement is being made to compensate for various voltage drops in the grounds throughout the circuit board.

Most of the voltmeter's uncertainty is due to the voltage reference, and the one used in this product has a long term drift of 0.018 % /year (0.0015dB/year) and 0.0004 % /°C, so measurement and calibration uncertainty is extremely low.

Factory Calibration

The factory calibration serves two purposes: 1) it provides a calibration that is traceable to international standards, and 2) it corrects for instrument errors due to finite component tolerances. Because of these finite component tolerances, the corrections can be fairly large, greater than ±2 %.

It is recommended that the EMX-1434 be returned annually to VTI Instruments for a factory (traceable) calibration to ensure that the instrument maintains its accuracy and precision.

Self-Calibration

Self-calibration is used to make small corrections to the factory calibration. These corrections may be necessary due to thermal changes and aging. Self-calibration will perform adjustments to ensure the highest instrument accuracy.

Changes in temperature affect gains and offsets of all electronic instrumentation. If the unit is operated at a temperature which is different from the temperature at which factory calibration or the last self-calibration had been performed, then a new self-calibration should be performed. To ensure the EMX-1434 meets the specified accuracy, users experiencing temperature differences as small as 5 °C should consider performing a self-calibration.

Gains and offsets change as instruments age as well. Mechanical stresses are induced when the components are made and installed on the board. Over time these stresses are relieved and the component values change slightly. From time to time these changes must be corrected for by re-calibration.

Since the factory calibration already compensated for component tolerances, self-calibration only corrects for small changes due to time and temperature. If a self-calibration detects a large change it is assumed to be due to a failing component and the user will be notified that their unit should be returned to VTI Instruments for service.

When a self-calibration is performed, the state of the instrument is stored; the signal outputs are temporarily disconnected from the output amplifiers and the output connectors are tied to ground through 100 Ω resistors. When self-calibration is complete the instrument is restored to its previous state.

SELF-TEST

Self-test exercises as many features of the instrument as is reasonably possible and makes various measurements to ensure, with a high level of confidence, that the instrument is functional.

- Entire signal path from signal DAC to protection relay is functional
- Entire signal path from offset DAC to protection relay is functional
- PGA gains are correct
- Tachometer level and hysteresis DACs are functional
- Tachometer amplifier output offset
- External Shutdown circuit is functional

Things that are not covered under self-test:

- Summing input
- Digital I/O
- Tachometer amplifier gain
- Tachometer amplifier AC/DC coupling
- The ability of the analog outputs to drive to maximum current

When a self-test is performed, the state of the instrument is stored; the signal outputs are temporarily disconnected from the output amplifiers and output connectors are tied to ground through 100 Ω resistors. When self-test is complete the instrument is restored to its previous state.

APPENDIX D

RANDOM NOISE GENERATION

OVERVIEW

EMX-1434 DAC output mode provides three distribution types of random noise, white-normal, white-uniform and pink-normal. While the white-normal and white-uniform random noise gives uniform distribution across the frequency range of the signal, the pink-normal random noise has -3dB/Octave spectral shape. The pink-noise generator is mainly used in acoustic analysis. The white noise generator is commonly used as a stimulus signal for the frequency response measurement of the system. The difference between white-normal and white-uniform is the distribution in the amplitude axis. While, white-normal random noise gives near Gaussian distribution of the amplitude, white-uniform random noise has uniform distribution for the entire amplitude range.

With EMX-1434, the user can specify a random seed for each DAC channel. This makes it possible to generate exactly the same random sequence repeatedly. More importantly, by choosing different random seed between DAC channels, EMX-1434 can generate uncorrelated random noise from each DAC channel simultaneously. This is especially useful to measure a frequency response matrix of multi-inputs multi-outputs system.

The output of EMX-1434 white-normal random noise can be band limited to a certain frequency range. Sometimes, this is important to avoid exciting the device under test beyond the frequency range of interest.

WHITE GAUSSIAN NOISE

EMX-1434 white-normal (or Gaussian) noise is generated from white-uniform random noise, by transforming the distribution by Box-Muller transformation algorithm. This gives a PDF that is Gaussian to within 0.2% out to 4.8 sigma.

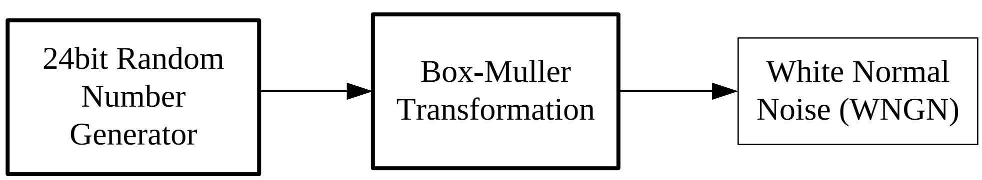


FIGURE D-1: WHITE NORMAL NOISE GENERATION

PINK NOISE

The pink noise is generated by applying a 3rd order IIR filter, with -3dB/Octave level to the white-normal random noise.

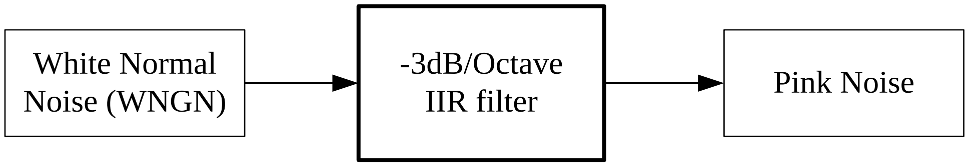


FIGURE D-2: PINK NOISE GENERATION

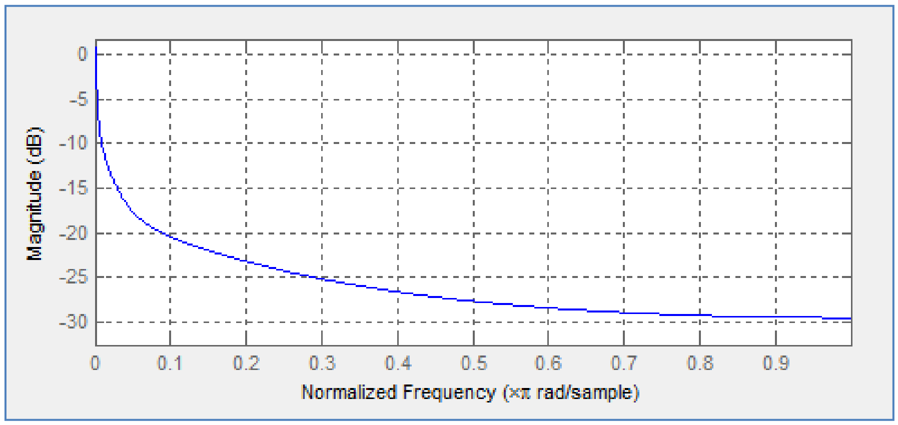


FIGURE D-3: -3DB/OCTAVE FILTER

BAND LIMITED NOISE

The band limited random noise is generated by applying multiple stages of x2 interpolation filter to the original base band white-normal random noise (WGNG) and mixing it with user specified center frequency (LO).

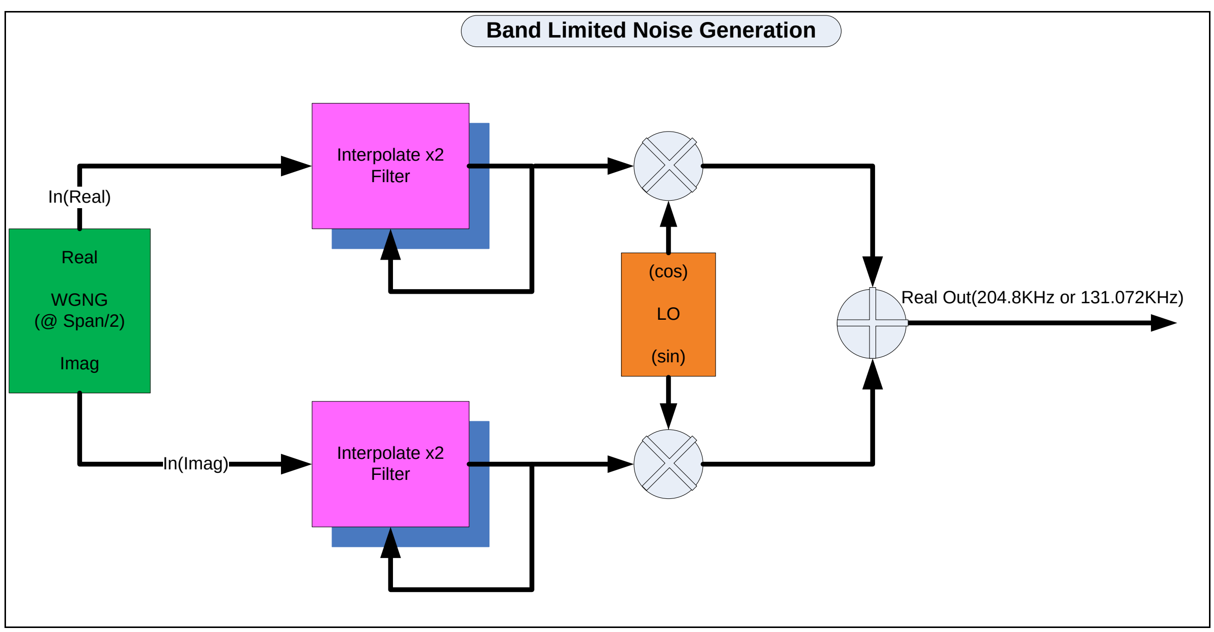


FIGURE D-4: BAND LIMITED NOISE GENERATOR

Stages	Sample Rate	BW	Sample Rate	BW
0	204800	80000	131072	51200
1	102400	40000	65536	25600
2	51200	20000	32768	12800
3	25600	10000	16384	6400
4	12800	5000	8192	3200
5	6400	2500	4096	1600
6	3200	1250	2048	800
7	1600	625	1024	400
8	800	312.5	512	200
9	400	156.25	256	100
10	200	78.125	128	50
11	100	39.0625	64	25
12	50	19.53125	32	12.5
13	25	9.765625	16	6.25
14	12.5	4.882813	8	3.125
15	6.25	2.441406	4	1.5625
16	3.125	1.220703	2	0.78125
17	1.5625	0.610352	1	0.390625
18	0.78125	0.305176	0.5	0.195313
19	0.390625	0.152588	0.25	0.097656

FIGURE D-5: BAND LIMITED NOSE BANDWIDTH

INDEX