ARTISAN®
TECHNOLOGY GROUP

Your **definitive** source for quality pre-owned equipment.

Artisan Technology Group
(217) 352-9330 | sales@artisantg.com | artisantg.com

Full-service, independent repair center
with experienced engineers and technicians on staff.

We buy your excess, underutilized, and idle equipment
along with credit for buybacks and trade-ins.

Custom engineering
so your equipment works exactly as you specify.

- Critical and expedited services
- In stock / Ready-to-ship
- Leasing / Rentals / Demos
- ITAR-certified secure asset solutions

Expert team | Trust guarantee | 100% satisfaction

All trademarks, brand names, and brands appearing herein are the property of their respective owners.

Find the *Analogic DBS9900* at our website: *Click HERE*

DBS9900

High-Performance VXI Carrier

User's Manual
Revision 01

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Proprietary Statement

The information contained in this publication is derived in part from proprietary and patent data of the Analogic Corporation. This information has been prepared for the express purpose of assisting operating and maintenance personnel in the efficient use of the instrument described herein. Publication of this information does not convey any rights to reproduce it or to use it for any purpose other than in connection with the installation, operation, and maintenance of the equipment described herein.

P/N 82-5134
Revision 01

Copyright © Analogic Corporation 2000. All rights reserved.
Printed in U.S.A.

DBS9900, DBS901 and DBS902 are trademarks of Analogic Corporation.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Warranty

Analogic warrants only to the original purchaser that this product, as purchased from Analogic or an Analogic distributor or dealer, will conform to the written specifications for a period of one year from the date of purchase. If the product fails to conform to these warranties, Analogic, as its sole and exclusive liability hereunder, will repair or replace the product and/or its components within a reasonable period of time if the product is returned to a Analogic service center within the warranty period. These warranties are made upon the express condition that:

a) The purchaser will promptly notify Analogic in writing of any non-conformity with the above warranty including a detailed explanation of the alleged deficiencies.
b) The product is returned to an Analogic service center at the buyer's expense after making suitable arrangements for performance of service.
c) When the product is returned for repair, a copy of the original bill of sale or invoice is sent with the product.
d) Analogic will not be liable for any incidental or consequential damages.
e) In the opinion of Analogic upon inspection, the product has not been misused, altered, or damaged due to abnormal handling and/or operation.
f) Repairs to the product and/or its components have not been made by anyone other than Analogic or one of its authorized repair agents.
g) The product has not been modified, altered, or changed in any manner by anyone other than Analogic or one of its authorized repair agents.

THIS WARRANTY EXCLUDES ALL OTHER WARRANTIES, WHETHER EXPRESSED OR IMPLIED, ORAL OR WRITTEN, INCLUDING WITHOUT LIMITATION WARRANTIES OF MERCHANTABILITY AND/OR FITNESS FOR A PARTICULAR PURPOSE.

No term, condition, understanding or agreement purporting to modify the terms of this warranty shall have any legal effect unless made in writing and signed by an authorized officer of Analogic and the purchaser.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Definition of Terms

Term	Definition
I/0 1, I/0 2	Also called daughterboards, and/or modules. There can be one or two modules installed in the DBS9900. I/O 1 refers to the first installed module and I/O 2 refers to the second.
DBS901	Arbitrary Waveform Generator daughterboard (purchased separately).
DBS902	Digitizer daughter board (purchased separately).
Sample Clock	Clock which drives the sample rate of each module. An *Internal* or *External* Sample Clock may be used. It is possible (using the driver) for one module to use an Internal Sample Clock and the other to be using an External Sample Clock.
Reference Clock	Timebase used by the DBS9900 as a reference for both installed modules (I/O 1 and I/O 2). This PLL 10MHz reference signal is provided internally by default but an *External* signal may be used. Can only be applied to the CLK 2 input on the Front Panel.

About this Manual

This manual is organized in a top-down manner.

This means that the first chapters provide overview and installation information.

Then, configuration and setup information is presented.

Finally, functionality is described. Each of the following sections describe similar content – but from a high-level perspective (Soft Front Panel) to a low-level perspective (hardware register descriptions).

- DBS9900 Soft Front Panel
- DBS9900 Driver (.DLL) Functions
- DBS9900 Registers

Reference material is presented in Appendices.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Safety Precautions

Warnings and Cautions

The terms **WARNING** and **CAUTION** have specific meanings in this manual:

WARNING



A WARNING advises against certain actions or situations that could result in personal injury or death.

CAUTION



A CAUTION advises against actions that could damage equipment, produce inaccurate data or invalidate a procedure.

Mesures de securite

Mises en garde

Les terms **AVERTISSEMENT** et **ATTENTION** ont des sens spécifiques dans cette notice.

AVERTISSEMENT

Un AVERTISSEMENT informe des actions ou situations qui peuvent présenter un risque de blessure ou de décès.

ATTENTION

Une mise en garde débutant par le terme ATTENTION informe des actions que peuvent endommager le metériel, produire des données incorrectes ou nuire au fonctionnement.

Sicherheitsvorkehrungen

Warnung und Vorsicht

Die Bezeichnungen **WARNUNG** und **VORSICHT** haben in diesen Sicherheitsvorschriften eine besondere Bedeutung.

WARNUNG

Eine WARNUNG rät gegen bestimmte Handlungen oder Situationen, die Verletzung der Person oder Tod zur Folge haben können.

VORSICHT

VORSICHT weist auf Handlungen hin, die das Gerät beschädigen könnten, unrichtige Daten verursachen oder einen Vorgang auslöschen können.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

SAFETY SUMMARY

The following safety precautions apply to both operating and maintenance personnel and must be observed during all phases of installation, operation, and service of the unit. Before applying power, follow the installation instructions and become familiar with the operating instructions for all components.



CAUTION

This product uses components which can be damaged by electrostatic discharge (ESD). To avoid damage, be sure to follow proper procedures for handling, storing, and transporting ESD-sensitive devices.

PRÉCAUTIONS

Ce produit utilise des composants pouvant être endommagés par décharge électrostatique (DES). Pour éviter les dégâts, suivre les procédures appropriées lors de la manipulation, du stockage, et du transport des dispositifs sensibles aux décharges électrostatiques.

VORSICHT

Dieses Produkt ist mit Komponenten ausgerüstet, die durch elektrostatische Entladung beschädigt werden können. Zur Vermeidung von Schäden muß dafür gesorgt werden, daß die maßgeblichen Vorschriften für die Behandlung, Lagerung und den Transport für elektrostatisch empfindliche Geräte beachtet werden.



WARNING

Do not operate the unit in the presence of flammable gases or fumes. Operation of the unit in any such environment constitutes a definite safety hazard.

AVERTISSEMENT

Ne pas faire fonctionner l'unité en présence de gaz ou de vapeurs inflammables. Le fonctionnement de l'unité dans un environnement de ce type constitue un danger évident pour la sécurité.

WARNUNG

Die Einheit darf nicht in Gegenwart von feuergefährlichen Gasen oder Dämpfen betrieben werden. Der Betrieb der Einheit in einer derartigen Umgebung stellt eine eindeutige Gefahrenquelle dar.



WARNING

Do not install substitute parts or perform any unauthorized modifications to this unit. Return the unit to Analogic for service and repair to ensure that the safety features are maintained.

AVERTISSEMENT - NE PAS SUBSTITUER LES PIÈCES OU MODIFIER L'UNITÉ

Ne pas installer de pièces de substitution ni effectuer des modifications non autorisées sur l'unité. Renvoyer l'unité à Analogic pour l'entretien et les réparations, de manière à assurer que les dispositifs de sécurité soient maintenus.

WARNUNG - NUR ORIGINALTEILE BENÜTZEN UND DIE EINHEIT NICHT ÄNDERN

Es sind nur Originalteile einzubauen, und ohne Erlaubnis darf keine Änderung an der Einheit vorgenommen werden. Zur Wartung und Reparatur ist die Einheit an Analogic zurückzusenden, damit die Beibehaltung der Sicherheitseigenschaften gewährleistet ist.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Contents

1 Overview 1-1

1.1 Introduction 1-1

1.2 Technical Support 1-2

2 Preparation & Installation 2-1

2.1 Overview 2-1

2.2 Host Computer Requirements 2-1

2.3 Power Consumption and Airflow Requirements 2-1

2.4 Front Panel Connectors and Indicators 2-2

2.5 Hardware Installation 2-4
- 2.5.1 Installation into a VXI Chassis 2-4
- 2.5.2 The DBS9900 Logical Address 2-4
- 2.5.3 A16, A24 and A32 Communications Mode 2-5

2.6 Software Installation 2-6
- 2.6.1 Installation Steps 2-6

2.7 Start-Up and Verification 2-6
- 2.7.1 Self-Test 2-6

2.8 DBS9900/Module Pin Connections 2-7

2.9 DBS9900 26 Pin Control Connector 2-8

3 Operation 3-1

3.1 Sample Clock and Timebase 3-2
- 3.1.1 Possible Clock Configurations 3-2
- 3.1.2 Internal Sample Clock Parameters 3-3
- 3.1.3 External Sample Clock Parameters 3-3
- 3.1.4 External Reference Parameters 3-3

3.2 Performance Specification 3-4
- 3.2.1 ANALOG I/O 3-4
- 3.2.2 INTERNAL SAMPLE CLOCK 3-4
- 3.2.3 CONTROL INPUTS 3-4
- 3.2.4 CONTROL OUTPUTS 3-5
- 3.2.5 POWER 3-6
- 3.2.6 RELIABILITY 3-6
- 3.2.7 RECOMMENDED WARM-UP TIME 3-6
- 3.2.8 ENVIRONMENTAL AND MECHANICAL 3-6
- 3.2.9 VME/VXI BUS COMPLIANCE 3-6
- 3.2.10 SOFTWARE DRIVER 3-6
- 3.2.11 EMC & SAFETY AGENCY COMPLIANCE 3-7
- 3.2.12 CALIBRATION 3-7

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

4 Using the DBS9900 Soft Front Panel 4-1

4.1 DBS9900 Main Panel 4-1
4.1.1 Main Panel Field Descriptions 4-2
Error! Bookmark not defined.

4.2 DBS9900 Sample Clock Panel 4-3
4.2.1 Sample Clock Field Descriptions 4-5

4.3 DBS9900 Trigger Panel 4-6
4.3.1 Trigger Panel Field Descriptions 4-7

5 DBS9900 Driver Software – Description of Functions 5-1

5.1 Initialize, Module and Close Functions 5-1
5.1.1 Initialize an9900_init 5-1
5.1.2 VIT Initialize an9900_VITinit 5-2
5.1.3 Set Module an9900_setModule 5-4
5.1.4 Get Module an9900_getModule 5-4
5.1.5 Identify Module an9900_identifyModule 5-5
5.1.6 Close an9900_close 5-6

5.2 Control 5-7
5.2.1 Device Type Query an9900_deviceType_Q 5-8
5.2.2 Control an9900_control 5-9
5.2.3 Status Query an9900_status_Q 5-10
5.2.4 Interrupt an9900_interrupt 5-11
5.2.5 Interrupt Query an9900_interrupt_Q 5-12

5.3 Module 5-13
5.3.1 Module Status Query an9900_moduleStatus_Q 5-14
5.3.2 Module Control an9900_moduleControl 5-15
5.3.3 Module Control Query an9900_moduleControl_Q 5-16
5.3.4 EEPROM Port Write an9900_eepromPortWrite 5-17
5.3.5 EEPROM Port Query an9900_eepromPort_Q 5-18
5.3.6 EEPROM Write Enable an9900_eepromWriteEnable 5-19

5.4 Memory 5-20
5.4.1 Memory Offset an9900_memoryOffset 5-20
5.4.2 Memory Offset Query an9900_memoryOffset_Q 5-21
5.4.3 Memory Attribute Query an9900_memoryAttribute_Q 5-22
5.4.4 Enable Memory an9900_enableMemory 5-23

5.5 Trigger 5-24
5.5.1 Set Trigger Source an9900_setTrigSource 5-24
5.5.2 Get Trigger Source an9900_getTrigSource 5-25
5.5.3 Set Trigger Output an9900_setTrigOut 5-26
5.5.4 Get Trigger Output an9900_getTrigOut 5-27
5.5.5 Set Trigger Polarity an9900_setTrigPolarity 5-28
5.5.6 Get Trigger Polarity an9900_getTrigPolarity 5-29
5.5.7 Set Trigger Threshold an9900_setTrigThreshold 5-30
5.5.8 Get Trigger Threshold an9900_getTrigThreshold 5-31
5.5.9 Trigger Source Control an9900_triggerSourceControl 5-32
5.5.10 Trigger Source Control Query an9900_triggerSourceControl_Q 5-33
5.5.11 Trigger Mode an9900_triggerMode 5-34
5.5.12 Trigger Mode Query an9900_triggerMode_Q 5-35

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.6 Clock 5-36
5.6.1 Set Clock Source — an9900_setClockSource 5-37
5.6.2 Get Clock Source — an9900_getClockSource 5-38
5.6.3 Set Internal Clock Reference — an9900_setIntClockRef 5-39
5.6.4 Get Internal Clock Reference — an9900_getIntClockRef 5-40
5.6.5 Set Clock Threshold — an9900_setClockThreshold 5-41
5.6.6 Get Clock Threshold — an9900_getClockThreshold 5-42
5.6.7 Set Sample Rate — an9900_setSampleRate 5-43
5.6.8 Get Sample Rate — an9900_getSampleRate 5-43
5.6.9 Await PLL Lock — an9900_awaitPLLLock 5-44
5.6.10 Clock Frequency — an9900_clockFreq 5-45
5.6.11 Clock Frequency Query — an9900_clockFreq_Q 5-46
5.6.12 Clock Control — an9900_clockControl 5-47
5.6.13 Clock Control Query — an9900_clockControl_Q 5-48
5.6.14 R Counter — an9900_rCounter 5-49
5.6.15 R Counter Query — an9900_rCounter_Q 5-49
5.6.16 N Counter — an9900_NCounter 5-50
5.6.17 High Speed Divider — an9900_highSpeedDivider 5-51
5.6.18 Low Speed Divider — an9900_lowSpeedDivider 5-51
5.6.19 Clock A/B Threshold — an9900_clockThreshold 5-52
5.6.20 Clock A/B Threshold Query — an9900_clockThreshold_Q 5-53
5.6.21 Trigger A/B Threshold — an9900_triggerThreshold 5-54
5.6.22 Trigger A/B Threshold Query — an9900_triggerThreshold_Q 5-55
5.6.23 Function Latch — an9900_functionLatch 5-56
5.6.24 Time Base Control — an9900_timeBaseControl 5-57

5.7 Utility 5-58
5.7.1 Get Error Text — an9900_getErrorText 5-58
5.7.2 Mask Write to Instrument — an9900_maskWriteInst 5-59
5.7.3 Write to Instrument — an9900_writeInst 5-60
5.7.4 Read from Instrument — an9900_readInst 5-61
5.7.5 Reset — an9900_reset 5-62
5.7.6 Error Message — an9900_errorMessage 5-63

DBS9900 VME / VXI Bus Interface - Registers 6-1

6.1 ID REGISTER 0x00 Read Only Logical Address Register 6-3

6.2 DEVICE TYPE REGISTER 6-3

6.3 Control Register 6-5

6.4 Status Register 6-5

6.5 Memory Offset Register 6-6

6.6 Memory Attribute Register 6-7

6.7 Module / Relay Control Register 6-7

6.8 Trigger Source Control Register 6-8

6.9 Trigger Mode Control Register 6-9

6.10 Interrupt Level Select Register 6-10

6.11 Interrupt Vector Select Register 6-10

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

6.12 JTAG Programming Register 6-11

7 Clock 7-1

7.1 Channel A/B Clock Selection: 7-1

7.2 Frequency Synthesizer: 7-1

7.3 Time Base Divider: 7-1

7.4 Program Sequence: 7-1
7.4.1 Sample Rate Calculation: 7-2

7.5 Phase Lock Loop 7-3

7.6 LMX2306 Communications port 7-3
7.6.1 TIME BASE RESET 7-4
7.6.2 PLL R-COUNTER 7-4
7.6.3 PLL N-COUNTER 7-6
7.6.4 PLL FUNCTION LATCH 7-8

7.7 Clock Control Register 7-11
7.7.1 Clock Control Description 7-12

7.8 LSDIV Low Speed Divider Register 7-12

7.9 Time Base Control Register 7-15
7.9.1 TIME BASE CLOCK Inputs 7-16
7.9.2 D[07:05] MISCELANEOUS Control bits 7-16

7.10 High Speed Divider Data Register 7-17

7.11 Clock A Threshold Register 7-18

7.12 Clock B Threshold Register 7-19

7.13 Trigger A Threshold Register 7-20

7.14 Trigger B Threshold Register 7-21

8 Programming Examples: 8-1

8.1 Clock Programming Example: 8-1

9 Time Base Truth Table: 9-1

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

1 Overview

1.1 Introduction

The Analogic DBS9900 is a high-speed, VXI 'C' size module. The DBS9900 accepts one or two daughterboards (called *modules)* from the DBS 9xx family of high-speed, high precision waveform generators and digitizers. Each module contains the circuitry necessary to operate independently for a complete channel of analog input or output. This motherboard/daughterboard concept allows a single VXI chassis slot to provide multiple functions thereby maximizing VXI resources and decreasing the cost per slot.

The DBS9900 contains the basic register set for a register-based VXI device and also the CLASS Dependent Register set (used for functions common to all modules resident on the DBS9900). It also contains all circuitry and connectors required for supporting functions associated with the plug-in modules. That the entire VXI interface is implemented in a single device provides increased reliability and lower cost.

Versatile clocking and triggering is provided via internal circuitry, front panel connectors, and VXI TTLTRG or ECLTRG lines. The internal clock may be used to drive all plug-in function modules simultaneously so that clock skew and delays are reduced to an absolute minimum. This allows coherent operation of multiple plug-in modules, such as an Arbitrary Waveform Generator and a Digitizer. Programmable trigger and clock thresholds are provided for each module.

Software drivers support all instrument functionality via a soft front panel and are compliant with current VXI Plug and Play specifications. These drivers operate under Windows 9x and Windows NT. The User makes a selection of operating system desired as part of software installation and setup. The driver, upon initialization, interrogates the DBS9900 and identifies installed modules. Appropriate .DLLs are associated with each type of module.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

1.2 Technical Support

If you need technical assistance, or if the unit fails to operate as expected upon arrival, please contact Analogic's Measurement and Control Division (MCD) Applications Engineering department:

Telephone: (978) 977-3000 Domestic Calls: Ext. 3859 or 3881

You will be connected with an T&M Applications Engineer who will work with you to solve your problem. If the unit needs to be returned to the factory, this engineer will refer you to our Customer Service Office.

If referred to Customer Service

Please provide the Customer Service Office with the following information:

- Model Number
- Serial Number
- Purchase Order number
- Quantity being returned
- Detailed description of the malfunction or reported failure
- Your "Bill To" and "Ship To" addresses
- Method of return shipment, with account number

Our Customer Service Coordinator will provide you a Customer Service Order number (CSO) that is an internal Analogic tracking number. The Service Coordinator will also inform you of the Warranty or Non-Warranty status of the units being returned, and the repair charge, if any.

The CSO number is your return authorization number. Please write this number on your Purchase Order and Shipping Label.

Send all authorized returns to:

Analogic Corporation
8 Centennial Drive
Peabody, MA 01960
Attn: Receiving Dock B
CSO# _____

After the material has been returned, you will receive an acknowledgment copy of the CSO that will be marked with the scheduled return date.

Analogic Customer Service Division offers the following means of communication:

Telephone Toll-free (Outside MA)	800-237-2200
Telephone (Inside MA)	978-977-3000
Spare Parts Only	Ext. 3614
Repairs Only	Ext. 3612, 3616, 3617
Facsimile	978-532-8913
Internet E-Mail Address	CSD@ANALOGIC.COM

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Your cooperation in contacting Analogic, Customer Service, and including the CSO Number with your return, as requested above, will allow us to properly track your returned goods and expedite repair and return.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2 Preparation & Installation

2.1 Overview

This section provides instructions for setting up the unit and installing the necessary hardware and software. After unpacking, carefully inspect the hardware for any damage that may have occurred during shipment. If necessary, contact your shipping carrier to file a claim. Save the shipping carton and packing materials if for any reason you need to return the product for repair or replacement.



CAUTION

This product contains components which are sensitive to electrostatic discharge (ESD). Be sure to follow proper procedures for handling, storing and transporting ESD-sensitive assemblies.

2.2 Host Computer Requirements

Minimum system requirements for running the driver and soft front panel applications:

- Pentium processor
- 32 MB RAM
- Win 9x/Win NT 4.0, Service Pack 3
- 20 MB free hard drive space

2.3 Power Consumption and Airflow Requirements

Power Consumption:

Idle.

Airflow Requirements

4 liters/minute

Volts	Amps	Watts
24	0.00	0.0
-24	0.00	0.0
12	0.01	0.1
-12	0.04	0.4
5	0.70	3.5
-5.2	0.98	5.1
-2	0.64	1.3

Supply Power Draw = 10.4 Watts

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.4 Front Panel Connectors and Indicators

The DBS9900 Front Panel is shown below.

I/O 1, I/O 2, CAL, CLK and TRIG are SMA connectors. An additional 26 pin, 3-row D connector is provided for module specific I/O. There are three LED indicators on the front panel. They are POWER, CH 1 ACTIVE, and CH 2 ACTIVE. The power LED is yellow during self-test. If DC power is OK at the end of the self-test, it will turn green. The CH1 and CH2 LEDs are dark if no module is installed, green when the module is active, and amber when idle.

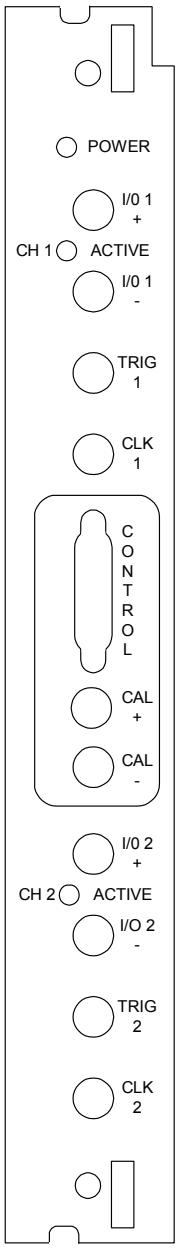


Figure 2-1 DBS9900 Front Panel

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Front Panel Connectors and Indicators, continued

Pin	Description	Pin	Description	Pin	Description
1	I/O_1_B Reserved	**10**	I/O_3_B Reserved	**19**	902 DATA_TAG_B I/O_2
2	I/O_2_B Reserved	**11**	I/O_4_B Reserved	**20**	902 DATA_TAG_A I/O_2
3	901 SYNC I/O_2	**12**	I/O_5_B Reserved	**21**	GND
4	901 MARKER I/O_2	**13**	GND	**22**	901 ADVANCE I/O_2
5	901 SYNC I/O_1	**14**	GND	**23**	901 ADVANCE I/O_1
6	901 MARKER I/O_1	**15**	GND	**24**	GND
7	I/O_3_A Reserved	**16**	GND	**25**	902 DATA_TAG_A I/O_1
8	I/O_2_A Reserved	**17**	I/O_5_A Reserved	**26**	902 DATA_TAG_A I/O_1
9	I/O_1_A Reserved	**18**	I/O_4_A Reserved	Note: All outputs are 50 ohm impedance. All inputs are high impedance.	

Figure 2-2 Front Panel 26 Pin Connector

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.5 Hardware Installation

2.5.1 Installation into a VXI Chassis

When selecting the appropriate VXI chassis, ensure that it meets the cooling and power requirements specified below. This instrument can be installed in any VXI chassis slot except Slot 0.

One empty slot is required.

The DBS9900 should be FIRMLY inserted into the slot to ensure proper operation.

2.5.2 The DBS9900 Logical Address

Registers in VXIbus devices are located in the A16 address space (within 64 byte blocks). To determine the unit's base address, the device's Logical Address must be known. The Logical Address is an 8-bit value, which maps to a unique A16 address. The Logical Address selector is an eight bit dip switch located as shown in the figure below.

The Logical Address value indicated by this selector corresponds to bits 6-13 of the device register base address. Bits 14 and 15 of the base address are both 1, and the base address is:

Logical Address * 64 + 49152

The DSB9900 is shipped with the base address set to 0xE000 (which maps to Logical Address 128).

To change the shipped DBS9900 Logical Address

Using a needle probe or similar instrument, each of the eight dip switches may be set to the desired logical address. Simply insert the needle probe through one of the holes in the case and push the dip switch to the desired setting.

For A16 Base Address 0xC000 – 0xFF80, set switch to the "ON" position for logic 1. Do not set to 0xFFC0. This represents Logical Address 255d, which is used for DC devices and should not be set for the DBS9900.

In the figure below, S1 is shown in the shipped position.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.5.3 A16, A24 and A32 Communications Mode

The DBS9900 contains jumpers that can be set for A24 or A32 Communications Mode. These jumpers are configured at the factory for A24 mode. To operate in A16 mode, do not set any jumpers.

In order to change this setting the DBS9900 boards must be removed from their case to access the jumpers. Proceed with caution.

Use the figure below to determine jumper location when setting the jumpers for A24 or A32 Communications Mode.

Note: A24 is only recommended when using a GPIB-VXI-C VXI controller.

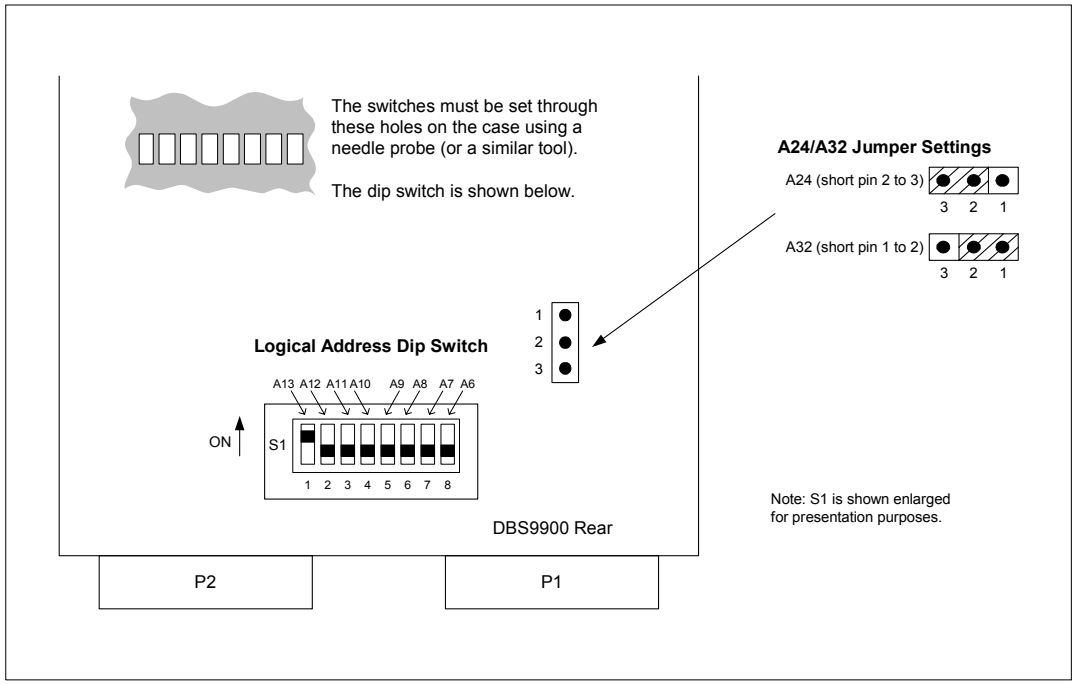


Figure 2-3 Jumper Settings for A24 and A32 Communications Mode

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.6 Software Installation

The DBS9900 software is capable of interfacing with a host system via one of the following options:

- Soft Front Panel
- .DLL (for application programming) – Plug and Play driver functions
- Direct manipulation of the hardware registers on the board
  (This is *not* recommended. It is recommended that the Plug and Play driver functions in the .DLL be used instead – via Soft Front Panel use or some other application).

All software is shipped on the CD/ROM provided.

2.6.1 Installation Steps

Follow the steps below to install the software from the CD/ROM.

Step	Action
1.	Insert CD/ROM.
2.	The installation should proceed automatically via an InstallShield wizard. Follow the instructions as they appear.

2.7 Start-Up and Verification

2.7.1 Self-Test

When the VME bus host asserts the Soft Reset bit, the DBS9900 performs a simple self-test of its circuitry. During the self-test cycle, the DC Power is checked to ensure supply voltage is within spec. If DC power is out of spec, the SYSFAIL signal will be asserted on the VXI bus. This can be masked to prevent the SYSFAIL from being asserted by setting the SYSFAIL INHIBIT bit in the CONTROL REGISTER (0x04). The power LED indicates that SYSFAIL is asserted if yellow. During self-test, the power LED shows yellow. If DC power is OK at the end of self-test, it will turn green.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.8 DBS9900/Module Pin Connections

NAME	DESCRIPTION	TYPE	# PINS
D[15:0]	Bidirectional data bus	BI	16
A[22:0]	Address bus in units of words	out	23
Write	Defines write cycle	out	1
/strobe	Enables data for read clocks write data	out	1
/msel	Identifies I/O cycle as memory	out	1
/clear	Bus reset or soft reset	out	1
/dtak_hold	Extends cycle by delaying DTACK	In O.C.	1
Interrupt	Interrupt line	In	1
/error	Self Test output signal	In O.C.	1
Int_clk	Interface clock	out	1
Sam_clock	Sample Clock	out	2
Size[3:0]	Memory size required	in	4
Done	Module Waveform terminated	in	1
Installed	Module installed in slot	in	1
/clk_hold	Forces Int_clk ON when low	In O.C.	1
Sync_out	Sync signal to Backplane	in	1
Trig_out	Trigger signal to Backplane	in	1
Gate_mode	Gate mode high true	out	1
/trig	Trigger low true	out	1
/adv	Advance low true	out	1
Arm	Generator / Digitizer is armed	out	1
Tclk	JTAG clk	out	1
Tdo	JTAG data out to module	out	1
Tdi	JTAG data in from module	in	1
Tms	JTAG control out to module	out	1
	TOTAL		66

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.9 DBS9900 26 Pin Control Connector

Pin	Description	Pin	Description	Pin	Description
1	Mod. B - 1	10	Mod. B - 3	19	Mod. B - 6
2	Mod. B - 2	11	Mod. B - 4	20	Mod. B - 7
3	Mod. B SYNC	12	Mod. B - 5	21	GND
4	Mod. B MARKER	13	GND	22	Mod. B ADVANCE
5	Mod. A SYNC	14	GND	23	Mod. A ADVANCE
6	Mod. A MARKER	15	GND	24	GND
7	Mod. A - 3	16	GND	25	Mod. A - 7
8	Mod. A - 2	17	Mod. A - 5	26	Mod. A - 6
9	Mod. A - 1	18	Mod. A - 4		

All pins labeled by the module name and a number, e.g., MOD. A-1, are uncommitted pins that may be used for general I/O purposes by the associated plugins.

Pins 3-6 are dedicated for the class of arbitrary waveform generator plugins (for example, DBS901).

Pins 22 and 23 are dedicated to digitizer class plugins (for example, DBS902).

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

3 Operation

The DBS9900 High-Performance VXI Carrier provides the structure, power, clock and trigger functionality for use with many different types of plug-in daughterboard modules.

Features include:

- A programmable threshold External Clock is supplied to each module through front panel connectors (labeled CLK1 and CLK2) to control the sample rate.
- A programmable threshold External Trigger is also supplied for each module through front panel connectors (labeled TRIG1 and TRIG2) to initiate activity of that module.
- Any one of eight TTLTRIG lines can be selected as an external backplane trigger. These can be used with the Internal or External Clock. One of two ECL backplane trigger lines can also be selected.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

3.1 Sample Clock and Timebase

The DBS9900 provides the required timebase (Reference Clock) for both installed modules. The limits of the frequency range are 60Hz to 100MHz depending on the plug-in module.

An Internal Sample Clock may be used for either one or both of the plugin modules. Additionally, independent External Sample Clocks can configured for each or both modules. When making Sample Clock and Reference Clock selections, refer to the following table for guidance.

Clock parameters (frequency, thresholds etc.) are described in more detail on the next page.

3.1.1 Possible Clock Configurations

The table below lists possible clock configurations. If you configure the clocks for I/O 1 in a certain way, that affects what are then clock configurations for I/O 2.

POSSIBLE CLOCK CONFIGURATIONS FOR I/O 1 AND I/O 2			
IF I/O 1 is set to use	**THEN I/O 2 can be set up to use:**	Driver	SFP
Internal Sample Clock (Internal Reference timebase)	Internal Sample Clock *(same as I/O 1)*	√	√
	External Sample Clock using CLK 1	√	
	External Sample Clock using CLK 2	√	
External Reference – CLK2 (Internal Sample Clock using external timebase)	External Sample Clock using CLK 1	√	
	External Reference CLK 2 *(same as I/O 1)*	√	√
External Sample Clock using CLK 1	Internal Sample Clock (Internal Reference)	√	
	External Sample Clock using CLK 1 *(same as I/O 1)*	√	√
	External Sample Clock using CLK 2	√	√
	External Reference CLK 2	√	
External Sample Clock using CLK 2	Internal Sample Clock	√	
	External Sample Clock using CLK 1	√	√
	External Sample Clock using CLK 2 *(same as I/O 1)*	√	√

Note: The DBS9900 Soft Front Panel implements the most common, simple clock configurations. To use a more complicated configuration (for example, for I/O 1 to use the Internal Sample Clock and I/O 2 to use an External Sample Clock) the driver functions must be used.

For more information, refer to documentation on the following driver functions.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

- an9900_clockControl
- an9900_clockFreq
- an9900_clockThreshold

3.1.2 Internal Sample Clock Parameters

Configuring the Internal Sample Clock requires the specification of a Desired Frequency. The Desired Frequency range is dependent upon the type of plug-in module installed into the DBS9900 carrier.

- The DBS9900 internal timebase is capable of a 60Hz to 100MHz range. The type(s) of plug-ins may restrict the use of this entire range.

The DBS9900 will calculate an *Actual Frequency* to a maximum error of better than 0.5% from the Desired Frequency. It is the Actual Frequency that is used by the DBS9900. The dividers used to calculate this actual frequency are available to the user in the software driver.

3.1.3 External Sample Clock Parameters

An External Sample Clock can be fed to either CLK 1 or CLK 2 for either module (I/O 1 and I/O 2).

3.1.4 External Reference Parameters

Defines an external signal to act as the reference signal for the PLL.

This signal may be applied to the DBS9900 Front Panel CLK 2 input SMA connector ONLY.

The default External Reference Voltage Threshold is 0V.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

3.2 Performance Specification

3.2.1 ANALOG I/O

Number of Channels:	One single ended or differential I/O port for each plug-in module via front panel mounted SMA connectors.
Signal Amplitude:	See DBS 9XX plug-in module specifications.
Impedance:	(Dependent on modules installed).
Overvoltage Protection:	(Dependent on modules installed).

3.2.2 INTERNAL SAMPLE CLOCK

Frequency Range:	60 Hz to 100 MHz. Useable range dependent on modules installed.
Frequency Setting Resolution:	<=500 PPM of set frequency. Worst case is 50 kHz at 100 MS/s.
Frequency Accuracy:	± 50 PPM of frequency setting.
Settling Time:	1 ms to ± 2% typ.
Phase Jitter:	(1 µs measurement window) 10 ps RMS max. from 12.5 MHz to 100 MHz. 30 ps RMS max. from 95 Hz to 12.5 MHz.

3.2.3 CONTROL INPUTS

External Sample Clock:

Number of Inputs:	2 front-panel-mounted SMA connectors programmable to control either or both plug-in modules.
Frequency Range:	100 Hz to 100 MHz.
Amplitude:	± 500 mV min. to ± 10V max.
Hysteresis:	± 40 mV.
Threshold Setting:	Programmable ± 10V, ± 5% with 78 mV resolution.
Impedance:	4 kΩ ± 5%.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Waveform:	5 µs min. pulse width. May be dependent on module installed.
Overvoltage Protection:	± 20V max.

External Reference for PLL:

Inputs:	One front-panel-mounted SMA connector shared with channel B external clock input.
Frequency Range:	10 MHz ± 1%.
Amplitude:	± 1V min. to ± 10V max.
Hysteresis:	± 40 mV
Threshold Setting:	Programmable ± 10V, ± 5% with 78 mV resolution.
Impedance:	4 kΩ ± 5%.
Overvoltage Protection:	± 20V max.

External Trigger:

Inputs:	One for each plug-in module via 2 front-panel-mounted SMA connectors or via VXI TTLTRG/ECLTRG lines.
Edge Trigger Modes:	Programmable Start on rising edge or start on falling edge.
Gated Trigger Modes:	Programmable. Run when high and stop when low. Or run when low and stop when high.
Amplitude:	± 1V min. to ± 10V max.
Hysteresis:	± 40 mV
Threshold Setting:	Programmable ± 10V, ± 5% with 78 mV resolution.
Impedance:	4 kΩ ± 5%.

Trigger to External Clock Setup:

Time:	5 ns min.

Trigger to Internal Clock:

Uncertainty:	<1 clock period.
Overvoltage Protection:	± 20V max.

3.2.4 CONTROL OUTPUTS

Front-panel-mounted 26 pin "D" connector. Supports DBS 9XX plug-in module specific functions.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

3.2.5 POWER

Total Power Dissapation: 20W max. + installed module power.

DBS 9900 Power Consumption Profile:

DBS 9900 Only

Idle

Volts	Amps	Watts
24	0.00	0.0
- 24	0.00	0.0
12	0.01	0.1
- 12	0.04	0.4
5	0.70	3.5
- 5.2	0.98	5.1
-2	0.64	1.3
Total		10.4

3.2.6 RELIABILITY

MTBF & Failure Rate: 20,000 Hrs.
Determined by the Generic Parts Count method of MIL-HDBK-217F for a ground benign environment at a temperature of 30°C.

3.2.7 RECOMMENDED WARM-UP TIME

20 minutes.

3.2.8 ENVIRONMENTAL AND MECHANICAL

Ambient Operating Temperature Range:	0°C to + 40°C for rated specifications.
Storage Temperature Range:	- 25°C to + 75°C non-condensing.
Cooling Air Flow:	4 litre/sec. For 10° rise at 10 mm H_20 back pressure typ.
Relative Humidity:	90% max. non-condensing.
Vibration:	2G sinusoidal at 55 Hz.
Shock:	30G/11 mS half sine pulse.
Weight:	1.5lbs., (0.68 kg).
Size:	Single width, VXI "C" size.

3.2.9 VME/VXI BUS COMPLIANCE

VXI spec. version 1.4:	Register based, servant device.
Interrupts:	Programmable for 1 of 7 levels, ROAK, maskable.
VME A24:	Address modifier codes 39, 3A, 3D, 3E.
VME A16:	Address modifier codes 29 and 2D.

3.2.10 SOFTWARE DRIVER

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

VXI plug&play® compliant for Windows 95 and NT.
Written using LabWindows CVI®.
Source code is included as well as .DLL files to allow porting to most popular programming environments.
Source code is also provided with the fully functional soft front panels.

3.2.11 EMC & SAFETY AGENCY COMPLIANCE

Meets all requirements for Safety and EMC for the US, Canada and Europe and is CE compliant.

3.2.12 CALIBRATION

See plug-in function module specifications.

Specifications are subject to change without notice.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

4 Using the DBS9900 Soft Front Panel

The DBS9900 Soft Front Panel can be used to control the instrument and perform basic operations.

4.1 DBS9900 Main Panel

The DBS9900 Main Panel is the first screen to appear when the DBS9900 Soft Front Panel software is opened. From this panel, the following operations may be performed:

- Initialization of DBS9900 units found in the VXI system.
- Selection and Connection to installed modules I/O 1 and I/O 2.
- Clock Configuration – Internal and External Sample Clocks, External Reference
- Trigger Selection

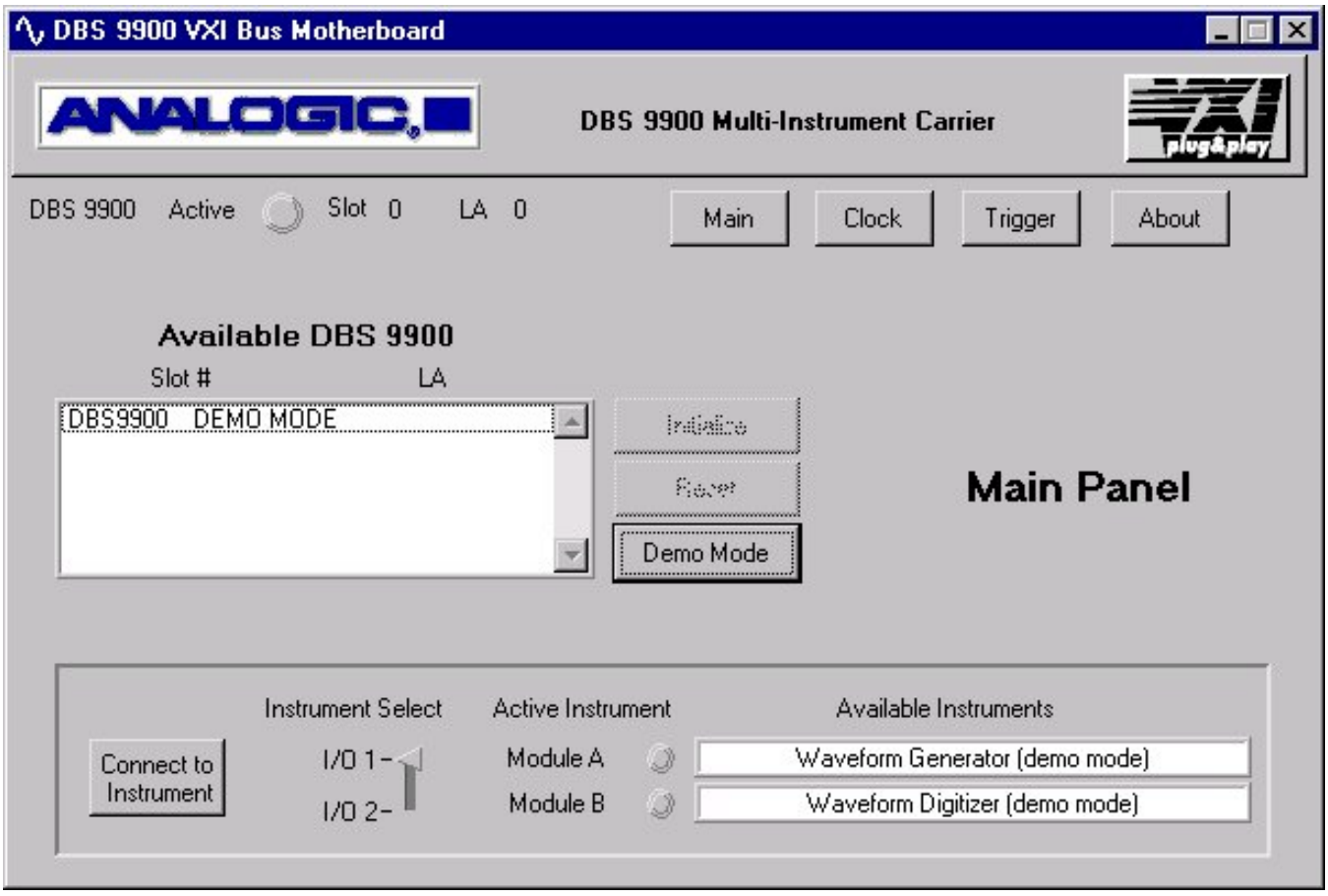


Figure 4-1 DBS9900 SFP Main Panel

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

4.1.1 Main Panel Field Descriptions

Demo Mode Button	Demo mode allows mock operation of most instrument functions & controls, as if the instrument were actually present.
Sample Clock	Brings up the Sample Clock panel.
Trigger Button	Brings up the Trigger panel.
Help Button	Allows access to the DBS9900 Soft Front Panel online help system.
Reset Button	When pressed, forces all soft controls to their default settings. The instrument will be reset to its initial power up conditions. Before initiating the reset function, a warning will appear stating that all present instrument settings and waveform data will be lost and you will be given an opportunity to Continue or Cancel.
About Button	Brings up the "About" screen detailing software version information.
Initiate Button	Interrogates the selected DBS9900 (in the Available DBS9900 Listing") and causes the installed module(s) to be shown in the "Available Instruments" display. If a DBS9900 was previously initiated, the initiate button brings up the message; "DBS9900 at LAxx already initiated". An "OK" button in the message box allows the message to disappear.
Available DBS9900	Shows the slot number and logical address of all DBS9900 installed in the VXI chassis. Allows selection of the desired DBS9900 to initialize.
Exit Button	Closes the Soft Front panel program.
Connect to Instrument	Brings up the selected instrument. If no instrument is installed, nothing happens when pressing the "Connect to Instrument" button.
Active Indicator	Illuminates when the DBS9900 that is highlighted in the "Available DBS9900" listing, has been initiated. When any DBS9900 that has not been initiated is highlighted, the Active Indicator is not illuminated.
Available Instruments Display	Shows the name and model number of any instruments (plug-in modules) that are installed in the DBS9900 that is presently highlighted in the "Available DBS9900" listing and has been initiated. If the DBS9900 that is highlighted has not been initiated, the "Active" indicator will be off and the Available Instruments Display will be blank.
Active Instrument Indicators	Labeled *Module A* and *Module B*". Illuminate green to show the instrument selected by the "Instrument Select" switch. If no instrument (module) is installed in the DBS9900, the Active Instrument indicator will remain red. If an instrument has previously been connected to, its "Active Instrument" indicator will remain green, even if the "Instrument Select" switch is not pointing to the instrument. These indicators are correlated with the CH1 and CH2 indicators that appear on the front panel hardware of the DBS9900.
Instrument Select Switch	Selects the installed instrument (module) that is connected when the "Connect To Instrument" button is pushed. If no instrument (module) is installed, nothing happens when the "Connect To Instrument" button is pushed.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

4.2 DBS9900 Sample Clock Panel

The DBS9900 Sample Clock Panel may be accessed from the Main Panel or via the *Sample Clock* button on any other panel.

From this panel, the following operations may be performed:

- Selection of Clock Mode: Internal, External or External Reference
  The Clock Mode box allows selection of the clock configuration to used to drive the installed modules I/O 1 and I/O 2. This may be Internal Sample Clock, External Sample Clock or External Reference.
  At this time, the Soft Front Panel requires that both modules use the same clock configuration.

- Configuration of the Internal Sample Clock
  The Internal Sample Clock allows a *Desired Frequency* to be entered (the sample rate desired). This may up to 100MHz for installed Generator modules and up to 80MHz for installed Digitizer modules. The DBS9900 will calculate an *Actual Frequency* to less than 0.4% from the *Desired Frequency* and display this in the Internal Sample Clock box. The dividers used to calculate this frequency will also be displayed.

- Configuration of the External Sample Clock
  An External Sample Clock can be fed to either CLK 1 or CLK 2 for either module (I/O 1 and I/O 2). If an External Sample Clock is used, the *Threshold* box allows the selection of a voltage threshold.
  If an External Sample Clock is selected, the *Internal Clock* box will become grayed out (inaccessible).

- Configuration of an External Reference
  An External Reference can be used for the PLL 10MHz reference signal. This signal may be applied to the DBS9900 Front Panel CLK 2 input ONLY. The default External Reference voltage threshold is 0V. This may be changed by changing the Ext. Ref. field in the *Threshold* box.
  If External Reference is selected, the Internal Clock box will remain active so that a desired sample rate can be entered.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

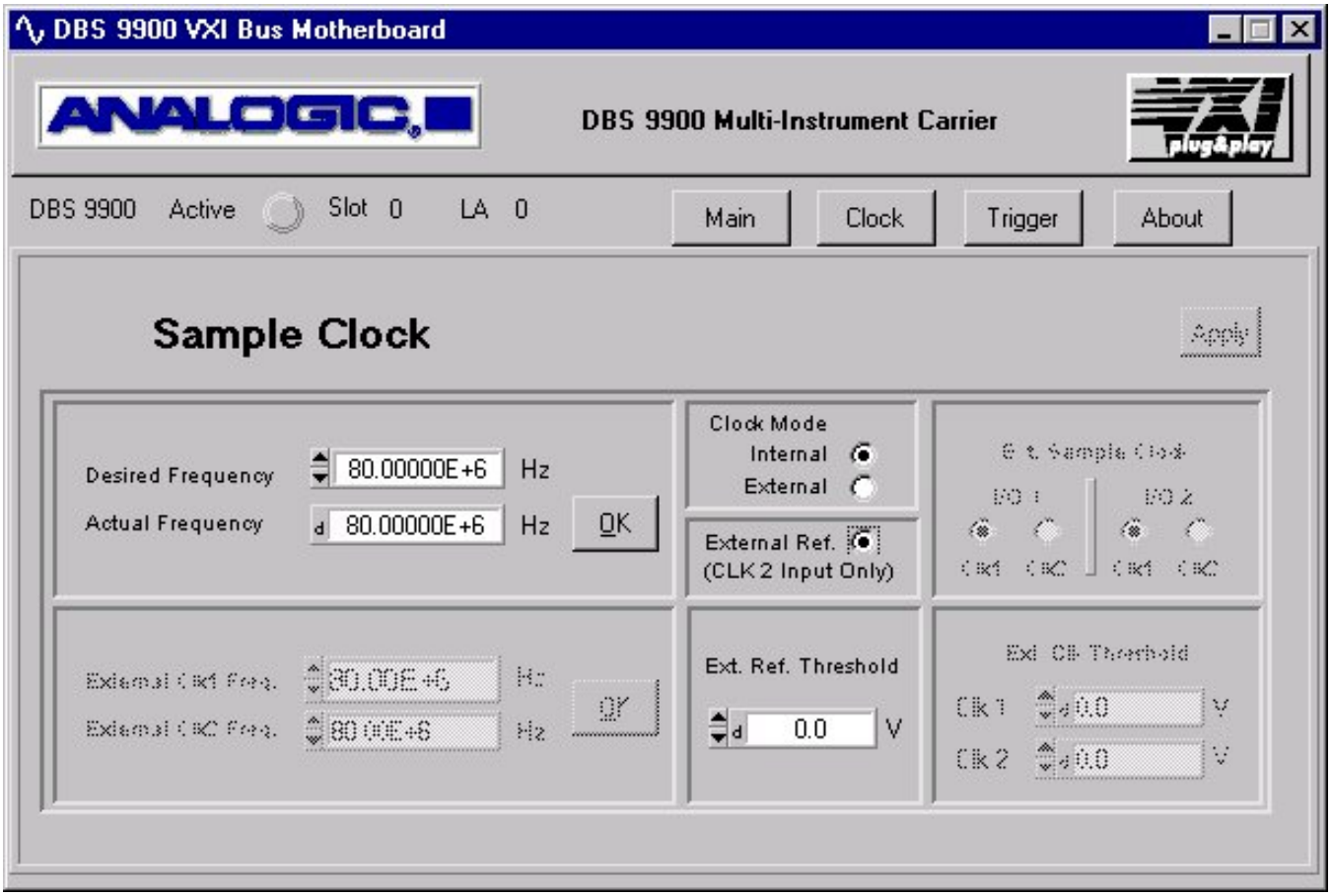


Figure 4-2 DBS9900 SFP Sample Clock Panel

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

4.2.1 Sample Clock Field Descriptions

Internal Clock Box	This entire box is grayed out except when "Clock Mode" is set to "Internal" or "External Ref".
Internal Clock / Desired Frequency	Eight digit control with scrolling. User enters desired sample clock frequency. Default is 80,000,000Hz.
Internal Clock / Actual Frequency	Eight digit display only of sample clock frequency actually produced. Display is updated when "OK" is pushed.
Internal Clock / OK Push Button	Updates "Actual Frequency", "R Counter", "N Counter", "HSDIV", and "LSDIV" displays to correlate with the "Desired Frequency".
Internal Clock / R Counter, N Counter, HSDIV & LSDIV	Displays the values of the clock registers used to produce the "Actual Frequency" after pressing 'OK".
Clock Mode	Three position target switch that selects "Internal" or "External" clock input or "External Ref" for the PLL 10MHz reference signal. Note that the reference signal can be applied only to the "CLK 2" input on the DBS9900 front panel. Default is 'Internal".
External Sample Clock Box	This entire box is grayed out except when "Clock Mode" is set to "External".
I/O-1	Two position target switch that selects either the "CLK-1" or "CLK-2" input on the DBS9900 front panel and feeds it to I/O-1 (module A). Default is "Clk-1".
I/O-2	Two position target switch that selects either the "CLK-1" or "CLK-2" input on the DBS9900 front panel and feeds it to I/O-2 (module B). Default is "Clk-1".
Threshold Box	This entire box is grayed out except when "Clock Mode" is set to "External".
Clk-1	Bipolar, scrolling control with X.X resolution. Sets threshold voltage for "Clk-1" external clock input. Default is 0V.
Clk-2	Bipolar, scrolling control with X.X resolution. Sets threshold voltage for "Clk-2" external clock input. Default is 0V.
Ext Ref	Bipolar, scrolling control with X.X resolution. Sets threshold voltage for the 10MHz external reference signal. Note that this signal is applied to the DBS9900 front panel Clk-2 input only. Default is 0V. This control is grayed out except when "Clock Mode" is set to "External Ref.".
Apply	Any user changes on the soft panel will not take effect until the "Apply" button is pushed.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

4.3 DBS9900 Trigger Panel

All of the following controls are duplicated in both the I/O-1 and I/O-2 boxes.

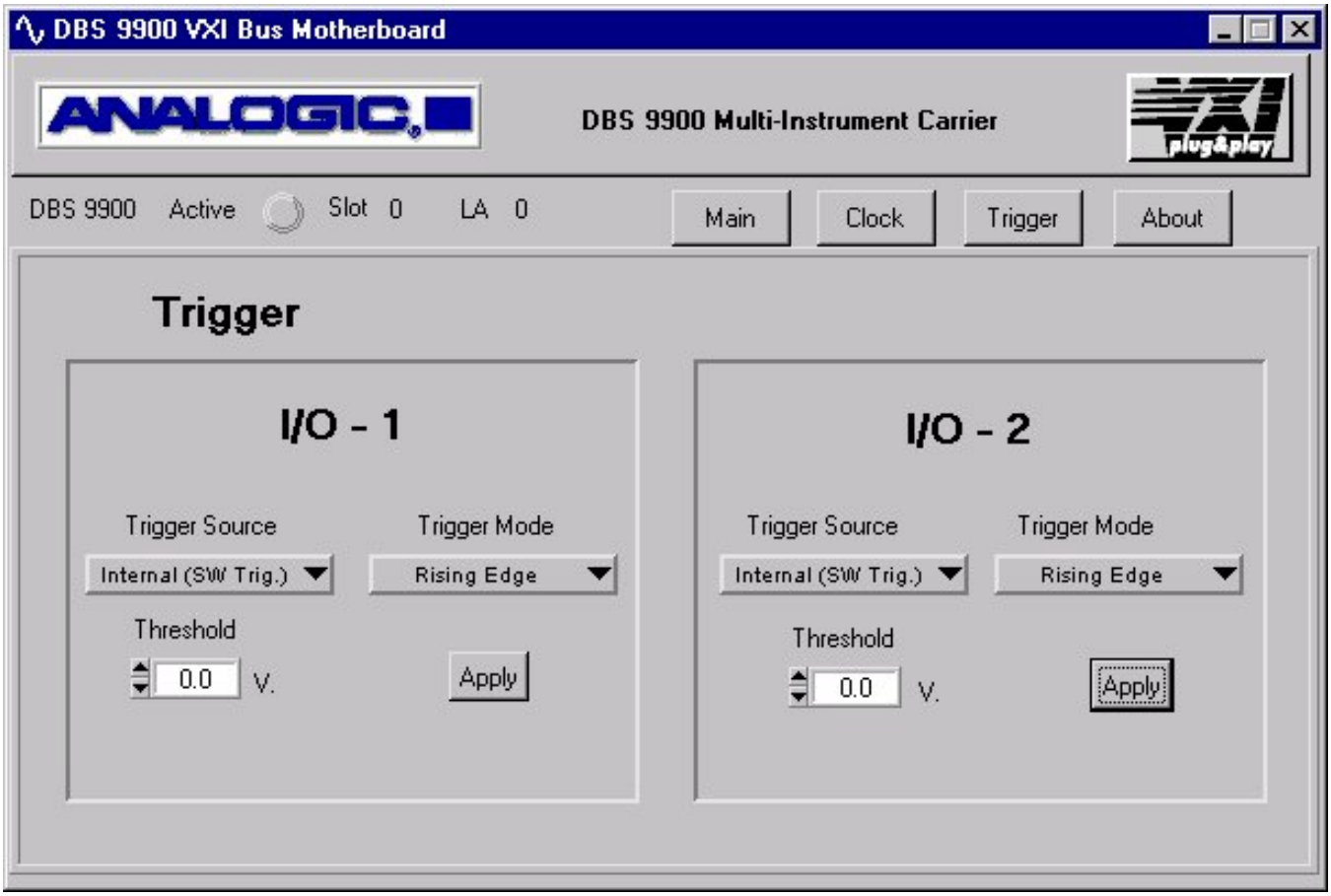


Figure 4-3 DBS9900 SFP Trigger Panel

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

4.3.1 Trigger Panel Field Descriptions

Trigger Source	Pull down menu. Default is "Internal (SW Trig.)". Other menu items are "Front Panel Ch. 1", "Front Panel Ch. 2", "TTLTRG-0", "TTLTRG-0", "TTLTRG-1", "TTLTRG-2", "TTLTRG-3", "TTLTRG-4", "TTLTRG-5", "TTLTRG-6", "TTLTRG-7", "ECLTRG-0" and "ECLTRG-1".
Trigger Mode	Pull down menu. Default is "Rising Edge". Other menu items are "Falling Edge", "Gated Active Hi" and "Gated Active Lo". The "Trigger Mode" control for I/O-2 will be grayed out if the user has selected "Front Panel Ch. 1" or "Front Panel Ch. 2" for both I/O-1 and I/O-2. (Only one mode setting can be used for a given trigger input.)
Threshold	Bipolar (+ & -), scrolling control with a resolution of X.X Volts. The threshold control for I/O-2 will be grayed out if the user has selected "Front Panel Ch. 1" or "Front Panel Ch. 2" for both I/O-1 and I/O-2. (Only one threshold setting can be used for a given trigger input.)
Apply	Any user changes on the soft panel will not take effect until the "Apply" button is pushed. Upon pushing apply, it must be checked and verified that the TTLTRG or ECLTRG line that is selected on this panel, is not already selected elsewhere. If already selected elsewhere, a message will appear saying; "The TTLTRG or ECLTRG line you selected is already in use. Please select another line or change the selection previously made."

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5 DBS9900 Driver Software – Description of Functions



WARNING

The application software that uses this driver should not try to access 9900 registers directly. Using functions other than those provided by this driver to access 9900 registers may end in unexpected results. However, *reading* register values or getting attribute values using VISA function calls with the Vi Session returned by 'an9900_init' is acceptable.

5.1 Initialize, Module and Close Functions

5.1.1 Initialize an9900_init

This function initializes the instrument by getting a session handle (ViSession) from VISA and copying calibration constants from the EEPROM to the instrument registers.

Function Prototype

ViStatus _VI_FUNC **an9900_init** (ViRsrc **resourceName**, ViBoolean **IDQuery**, ViBoolean **resetDevice**, ViPSession **vi**);

an9900_init		
Parameters	**Variable Type**	**Description**
<INPUT>		
resourceName	ViRsrc	Specifies the name of the VXI resource. Example: "VXI::128" where 128 represents the logical address of the instrument.
IDQuery	ViBoolean	Specifies whether an ID query is sent to the instrument during initialization. IDQuery ensures that the instrument at the given address is indeed a DBS9900.
resetDevice	ViBoolean	Specifies whether the instrument is to be reset to its power-on settings during the initialization procedure.
vi	ViPSession	Session handle.
<OUTPUT>		
None		
<RETURN>		
= 0 > 0 < 0	ViStatus	"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.1.2 VIT Initialize an9900_VITinit

This function is used only for the Soft Front Panel and performs the following initialization actions:

- Opens a session to the Default Resource Manager resource and a session to the specified device using the interface and address specified in the Resource_Name control.
- Performs an identification query on the Instrument.
- Resets the instrument to a known state.
- Sends initialization commands to the instrument that set any necessary programmatic variables such as Headers Off, Short Command form, and Data Transfer Binary to the state necessary for the operation of the instrument driver.
- Returns an Instrument Handle which is used to differentiate between different sessions of this instrument driver.

Each time this function is invoked, a Unique Session is opened. It is possible to have more than one session open for the same resource.

Function Prototype

ViStatus _VI_FUNC **an9900_VITinit** (ViRsrc **resourceName**, ViBoolean **IDQuery**, ViBoolean **resetDevice**, ViSession **instrumentHandle**);

an9900_VITinit		
Parameters	**Variable Type**	**Description**
<INPUT>		
resourceName	ViRsrc	Specifies the interface and address of the device that is to be initialized. Default Value: *GPIB-VXI : : 144* Note: Based on the instrument Descriptor, this operation establishes a communication session with a device. The grammar for the Instrument Descriptor is shown below. Optional parameters are shown in square brackets ([ ]) VXI — VXI [board] : : VXI logical address [: : INSTR] GPIB-VXI — GPIB-VXI [board] [: : GPIB-VXI primary address] : : VXI logical address [: : INSTR] The VXI keyword is used for either embedded or MXIbus controllers in VXI instruments. The default value for optional parameters are: board — 0 GPIB-VXI primary address — 1
IDQuery	ViBoolean	Specifies if an ID Query is sent to the instrument during the initialization procedure. Valid Range: VI_OFF (0) — Skip Query VI_ON (1) — Do Query (Default Value)
resetDevice	ViBoolean	This control specifies if the instrument is to be reset to its power-on settings during the initialization procedure. Valid Range:

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

		VI_OFF (0) Don't Reset VI_ON (1) Reset Device (Default Value) Notes: If you do not want the instrument reset. Set this control to *Don't Reset* while initializing the instrument.
instrumentHandle	ViSession (passed by reference)	This control returns an Instrument Handle that is used in all subsequent function calls to differentiate between different sessions of this instrument driver. Note: Each time this function is invoked a Unique Session is opened. It is possible to have more than one session open for the same resource.
<OUTPUT>		
None	None	None
<RETURN>		
0	ViStatus	No error (the call was successfully).
BFFC0002		Parameter 2 (ID Query) out of range.
BFFC0003		Parameter 3 (Reset Device) out of range.
BFFF0000		Miscellaneous or system error occurred.
BFFF000E		Invalid session handle.
BFFF0015		Timeout occurred before operation could complete.
BFFF0034		Violation of raw write protocol occurred.
BFFF0035		Violation of raw read protocol occurred.
BFFF0036		Device reported an output protocol error.
BFFF0037		Device reported an input protocol error.
BFFF0038		Bus error occurred during transfer.
BFFF003A		Invalid setup (attributes are not consistent).
BFFF005F		No listeners condition was detected.
BFFF0060		This interface is not the controller in charge.
BFFF0067		Operation is not supported on this session.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.1.3 Set Module — an9900_setModule

The DBS9900 can host two modules, any combination of DBS901s and DBS902s. Some DBS9900 functionality, such as clock generation is common to both modules, whereas other functionality such as clock and trigger source selection can be controlled on a per-module basis.

The AN9900_MODULE enumerated constant is a parameter in all DBS9900 driver functions that need to "know" which module they are acting upon. It is also used by the "register select" function, an9900setModule, to specify which module subsequent an901_xxx or an902_xxx calls will act upon.

Function Prototype

ViStatus _VI_FUNC **an9900_setModule** (ViSession **instrumentHandle**, AN9900_MODULE **module**);

an9900_setModule		
Parameters	**Variable Type**	**Description**
<INPUT>		
instrumentHandle	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
module	AN9900_MODULE	AN9900_MOD_A Module A selected. AN9900_MOD_B Module B selected.
<OUTPUT>		
None	None	None
<RETURN>		
= 0 > 0 < 0	ViStatus	"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully.

5.1.4 Get Module — an9900_getModule

This function reads the current module selected.

Function Prototype

ViStatus _VI_FUNC **an9900_getModule** (ViSession **instrumentHandle**, AN9900_MODULE ***module**);

an9900_getModule		
Parameters	**Variable Type**	**Description**
<INPUT>		
vi	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
<OUTPUT>		
***module**	AN9900_MODULE	AN9900_MOD_A Module A selected. AN9900_MOD_B Module B selected.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

<RETURN>		
= 0	ViStatus	"VI_SUCCESS"
> 0		Warning: The function completed, but an exception condition occurred which may require attention.
< 0		Error: The function did not complete successfully.

5.1.5 Identify Module an9900_identifyModule

This function specifies whether a given module is a DBS901, a DBS902 or is not installed. By calling this function twice, once with each AN9900_MODULE value, a program can determine which plug-in daughterboards are installed.

Function Prototype

ViStatus _VI_FUNC **an9900_identifyModule** (ViSession **vi**, AN9900_MODULE **mod**,
AN9900_MOD_TYPE ***type**);

an9900_identifyModule		
Parameters	**Variable Type**	**Description**
<INPUT>		
vi	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
mod	AN9900_MODULE	AN9900_MOD_A Check Module A. AN9900_MOD_B Check Module B.
<OUTPUT>		
***type**	AN9900_MOD_TYPE	AN9900_MOD_EMPTY No module is installed at this position. AN9900_MOD_DBS901 DBS901 waveform generator is installed. AN9900_MOD_DBS902 DBS902 digitizer is installed. AN9900_MOD_UNKNOWN Invalid module type, EEPROM may be corrupt or invalid.
<RETURN>		
= 0	ViStatus	"VI_SUCCESS"
> 0		Warning: The function completed, but an exception condition occurred which may require attention.
< 0		Error: The function did not complete successfully.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.1.6 Close an9900_close

This function takes the instrument offline and performs the following operations:

- viClose (instrSession) and
- viClose (rmSession).

The instrument must be re-initialized before it can be used again.

Function Prototype

ViStatus **an9900_close** (ViSession **instrumentHandle**);

an9900_close		
Parameters	**Variable Type**	**Description**
<INPUT>		
instrumentHandle	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session. Default Value: None
<OUTPUT>	None	None
<RETURN>		
= 0 > 0 < 0		"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfullly. NOTE: Use the an902_error_message() function to get a textual description of errors and warnings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.2 Control

This class provides control functions to communicate with the instrument, and indicate

- the type of memory requirements,
- the device type code,
- the revision level of the PCB,
- generate the SOFT RESET,
- inhibit SYSFAIL generation,
- enable A24/A32 memory access,
- return the value of the status register,
- and set and query the interrupt level and interrupt vector.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.2.1 Device Type Query — an9900_deviceType_Q

Indicates the type of memory requirements as defined in the VXI specifications, the device type code, and revision level of the PCB.

Function Prototype

ViStatus **an9900_deviceType_Q** (ViSession **instrumentHandle**,
ViUInt16 ***memoryRequirements**,
ViUInt16 ***deviceTypeCode**,
ViUInt16 ***revisionLevel**);

an9900_deviceType_Q		
Parameters	**Variable Type**	**Description**
<INPUT>		
instrumentHandle	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
<OUTPUT>		
*memoryRequirements	ViUInt16 (passed by reference)	The Value will be 0xm21r, where m is dependent on the memory requirements of the DBS9900 (with any daughter boards installed) and r is the revision level of the motherboard PCB.
*deviceTypeCode	ViUInt16 (passed by reference)	Indicates the device type code.
*revisionLevel	ViUInt16 (passed by reference)	The revision level of the printed circuit board.
<RETURN>		
= 0 > 0 < 0		"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfullly. NOTE: Use the an902_error_message() function to get a textual description of errors and warnings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.2.2 Control an9900_control

Used to generate the SOFT RESET, inhibit SYSFAIL generation, and to enable A24/A32 memory access. If the reset flag is set, it will be set and cleared by the driver to cause a Soft Reset.

Function Prototype

ViStatus **an9900_control** (ViSession **instrumentHandle**,
ViUInt16 **softReset**,
ViUInt16 **sysfailInhibit**,
ViUInt16 **memoryAccess**);

an9900_control		
Parameters	**Variable Type**	**Description**
<INPUT>		
instrumentHandle	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
softReset	ViUInt16	Selects to RESET the DBS9900 registers, or to enable access to them. 1 Resets DBS9900 Registers 0 Enables access to DBS9900
sysfailInhibit	ViUInt16	Selects to prevent or enable sysfail assertion. 1 Prevent SYSFAIL assertion 0 Enables SYSFAIL assertion
memoryAccess	ViUInt16	Selects to enable or disconnect the DBS9900 memory access. 1 Enables DBS9900 memory access 0 Disconnects DBS9900 memory
<OUTPUT>	None	None
<RETURN>		
= 0 > 0 < 0		"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully. NOTE: Use the an902_error_message() function to get a textual description of errors and warnings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.2.3 Status Query — an9900_status_Q

Returns the value of the status register.

Function Prototype

ViStatus **an9900_status_Q** (ViSession **instrumentHandle**,
ViUInt16 ***status**);

an9900_status_Q		
Parameters	**Variable Type**	**Description**
<INPUT>		
instrumentHandle	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
<OUTPUT>		
*status	ViUInt16 (passed by reference)	Returns the value of the status/control register.
<RETURN>		
= 0 > 0 < 0		"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully. NOTE: Use the an902_error_message() function to get a textual description of errors and warnings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.2.4 Interrupt an9900_interrupt

Sets the interrupt level and interrupt vector.

Function Prototype

ViStatus **an9900_interrupt** (ViSession **instrumentHandle**,
ViUInt16 **interruptLevel**,
ViUInt16 **interruptVector**);

an9900_interrupt		
Parameters	**Variable Type**	**Description**
<INPUT>		
instrumentHandle	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
interruptLevel	ViUInt16	Selects one of seven interrupt levels that the DBS9900 will interrupt on. 0 No Interrupt 1 Level 1 2 Level 2 3 Level 3 4 Level 4 5 Level 5 6 Level 6 7 Level 7
interruptVector	ViUInt16	Contains the lower 8 bits of the STATUS/ID or interrupt vector that will be returned to the host during interrupt Acknowledge cycles.
<OUTPUT>	None	None
<RETURN>		
= 0 > 0 < 0		"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully. NOTE: Use the an902_error_message() function to get a textual description of errors and warnings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.2.5 Interrupt Query — an9900_interrupt_Q

Queries the interrupt level and interrupt vector.

Function Prototype

ViStatus **an9900_interrupt_Q** (ViSession **instrumentHandle**,
ViUInt16 ***interruptLevel**,
ViUInt16 ***interruptVector**,
ViUInt16 ***module**);

an9900_interrupt_Q		
Parameters	**Variable Type**	**Description**
<INPUT>		
instrumentHandle	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
<OUTPUT>		
*interruptLevel	ViUInt16 (passed by reference)	Returns the Interrupt Level on the DBS9900.
*interruptVector	ViUInt16 (passed by reference)	Contains the lower 8 bits of the STATUS/ID or interrupt vector that will be returned to the host during interrupt Acknowledge cycles.
*module	ViUInt16 (passed by reference)	2 bit mask specifying the module tha triggered the interrupt. A 01b mask indicates that module B generated the interrupt, a 10b mask indicates that module A generated the interrupt, and a 11b mask indicates simultaneous interruptions by both modules.
<RETURN>		
= 0 > 0 < 0		"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully. NOTE: Use the an902_error_message() function to get a textual description of errors and warnings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.3 Module

This class provides functions to communicate with the instrument and

- returns information about specific module status,
- selects one of two modules, and calibration port,
- writes data to the specified EEPROM location of the active module,
- reads data back from the specified EEPROM location of the active module, and
- enables or disables the active module.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.3.1 Module Status Query an9900_moduleStatus_Q

Returns information about specific module status. If present, done operation, and spare.

Function Prototype

ViStatus **an9900_moduleStatus_Q** (ViSession **instrumentHandle**,
ViUInt16 **module**,
ViUInt16 ***present**,
ViUInt16 ***done**);

an9900_moduleStatus_Q		
Parameters	**Variable Type**	**Description**
<INPUT>		
instrumentHandle	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
module	ViUInt16	Selects the module from which information is desired. 1 Module 2 0 Module 1
<OUTPUT>		
*present	ViUInt16 (passed by reference)	Flag specifying if module is present.
*done	ViUInt16 (passed by reference)	Flag specifying if module is done executing.
<RETURN>		
= 0 > 0 < 0		"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully. NOTE: Use the an902_error_message() function to get a textual description of errors and warnings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.3.2 Module Control — an9900_moduleControl

Selects one of two modules, and calibration port.

Function Prototype

ViStatus **an9900_moduleControl** (ViSession **instrumentHandle**,
ViUInt16 **module**,
ViUInt16 **calPortRelay**,
ViUInt16 **SPRRLY1**,
ViUInt16 **SPRRLY2**,
ViUInt16 **SPRRLY3**);

an9900_moduleControl		
Parameters	**Variable Type**	**Description**
<INPUT>		
instrumentHandle	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
module	ViUInt16	Specifies the module to be selected from the register set. 1 Module B 0 Module A
calPortRelay	ViUInt16	Selects the channel to be connected to the cal port. 1 Channel 2 connected to CAL port 0 Channel 1 connected to CAL port
SPRRLY1	ViUInt16	Opens or closes the specific SPR_RLY 1 Close 0 Open
SPRRLY2	ViUInt16	Opens or closes the specific SPR_RLY 1 Close 0 Open
SPRRLY3	ViUInt16	Opens or closes the specific SPR_RLY 1 Close 0 Open
<OUTPUT>	None	None
<RETURN>		
= 0 > 0 < 0		"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully. NOTE: Use the an902_error_message() function to get a textual description of errors and warnings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.3.3 Module Control Query — an9900_moduleControl_Q

Queries information about module and relay selection.

Function Prototype

ViStatus **an9900_moduleControl_Q** (ViSession **instrumentHandle**,
ViUInt16 ***module**,
ViUInt16 ***calPortRelay**,
ViUInt16 ***SPRRLY1**,
ViUInt16 ***SPRRLY2**,
ViUInt16 ***SPRRLY3**);

an9900_moduleControl_Q		
Parameters	**Variable Type**	**Description**
<INPUT>		
instrumentHandle	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
<OUTPUT>		
*module	ViUInt16 (passed by reference)	Returns the module selected for the register set.
*calPortRelay	ViUInt16 (passed by reference)	Returns the channel connected to the cal port.
*SPRRLY1	ViUInt16 (passed by reference)	Returns if the specific SPR_RLY is closed or open.
*SPRRLY2	ViUInt16 (passed by reference)	Returns if the specific SPR_RLY is closed or open.
*SPRRLY3	ViUInt16 (passed by reference)	Returns if the specific SPR_RLY is closed or open.
<RETURN>		
= 0 > 0 < 0		"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully. NOTE: Use the an902_error_message() function to get a textual description of errors and warnings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.3.4 EEPROM Port Write an9900_eepromPortWrite

Writes data to the specified EEPROM location of the active module. The EEPROM must be in write-enabled mode.

Function Prototype

ViStatus **an9900_eepromPortWrite** (ViSession **instrumentHandle**,
ViUInt16 **offset**,
ViUInt16 **eeData**);

an9900_eepromPortWrite		
Parameters	**Variable Type**	**Description**
<INPUT>		
instrumentHandle	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
offset	ViUInt16	Specifies the EEPROM offset to which to write data.
eeData	ViUInt16	Specifies the hex data value to write to the EEPROM.
<OUTPUT>	None	None
<RETURN>		
= 0 > 0 < 0		"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully. NOTE: Use the an902_error_message() function to get a textual description of errors and warnings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.3.5 EEPROM Port Query an9900_eepromPort_Q

Reads back data from the specified EEPROM location of the active module.

Function Prototype

ViStatus **an9900_eepromPort_Q** (ViSession **instrumentHandle**,
ViUInt16 **offset**,
ViUInt16 ***eeData**);

an9900_eepromPort_Q		
Parameters	**Variable Type**	**Description**
<INPUT>		
instrumentHandle	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
offset	ViUInt16	Specifies the EEPROM offset from which to read data.
<OUTPUT>		
*eeData	ViUInt16	Data read from the specified EEPROM port.
<RETURN>		
= 0 > 0 < 0		"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully. NOTE: Use the an902_error_message() function to get a textual description of errors and warnings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.3.6 EEPROM Write Enable an9900_eepromWriteEnable

This function enables or disables data write to the EEPROM of the active module.

Function Prototype

ViStatus **an9900_eepromWriteEnable** (ViSession **instrumentHandle**,
ViUInt16 **writeEnable**);

an9900_eepromWriteEnable		
Parameters	**Variable Type**	**Description**
<INPUT>		
instrumentHandle	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
writeEnable	ViUInt16	Selects to enable or disable data write to the EEPROM. 1 Enable EEPROM write. 0 Disable EEPROM write
<OUTPUT>		
None	None	None
<RETURN>		
= 0 > 0 < 0		"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully. NOTE: Use the an902_error_message() function to get a textual description of errors and warnings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.4 Memory

This class provides functions to communicate with the instrument and to define the base address of the A24 or A32 operational registers. These functions also return important characteristics of the memory of the DBS9900's daughter board modules.

5.4.1 Memory Offset — an9900_memoryOffset

This function defines the base address of the A24 or A32 operational registers.

Function Prototype

ViStatus **an9900_memoryOffset** (ViSession **instrumentHandle**,
ViUInt16 **baseAddress**);

an9900_memoryOffset		
Parameters	**Variable Type**	**Description**
<INPUT>		
instrumentHandle	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
baseAddress	ViUInt16	Defines the base address of the A24 or A32 operational registers. The m+1 most significant bits are the values of the m+1 most significant bits of the A24 or A32 register addresses, where m is the value of the "required memory" field of the device type register.
<OUTPUT>		
None	None	None
<RETURN>		
= 0 > 0 < 0		"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully. NOTE: Use the an902_error_message() function to get a textual description of errors and warnings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.4.2 Memory Offset Query an9900_memoryOffset_Q

This function returns the base address of the A24 or A32 operational registers.

Function Prototype

ViStatus **an9900_memoryOffset_Q** (ViSession **instrumentHandle**,
ViUInt16 ***baseAddress**);

an9900_memoryOffset_Q		
Parameters	**Variable Type**	**Description**
<INPUT>		
instrumentHandle	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
<OUTPUT>		
*baseAddress	ViUInt16 (passed by reference)	Returns the base address of the A24 or A32 operational registers. The m+1 most significant bits are the values of the m+1 most significant bits of the A24 or A32 register addresses, where m is the value of the "required memory" field of the device type register.
<RETURN>		
= 0 > 0 < 0		"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully. NOTE: Use the an902_error_message() function to get a textual description of errors and warnings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.4.3 Memory Attribute Query an9900_memoryAttribute_Q

This function returns important characteristics of the memory of the DBS9900's daughter board modules.

Function Prototype

ViStatus **an9900_memoryAttribute_Q** (ViSession **instrumentHandle**,
ViUInt16 ***memoryAttributes**);

an9900_memoryAttribute_Q		
Parameters	**Variable Type**	**Description**
<INPUT>		
instrumentHandle	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
<OUTPUT>		
*memoryAttributes	ViUInt16 (passed by reference)	Lists important characteristics of the memory on the daughter board modules.
<RETURN>		
= 0 > 0 < 0		"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully. NOTE: Use the an902_error_message() function to get a textual description of errors and warnings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.4.4 Enable Memory an9900_enableMemory

This function either enables or disables the memory.

Function Prototype

ViStatus **an9900_enableMemory** (ViSession **instrumentHandle**,
int memoryAccess);

an9900_enableMemory		
Parameters	**Variable Type**	**Description**
<INPUT>		
instrumentHandle	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
memoryAccess	int	Defines whether memory is enabled or not.
<OUTPUT>		
None	None	None
<RETURN>		
= 0 > 0 < 0		"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully. NOTE: Use the an902_error_message() function to get a textual description of errors and warnings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.5 Trigger

This class provides functions to communicate with the instrument

- to change and query the trigger configuration for each module and
- to set and query the trigger mode for each module.

5.5.1 Set Trigger Source an9900_setTrigSource

This function sets the trigger source.

Function Prototype

ViStatus _VI_FUNC **an9900_setTrigSource** (ViSession **vi**, AN9900_MODULE **mod**,
AN9900_TRIGSRC **src**);

an9900_setTrigSource		
Parameters	**Variable Type**	**Description**
<INPUT>		
vi	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
mod	AN9900_MODULE	AN9900_MOD_A Module A selected. AN9900_MOD_B Module B selected.
src	AN9900_TRIGSRC	AN9900_TRIGSRC_REG Select Software Trigger. AN9900_TRIGSRC_ECL0 Select Backplane ECL trigger. AN9900_TRIGSRC_ECL0 Select Backplane ECL trigger. AN9900_TRIGSRC_FP_A Select Front Panel trigger. AN9900_TRIGSRC_FP_B Select Front Panel trigger. AN9900_TRIGSRC_TTL0 Select backplane TTL trigger (TTL 0 – 7). AN9900_TRIGSRC_TTL1, AN9900_TRIGSRC_TTL2 AN9900_TRIGSRC_TTL3, AN9900_TRIGSRC_TTL4 AN9900_TRIGSRC_TTL5, AN9900_TRIGSRC_TTL6 AN9900_TRIGSRC_TTL7
<OUTPUT>		
None		
<RETURN>		
= 0 > 0 < 0	ViStatus	"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.5.2 Get Trigger Source an9900_getTrigSource

This function gets the currently selected trigger source.

Function Prototype

ViStatus _VI_FUNC **an9900_setTrigSource** (ViSession **vi**, AN9900_MODULE **mod**,
AN9900_TRIGSRC ***src**);

an9900_getTrigSource		
Parameters	**Variable Type**	**Description**
<INPUT>		
vi	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
mod	AN9900_MODULE	AN9900_MOD_A Module A selected. AN9900_MOD_B Module B selected.
<OUTPUT>		
***src**	AN9900_TRIGSRC	AN9900_TRIGSRC_REG Software Trigger selected. AN9900_TRIGSRC_ECL0 Backplane ECL trigger selected. AN9900_TRIGSRC_ECL0 Backplane ECL trigger selected. AN9900_TRIGSRC_FP_A Front Panel trigger selected. AN9900_TRIGSRC_FP_B Front Panel trigger selected. AN9900_TRIGSRC_TTL0 Backplane TTL trigger (TTL 0 – 7) selected. AN9900_TRIGSRC_TTL1, AN9900_TRIGSRC_TTL2 AN9900_TRIGSRC_TTL3, AN9900_TRIGSRC_TTL4 AN9900_TRIGSRC_TTL5, AN9900_TRIGSRC_TTL6 AN9900_TRIGSRC_TTL7
<RETURN>		
= 0	ViStatus	"VI_SUCCESS"
> 0		Warning: The function completed, but an exception condition occurred which may require attention.
< 0		Error: The function did not complete successfully.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.5.3 Set Trigger Output an9900_setTrigOut

This function sets the trigger output.

Function Prototype

ViStatus _VI_FUNC **an9900_setTrigOut** (ViSession **vi**, AN9900_MODULE **mod**,
AN9900_TRIGOUT **out**);

an9900_setTrigOut		
Parameters	**Variable Type**	**Description**
<INPUT>		
vi	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
mod	AN9900_MODULE	AN9900_MOD_A Module A selected. AN9900_MOD_B Module B selected.
out	AN9900_TRIGOUT	AN9900_TRIGOUT_NONE Not driving trigger. AN9900_TRIGSRC_ECL0 Select Backplane ECL output. AN9900_TRIGSRC_ECL0 Select Backplane ECL output. AN9900_TRIGSRC_TTL0 Select backplane TTL output (TTL 0 – 7). AN9900_TRIGSRC_TTL1, AN9900_TRIGSRC_TTL2 AN9900_TRIGSRC_TTL3, AN9900_TRIGSRC_TTL4 AN9900_TRIGSRC_TTL5, AN9900_TRIGSRC_TTL6 AN9900_TRIGSRC_TTL7
<OUTPUT>		
None		
<RETURN>		
= 0	ViStatus	"VI_SUCCESS"
> 0		Warning: The function completed, but an exception condition occurred which may require attention.
< 0		Error: The function did not complete successfully.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.5.4 Get Trigger Output an9900_getTrigOut

This function gets the currently selected trigger output.

Function Prototype

ViStatus _VI_FUNC **an9900_getTrigOut** (ViSession **vi**, AN9900_MODULE **mod**,
AN9900_TRIGOUT ***out**);

an9900_getTrigOut		
Parameters	**Variable Type**	**Description**
<INPUT>		
vi	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
mod	AN9900_MODULE	AN9900_MOD_A Module A selected. AN9900_MOD_B Module B selected.
<OUTPUT>		
***out**	AN9900_TRIGOUT	AN9900_TRIGSRC_NONE No trigger output selected. AN9900_TRIGSRC_ECL0 Backplane ECL output selected. AN9900_TRIGSRC_ECL0 Backplane ECL output selected. AN9900_TRIGSRC_TTL0 Backplane TTL output (TTL 0 – 7) selected. AN9900_TRIGSRC_TTL1, AN9900_TRIGSRC_TTL2 AN9900_TRIGSRC_TTL3, AN9900_TRIGSRC_TTL4 AN9900_TRIGSRC_TTL5, AN9900_TRIGSRC_TTL6 AN9900_TRIGSRC_TTL7
<RETURN>		
= 0	ViStatus	"VI_SUCCESS"
> 0		Warning: The function completed, but an exception condition occurred which may require attention.
< 0		Error: The function did not complete successfully.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.5.5 Set Trigger Polarity an9900_setTrigPolarity

This function sets trigger polarity for the specified module. Whether the trigger is "edge" or "level" sensitive depends on the mode. Gated trigger is level sensitive; all others are edge-sensitive.

Function Prototype

ViStatus _VI_FUNC **an9900_setTrigPolarity** (ViSession **vi**, AN9900_MODULE **mod**,
AN9900_TRIGPOL **pol**);

an9900_setTrigPolarity		
Parameters	**Variable Type**	**Description**
<INPUT>		
vi	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
mod	AN9900_MODULE	AN9900_MOD_A Set trigger polarity for Module A. AN9900_MOD_B Set trigger polarity for Module B.
pol	AN9900_TRIGPOL	AN9900_TRIGPOL_RISING Trigger on rising edge. AN9900_TRIGPOL_FALLING Trigger on falling edge. AN9900_TRIGPOL_HILEVEL =AN9900_TRIGPOL_RISING Level sensitive interpretation. AN9900_TRIGPOL_LOLEVEL =AN9900_TRIGPOL_FALLING If trigger mode is gated.
<OUTPUT>		
NONE		
<RETURN>		
= 0 > 0 < 0	ViStatus	"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.5.6 Get Trigger Polarity an9900_getTrigPolarity

Get trigger polarity. whether the trigger is "edge" or "level" sensitive depends on the mode. Gated trigger is level sensitive; all others are edge-sensitive.

Function Prototype

ViStatus _VI_FUNC **an9900_getTrigPolarity** (ViSession **vi**, AN9900_MODULE **mod**, AN9900_TRIGPOL ***pol**);

an9900_getTrigPolarity		
Parameters	**Variable Type**	**Description**
<INPUT>		
vi	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
mod	AN9900_MODULE	AN9900_MOD_A Get trigger polarity set for Module A. AN9900_MOD_B Get trigger polarity set for Module B.
<OUTPUT>		
***pol**	AN9900_TRIGPOL	AN9900_TRIGPOL_RISING Trigger on rising edge. AN9900_TRIGPOL_FALLING Trigger on falling edge. AN9900_TRIGPOL_HILEVEL =AN9900_TRIGPOL_RISING Level sensitive interpretation. AN9900_TRIGPOL_LOLEVEL =AN9900_TRIGPOL_FALLING If trigger mode is gated.
<RETURN>		
= 0 > 0 < 0	ViStatus	"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.5.7 Set Trigger Threshold an9900_setTrigThreshold

This function sets the trigger threshold voltage for the specified module.

Function Prototype

ViStatus _VI_FUNC **an9900_setTrigThreshold** (ViSession **vi**, AN9900_MODULE **mod**,
ViReal64 **thresh**);

an9900_setTrigThreshold		
Parameters	**Variable Type**	**Description**
<INPUT>		
vi	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
mod	AN9900_MODULE	AN9900_MOD_A Set trigger threshold for Module A. AN9900_MOD_B Set trigger threshold for Module B.
thresh	ViReal64	
<OUTPUT>		
NONE		
<RETURN>		
= 0 > 0 < 0	ViStatus	"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.5.8 Get Trigger Threshold an9900_getTrigThreshold

This function gets the trigger threshold voltage for the specified module.

Function Prototype

ViStatus _VI_FUNC **an9900_getTrigThreshold** (ViSession **vi**, AN9900_MODULE **mod**,
ViReal64 ***thresh**);

an9900_getTrigThreshold		
Parameters	**Variable Type**	**Description**
<INPUT>		
vi	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
mod	AN9900_MODULE	AN9900_MOD_A Get trigger threshold setting for Module A. AN9900_MOD_B Get trigger threshold setting for Module B.
<OUTPUT>		
***thresh**	ViReal64	
<RETURN>		
= 0	ViStatus	"VI_SUCCESS"
> 0		Warning: The function completed, but an exception condition occurred which may require attention.
< 0		Error: The function did not complete successfully.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.5.9 Trigger Source Control an9900_triggerSourceControl

This function changes the trigger configuration for each module.

Function Prototype

ViStatus **an9900_triggerSourceControl** (ViSession **instrumentHandle**,
ViUInt16 **modAtriggerInput**,
ViUInt16 **modAtriggerOutput**,
ViUInt16 **modBtriggerInput**,
ViUInt16 **modBtriggerOutput**);

an9900_triggerSourceControl		
Parameters	**Variable Type**	**Description**
<INPUT>		
instrumentHandle	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
modAtriggerInput	ViUInt16	Selects the trigger input for Module A. 0x0 Register Trigger 0x4 ECL0 0x5 ECL1 0x6 FPA 0x7 FPB 0x8 TTL0 0x9 TTL1 0xA TTL2 0xB TTL3 0xC TTL4 0xD TTL5 0xE TTL6 0xF TTL7
modAtriggerOuput	ViUInt16	Selects the trigger output for Module A. 0x0 Not Driving Trigger 0x6 ECL0 0x7 ECL1 0x8 TTL0 0x9 TTL1 0xA TTL2 0xB TTL3 0xC TTL4 0xD TTL5 0xE TTL6 0xF TTL7
modBtriggerInput	ViUInt16	Selects the trigger input for Module B. 0x0 Register Trigger 0x4 ECL0 0x5 ECL1 0x6 FPA 0x7 FPB 0x8 TTL0 0x9 TTL1 0xA TTL2 0xB TTL3 0xC TTL4 0xD TTL5 0xE TTL6 0xF TTL7
modBtriggerOutput	ViUInt16	Selects the trigger output for Module B. 0x0 Not Driving Trigger 0x6 ECL0 0x7 ECL1 0x8 TTL0 0x9 TTL1 0xA TTL2 0xB TTL3 0xC TTL4 0xD TTL5 0xE TTL6 0xF TTL7
<OUTPUT>		
None	None	None
<RETURN>		
= 0 > 0 < 0		"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully. NOTE: Use the an902_error_message() function to get a textual description of errors and warnings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.5.10 Trigger Source Control Query an9900_triggerSourceControl_Q

This function queries trigger information for each module.

Function Prototype

ViStatus **an9900_triggerSourceControl_Q** (ViSession **instrumentHandle**,
ViUInt16 ***modAtriggerInput**,
ViUInt16 ***modAtriggerOutput**,
ViUInt16 ***modBtriggerInput**,
ViUInt16 ***modBtriggerOutput**);

an9900_triggerSourceControl_Q		
Parameters	**Variable Type**	**Description**
<INPUT>		
instrumentHandle	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
<OUTPUT>		
*modAtriggerInput	ViUInt16 (passed by reference)	Returns the trigger input for Module A.
*modAtriggerOuput	ViUInt16 (passed by reference)	Returns the trigger output for Module A.
*modBtriggerInput	ViUInt16 (passed by reference)	Returns the trigger input for Module B.
*modBtriggerOutput	ViUInt16 (passed by reference)	Returns the trigger output for Module B.
<RETURN>		
= 0 > 0 < 0		"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully. NOTE: Use the an902_error_message() function to get a textual description of errors and warnings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.5.11 Trigger Mode an9900_triggerMode

This function sets the trigger mode for each module.

Function Prototype

ViStatus **an9900_triggerMode** (ViSession **instrumentHandle**,
ViUInt16 **gateEnableA**, ViUInt16 **slopeA**,
ViUInt16 **softTriggerA**, ViUInt16 **gateEnableB**,
ViUInt16 **slopeB**, ViUInt16 **softTriggerB**,
ViUInt16 **softAdvA**, ViUInt16 **softAdvB**);

an9900_triggerMode		
Parameters	**Variable Type**	**Description**
<INPUT>		
instrumentHandle	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
gateEnableA	ViUInt16	Selects between edge, or gated trigger. 1 Gated Trigger 0 Edge Trigger
slopeA	ViUInt16	Selects between falling or rising edge trigger. 1 Rising Edge Trigger 0 Falling Edge Trigger
softTriggerA	ViUInt16	Specifies the condition of the soft trigger. 1 Trigger 0 Inactive
gateEnableB	ViUInt16	Selects between edge, or gated trigger. 1 Gated Trigger 0 Edge Trigger
slopeB	ViUInt16	Selects between falling or rising edge trigger. 1 Rising Edge Trigger 0 Falling Edge Trigger
softTriggerB	ViUInt16	Specifies the condition of the soft trigger. 1 Trigger 0 Inactive
softAdvA	ViUInt16	Selects between advance or inactive soft-advance. 1 Advance 0 Inactive
softAdvB	ViUInt16	Selects between advance or inactive soft-advance. 1 Advance 0 Inactive
<OUTPUT>		
None	None	None
<RETURN>		
= 0 > 0 < 0		"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully. NOTE: Use the an902_error_message() function to get a textual description of errors and warnings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.5.12 Trigger Mode Query an9900_triggerMode_Q

This function queries trigger mode information for each module.

Function Prototype

ViStatus **an9900_triggerMode_Q** (ViSession **instrumentHandle**,
ViUInt16 ***gateEnableA**, ViUInt16 ***slopeA**,
ViUInt16 ***softTriggerA**, ViUInt16 ***gateEnableB**,
ViUInt16 ***slopeB**, ViUInt16 ***softTriggerB**,
ViUInt16 ***softAdvA**, ViUInt16 ***softAdvB**);

an9900_triggerMode_Q		
Parameters	**Variable Type**	**Description**
<INPUT>		
instrumentHandle	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
<OUTPUT>		
*gateEnableA	ViUInt16 (passed by reference)	Outputs edge, or gated trigger. 1 Gated Trigger 0 Edge Trigger
*slopeA	ViUInt16 (passed by reference)	Outputs falling or rising edge trigger. 1 Rising Edge Trigger 0 Falling Edge Trigger
*softTriggerA	ViUInt16 (passed by reference)	Outputs the condition of the soft trigger. 1 Trigger 0 Inactive
*gateEnableB	ViUInt16 (passed by reference)	Outputs edge, or gated trigger. 1 Gated Trigger 0 Edge Trigger
*slopeB	ViUInt16 (passed by reference)	Outputs falling or rising edge trigger. 1 Rising Edge Trigger 0 Falling Edge Trigger
*softTriggerB	ViUInt16 (passed by reference)	Outputs the condition of the soft trigger. 1 Trigger 0 Inactive
*softAdvA	ViUInt16 (passed by reference)	Outputs advance or inactive soft-advance. 1 Advance 0 Inactive
*softAdvB	ViUInt16 (passed by reference)	Outputs advance or inactive soft-advance. 1 Advance 0 Inactive
<RETURN>		
= 0 > 0 < 0		"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully. NOTE: Use the an902_error_message() function to get a textual description of errors and warnings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.6 Clock

This class provides functions to communicate with the instrument to set the clock frequency to less than .05% from the desired frequency, and to return important parameters and frequency about the

- clock
- N counter,
- Low Speed Divider,
- High Speed Divider, and the
- value for the Time Base Select.

These functions also route no clock, time base clock, Channel A or Channel B clock information to the daughter boards.

Additionally, functions are provided which

- set the value of the R and N counter,
- sets the value of the High Speed and Low Speed Dividers,
- specifies the threshold values for Clock A and Clock B,
- sets the values needed to set the Trigger A and Trigger B zero crossing thresholds,
- configures the FO_LD control bits,
- facilitates power down and counter reset, and
- controls the TB select lines on the Time Base control register.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.6.1 Set Clock Source an9900_setClockSource

This function sets the clock source for the specified module.

Function Prototype

ViStatus _VI_FUNC **an9900_setClockSource** (ViSession **vi**, AN9900_MODULE **mod**,
AN9900_CLKSRC **src**);

an9900_setClockSource		
Parameters	**Variable Type**	**Description**
<INPUT>		
vi	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
mod	AN9900_MODULE	AN9900_MOD_A Set clock source for Module A. AN9900_MOD_B Set clock source for Module B.
src	AN9900_CLKSRC	AN9900_CLKSRC_NONE No clock. No wave will be generated. AN9900_CLKSRC_INT Internal clock, with programmable frequency. AN9900_CLKSRC_EXT_A External channel 'A' clock. AN9900_CLKSRC_EXT_B External channel 'B' clock.
<OUTPUT>		
NONE		
<RETURN>		
= 0 > 0 < 0	ViStatus	"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.6.2 Get Clock Source — an9900_getClockSource

This function gets the clock source for the specified module.

Function Prototype

ViStatus _VI_FUNC **an9900_getClockSource** (ViSession **vi**, AN9900_MODULE **mod**,
AN9900_CLKSRC ***src**);

an9900_getClockSource		
Parameters	**Variable Type**	**Description**
<INPUT>		
vi	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
mod	AN9900_MODULE	AN9900_MOD_A Get clock source set for Module A. AN9900_MOD_B Get clock source set for Module B.
<OUTPUT>		
***src**	AN9900_CLKSRC	AN9900_CLKSRC_NONE No clock. No wave will be generated. AN9900_CLKSRC_INT Internal clock, with programmable frequency. AN9900_CLKSRC_EXT_A External channel 'A' clock. AN9900_CLKSRC_EXT_B External channel 'B' clock.
<RETURN>		
= 0 > 0 < 0	ViStatus	"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.6.3 Set Internal Clock Reference an9900_setIntClockRef

This function specifies whether the internal sample rate generator uses its internal reference oscillator or uses an external 10MHz reference applied to clock *B*.

**Note: The external reference clock must be applied to the channel B input and its threshold may be set using an9900_setClockThreshold. Note that this may interact with external clock selection for the other module. It is up to the programmer to be aware of this.*

Function Prototype

ViStatus _VI_FUNC **an9900_setIntClockRef** (ViSession **vi**, AN9900_CLKREF **ref**);

an9900_setIntClockRef		
Parameters	**Variable Type**	**Description**
<INPUT>		
vi	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
ref	AN9900_CLKREF	Internal clock reference: AN9900_CLKREF_INT — Use internal reference AN9900_CLKREF_EXT — Use external 10MHz reference
<OUTPUT>		
None	None	None
<RETURN>		
= 0 > 0 < 0	ViStatus	"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.6.4 Get Internal Clock Reference an9900_getIntClockRef

This function gets clock reference selection for 9900 internal clock synthesizer.

Function Prototype

ViStatus _VI_FUNC **an9900_setIntClockRef** (ViSession **vi**, AN9900_CLKREF ***ref**);

an9900_setIntClockRef		
Parameters	**Variable Type**	**Description**
<INPUT>		
vi	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
<OUTPUT>		
***ref**	AN9900_CLKREF (passed by reference)	Internal clock reference: AN9900_CLKREF_INT — Use internal reference AN9900_CLKREF_EXT — Use external 10MHz reference
<RETURN>		
= 0	ViStatus	"VI_SUCCESS"
> 0		Warning: The function completed, but an exception condition occurred which may require attention.
< 0		Error: The function did not complete successfully.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.6.5 Set Clock Threshold an9900_setClockThreshold

This function sets the external clock threshold voltage. Each front-panel clock input connector has a programmable threshold comparator. The clock source multiplexer comes after this comparator, therefore either clock input can be used for either or both modules.

Function Prototype

ViStatus _VI_FUNC **an9900_setClockThreshold** (ViSession **vi**, AN9900_EXTCLK **mod**, ViReal64 **thresh**);

an9900_setClockThreshold		
Parameters	**Variable Type**	**Description**
<INPUT>		
vi	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
mod	AN9900_EXTCLK	Specifies an external clock input. AN9900_EXTCLK_A External clock 'A' input AN9900_EXTCLK_B External clock 'B' input
thresh	ViReal64	
<OUTPUT>		
NONE		
<RETURN>		
= 0 > 0 < 0	ViStatus	"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.6.6 Get Clock Threshold an9900_getClockThreshold

Get external clock threshold voltage. Each front-panel clock input connector has a programmable threshold comparator. The clock source multiplexer comes after this comparator, therefore either the clock input can be used for either or both modules.

Function Prototype

ViStatus _VI_FUNC **an9900_getClockThreshold** (ViSession **vi**, AN9900_EXTCLK **mod**, ViReal64 ***thresh**);

an9900_getClockThreshold		
Parameters	**Variable Type**	**Description**
<INPUT>		
vi	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
mod	AN9900_EXTCLK	Specifies for which external clock input to check threshold. AN9900_EXTCLK_A External clock 'A' input AN9900_EXTCLK_B External clock 'B' input
<OUTPUT>		
***thresh**	ViReal64	
<RETURN>		
= 0	ViStatus	"VI_SUCCESS"
> 0		Warning: The function completed, but an exception condition occurred which may require attention.
< 0		Error: The function did not complete successfully.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.6.7 Set Sample Rate an9900_setSampleRate

This function accepts a floating point number in hertz and sets the internal clock to the nearest available value.

Function Prototype

ViStatus _VI_FUNC **an9900_setSampleRate** (ViSession **vi**, ViReal64 **rate**);

an9900_setSampleRate		
Parameters	**Variable Type**	**Description**
<INPUT>		
vi	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
rate	ViReal64	10,000,000 – 95.36743Hz
<OUTPUT>		
NONE		
<RETURN>		
= 0 > 0 < 0	ViStatus	"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully.

5.6.8 Get Sample Rate an9900_getSampleRate

This function returns the actual sample rate value based on instrument registers.

Function Prototype

ViStatus _VI_FUNC **an9900_getSampleRate** (ViSession **vi**, ViReal64 ***rate**);

an9900_getSampleRate		
Parameters	**Variable Type**	**Description**
<INPUT>		
vi	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
<OUTPUT>		
***rate**	ViReal64	
<RETURN>		
= 0 > 0 < 0	ViStatus	"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.6.9 Await PLL Lock an9900_awaitPLLLock

This function waits up to *timeout* seconds for the PLL to lock. Returns VI_SUCCESS if it locks, or AN9900_ERROR_PLL_LOCK_TIMEOUT if it is timed out. It may be used after AN9900_setSampleRate to ascertain that the PLL has locked. AN9900_setSampleRate does not automatically wait for lock, since to do so would slow down the calling program unnecessarily in most cases.

Function Prototype

ViStatus _VI_FUNC **an9900_awaitPLLLock** (ViSession **vi**, ViUInt16 **timeout**);

an9900_awaitPLLLock		
Parameters	**Variable Type**	**Description**
<INPUT>		
vi	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
timeout	VIUInt16	> 100 micro-seconds
<OUTPUT>		
NONE		
<RETURN>		
= 0 > 0 < 0	ViStatus	"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.6.10 Clock Frequency an9900_clockFreq

This function sets the clock frequency to less than 0.5% from the desired frequency.

Function Prototype

ViStatus **an9900_clockFreq** (ViSession **instrumentHandle**,
ViReal64 **frequency**);

an9900_clockFreq		
Parameters	**Variable Type**	**Description**
<INPUT>		
instrumentHandle	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
frequency	ViReal64	Desired frequency for the clock.
<OUTPUT>		
None	None	None
<RETURN>		
= 0 > 0 < 0		"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully. NOTE: Use the an902_error_message() function to get a textual description of errors and warnings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.6.11 Clock Frequency Query an9900_clockFreq_Q

This function returns important parameters about the clock. Returns the clock frequency, N counter, Low Speed Divider, High Speed Divider and the value f of the Time Base Select.

Function Prototype

ViStatus **an9900_clockFreq_Q** (ViSession **instrumentHandle**,
ViReal64 ***frequency**,
ViUInt32 ***NCounter**,
ViUInt32 ***lowSpeedDivider**,
ViUInt16 ***highSpeedDivider**,
ViUInt16 ***timeBaseSelect**);

an9900_clockFreq_Q		
Parameters	**Variable Type**	**Description**
<INPUT>		
instrumentHandle	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
<OUTPUT>		
*frequency	ViReal64 (passed by reference)	Returns the actual clock frequency.
*NCounter	ViUInt32 (passed by reference)	Returns the value of the N counter.
*lowSpeedDivider	ViUInt32 (passed by reference)	Returns the value of the Low Speed Divider
*highSpeedDivider	ViUInt16 (passed by reference)	Returns the value of the High Speed Divider
*timeBaseSelect	ViUInt16 (passed by reference)	Returns the value of the Time Base Select lines.
<RETURN>		
= 0 > 0 < 0		"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully. NOTE: Use the an902_error_message() function to get a textual description of errors and warnings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.6.12 Clock Control an9900_clockControl

This function routes no clock, time base clock, Channel A or Channel B clock to daughter boards.

Function Prototype

ViStatus **an9900_clockControl** (ViSession **instrumentHandle**,
ViUInt16 **EXTCLKAenable**,
ViUInt16 **EXTCLKBenable**,
ViUInt16 **CHACLKSelect**,
ViUInt16 **CHBCLKSelect**);

an9900_clockControl		
Parameters	**Variable Type**	**Description**
<INPUT>		
instrumentHandle	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
EXTCLKAenable	ViUInt16	Enables the signal present on the input of Clock A to pass through the Channel A threshold comparator to the Clock selector multiplexers. 1 Enable 0 Disable
EXTCLKBenable	ViUInt16	Enables the signal present on the input of Clock B to pass through the Channel B threshold comparator to the Clock selector multiplexers. 1 Enable 0 Disable
CHACLKSelect	ViUInt16	Routes the clock to daughter board A. 0 Select No Clock 1 Select Internal Time Base Clock 2 Select Channel A Clock 3 Select Channel B Clock
CHBCLKSelect	ViUInt16	Routes the clock to daughter board B. 0 Select No Clock 1 Select Internal Time Base Clock 2 Select Channel A Clock 3 Select Channel B Clock
<OUTPUT>		
None	None	None
<RETURN>		
= 0 > 0 < 0		"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully. NOTE: Use the an902_error_message() function to get a textual description of errors and warnings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.6.13 Clock Control Query an9900_clockControl_Q

This function returns clock routing information from the daughter boards.

Function Prototype

ViStatus **an9900_clockControl_Q** (ViSession **instrumentHandle**,
ViUInt16 ***EXTCLKAenable**,
ViUInt16 ***EXTCLKBenable**,
ViUInt16 ***CHACLKSelect**,
ViUInt16 ***CHBCLKSelect**);

an9900_clockControl_Q		
Parameters	**Variable Type**	**Description**
<INPUT>		
instrumentHandle	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
<OUTPUT>		
*EXTCLKAenable	ViUInt16 (passed by reference)	Signal present on the input of Clock A enable.
*EXTCLKBenable	ViUInt16 (passed by reference)	Signal present on the input of Clock B enable.
*CHACLKSelect	ViUInt16 (passed by reference)	Clock routing to daughter board A.
*CHBCLKSelect	ViUInt16 (passed by reference)	Clock routing to daughter board B.
<RETURN>		
= 0 > 0 < 0		"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully. NOTE: Use the an902_error_message() function to get a textual description of errors and warnings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.6.14 R Counter an9900_rCounter

This function sets the value of the R counter.

Function Prototype

ViStatus **an9900_rCounter** (ViSession **instrumentHandle**,
ViUInt16 **counter**);

an9900_rCounter		
Parameters	**Variable Type**	**Description**
<INPUT>		
instrumentHandle	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
counter	ViUInt16	Sets the value of the R counter.
<OUTPUT>		
None	None	None
<RETURN>		
= 0 > 0 < 0		"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully. NOTE: Use the an902_error_message() function to get a textual description of errors and warnings.

5.6.15 R Counter Query an9900_rCounter_Q

This function returns the value of the R counter.

Function Prototype

ViStatus **an9900_rCounter_Q** (ViSession **instrumentHandle**,
ViUInt16 ***counter**);

an9900_rCounter_Q		
Parameters	**Variable Type**	**Description**
<INPUT>		
instrumentHandle	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
<OUTPUT>		
*counter	ViUInt16 (passed by reference)	Returns the bit mask for the R counter.
<RETURN>		
= 0 > 0 < 0		"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully. NOTE: Use the an902_error_message() function to get a textual description of errors and warnings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.6.16 N Counter an9900_NCounter

This function sets the value of the N counter.

Function Prototype

ViStatus **an9900_NCounter** (ViSession **instrumentHandle**,
ViUInt16 **Ncounter**);

an9900_NCounter		
Parameters	**Variable Type**	**Description**
<INPUT>		
instrumentHandle	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
NCounter	ViUInt16	Sets the value of the N counter. // valid range? //
<OUTPUT>		
None	None	None
<RETURN>		
= 0 > 0 < 0		"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully. NOTE: Use the an902_error_message() function to get a textual description of errors and warnings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.6.17 High Speed Divider an9900_highSpeedDivider

This function sets the value of the High Speed Divider.

Function Prototype

ViStatus **an9900_highSpeedDivider** (ViSession **instrumentHandle**,
ViUInt16 **divider**);

an9900_highSpeedDivider		
Parameters	**Variable Type**	**Description**
<INPUT>		
instrumentHandle	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
divider	ViUInt16	Bit mask of the High Speed Divider value.
<OUTPUT>		
None	None	None
<RETURN>		
= 0 > 0 < 0		"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully. NOTE: Use the an902_error_message() function to get a textual description of errors and warnings.

5.6.18 Low Speed Divider an9900_lowSpeedDivider

This function sets the value of the Low Speed Divider.

Function Prototype

ViStatus **an9900_lowSpeedDivider** (ViSession **instrumentHandle**,
ViUInt32 **divider**);

an9900_lowSpeedDivider		
Parameters	**Variable Type**	**Description**
<INPUT>		
instrumentHandle	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
divider	ViUInt32	Bit mask of the Low Speed Divider value.
<OUTPUT>		
None	None	None
<RETURN>		
= 0 > 0 < 0		"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully. NOTE: Use the an902_error_message() function to get a textual description of errors and warnings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.6.19 Clock A/B Threshold an9900_clockThreshold

This function specifies the threshold values for Clock A and B.

Function Prototype

ViStatus **an9900_clockThreshold** (ViSession **instrumentHandle**,
ViReal64 **CLKAThreshold**,
ViReal64 **CLKBThreshold**);

an9900_clockThreshold		
Parameters	**Variable Type**	**Description**
<INPUT>		
instrumentHandle	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
CLKAThreshold	ViReal64	Specifies the Clock A zero crossing threshold.
CLKBThreshold	ViReal64	Specifies the Clock B zero crossing threshold.
<OUTPUT>		
None	None	None
<RETURN>		
= 0 > 0 < 0		"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully. NOTE: Use the an902_error_message() function to get a textual description of errors and warnings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.6.20 Clock A/B Threshold Query — an9900_clockThreshold_Q

Returns the threshold values for clocks A and B.

Function Prototype

ViStatus **an9900_clockThreshold_Q** (ViSession **instrumentHandle**,
ViReal64 ***CLKAThreshold**,
ViReal64 ***CLKBThreshold**);

an9900_clockThreshold_Q		
Parameters	**Variable Type**	**Description**
<INPUT>		
instrumentHandle	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
<OUTPUT>		
*CLKAThreshold	ViReal64 (passed by reference)	Returns the bit mask of the Clock A zero crossing threshold.
*CLKBThreshold	ViReal64 (passed by reference)	Returns the bit mask of the Clock B zero crossing threshold.
<RETURN>		
= 0 > 0 < 0		"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully. NOTE: Use the an902_error_message() function to get a textual description of errors and warnings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.6.21 Trigger A/B Threshold an9900_triggerThreshold

This function sets the values needed to set the Trigger A and Trigger B zero crossing thresholds.

Function Prototype

ViStatus **an9900_triggerThreshold** (ViSession **instrumentHandle**,
ViReal64 **triggerAThreshold**,
ViReal64 **triggerBThreshold**);

an9900_triggerThreshold		
Parameters	**Variable Type**	**Description**
<INPUT>		
instrumentHandle	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
triggerAThreshold	ViReal64	Bit mask with Trigger A zero crossing threshold.
triggerBThreshold	ViReal64	Bit mask with Trigger B zero crossing threshold.
<OUTPUT>		
None	None	None
<RETURN>		
= 0 > 0 < 0		"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully. NOTE: Use the an902_error_message() function to get a textual description of errors and warnings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.6.22 Trigger A/B Threshold Query an9900_triggerThreshold_Q

This function returns the threshold values for the Trigger A and Trigger B.

Function Prototype

ViStatus **an9900_triggerThreshold_Q** (ViSession **instrumentHandle**,
ViReal64 ***triggerAThreshold**,
ViReal64 ***triggerBThreshold**);

an9900_triggerThreshold_Q		
Parameters	**Variable Type**	**Description**
<INPUT>		
instrumentHandle	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
<OUTPUT>		
*triggerAThreshold	ViReal64 (passed by reference)	Bit mask with Trigger A zero crossing threshold.
*triggerBThreshold	ViReal64 (passed by reference)	Bit mask with Trigger B zero crossing threshold.
<RETURN>		
= 0 > 0 < 0		"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully. NOTE: Use the an902_error_message() function to get a textual description of errors and warnings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.6.23 Function Latch an9900_functionLatch

This function configures the FO_LD control bits, the power down, and the counter reset.

Function Prototype

ViStatus **an9900_functionLatch** (ViSession **instrumentHandle**,
ViUInt16 **FO_LD**,
ViUInt16 **counterReset**);

an9900_functionLatch		
Parameters	**Variable Type**	**Description**
<INPUT>		
instrumentHandle	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
FO_LD	ViUInt16	Selects the FO_LD configuration. 0 Tri State 1 Digital Lock Detect 2 N Divider Output 3 Active High 4 R Divider Output 5 N Channel Open Drain Lock Detect 6 Serial Data Output 7 Active Low
counterReset	ViUInt16	If enabled, will set the Counter reset bit ON and then to OFF. 1 Enable 0 Disable
<OUTPUT>		
None	None	None
<RETURN>		
= 0 > 0 < 0		"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully. NOTE: Use the an902_error_message() function to get a textual description of errors and warnings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.6.24 Time Base Control an9900_timeBaseControl

This function controls the TB select lines on the Time Base Control register.

Function Prototype

ViStatus **an9900_timeBaseControl** (ViSession **instrumentHandle**,
ViUInt16 **frequencySelect**, ViUInt16 **internalReference**,
ViUInt16 **enablePLL**, ViUInt16 **externalReferenceSel**);

an9900_timeBaseControl		
Parameters	**Variable Type**	**Description**
<INPUT>		
instrumentHandle	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
frequencySelect	ViUInt16	Specifies the Frequency select options for the clock. 0 Disable Time Base Clock 1 Enables LSDIV 2 Enables HSDIV FF 3 Enables HSDIV Divider
internalReference	ViUInt16	Enable Internal Reference. 1 Select Internal Ref. 0 Clear Internal Ref.
enablePLL	ViUInt16	Set or clear the PLL. 1 Enable PLL 0 Disable PLL
externalReferenceSel	ViUInt16	Set or clear the External Reference. 1 Set External Reference 0 Clear External Reference
<OUTPUT>		
None	None	None
<RETURN>		
= 0 > 0 < 0		"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully. NOTE: Use the an902_error_message() function to get a textual description of errors and warnings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.7 Utility

This class of functions provides lower level functions to communicate with the instrument, and to change instrument parameters.

5.7.1 Get Error Text an9900_getErrorText

This function converts a ViStatus constant into a text string. *Desc* must have room for up to 256 chars.

Function Prototype

ViStatus _VI_FUNC **an9900_getErrorText** (ViSession **vi**, ViStatus **status**, ViChar **desc**);

an9900_getErrorText		
Parameters	**Variable Type**	**Description**
<INPUT>		
vi	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
status	ViStatus	
desc	ViChar	
<OUTPUT>		
NONE		
<RETURN>		
= 0 > 0 < 0	ViStatus	"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.7.2 Mask Write to Instrument an9900_maskWriteInst

This function writes data to an instrument register to modify device settings. It performs a read and then a write to preserve actual values.

Function Prototype

ViStatus **an9900_maskWriteInst** (ViSession **instrumentHandle**,
ViUInt16 **registerNumber**,
ViUInt16 **mask**,
ViUInt16 **value**);

an9900_maskWriteInst		
Parameters	**Variable Type**	**Description**
<INPUT>		
instrumentHandle	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
registerNumber	ViUInt16	Specifies which register is to be written. Valid range: 0x00 – 0x3F (Default value = 0x00).
mask	ViUInt16	Masks the data (in hex) to write to a register. Valid range: 0x00 – 0xFF (Default value = 0x00).
value	ViUInt16	Writes data (in hex0 to a register). Valid range: 0x00 – 0xFFFF (Default value = 0x00).
<OUTPUT>		
None	None	None
<RETURN>		
= 0 > 0 < 0		"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully. NOTE: Use the an902_error_message() function to get a textual description of errors and warnings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.7.3 Write to Instrument an9900_writeInst

This function writes data to an instrument register to modify device settings.

Function Prototype

ViStatus **an9900_writeInst** (ViSession **instrumentHandle**,
ViUInt16 **registerNumber**,
ViUInt16 **value**);

an9900_writeInst		
Parameters	**Variable Type**	**Description**
<INPUT>		
instrumentHandle	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
registerNumber	ViUInt16	Specifies register to be written. Valid range: 0x00 - 0x3F. (Default is 0x00).
value	ViUInt16	This control writes data (in hex) to a register. Valid range: 0x00-0xFFFF. (Default is 0x00).
<OUTPUT>		
None	None	None
<RETURN>		
= 0 > 0 < 0		"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully. NOTE: Use the an902_error_message() function to get a textual description of errors and warnings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.7.4 Read from Instrument an9900_readInst

Given a hexadecimal register number and a bit number, this function returns the value of the register and bit.

Function Prototype

ViStatus **an9900_readInst** (ViSession **instrumentHandle**,
ViUInt16 **registerNumber**,
ViUInt16 ***value**);

an9900_readInst		
Parameters	**Variable Type**	**Description**
<INPUT>		
instrumentHandle	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
registerNumber	ViUInt16	Number, in hex, of the register to be accessed. Valid range: 0x00 - 0x2F. (Default is 0x00).
<OUTPUT>		
*value	ViUInt16 (passed by reference)	Displays the hex value of word queried. Valid Range: 0x00 – 0xFFFF.
<RETURN>		
= 0 > 0 < 0		"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully. NOTE: Use the an902_error_message() function to get a textual description of errors and warnings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.7.5 Reset an9900_reset

This function resets the instrument to a known state and sets any necessary programmatic variables necessary for the operation of the instrument driver. It will disable all output ports and set all other registers to default values. Reads and writes to other module registers will not transfer valid data during this time.

Function Prototype

ViStatus **an9900_reset** (ViSession **instrumentHandle**);

an9900_reset		
Parameters	**Variable Type**	**Description**
<INPUT>		
instrumentHandle	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
<OUTPUT>		
None	None	None
<RETURN>		
= 0 > 0 < 0		"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully. NOTE: Use the an902_error_message() function to get a textual description of errors and warnings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.7.6 Error Message an9900_errorMessage

This function takes the Status Code returned by the instrument driver functions, interprets it, and returns it as a user-readable string.

Function Prototype

ViStatus **an9900_errorMessage** (ViSession **instrumentHandle**,
ViStatus **errorCode**,
ViChar **errorMessage[]**);

an9900_errorMessage		
Parameters	**Variable Type**	**Description**
<INPUT>		
instrumentHandle	ViSession	This control accepts the Instrument Handle returned by the Initialize function to select the desired instrument driver session.
errorCode	ViStatus	This control accepts the Status Code returned from the instrument driver functions. Default Value: 0 – VI_SUCCESS
<OUTPUT>		
errorMessage	ViChar[]	This control returns the interpreted Status Code as a user-readable message string. Note: The array must contain at least 256 elements – ViChar[256].
<RETURN>		
= 0 > 0 < 0		"VI_SUCCESS" Warning: The function completed, but an exception condition occurred which may require attention. Error: The function did not complete successfully. NOTE: Use the an902_error_message() function to get a textual description of errors and warnings.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

6 DBS9900 VME / VXI Bus Interface - Registers

The VME / VXI Interface conforms to **Version 1.4** of the VXI specification.

Per VXI definition, the DBS9900 is a *register based servant* device. *Message based protocols* will not be supported. *Dynamic Configuration* will be supported. *MODID slot detection* and *TTL backplane triggers* will be supported. The standard *Interface Register Offset Mapping* will be supported. *Unaligned Transfers* and *Read-Modify-Write Capability* will not be supported. MODID and TTL Trigger pinouts are per VXI Specification. Operation of the SYSFAIL and INHIBIT are also per section 3.2.1.1.2 of the VXI Specification.

In terms of the VME specifications, the DBS9900 is an A32: A24: A16: D16: D32: Slave that will support the following:

- 16-bit word transfers.
- 32-bit word transfers
- Standard (A24) 16 Mbyte address space (AM codes 39, 3A, 3D, and 3E) jumper selectable.
- Extended (A32) 4 Gbyte address space (AM codes 09, 0A, 0D, and 0E) jumper selectable.
- Short (A16) 64K address space (AM codes 29 and 2D)
- Block Transfers (BLT) for memory using AM codes 3B, 3F, 0F, and 0B
- Maskable interrupt generation on one of 7 register programmable levels
- The DBS9900 *does not* support Bus Request or Bus Arbitration.
- The DBS9900 *is never* a Bus Master Device.

The DBS9900 allows for two independent daughter board modules, accessed via module interface connectors (on the DBS9900). The DBS9900 is ONE DEVICE as defined by VXI conventions, but is actually two independent devices functionally.

The DBS9900 supports D32 transfers to the daughter board module registers/memory only (0x20 – 0x3E). D32 transfers are not supported to the registers physically located on the DBS9900 mother board (0x00 – 0x1E). D16 transfers must be used to access these mother board registers.

The DBS9900 can generate interrupts, but CANNOT request data bus operations as a bus master. Interrupts may be generated on one of seven levels. Interrupt sources are maskable and readable. The Interrupt Level is register programmable, and the interrupt will be cleared during the interrupt service STATUS ID cycle, in accordance with the VME-defined *ROAK* protocol. Because the DBS9900 can accommodate two daughter board modules, the interrupt service routine checks and clears *both* interrupt registers on the rider modules when servicing an interrupt.

The DBS9900 registers are accessed in A16 mode at a base address set by Dynamic Configuration.

The A16 addresses from 0x00 to 0x1E reside on the DBS9900. The A16 addresses from 0x20 to 0x3E reside on each module and the functions are defined in the individual module specs. Each module can be accessed by switching in its DEVICE dependent Registers into this space. This is done by setting a bit in the Module / Relay Select Register. This register also controls relays to program the CAL front panel connectors.

The data and address busses will be static if the VXI bus is not accessing registers on the module. The interface clock will operate in a burst mode, only during VXI bus cycles involving the

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

DBS9900 module. The sample clocks will be double multiplexed on the individual modules to reduce analog noise.

The Module RUN Control Register contains a RUN (or GO) bit for each module installed on the DBS9900. This allows simultaneous control of the module functions without the need to address each module individually.

Table 1 - Interface Register Map

Base +	Description	R/W	# Bits
0x00	ID Register (e.g. 0xCFF5) / Logical Address Register	R/W	16
0x02	Device Type (0x***m***21***r***)	R only	16
0x04	Status / Control Register	R/W	16
0x06	Memory Offset Register	R/W	*
0x08	Memory Attribute Register (0xEE70)	R only	16
0x0A	Module Select / RELAY Register	R/W	16
0x0C	Trigger Source Control Register	R/W	16
0x0E	Trigger Mode Control Register	R/W	6
0x10	Interrupt Level Select Register	R/W	3
0x12	Interrupt Vector Register	R/W	10
0x14	PLL 1 Data Register (MSB)	R/W	16
0x16	PLL 0 Data Register (LSB)	R/W	5
0x18	PLL Address Register	R/W	4
0x1A	JTAG Programming Register	R/W	2
0x1C	Not Used		
0x1E	Not Used		
0x20 to 0x3E	Module DEVICE Dependent Registers		

* Length determined by memory requirements

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

6.1 ID REGISTER 0x00 Read Only Logical Address Register

The ID REGISTER indicates the ANALOGIC manufacturer's code and the status of the Mode Select Jumper (A32 or A24).

Jumper JP1	MODE	ID REGISTER Value
Pins 1&2 jumpered	A32	0xDFF5
Pins 2&3 jumpered	A24	0xCFF5

The Logical Address Register shares the same address as the ID Register and may only be loaded during the Dynamic configuration process. The contents of the Logical Address Register cannot be read back. // say more here about logical address? //

6.2 DEVICE TYPE REGISTER

The DEVICE TYPE REGISTER (0x02), READ only, indicates the type of memory requirements as defined in the VXI specifications: the device type code, and the revision level of the PCB. The value will be 0x***m***21***r***, where ***m*** is dependent on the memory requirements of the DBS9900 (with any daughter boards installed) and ***r*** is the revision level of the printed circuit board (PCB).

The Device Type Register contains a 4-bit field to define the memory size requirements of the device. This is the value ***m*** defined in the VXI spec. Four pins on the daughter board connector are used to request an ***m*** value from the VXI bus as if the device were in A32 mode. The DBS9900 will compare the ***m*** values for the two daughter board modules and select the lowest (largest amount of memory) non-zero value of ***m*** for this bit field. If no module is present, resistors will force these pins to ground, indicating that there is no memory required on this module.

Each daughter board module will have a JTAG port that will be a communication port to a continuous JTAG chain for programming and testing any ISP programmable devices that may be resident on the modules. Programming the ISP devices can be accomplished via the JTAG register inside the VXI interface PLD or via the ByteBlaster™ header on the DBS9900. The only exception will be the VXI interface PLD itself, which will be configured during power-up via an EPROM device and which can be programmed via a separate ByteBlaster header as well.

Each daughter board module will have a small serial EEPROM that will contain serial number, PCB revision number, firmware revision numbers, calibration data, etc. The size of the EEPROM is 128 words by 16 bits.

Any functions that are repeated among modules will be defined if possible at the same address with the same bit field. This includes the JTAG and EEPROM ports, clock divisor, trigger select register, etc. The module select register will map these registers into the device dependent register space. This allows for some commonality of low-level register control software.

The interface clock will extend to the daughter board for the purpose of I/O and serial data transfers to DACs and relays. The clock will operate in a burst mode only for the duration of the VXI bus cycle to reduce analog noise. The interface clock, data, control and address busses will all remain static when the DBS9900 is not engaged in data bus transfers or an interrupt acknowledge cycle. These busses will also be buffered as they enter the daughter board to reduce noise while the VXI bus is accessing the adjacent channel.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 2 - Module memory Sizing to VXI Device Type register

SIZE SELECT pins	MEMORY REQUIRED in Words	m (A32)	m (A24)
0000	none		
0001	512 M	1	N/A
0010	256 M	2	N/A
0011	128 M	3	N/A
0100	64 M	4	N/A
0101	32 M	5	N/A
0110	16 M	6	N/A
0111	8M	7	N/A
1000	4 M	8	0
1001	2 M	9	1
1010	1024 k	A	2
1011	512 k	B	3
1100	256 k	C	4
1101	128 k	D	5
1110	64 k	E	6
1111	32 k	F	7

These definitions are put forth to fulfill the VXI specification and not meant to imply the actual memory capacity of a module.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

6.3 Control Register

The CONTROL REGISTER (0x04) is used to generate a SOFT RESET,

- inhibit SYSFAIL generation
- to enable A24/A32 memory access.

When the SOFT RESET bit is set, it should be cleared again with the next operation to the DBS9900. No other operations should be attempted with the SOFT RESET bit SET, or a bus error will result.

Table 3 - CONTROL REGISTER - WRITE (0x04)

BIT	Name	Set Function	Clear Function
D0	SOFT RESET	Resets DBS9900 registers	Enables access to DBS9900
D1	SYSFAIL INHB	Prevents SYSFAIL assertion	Enables SYSFAIL assertion
D2 –D14		Not Used	Not Used
D15	A24/A32 ENABLE	Enables DBS9900 memory access	Disconnects DBS9900 memory

6.4 Status Register

The STATUS REGISTER (0x04) is used to read back the CONTROL REGISTER bits: READY, PASSED, and VXI MODID. It is also used to detect that a daughter board is present and its operational status. The Status Register has four additional bits added to the basic VXI functions. These are read only. Two bits are used to indicate that a module is present (i.e. installed) on the DBS9900. One bit is assigned for each possible module. If the module is present its corresponding bit will be SET. This defines up to two modules per VXI Motherboard. Two additional bits are used as DONE bits, one for each module. The Done bits will be duplicated in the module STATUS registers for flexibility.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 4 - STATUS REGISTER - READ ONLY (0x04)

BIT	Name	SET means:	Cleared means:
D0	SOFT RESET	Resets DBS9900 registers	Enables access to DBS9900
D1	SYSFL INHIB	Prevents SYSFAIL assertion	Enables SYSFAIL assertion
D2	PASSED	DC POWER OK	DC POWER FAIL
D3	READY	Self Test is complete	Self Test is in progress
D4		Not Used	Not Used
D5	MOD A INST	Module A is present	Module A is not installed
D6	MOD B INST	Module B is present	Module B is not installed
D7		Not Used	Not Used
D8	MOD A DONE	Module A is done	Module A is busy
D9	MOD B DONE	Module B is done	Module B is busy
D10		Not Used	Not Used
D11	MOD A SPARE	Module A spare	Module A spare
D12	MOD B SPARE	Module B spare	Module B spare
D13		Not Used	Not Used
D14	MODID*	Device not selected	Device selected
D15	A24/A32 ACTIVE	DBS9900 memory access enabled	DBS9900 memory access disabled

6.5 Memory Offset Register

The 16-bit Memory Offset Register (0x06) defines the base address of the DBS9900's A24 or A32 operational registers. The *m*+1 most significant bits of the offset register are the values of the *m*+1 most significant bits of the A24 or A32 register addresses. *M* is the value of the *required memory* field of the device type register (0x02).

Table 5- Offset Register Bit assignments in A24 and A32 Mode (0x06) R/W

JP1	MODE	D:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Pins 2&3	A24	A:	23	22	21	20	19	18	17	16	NA	NA	NA	NA	NA	NA	NA	NA
Pins 1&2	A32	A:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

6.6 Memory Attribute Register

The Memory Attribute Register (0x08) lists important characteristics of the memory on the DBS9900 daughter board modules. The details of the memory attribute register format can be found in the VXI specification, section 3.2.3.1.1.

6.7 Module / Relay Control Register

The Module / Relay Control Register (0x0A) is used to select one of two modules. The DBS9900 has two sets of A16 registers in the space 0x20 to 0x3E. One set is dedicated to each daughter board module. The SELECT bit switches in one set or the other so that both devices can be controlled. The memory is also switched at the same time to allow data to be transferred, one at a time, to both devices.

Table 6 - Module / Relay Select Register – R/W (0xA)

	Name	SET	Cleared
D0	SELECT	Select Module B Registers	Select Module A Registers
D1		Not Used	Not Used
D2		Not Used	Not Used
D3		Not Used	Not Used
D4	CAL PORT RELAY	Channel 2 connected to CAL PORT	Channel 1 connected to CAL PORT
D5	SPR_RLY 1	Close	Open
D6	SPR_RLY 2	Close	Open
D7	SPR_RLY 3	Close	Open

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

6.8 Trigger Source Control Register

The Trigger Source Control Register (0x0C) is used to multiplex the eight TTLTRG lines and the two ECLTRG lines from the backplane and the two front panel trigger inputs (FPA and FPB) to Module A and Module B independently. Both modules can use any one of the 12 lines as an input to its trigger circuit. Also available are register trigger bits in the Trigger Mode Register that can be used to trigger the module via software. Each module can also drive any one of the 10 backplane lines, or not drive any of them. The modules cannot drive the front panel trigger connectors. A module should not be set to receive and drive a trigger from the same line. Module A and Module B may be set to drive the same backplane trigger line. This, however, will result in a wired OR function for that trigger output.

- FPA = Front Panel Trigger A
- FPB = Front Panel Trigger B

Some modules can drive an additional SYNChronization signal onto one of the 8 TTL backplane trigger lines. This is selected by the same Trigger output Select field in the Trigger Source Control Register that selects the primary trigger output. The SYNC signal will always drive BP_TRIGOUT(N+4) where N is the PRIMARY TRIGGER output. For example, if a code of 1001 is set, the PRIMARY TRIGGER will drive BP_TRIGOUT1 and SYNC will drive BP_TRIGOUT5. Or, if a code of 1111 is set, the PRIMARY TRIGGER will drive BP_TRIGOUT7 and SYNC will drive BP_TRIGOUT3.

Table 7 - Trigger Source Control Register – R/W (0xC)

	Name	
D0	BP_TRIG0IN0	Module A Trigger Input Select - 0000 = Register Trigger, 0100 = ECL0, 0101 = ECL1, 0110 = FPA*, 0111 = FPB*, 1000 = TTL0, 1001= TTL1, 1010 = TTL2, 1011 = TTL3, 1100 = TTL4, 1101 = TTL5, 1110 = TTL6, 1111 = TTL7
D1	BP_TRIG0IN1	
D2	BP_TRIG0IN2	
D3	BP_TRIG0IN3	
D4	BP_TRIG0OUT0	Module A Trigger Output Select 0000 = Not driving Trigger, 0110 = ECL0, 0111 = ECL1, 1000 = TTL0, 1001= TTL1, 1010 = TTL2, 1011 = TTL3, 1100 = TTL4, 1101 = TTL5, 1110 = TTL6, 1111 = TTL7
D5	BP_TRIG0OUT1	
D6	BP_TRIG0OUT2	
D7	BP_TRIG0OUT3	
D8	BP_TRIG1IN0	Module B Trigger Input Select - 0000 = Register Trigger, 0100 = ECL0, 0101 = ECL1, 0110 = FPA*, 0111 = FPB*, 1000 = TTL0, 1001= TTL1, 1010 = TTL2, 1011 = TTL3,
D9	BP_TRIG1IN1	
D10	BP_TRIG1IN2	

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

D11	BP_TRIG1IN3	1100 = TTL4, 1101 = TTL5, 1110 = TTL6, 1111 = TTL7
D12	BP_TRIG1OUT0	Module B Trigger Output Select 0000 = Not driving Trigger, 0110 = ECL0, 0111 = ECL1, 1000 = TTL0, 1001= TTL1, 1010 = TTL2, 1011 = TTL3, 1100 = TTL4, 1101 = TTL5, 1110 = TTL6, 1111 = TTL7
D13	BP_TRIG1OUT1	
D14	BP_TRIG1OUT2	
D15	BP_TRIG1OUT3	

6.9 Trigger Mode Control Register

The Trigger Mode Control Register (0x0E) is used to set the trigger mode for modules A and B. Gate mode / Edge mode and trigger polarity is set for each module as well as a software (register) trigger bit. The software trigger bit will not have any effect unless the trigger input field of the Trigger source register is set to 0x0. The SOFT_TRIG function is not affected by the SLOPE bit. It is always high true regardless of the level of the SLOPE bit.

A Software Advance function is also provided that can be used to Advance to the next segment or to Branch to the Start Segment. To enable this bit, the front panel Advance input must be a logic high level or left disconnected. To enable the Advance from the front panel, this bit must be in the logic low state.

Table 8 - Trigger Mode Register– R/W (0xE)

	Name	Description
D0	GATE_ENA_A	0 = Edge trigger, 1 = gated trigger (start / stop)
D1	SLOPE_A	0 = falling edge trigger or low true gate trigger
D2	SOFT_TRIG_A	0 = inactive, 1 = trigger
D3	GATE_ENA_B	0 = Edge trigger, 1 = gated trigger (start / stop)
D4	SLOPE_B	0 = falling edge trigger or low true gate trigger
D5	SOFT_TRIG_B	0 = inactive, 1 = trigger
D6	SOFT_ADV_A	0 = inactive, 1 = Advance (Branch)
D7	SOFT_ADV_B	0 = inactive, 1 = Advance (Branch)

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

6.10 Interrupt Level Select Register

The 3-bit Interrupt Level Select (0x10) selects one of seven interrupt levels that the DBS9900 will interrupt. This register is set to all zeros on power up or reset.

Table 9

D7	D6	D5	D4	D3	D2	D1	D0	
THESE BITS NOT USED					INTERRUPT LEVEL			
					0	0	0	NO INTERRUPT
					0	0	1	Level 1
					0	1	0	Level 2
					0	1	1	Level 3
					1	0	0	Level 4
					1	0	1	Level 5
					1	1	0	Level 6
					1	1	1	Level 7

6.11 Interrupt Vector Select Register

The 8-bit Interrupt Vector Select Register (0x12) contains the lower 8 bits of the STATUS / ID, or interrupt vector, that will be returned to the host during Interrupt Acknowledge cycles. The DBS9900 responds to D16 Interrupt Acknowledge cycles at the level set by the interrupt level register. The upper 6 bits of the 16-bit vector are always driven to zero. If D9 is set, it will indicate the interrupt was triggered by Module A. If D8 is set, it indicates that Module B is interrupting. Note that both modules can interrupt at the same time. Finally, note that a second module could initiate an interrupt after the first module has interrupted but before the interrupt service routine is completed. Therefore, to ensure correct operation, the interrupt service routine needs to check and clear the interrupt register bits on *each* daughter board module whenever servicing an interrupt. This register is set to 0x00 on power up or reset.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 10 - Interrupt Vector Select Register

D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
MOD A	MOD B	STATUS / ID --- INTERRUPT VECTOR							

6.12 JTAG Programming Register

The JTAG Programming Register (0x1A) is used to re-program all of the in-circuit programmable devices on the DBS9900 and daughter board modules. This feature will be especially useful for correcting design errors or adding features to these programmable devices via a software driver update. All of the programmable devices on the DBS9900 and daughter board modules are connected together in one large JTAG chain. A provision will be made on the DBS9900 to automatically connect the JTAG chain as daughter board modules are plugged in and removed. Bit D1 is the JTAG TMS (Time Multiplexed Signal) and bit D0 is the JTAG Data signal. The JTAG clock signal is automatically generated by the VXI interface IC upon a write to the JTAG register. A read of the JTAG register will read back the status of the JTAG Data bit in position D0. A JTAG header on the DBS9900 provides an additional path for programming these devices. Note that the only device, which cannot be re-programmed via the JTAG Programming Register, is the VXI interface itself. The programming information for the VXI interface IC will reside inside an EPROM device that will be socketed (8-pin DIP) on the DBS9900. The VXI interface IC can also be re-programmed via a separate dedicated serial port header, as long as the EPROM is first removed from its socket.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

7 Clock

7.1 Channel A/B Clock Selection:

The DBS9900 has two clock inputs: Channel A and Channel B. Each channel enters the DBS9900 through a non-isolated SMA connector on the front panel and is routed to a high speed comparator. The threshold can be programmed to any voltage between –10V and +9.921875V with 8 bit resolution. There is a fixed 40mV hysteresis built into the input of the comparators to prevent inadvertent pulses from noisy inputs. Each comparator output is routed to a 4:1 Mux which select no clock, Time Base Clock, Channel A input or Channel B input clock. The output of the Channel B comparator is also routed to a 2:1 Mux which selects either a internal 10MHz reference oscillator (default) or the Channel B input. This allows the connection of an external 10MHz reference clock to the PLL reference input. This allows multiple daughter boards to run off of one reference, being synchronized together.

Since the input impedance is 4kΩ, it is expected that the user will source terminate their drivers with 50Ω series terminators to prevent reflections.

7.2 Frequency Synthesizer:

The internal clock generation circuitry consists of a 550MHz Phase Lock Loop (National LMX2306), which is programmed to generate frequencies (**Fvco**) from 100MHz to 200MHz in 50kHz steps. The wide adjustment range (over two to one) is achieved by a wide frequency range voltage controlled oscillator (Mini-Circuits JTOS-200). A 10MHz, 25PPM accurate crystal oscillator is used as a reference input to the PLL.

7.3 Time Base Divider:

The internal synthesizer drives a 2-tiered clock divider chain consisting of high and low speed divider elements to give frequency division ratios from 2 to 1,048,592. The high-speed divider (**HSDIV**) provides time base clock division ratios of 2 to 32 in steps of two. The low-speed divider (**LSDIV**) provides clock division ratios of 32 to 1,048,592, in steps of 16. The synthesizer/time base is able to generate sample rates from 95.36598Hz to 100MHz, with a worst case resolution of 500PPM of the programmed frequency. The lower divider chain is turned on only as needed in order to keep sub-harmonic beat frequencies out of the clock signal. A 4:1 Mux selects no clock, LSDIV Clock, HSDIV Clock. This clock (**TBCLOCK**) is routed to the daughter board clock selection muxes, which in turn are routed to the daughter board, and the trigger FPGA controller.

7.4 Program Sequence:

The programming model should look like the following process:

Program PLL:

- Program R-COUNTER (0x18, = 0h, 0x14, 0x16).
- Program N-COUNTER (0x18 = 1h, 0x14, 0x16).
- Program FUNCTION LATCH (0x18 = 2h or 3h, 0x14, 0x16)

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Program Time Base:

- Program High Speed Divider (0x18 = 7h, 0x14)
- Program Low Speed Divider (0x18 = 5h, 0x14)
- Program Time Base Control Register (0x18 = 6h, 0x14)
- Program Clock A/B Threshold (0x18 = 8h or 9h, 0x14)
- Program Trigger A/B Threshold (0x18 = Ah or Bh, 0x14)
- Program Clock Control Register (0x18 = 4h, 0x14)

More will be said about the function of these registers.

NOTE: The indirect addressing register 0X18 should be loaded followed by 0X14 and 0X16.

7.4.1 Sample Rate Calculation:

The table below shows the various dividers available on the DBS9900 Mother Board Time Base. Column 1 shows which divider output can be selected. Column 2 shows the formula for the Time Base Output frequency as a function of the programmable divide ratios. Columns 3 and 4 show the ranges of each divide ratio and output frequencies. The formula for calculating output frequency can be summarized as **Ftb** = **Fpd * N / D**; where **N** can take on values from 2000 to 4000 and **D** can take on values of 2 to 1,048,592. As a result, there are multiple solutions for a single desired output frequency. To determine an output frequency that is as close as possible to the desired frequency you could check every possible solution and look for the minimum error or you could follow the algorithm following Table 12. This algorithm guarantees that you will get an actual Sample Rate that has a frequency error, Fε <= 1/Nmin. = 1/2000 = .05% of the desired frequency within the range of 95.36598Hz to 100MHz.

Table 1 – Frequency Ranges

PLL Frequency Output			
Range	**Frequency Formula**	**Divider Ratio Range**	**Frequency Range**
Fvco	Fvco = Fpd * N	N = 2000 to 4000, steps of 1	100MHz to 200MHz, steps of Fpd = 50kHz
Time Base Freq. Output	**Frequency Formula**	**Divider Ratio Range**	**Sample rate**
TBCLK_DIV1	Fo = Fvco * 1/2	2	50MHz to 100MHz
TBCLK_DIV2	Fo = Fvco * 1/HSDIV * 1/2	4 to 32 in steps of 2	3.125MHz to 50MHz
LSDIV	Fo = Fvco * 1/8 * 1/LSDIV * 1/2	32 to 1,048,592 in steps of 16	95.36598Hz to 6.25MHz

Sample Rate calculation algorithm in a nutshell:

1. Compare Fw (wanted frequency) to Frequency range in table 12. Select one of three procedures to calculate Frequency Synthesizer parameters.

   If the TBCLK_DIV1 range is selected for Sample Rates of 50MHz to 100MHz.

1. Calculate Nw = ROUND((Fw / Fpd) * 2)

   2. Program N = Nw and select TBCLK_DIV1range.

      If the TBCLK_DIV2 range is selected for Sample Rates of 3.125MHz to 50MHz.

      Fpd = 50kHz, Nmax = 4000.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

1. Calculate HSDIVw = ROUND((Fpd / Fw) * 1/2 * Nmax).
2. Calculate Nw = ROUND((Fw / Fpd) * 2 * HSDIVw).
3. Program N = Nw, HSDIV = HSDIVw and select TBCLK_DIV2 range.

If the LSDIV range is selected for Sample Rates of 95.36598Hz to 6.25MHz.

Fpd = 50kHz, Nmax = 4000, HSDIV = 8.

1. Calculate LSDIVw = ROUND((Fpd / Fw) * 1/2 * 1/8 * Nmax). If the calculated LSDIVw is greater than 2^16 then make LSDIVw = 2^16.

- Calculate Nw = ((Fw / Fpd) * 2 * 8 * LSDIVw).
- Calculate ROUND(Nw) = ROUND_DOWN((Fw / Fpd) * 2 * 8 * LSDIVw).

2. Rounding down Nw will lower the output frequency by the amount,

freq.error = (Round(Nw) – Nw) / Nw. In this case, LSDIVadj. = freq.error * (2^16).

3. Program N = Nw, HSDIV = 8, LSDIV = (LSDIVw + LSDIVadj). and select LSDIV range.

Note: If the desired frequency is an integer, it is best to pick values for Nw and LSDIVw that come out as integers.

See programming example in section **Error! Reference source not found.**.

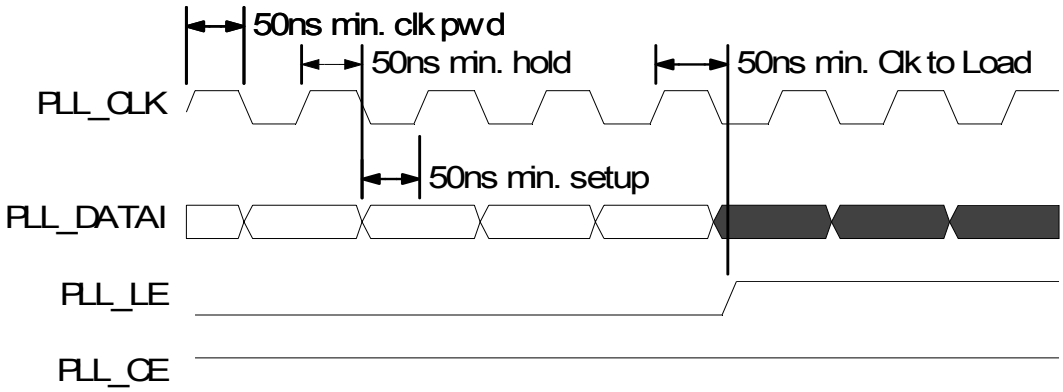
7.5 Phase Lock Loop

The National LMX2306 is a 550MHz Phase Lock Loop. It consists of a 21 bit serial to parallel data register, 14 bit R-COUNTER, 18 bit N-COUNTER, divide by 8/9 dual modulus pre-scalar, 18 bit FUNCTION LATCH, phase detector and an output charge pump to integrate phase detector pulses. There is a Lock Detect circuit to indicate when frequency lock has occurred. The R-COUNTER divides the reference clock down to the phase detector comparison frequency. The N-COUNTER divides the VCO frequency (Fvco) down to the phase detector comparison frequency. The FUNCTION LATCH performs various configuration duties within the LMX2306 chip.

7.6 LMX2306 Communications port

Data is loaded using National's MICROWIRE™ Serial Interface. The interface consists of Clock (**PLL_CLK**), Data (**PLL_DATAI**), Latch Enable (**PLL_LE**) and a Chip Enable (**PLL_CE**). **PLL_CE** goes high on power-up. When **PLL_LE** goes low, data is clocked into the LMX2306 on the rising edge of the clock. Where **PLL_LE** goes high and the 21-bit shift register internal to the LMX2306 loads the internal registers and counters. A timing diagram is shown below:

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com



7.6.1 TIME BASE RESET

The time base registers should all be set to their default settings as shown in the default column of each register upon power up, Hard Reset, or Soft Reset.

Since the LMX2306 PLL is a serial load device it will be up to the software driver to initialize the internal registers to their default state as defined in the default columns of the PLL registers (**R-COUNTER, N-COUNTER, FUNCTION LATCH**). Notice that the **FUNCTION LATCH** should be written twice in order to reset internal counters. **F[1]** set high resets the counters, and **F[1]** set low starts the counters counting in synchronism. The **FO_LD** line should be configured (**F[5:3] = 1h**) to output Lock Detect.

7.6.2 PLL R-COUNTER

The **R-COUNTER (0x14, 0x16, Base + 0x18 = 0h) R/W** is a 21 bit control word that is transmitted to the DBS9900 VXI Interface FPGA in one 16-bit and one 5-bit write. The software driver writes to the control bits located in **0x18 = 0h** of the VXI FPGA. This register instructs the FPGA that the next two register writes, **0x14** and **0x16** will be directed through a parallel to serial shift register to the **R-COUNTER**. The control word **C[4:1]**, is concatenated to the end of the bit string, and controls the register inside the PLL that is to be loaded upon completion of the serial transfer. The MSB's of the **R-COUNTER** are to be loaded into **BASE + 0x14**. The five LSB's are transmitted to **BASE + 0x16**. The 21-bit serial shift register (internal to the LMX2306) is loaded MSB first, via the data input on the rising edge of the clock when the Latch Enable (**PLL_LE**) goes low.

Loading is complete after 21 rising edges of the clock. The internal register is latched on the rising edge of **PLL_LE**. The **R-COUNTER** divides the reference clock down to the phase detector comparison frequency (**Fpd**). The legitimate division ratios for the **R-COUNTER** are $3 \leq \mathbf{R} \leq 16383$. The DBS9900 uses a fixed 10MHz reference oscillator so the **R-COUNTER** should be programmed to a divide ratio of **R** = 200d = c8h for **Fpd** = 50kHz. The data format for the **R-COUNTER** is shown in the table below. **R19** determines the lock detect precision.

When **R19** is set low, the PLL decides that it is only in frequency lock after three consecutive reference clock cycles where the reference and VCO input edges are within 15nsec of each other. If set high, the PLL decides that it is in frequency lock after five consecutive reference clock cycles.

R[18:15] set test modes used by the manufacturer and should be set low.

R[14:1] are the bits of the **R-COUNTER** where **R14** is the MSB and **R1** is the LSB.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

0x18(C[4:1]) are control bits and should be set to 0h to allow the data to be loaded in the **R-COUNTER**. The **R-COUNTER** will be latched at the end of the serial transfer on the rising edge of **PLL_LE**.

R-COUNTER should be initialized to the following state:

- **R[14:1] = c8h =** 200d where Fref / 200 = 50kHz = **Fpd**

Table 2 - R-Counter Register Map PLL1 Base + 0x14

Base + 0X14	R-Counter (PLL 1)	FUNCTION	Default	
Data Bus	**Lock Detect**			
D[15]	R[19]	Lock Detect Precision	Set to 0	R/W
	Test Bits			
D[14]	R[18]	Test Mode	Set to 0	R/W
D[13]	R[17]	Test Mode	Set to 0	R/W
D[12]	R[16]	Test Mode	Set to 0	R/W
D[11]	R[15]	Test Mode	Set to 0	R/W
	R-COUNTER		R = 00c8h	
D[10]	R[14]	B13 MSB	Set to 0	R/W
D[09]	R[13]	B12	Set to 0	R/W
D[08]	R[12]	B11	Set to 0	R/W
D[07]	R[11]	B10	Set to 0	R/W
D[06]	R[10]	B9	Set to 0	R/W
D[05]	R[9]	B8	Set to 0	R/W
D[04]	R[8]	B7	Set to 1	R/W
D[03]	R[7]	B6	Set to 1	R/W
D[02]	R[6]	B5	Set to 0	R/W
D[01]	R[5]	B4	Set to 0	R/W
D[00]	R[4]	B3	Set to 1	R/W

Table 3 - R-Counter Register Map PLL0 Base + 0x16

Base + 0X16	R-Counter (PLL 0)	FUNCTION	Default	
Data Bus				
D[04]	Not Used	Not Used	Set to 0	R/W
D[03]	Not Used	Not Used	Set to 0	R/W
	R-Counter			
D[02]	R[3]	B2	Set to 0	R/W
D[01]	R[2]	B1	Set to 0	R/W
D[00]	R[1]	B0 LSB	Set to 0	R/W

Table 4 - R-Counter Address Base + 0x18 (0h)

Base + 0X18	PLL Address	FUNCTION	Default	
Data Bus		C=0h		
D[03]	C[4]	Control bit, Set to 0	Set to 0	R/W
D[02]	C[3]	Control bit, Set to 0	Set to 0	R/W

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

D[01]	C[2]	Control bit, Set to 0	Set to 0	R/W
D[00]	C[1]	Control bit, Set to 0	Set to 0	R/W

7.6.3 PLL N-COUNTER

The **N-COUNTER (0x14, 0x16, Base + 0x18 = 1h) R/W** is a 21 bit control word that is transmitted to the DBS9900 VXI Interface FPGA in one 16-bit and one 5-bit write. The software driver writes to the control bits located in **0x18 = 1h** of the VXI FPGA. This register instructs the FPGA that the next two register writes, **0x14** and **0x16** will be directed through a parallel to serial shift register to the **N-COUNTER**. The control word **C[4:1]**, is concatenated to the end of the bit string, and controls the register inside the PLL that is to be loaded upon completion of the serial transfer. The software driver shall writes the MSB's of the **N-COUNTER** to **BASE + 0x14**. The five LSB's are transmitted to **BASE + 0x16**.

The 21-bit serial shift register is loaded MSB first, via the data input on the rising edge of the clock when the Latch Enable (**PLL_LE**) goes low. Loading is complete after 21 rising edges. The internal register is latched on the rising edge of **PLL_LE**. The **N-COUNTER** divides the VCO clock down to the Phase Detector comparison frequency (**Fpd**). The **N-COUNTER** is loaded when the control bits **C[4:1]** are set to **1h**. When the **N-COUNTER** is selected, 5-bits of the 21 bit, shift register is transferred to a 5-bit latch which sets a 5-bit pulse swallow counter (**A-COUNTER**).

A 13-bit latch transfers data to a 13-bit counter (**B-COUNTER**). Together, these counters act in consort to give a division ratio from 24 ≤ **N** ≤ 65,535, where 0 ≤ **A** ≤ 7, 3 ≤ **B** ≤ 8191 and **A** ≤ **B**. The formula for this division ratio is: **N = [(P X B) + A]** where **P = 8**. When used with the JTOS-200 the values for **N** will range from 2000 ≤ **N** ≤ 4000. The VCO frequency can be calculated with the following formula: **Fvco = Fref X [(P X B) + A] / R**.

The **N-COUNTER** should be initialized to the following state:

- B[12:0] = 1F4h = 500d, A[3:0] = 0h.

200MHz / 50kHz = 4000 => N = [(P X B) + A] => P = 8, B = 500, A = 0.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

The data format for the **N-COUNTER** is shown in the tables below.

Table 5 - N-Counter Register Map PLL 1 Base + 0x14

Base + 0X14	N-Counter (PLL 1)	FUNCTION	Default	
Data Bus	**GO bit**			
D[15]	N[19]	Not Used	Set to 0	R/W
	B-COUNTER		B=01F4h	
D[14]	N[18]	B12 MSB	Set to 0	R/W
D[13]	N[17]	B11	Set to 0	R/W
D[12]	N[16]	B10	Set to 0	R/W
D[11]	N[15]	B9	Set to 0	R/W
D[10]	N[14]	B8	Set to 1	R/W
D[09]	N[13]	B7	Set to 1	R/W
D[08]	N[12]	B6	Set to 1	R/W
D[07]	N[11]	B5	Set to 1	R/W
D[06]	N[10]	B4	Set to 1	R/W
D[05]	N[9]	B3	Set to 0	R/W
D[04]	N[8]	B2	Set to 1	R/W
D[03]	N[7]	B1	Set to 0	R/W
D[02]	N[6]	B0 LSB	Set to 0	R/W
D[01]	N[5]	Not Used	Set to 0	R/W
D[00]	N[4]	Not Used	Set to 0	R/W

Table 6 - N-Counter Register Map PLL 0 Base + 0x16

Base + 0X16	N-Counter (PLL 0)	FUNCTION	Default	
Data Bus	**Lock Detect**			
D[04]	Not Used	Not Used	Set to 0	R/W
D[03]	Not Used	Not Used	Set to 0	R/W
	A-COUNTER		A=0h	
D[02]	N[3]	A2 MSB	Set to 0	R/W
D[01]	N[2]	A1	Set to 0	R/W
D[00]	N[1]	A0 LSB	Set to 0	R/W

Table 7 - N-Counter Address Base + 0x18 (1h)

Base + 0X18	PLL Address	FUNCTION	Default	
Data Bus	**Control**	C=1h		
D[03]	C[4]	Control bit, set to 0	Set to 0	R/W
D[02]	C[3]	Control bit, set to 0	Set to 0	R/W
D[01]	C[2]	Control bit, set to 0	Set to 0	R/W
D[00]	C[1]	Control bit, set to 1	Set to 0	R/W

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

N19 enables one of four **FastLock Modes**. This bit is not used, and should be set low.

The data latched into **N[18:6]** programs the **B-COUNTER** divider. **N18** is the MSB and **N6** is the LSB. The count in **B** can take on values of 3 ≤ **B** ≤ 8191 where **A** ≤ **B.**

The data latched into **N[5:4]** performs no function in the LMX2306.

Data latched into **N[3:1]** programs the **A-COUNTER** Divider. **N3** is the MSB and **N1** is the LSB. The count in **A** can take on values of 0 ≤ **A** ≤ 7 where **A** ≤ **B**

7.6.4 PLL FUNCTION LATCH

The **FUNCTION LATCH (0x14, 0x16, Base + 0x18 = 2h or 3h) R/W** is a 21 bit control word that is transmitted to the Mother Board VXI Interface FPGA in one 16-bit and one 5-bit write. The software driver writes to the control bits located in **0x18 = 2h or 3h** of the VXI FPGA. This register instructs the FPGA that the next two register writes **0x14** and **0x16** will be directed through a parallel to serial shift register to the **FUNCTION LATCH**. The control word **C[4:1]**, is concatenated to the end of the bit string, and controls the register inside the PLL that is to be loaded upon completion of the serial transfer. The software driver writes the MSB's of the Function Latch to **BASE + 0x14**. The five LSB's are transmitted to **BASE + 0x16**.

The 21-bit serial shift register is loaded MSB first, via the data input on the rising edge of the clock when the Latch Enable (**PLL_LE**) goes low. Loading is complete after 21 rising edges of the clock. The internal register is latched on the rising edge of **PLL_LE**. The Function Latch configures the LMX2306 for various modes of operation.

The data format for the Function Latch is shown in the tables below. **F19** is used for Test Modes, and should be set to 0.

The Function Latch should be initialized to the following state:

- **F[1]** should be set high, then low in two consecutive writes, in order to reset and re-synchronize the internal counters.
- **FO_LD** MUX control should be set to **F[5:3] = 1h** so that **Digital Lock Detect** is available at the **FO_LD** pin to detect when the PLL is in lock.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 8 - FUNCTION LATCH PLL1 Base + 0x14

Base + 0X14	F Latch (PLL 1)	FUNCTION	Default	
Data Bus	**Test Mode**			
D[15]	F[19]	Not Used	Set to 0	R/W
	Power Down Mode			
D[14]	F[18]	Not Used	Set to 0	R/W
	Test Mode			
D[13]	F[17]	Not Used	Set to 0	R/W
D[12]	F[16]	Not Used	Set to 0	R/W
D[11]	F[15]	Not Used	Set to 0	R/W
	TimeOut Counter	**Fast Lock Time Out Counter**		
D[10]	F[14]	Not Used	Set to 0	R/W
D[09]	F[13]	Not Used	Set to 0	R/W
D[08]	F[12]	Not Used	Set to 0	R/W
D[07]	F[11]	Not Used	Set to 0	R/W
D[06]	F[10]	Not Used	Set to 0	R/W
D[05]	F[9]	Not Used	Set to 0	R/W
D[04]	F[8]	Not Used	Set to 0	R/W
D[03]	F[7]	**CP Tri-State**	Set to 0	R/W
D[02]	F[6]	**PD Polarity**	Set to 0	R/W
	FO_LD Control		**Lock Detect**	
D[01]	F[5]	FO_LD[2]	Set to 0	R/W
D[00]	F[4]	FO_LD[1]	Set to 0	R/W

Table 9 - FUNCTION LATCH PLL Base + 0x16

Base + 0X16	F Latch (PLL 0)	FUNCTION	Default	
Data Bus				
D[04]	Not Used	Not Used	Set to 0	R/W
D[03]	Not Used	Not Used	Set to 0	R/W
Data Bus	**FO_LD Control**			
D[02]	F[3]	FO_LD[0]	Set to 1	R/W
D[01]	F[2]	**Power Down**	Set to 0	R/W
D[00]	F[1]	**Counter Reset**	Set to 1 then 0	R/W

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 10 - FUNCTION LATCH Address Base + 0x18 (2h or 3h)

Base + 0X18	F Latch (PLL 0)	FUNCTION	Default	
Data Bus		C=2h or 3h		
D[03]	C[4]	Control bit, set to 0	Set to 0	R/W
D[02]	C[3]	Control bit, set to 0	Set to 0	R/W
D[01]	C[2]	Control bit, set to 1	Set to 0	R/W
D[00]	C[1]	Control bit, set to 0 or 1	Set to 0	R/W

If **C[4:1]** is set to **3h,** the same data as **C[4:1] = 2h** will be loaded into the function latch with the addition of providing a reset pulse to reset the **R-COUNTER**, **N-COUNTER** and **TIMEOUT** counters. This re-synchronizes the counters.

The power down feature of the LMX2306 is not used on the DBS9900. **F2, F10, F18** should be set low.

The FastLock function on the LMX2306 is not used on the DBS9900. **F[14:8]** should be set low.

F7 tri-states the charge pump integrator output. For normal operation, set **F7** low.

F6 selects the polarity of the phase detector. **F6** should be set high when the VCO gain slope is positive. **F6** should be set low when the VCO gain slope is negative. The JTOS-200 VCO has a positive gain slope. The integrator loop inverts the polarity of the control signal; therefore, **F6** should be set low.

F5 to **F3** control the function of the **FO_LD** (pin14). The internal MUX routes several internal nodes to pin14. The nodes are listed in the following table.

Bits **F[5:3]** are decoded in the following way:

Table 11 – FO_LD Register Map Base + 0x14

F[5:3]	FO_LD
000	TRI-STATE
001	Digital Lock Detect (Default)
010	N Divider Output (Fpd)
011	Active High
100	R Divider Output (Fpd)
101	N Channel Open Drain Lock Detect
110	Serial Data Output
111	Active Low

Setting **F[5:3]** to **2h** and **4h** is useful to determine if the reference oscillator and VCO are being properly divided down to **Fpd**. This aids in trouble shooting the PLL. **F[5:3]** set to **1h (Default)** makes digital lock detect available to the host. Under normal operation, **F[5:3]** should be programmed to **1h** to determine if the PLL is in frequency lock.

F[1] enables and disables the counter RESET. This function resets the counters internal to the LMX2306 to their initial reset state. When set high, the counter RESET is enabled. When set low, the counter RESET is disabled.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

C[4:1] must be set to **2h** in order for the LATCH FUNCTION to be loaded in the LMX2306. **3h** is used when initializing the PLL.

7.7 Clock Control Register

The Clock Control Register **(0x14, Base + 0x18 = 4h) R/W** routes No clock, Time Base Clock, Channel A or Channel B clock (to daughter boards A and B). To access this register, the software driver writes to the control bits located in **0x18 = 4h** of the VXI FPGA. This register instructs the FPGA that the next register writes the Clock Control Register at **0x14**.

The Clock Control Register should be initialized to the following state:

- All bits set low

Table 12 - Clock Control Register Map Base + 0x14

Base + 0X14	Clock Control Register	FUNCTION	Default	
Data Bus				
D[15]	Not Used	Not Used	Set to 0	R/W
D[14]	Not Used	Not Used	Set to 0	R/W
D[13]	Not Used	Not Used	Set to 0	R/W
D[12]	Not Used	Not Used	Set to 0	R/W
D[11]	Not Used	Not Used	Set to 0	R/W
D[10]	Not Used	Not Used	Set to 0	R/W
D[09]	Not Used	Not Used	Set to 0	R/W
D[08]	Not Used	Not Used	Set to 0	R/W
Data Bus	**CHA/B Control**			
D[07]	Not Used	Not Used	Set to 0	R/W
D[06]	Not Used	Not Used	Set to 0	R/W
D[05]	EXTCLKA_EN	Routes CHA Clk to Mod. A and B Clk selector MUX	Set to 0	R/W
D[04]	EXTCLKB_EN	Routes CHB Clk to Mod. A and B Clk selector MUX	Set to 0	R/W
D[03]	CHA_CLKSEL[1]	CHA Clock source select, MSB	Set to 0	R/W
D[02]	CHA_CLKSEL[0]	CHA Clock source select, LSB	Set to 0	R/W
D[01]	CHB_CLKSEL[1]	CHB Clock source select, MSB	Set to 0	R/W
D[00]	CHB_CLKSEL[0]	CHB Clock source select, LSB	Set to 0	R/W

Table 13 - Clock Control Address Base + 0x18

Base + 0X18	PLL Address	FUNCTION	Default	
Data Bus				
D[03]	C[4]	Control bit, set to 0	Set to 0	R/W
D[02]	C[3]	Control bit, set to 1	Set to 0	R/W

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

D[01]	C[2]	Control bit, set to 0	Set to 0	R/W
D[00]	C[1]	Control bit, set to 0	Set to 0	R/W

7.7.1 Clock Control Description

EXTCLKA_EN is a control output that enables the signal present on the input of Clock A (J4) to pass through the Channel A threshold comparator to the Clock selector multiplexers. This line is decoded in Software. See section **Error! Reference source not found.** for truth table.

EXTCLKB_EN is a control output that enables the signal present on input of Clock B (J10) to pass through the Channel B threshold comparator to the Clock selector multiplexers. This line is decoded in Software. See section **Error! Reference source not found.** for truth table.

CHA_CLKSEL[1:0] are control lines to route No Clock, Time Base Clock, External channel A Clock or External channel B Clock to daughter board A Clock. These lines are decoded in hardware. See section **Error! Reference source not found.** for truth table.

Table 14 - CHA_CLKSEL[1:0] Decode

CHA_CLKSEL[1:0]	FUNCTION	
00	Route No Clock to Module A (Default)	R/W
01	Route Internal Time Base Clock to Module A	R/W
10	Route Channel A Clock to Module A	R/W
11	Route Channel B Clock to Module A	R/W

CHB_CLKSEL[1:0] are control lines to route No Clock, Time Base Clock, External channel A Clock or External channel B Clock to daughter board B Clock. These lines are decoded in hardware. See section **Error! Reference source not found.** for truth table.

CHB_CLKSEL[1:0] are decoded in the following way:

Table 15 - CHB_CLKSEL[1:0] Decode

CHB_CLKSEL[1:0]	FUNCTION	
00	Route No Clock to Module B (Default)	R/W
01	Route Internal Time Base Clock to Module B	R/W
10	Route Channel A Clock to Module B	R/W
11	Route Channel B Clock to Module B	R/W

7.8 LSDIV Low Speed Divider Register

The **LSDIV** Clock Divider Register **(0x14, Base + 0x18 = 5h) R/W,** stores the count of the low frequency 16-bit counter. The address to write to this register is set when the host writes to **0x18 = 5h**. Data is loaded into the register when the host writes to address **0x14**. This counter is

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

clocked with the divide by 8 output of the **HSDIV** counter. It runs at one-eighth of the PLL clock frequency (**Fvco**), 12.5MHz to 25MHz. The **LSDIV** output then goes to a flip-flop to square up the output. The **HSDIV** divider should be programmed to 7h when the low speed divider is being used. **LSDIV** divides **Fvco** by the formula:

Ftb = Fvco / (16 * (LSDIV[15:0] + 2) where 0 ≤ **LSDIV[15:0]** ≤ 65,535. The possible division ratios available from **LSDIV** range from 32 to 1,048,592 in steps of 16.

The **LSDIV Register** should be initialized to the following state:

- All bits set low

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 16 - LSDIV Register MapBase + 0x14

Base + 0X14	LSDIV Register Map		Default	
Data Bus	**Function**	**Divisor**		
D[15]	LSDIV[15] , MSB	32,770	Set to 0	R/W
D[14]	LSDIV[14]	16,386	Set to 0	R/W
D[13]	LSDIV[13]	8,194	Set to 0	R/W
D[12]	LSDIV[12]	4,098	Set to 0	R/W
D[11]	LSDIV[11]	2,050	Set to 0	R/W
D[10]	LSDIV[10]	1,026	Set to 0	R/W
D[09]	LSDIV[9]	514	Set to 0	R/W
D[08]	LSDIV[8]	258	Set to 0	R/W
D[07]	LSDIV[7]	130	Set to 0	R/W
D[06]	LSDIV[6]	66	Set to 0	R/W
D[05]	LSDIV[5]	34	Set to 0	R/W
D[04]	LSDIV[4]	18	Set to 0	R/W
D[03]	LSDIV[3]	10	Set to 0	R/W
D[02]	LSDIV[2]	6	Set to 0	R/W
D[01]	LSDIV[1]	4	Set to 0	R/W
D[00]	LSDIV[0], LSB	3	Set to 0	R/W

Table 17 - LSDIV Address Map Base + 0x18

Base + 0X18	LSDIV Address		Default	
Data Bus	**Function**	C=5h		
D[03]	C[4]	Control bit, set to 0	Set to 0	R/W
D[02]	C[3]	Control bit, set to 1	Set to 0	R/W
D[01]	C[2]	Control bit, set to 0	Set to 0	R/W
D[00]	C[1]	Control bit, set to 1	Set to 0	R/W

LOW SPEED DATA REGISTER CONTROL Lines (5-28993)

To write to the LSDIV Counter, write 5h to Base + 0x18 to set address to write data. FREQSEL1, PLL_EN and INTREF should be set high and REFSEL set low. (clock running). CNTR_WR should be asserted high before the rising edge of CNTR_CLK.

To read the contents of the LSDIV counter, write 5h to 0x18 to set address to read data. FREQSEL1 should be set high. PLL_EN, INTREF and REFSEL set low (clock stopped) set low no sooner than .1uS after setting FREQSEL1. CNTR_WR should be asserted low before the rising edge of CNTR_CLK. Two reads are required to get valid data because of internal pipelining of the LSDIV Latch. The host can read 16 bit data from the counter register located at the Time Base address 0x14.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

7.9 Time Base Control Register

The Time Base Control Register **(0x14, Base + 0x18 = 6h)** R/W selects the clock inputs and enables the various dividers depending on the desired output frequency. Access to this register is enabled when the Software Driver writes to **0x18 = 6h**.

The Time Base Control Register should be initialized to the following state:

- All bits set low

Table 18 - Time Base Control Register Map Base + 0x14

Base + 0X14	Time Base Control Register	FUNCTION	Default	
Data Bus	**TB CONTROL**			
D[15]	Not Used	Not Used	Set to 0	R/W
D[14]	REFSEL	Selects External Ref. Oscillator	Set to 0	R/W
D[13]	PLL_EN	Enables PLL VCO and Integration Amp	Set to 0	R/W
D[12]	INTREF	Turns on Internal Reference Oscillator	Set to 0	R/W
D[11]	Not Used	Not Used	Set to 0	R/W
D[10]	Not Used	Not Used	Set to 0	R/W
D[09]	FREQSEL[1]	TB select line, MSB	Set to 0	R/W
D[08]	FREQSEL[0]	TB select line, LSB	Set to 0	R/W
	MISCELANEOUS			
D[07]	UP_DOWN	Direction Control for PLL Spur Trim	Set to 0	R/W
D[06]	CS	Spur Trim Chip select	Set to 0	R/W
D[05]	INC	Spur Trim Clock	Set to 0	R/W
D[04]	Lock Detect	Detects Frequency Lock		ROnly
D[03]	Not Used	Not Used	Set to 0	R/W
D[02]	Not Used	Not Used	Set to 0	R/W
D[01]	Not Used	Not Used	Set to 0	R/W
D[00]	Not Used	Not Used	Set to 0	R/W

Table 19 - Time Base Address Map Base + 0x18

Base + 0X18	Time Base Address	FUNCTION	Default	
Data Bus	**TB CONTROL**	C=6h		
D[03]	C[4]	Control bit, set to 0	Set to 0	R/W
D[02]	C[3]	Control bit, set to 1	Set to 0	R/W
D[01]	C[2]	Control bit, set to 1	Set to 0	R/W
D[00]	C[1]	Control bit, set to 0	Set to 0	R/W

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

7.9.1 TIME BASE CLOCK Inputs

REFSEL is a signal output that controls a 2:1 mux to select either External or Internal Reference Oscillators. REFSEL should be set high when External Reference is selected and PLL_EN is asserted. See section **Error! Reference source not found.** for truth table.

PLL_EN is a control output that enables the VCO and PLL Integration Amplifier. This is used when the PLL is to be used with either External or Internal Reference Oscillators. This line is decoded in Software. See section **Error! Reference source not found.** for truth table.

INTREF is a signal that enables the Internal Reference Oscillator. **INTREF** should be asserted when **PLL_EN** is asserted and Internal reference oscillator is to be used. See section **Error! Reference source not found.** for truth table.

FREQSEL[1:0] is decoded according to the following table.

Table 20 - D[09:08] Frequency Select Truth Table

	TB CONTROL	
D[09:08]	FREQSEL[1:0]	
11	HSDIV_EN	Enables HSDIV divider
10	HSDIV2_EN	Enables HSDIV FF
01	LSDIV_EN	Enables LSDIV
00	No Clock	Disable Time Base Clock (Default)

When **FREQSEL[1:0] = 0h** the time base selects no clock. Under this condition, **PLL_EN, INTREF** should be disabled and **REFSEL** should be set low.

When **FREQSEL[1:0]** = **1h, PLL_EN** should be enabled and **REFSEL** and **INTREF** should be asserted depending on whether internal or external reference oscillator is to be used. The **HSDIV** divider should be programmed to **HSDIV[3:0] = 7h** (divide ratio of 8). **LSDIV** can be programmed to modify the output clock frequency with divide ratios of 32 to 1,048,592, in steps of 16 (95.36598Hz to 6.25MHz frequency range).

When **FREQSEL[1:0]** = **2h**, **PLL_EN** should be enabled along with either external or internal oscillator and the **HSDIV** divider can be programmed to modify the output clock frequency with divide ratios of 4 to 32, in steps of 2 (3.125MHz to 50MHz frequency range).

When **FREQSEL[1:0]** is set to **3h**, **PLL_EN** should be enabled along with either external or internal oscillator. The **HSDIV** divider should be programmed to **1h** (divide ratio of 2) to modify the output clock frequency with divide ratio of 2 (50MHz to 100MHz frequency range).

7.9.2 D[07:05] MISCELANEOUS Control bits

When **UP_DOWN D[07]** is set high, and **CS D[06]** is set high, the EEPROM ***Spur trim pot*** will advance it's wiper towards the high rail (+2.5V + .25V) of the trim pot on the rising edge of **INC**. This increases compensation current applied to the PLL loop amplifiers negative input to a maximum of +100nA. When **UP_DOWN D[07]** is set low and **CS D[06]** is set high, the EEPROM ***Spur trim pot*** will advance it's wiper towards the low rail (+2.5V - .25V) of the trim pot on the rising edge of **INC D[05]**. This decreases compensation current to a minimum of -100nA. This compensation current acts to null mismatch between source/sink current output of the PLL charge pump. This in turn nulls out the frequency spurs located at offset frequencies spaced **Fpd** apart around the PLL carrier frequency (**Fvco**).

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

D[04] is the **Lock Detect** signal from the PLL. This output is available when **F[5:3]** is programmed to **1h** in the function latch. **D[04]** is set high by the PLL when the absolute phase error is less than 15nsec for at least 3 or 5 consecutive phase detector cycles (3 or 5 * 1/Fpd = 60μsec or 100μsec), depending on what **R[19]** is programmed to. **R[19]** should be programmed low, which forces three phase detector cycles to be used to determine frequency lock. **FO_LD** (**D[04]**) will go low if the phase detector error exceeds 30nsec. for more than one phase detector cycle. It is set to a low state when the charge pump is tri-stated or the PLL is powered down (**F[2], F[18]**). This is a Read Only signal designed to indicate phase lock.

7.10 High Speed Divider Data Register

The **HSDIV Register (0x14, BASE + 0x18 = 7h) R/W** is a 4 bit register designed to program the **HSDIV** divider. To access this register, The software driver writes to the control bits located in **BASE + 0x18 = 7h** of the VXI FPGA. This register instructs the FPGA that the next register writes the **HSDIV Register** at **0x14**.

The hardware transfers data to the 6 bit ECL counter. When the **PLL_EN**, **INTREF** is active high and **REFSEL**, **FREQSEL0** are low the **HSDIV** divider will clock this data into the counter on the rising edge of the clock when **! COUT** (terminal count) drives S1 low. The counter will count down toward zero on each clock. Once the count reaches zero, the terminal count will stay low for one clock cycle and will re-load the data into the counter on the clocks next rising edge. When the divider is programmed to count by two, the output is a square wave and is sent directly to the time base frequency selector MUX. For counts greater than two, the terminal count is routed to an external flip-flop and then on to the time base frequency selector. This flip-flop will only toggle on the rising edge of terminal count, thereby making the output wave form a 50% duty cycle square wave.

The **HSDIV Register** should be initialized to the following state:

- All bits read back low when the board is initialized.
  The data will read back in the following format:

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 21 - HSDIV Register Base + 0x14

Base + 0X14	HSDIV (PLL 1)	FUNCTION	Default	
Data Bus				
D[15]	Not Used	Not Used	Set to 0	R/W
D[14]	Not Used	Not Used	Set to 0	R/W
D[13]	Not Used	Not Used	Set to 0	R/W
D[12]	Not Used	Not Used	Set to 0	R/W
D[11]	Not Used	Not Used	Set to 0	R/W
D[10]	Not Used	Not Used	Set to 0	R/W
D[09]	Not Used	Not Used	Set to 0	R/W
D[08]	Not Used	Not Used	Set to 0	R/W
D[07]	Not Used	Not Used	Set to 0	R/W
D[06]	Not Used	Not Used	Set to 0	R/W
D[05]	Not Used	Not Used	Set to 0	R/W
D[04]	Not Used	Not Used	Set to 0	R/W
D[03]	HSDIV [3]	HSDIV, MSB	R = 0	R/W
D[02]	HSDIV [2]	HSDIV	R = 0	R/W
D[01]	HSDIV [1]	HSDIV	R = 0	R/W
D[00]	HSDIV [0]	HSDIV, LSB	R = 0	R/W

Table 22 - HSDIV Address Base + 0x18 (7h)

Base + 0X18	Address	FUNCTION	Default	
Data Bus		C=7h		
D[03]	C[4]	Control bit, set to 0	Set to 0	R/W
D[02]	C[3]	Control bit, set to 1	Set to 0	R/W
D[01]	C[2]	Control bit, set to 1	Set to 0	R/W
D[00]	C[1]	Control bit, set to 1	Set to 0	R/W

7.11 Clock A Threshold Register

The Clock A Threshold Register **(0x14, Base + 0x18 = 8h) R/W,** stores the bits needed to set the Clock A zero crossing thresholds. The byte, **D[07:00]** contains programming bits for clock A zero crossing. To access this register, the software driver writes to the control bits located in **Base + 0x18 = 8h** of the VXI FPGA. This register instructs the FPGA that the next register writes the **Clock A Threshold Register** at **0x14** in the Time Base address space. The zero crossing of the clock threshold is set by programming one of four DAC's in a quad 8 bit DAC to a voltage, from –10V to +9.921875V in 78.125mV steps. The coding is offset binary and the code map is as follows:

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 23 – DAC Codes

DAC Code	Analog Output
FF	+9.921875
81	+78.125mV
80	0V
7F	-78.125mV
01	-9.921875V
00	-10.0V

Table 24 - Clock A Threshold Register Map Base + 0x14

Base + 0X14	Clock A Register Map	Default	
Data Bus	**Function**		
D[07]	CLKA_THRESH[7] , MSB	Set to 0	R/W
D[06]	CLKA_THRESH[6]	Set to 0	R/W
D[05]	CLKA_THRESH[5]	Set to 0	R/W
D[04]	CLKA_THRESH[4]	Set to 0	R/W
D[03]	CLKA_THRESH[3]	Set to 0	R/W
D[02]	CLKA_THRESH[2]	Set to 0	R/W
D[01]	CLKA_THRESH[1]	Set to 0	R/W
D[00]	CLKA_THRESH[0], LSB	Set to 0	R/W

Table 25 - Time Base Address Map Base + 0x18 (8h)

Base + 0X18	Time Base Address	FUNCTION	Default	
Data Bus	**TB CONTROL**	C=8h		
D[03]	C[4]	Control bit, set to 1	Set to 0	R/W
D[02]	C[3]	Control bit, set to 0	Set to 0	R/W
D[01]	C[2]	Control bit, set to 0	Set to 0	R/W
D[00]	C[1]	Control bit, set to 0	Set to 0	R/W

7.12 Clock B Threshold Register

The Clock B Threshold Register **(0x14, Base + 0x18 = 9h) R/W,** stores the bits needed to set the Clock B zero crossing thresholds. The byte, **D[07:00]** contains programming bits for clock B zero crossing. To access this register, the software driver writes to the control bits located in **Base + 0x18 = 9h** of the VXI FPGA. This register instructs the FPGA that the next register writes the **Clock B Threshold Register** at **0x14**. The zero crossing of the clock threshold is set by programming one of four DAC's in a quad 8 bit DAC to a voltage, from –10V to +9.921875V in 78.125mV steps. The coding is offset binary and the code map is as follows:

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 26 – DAC Codes

DAC Code	Analog Output
FF	+9.921875
81	+78.125mV
80	0V
7F	-78.125mV
01	-9.921875V
00	-10.0V

Table 27 - Clock B Threshold Register Map Base + 0x14

Base + 0X14	Clock B Register Map	Default	
Data Bus	**Function**		
D[07]	CLKB_THRESH[7] , MSB	Set to 0	R/W
D[06]	CLKB_THRESH[6]	Set to 0	R/W
D[05]	CLKB_THRESH[5]	Set to 0	R/W
D[04]	CLKB_THRESH[4]	Set to 0	R/W
D[03]	CLKB_THRESH[3]	Set to 0	R/W
D[02]	CLKB_THRESH[2]	Set to 0	R/W
D[01]	CLKB_THRESH[1]	Set to 0	R/W
D[00]	CLKB_THRESH[0], LSB	Set to 0	R/W

Table 28 - Time Base Address Map Base + 0x18 (9h)

Base + 0X18	Time Base Address	FUNCTION	Default	
Data Bus	**TB CONTROL**	C=9h		
D[03]	C[4]	Control bit, set to 1	Set to 0	R/W
D[02]	C[3]	Control bit, set to 0	Set to 0	R/W
D[01]	C[2]	Control bit, set to 0	Set to 0	R/W
D[00]	C[1]	Control bit, set to 1	Set to 0	R/W

7.13 Trigger A Threshold Register

The Trigger A Threshold Register **(0x14, Base + 0x18 = Ah) R/W,** stores the bits needed to set the Trigger A zero crossing thresholds. The byte, **D[07:00]** contains programming bits for Trigger A. To access this register, the software driver writes to the control bits located in **Base + 0x18 = Ah** of the VXI FPGA. This register instructs the FPGA that the next register writes the **Trigger A Threshold Register** at **0x14**. The zero crossing of the Trigger threshold is set by programming one of four DAC's in a quad 8 bit DAC to a voltage from –10V to +9.921875V in 78.125mV steps. The coding is offset binary and the code map is as follows:

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 29 – DAC Codes

DAC Code	Analog Output
FF	+9.921875
81	+78.125mV
80	0V
7F	-78.125mV
01	-9.921875V
00	-10.0V

Table 30 - Trigger A Threshold Register Map Base + 0x14

Base + 0X14	Trigger A Register Map	Default	
Data Bus	**Function**		
D[07]	TRGA_THRESH[7] , MSB	Set to 0	R/W
D[06]	TRGA_THRESH[6]	Set to 0	R/W
D[05]	TRGA_THRESH[5]	Set to 0	R/W
D[04]	TRGA_THRESH[4]	Set to 0	R/W
D[03]	TRGA_THRESH[3]	Set to 0	R/W
D[02]	TRGA_THRESH[2]	Set to 0	R/W
D[01]	TRGA_THRESH[1]	Set to 0	R/W
D[00]	TRGA_THRESH[0], LSB	Set to 0	R/W

Table 31 - Time Base Address Map Base + 0x18 (Ah)

Base + 0X18	Time Base Address	FUNCTION	Default	
Data Bus	**TB CONTROL**	C=Ah		
D[03]	C[4]	Control bit, set to 1	Set to 0	R/W
D[02]	C[3]	Control bit, set to 0	Set to 0	R/W
D[01]	C[2]	Control bit, set to 1	Set to 0	R/W
D[00]	C[1]	Control bit, set to 0	Set to 0	R/W

7.14 Trigger B Threshold Register

The Trigger B Threshold Register **(0x14, Base + 0x18 = Bh) R/W,** stores the bits needed to set the Trigger B zero crossing thresholds. The byte, **D[07:00]** contains programming bits for Trigger B. To access this register, the software driver writes to the control bits located in **Base + 0x18 = Bh** of the VXI FPGA. This register instructs the FPGA that the next register writes the **Trigger B Threshold Register** at **0x14**. The zero crossing of the Trigger threshold is set by programming one of four DAC's in a quad 8 bit DAC to a voltage from –10V to +9.921875V in 78.125mV steps. The coding is offset binary and the code map is as follows:

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 32 – DAC Codes

DAC Code	Analog Output
FF	+9.921875
81	+78.125mV
80	0V
7F	-78.125mV
01	-9.921875V
00	-10.0V

Table 33 - Trigger B Threshold Register Map Base + 0x14

Base + 0X14	Trigger B Register Map	Default	
Data Bus	**Function**		
D[07]	TRGB_THRESH[7] , MSB	Set to 0	R/W
D[06]	TRGB_THRESH[6]	Set to 0	R/W
D[05]	TRGB_THRESH[5]	Set to 0	R/W
D[04]	TRGB_THRESH[4]	Set to 0	R/W
D[03]	TRGB_THRESH[3]	Set to 0	R/W
D[02]	TRGB_THRESH[2]	Set to 0	R/W
D[01]	TRGB_THRESH[1]	Set to 0	R/W
D[00]	TRGB_THRESH[0], LSB	Set to 0	R/W

Table 34 - Time Base Address Map Base + 0x18 (Bh)

Base + 0X18	Time Base Address	FUNCTION	Default	
Data Bus	**TB CONTROL**			
D[03]	C[4]	Control bit, set to 1	Set to 0	R/W
D[02]	C[3]	Control bit, set to 0	Set to 0	R/W
D[01]	C[2]	Control bit, set to 1	Set to 0	R/W
D[00]	C[1]	Control bit, set to 1	Set to 0	R/W

Clock A/B and Trigger A/B Threshold Register CONTROL Lines

DAC_WR, when set low makes the selected DAC latch transparent to data on the data bus until **DAC_WR** goes high. The selected DAC is determined by the **DAC_A1** and **DAC_A0** control lines. The following table shows MAX506 DAC coding:

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 35 - Threshold DAC Decoding

DAC_WR	DAC_A[1:0]	Latch State
1	XX	Input Data Latched
0	00	Clock A Threshold
0	01	Clock B Threshold
0	10	Trigger A Threshold
0	11	Trigger B Threshold

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

8 Programming Examples:

8.1 Clock Programming Example:

Set the Sample rate to 324Hz of Time Base and connect to Channel A. Connect External Clock B to Channel B with the Clock B Threshold set to +1.8V (TTL).

Calculate Time Base Frequency:

1. Fw = 324Hz. Since the frequency is in the range of 95.36598Hz to 6.25MHz select the LSDIV divider.
2. Calculate LSDIVw = ROUND((50kHz / 324) * 1/2 * 1/8 * 4000) = 38,580.
3. Calculate Nw = ROUND((324 / 50kHz) * 2 * 8 * 38,580) = 4000.
4. Program N = 4000, HSDIV = 8, LSDIV = 38,580 and select LSDIV range.
5. Factual = Fpd * Nw / (2 * 8 * LSDIVw) = 324.0021Hz, Fε = (324.0021 – 324) / 324 = 6.4ppm <= .02%.
6. Program R Register: Since the reference frequency Fref = 10MHz and Fpd = 50kHz. R = Fref / Fpd = 10MHz / 50kHz = 200. Set R = c8h.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

7. Program N Register: Since N = Fvco / Fpd, N = [(P X B) + A]; where P = 8, N = 4000. From the formula B = 500 and A = 0. See section **Error! Reference source not found.** for restrictions on A, B and N.

Program PLL Registers:	
Program R Register:	
Write 0h to Addr 0x18	Write to R Register
Write 0019h to Addr 0x14	(200d = R => HEX(R) = c8h = R[14:1]) => D[15:0] = 0019h
Write 00h to Addr 0x16	
Program N Register:	
Write 1h to Addr 0x18	Set Address to N Register
Write 07D0h to Addr 0x14	(500d = B => HEX(B) = 1F4h = B[12:0]) => D[15:0] = 7D0h
Write 00h to Addr 0x16	(0d = A => HEX(A) = 0h = A[2:0]) => D[4:0] = 00h
Program F Register:	
Write 3h to Addr 0x18	Set Address to F Register
Write 0h to Addr 0x14	((FO_LD=1h, Counter Reset=1h => F[3:1] = 5h) => 05h) => D[4:0] = 5h
Write 04h to Addr 0x16	((FO_LD=1h, Counter Reset=0h => F[3:1] = 4h) => 04h) => D[4:0] = 4h
Program Time Base Divider:	
Program HSDIV Register:	
Write 0007h to Addr 0x18	Set Address to HSDIV Register
Write 7h to Addr 0x14	(8 = HSDIV => HEX(HSDIV - 1) =7h = HSDIV[3:0]) => D[15:0] =0007h
Program LSDIV Register:	
Write 5h to Addr 0x18	Set Address to HSDIV Register
Write 96b2h to Addr 0x14	(38,580 = LSDIV => HEX(LSDIV - 2) =96b2h = LSDIV[15:0]) => D[15:0] = 96b2h
Program Time Base Control Register:	
Write 6h to Addr 0x18	Set Address to TBC Register
Write 3100h to Addr 0x14	(REFSEL = 0, PLL_EN = 1, INTREF = 1, FREQSEL[1:0] = 1) => D[15:0] = 3100h
Program CLK B Threshold Register:	
Write 9h to Addr 0x18	Set Address to Clock Threshold Register
Write 0097h to Addr 0x14	(+1.8V = -10 + THRESH_A * .078125 => THRESH_A = 151d => 97h) => D[7:0] = 97h
Program Clock Control:	
Write 4h to Addr 0x18	Set Address to CLK Control Register
Write 0017h to Addr 0x14	EXTCLK A_EN = 0, EXTCLK B_EN = 1, CHA_CLKSEL[1:0] = 1, CHB_CLKSEL[1:0] = 3 => D[15:0] = 0017h,

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

9 Time Base Truth Table:

The following table shows the typical Clock modes that can be anticipated to be used when setting up the Time Base and Clock routing.

Table 1 – Time Base Truth table

DBS9900 Time Base Control Bits	**CLOCK CONTROL REGISTER** (0x18=4h)				**TIME BASE CONTROL REGISTER** (0x18=6h)			
TIME BASE CONTROL MODES								
	0x14(D5)	014x(D4)	0x14(D[3:2])	0x14(D[1:0])	0x14(D14)	0x14(D13)	0x14(D12)	014x(D[9:8])
INTERNAL REFERENCE	**EXTCLKA_EN**	**EXTCLKB_EN**	**CHA_CLKSEL[1:0]**	**CHB_CLKSEL[1:0]**	**REFSEL**	**PLL_EN**	**INTREF**	**FREQSEL[1:0]**
MODA off, MODB off	0	0	0h	0h	0	0	0	0h
TB => MODA, MODB off	0	0	1h	0h	0	1	1	1h, 2h, 3h
CHA => MODA, MODB off	1	0	2h	0h	0	0	0	0h
CHB => MODA, MODB off	0	1	3h	0h	0	0	0	0h
MODA off, TB => MODB	0	0	0h	1h	0	1	1	1h, 2h, 3h
TB => MODA, TB => MODB	0	0	1h	1h	0	1	1	1h, 2h, 3h
CHA => MODA, TB => MODB	1	0	2h	1h	0	1	1	1h, 2h, 3h
CHB => MODA, TB => MODB	0	1	3h	1h	0	1	1	1h, 2h, 3h
MODA off, CHA => MODB	1	0	0h	2h	0	0	0	0h
TB => MODA, CHA => MODB	1	0	1h	2h	0	1	1	1h, 2h, 3h
CHA => MODA, CHA => MODB	1	0	2h	2h	0	0	0	0h
CHB => MODA, CHA => MODB	1	1	3h	2h	0	0	0	0h
MODA off, CHB => MODB	0	1	0h	3h	0	0	0	0h
TB => MODA, CHB => MODB	0	1	1h	3h	0	1	1	1h, 2h, 3h
CHA => MODA, CHB => MODB	1	1	2h	3h	0	0	0	0h
CHB => MODA, CHB => MODB	0	1	3h	3h	0	0	0	0h

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

DBS9900 Time Base Control Bits	CLOCK CONTROL REGISTER (0x18=4h)				
TIME BASE CONTROL MODES					
	0x14(D5)	014x(D4)	0x14(D[3:2])	0x14(D[1:0])	0x14(D14)
EXTERNAL REFERENCE	**EXTCLKA_EN**	**EXTCLKB_EN**	**CHA_CLKSEL[1:0]**	**CHB_CLKSEL[1:0]**	**REFSEL**
MODA off, MODB off	0	0	0h	0h	1
TB => MODA, MODB off	0	1	1h	0h	1
CHA => MODA, MODB off	1	0	2h	0h	1
CHB => MODA, MODB off	0	1	3h	0h	1
MODA off, TB => MODB	0	1	0h	1h	1
TB => MODA, TB => MODB	0	1	1h	1h	1
CHA => MODA, TB => MODB	1	1	2h	1h	1
CHB => MODA, TB => MODB	0	1	3h	1h	1
MODA off, CHA => MODB	1	0	0h	2h	1
TB => MODA, CHA => MODB	1	1	1h	2h	1
CHA => MODA, CHA => MODB	1	0	2h	2h	1
CHB => MODA, CHA => MODB	1	1	3h	2h	1
MODA off, CHB => MODB	0	1	0h	3h	1
TB => MODA, CHB => MODB	0	1	1h	3h	1
CHA => MODA, CHB => MODB	1	1	2h	3h	1
CHB => MODA, CHB => MODB	0	1	3h	3h	1

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Artisan Technology Group is an independent supplier of quality pre-owned equipment

Gold-standard solutions

Extend the life of your critical industrial, commercial, and military systems with our superior service and support.

We buy equipment

Planning to upgrade your current equipment? Have surplus equipment taking up shelf space? We'll give it a new home.

Learn more!

Visit us at **artisantg.com** for more info on price quotes, drivers, technical specifications, manuals, and documentation.

Artisan Scientific Corporation dba Artisan Technology Group is not an affiliate, representative, or authorized distributor for any manufacturer listed herein.

We're here to make your life easier. How can we help you today?

(217) 352-9330 | sales@artisantg.com | artisantg.com

ARTISAN® TECHNOLOGY GROUP