

---

# DigESwitch

Networked Test Solutions





---

# Table of Contents

Hardware Installation .....	12
Hardware Installation - 19" rack .....	12
Software Installation.....	13
For IVI Installation Only:.....	13
Software Installation:.....	14
IVI Configuration.....	14
To Create A Hardware Asset.....	15
To Create A Driver Session.....	15
To Create A Logical Name .....	17
Configuration and Setup .....	19
DigESwitch Home Page .....	19
IP Configuration Page.....	21
Changing the IP settings on the DigESwitch box .....	25

---

Customer Support Page .....	28
Factory IP Configuration Default Settings .....	28
8-Bus Block Diagram.....	32
8-Bus Block Connector Pinout & Layout.....	33
4-Bus Block Diagram.....	34
4-Bus Block Connector Pinout & Layout.....	35
Overview.....	36
Features:.....	36
Front Panel .....	40
Power LED.....	40
Active LED.....	40
Link LED.....	41
Signal Connectors.....	41
Connector J1 (8-Bus Model).....	42

---

---

Connector J2 (8-Bus Model).....	43
Connector J3 (8-Bus Model).....	44
Connector J4 (8-Bus Model).....	45
Connector J5 (8-Bus Model).....	46
Connector J1 (4-Bus Model).....	47
Connector J2 (4-Bus Model).....	48
Connector J3 (4-Bus Model).....	49
Connector J4 (4-Bus Model).....	50
Connector J5 (4-Bus Model).....	51
Rear Panel .....	52
Specifications.....	53
DigESwitch IO Connectors (J1 – J5).....	53
Relays.....	53
System Operating Environment.....	54

---

DigESwitch Configurations .....	54
Power Requirements .....	54
Fuses .....	55
Switching Speeds.....	55
High Frequency Specifications .....	55
Box Dimensions.....	56
IVI Compliance Category .....	59
IVI-C Custom Specific Instrument Driver .....	59
Software Information:.....	60
IVI Switch Class Software .....	60
Supported Calls - Required Functions.....	60
Supported Calls - IviSwtchBase Capability Group Functions.....	61
Supported Calls - Utility Functions.....	61
Supported Calls - Set and Get Check Attribute Functions.....	61

---

---

Supported Calls - Lock and Unlock Functions.....	62
Supported Calls - Error Information Functions .....	62
Unsupported Calls - Interchangeability Checking Functions.....	63
Unsupported Calls - IviSwchScanner Extension Group Functions.....	63
Unsupported Calls - IviSwchSoftwareTrigger Extension Group Functions.....	63
Unsupported Calls - Utility Functions.....	63
IVI-C Switch Class-Compliant Specific Software .....	64
Library (.dll) Software .....	67
dlesw_Open.....	68
dlesw_Close .....	69
dlesw_GetFirmwareRev.....	70
dlesw_GetInstrumentModel.....	71
dlesw_Reset.....	72
dlesw_ConnectChan.....	73

---

dlesw_DisconnectChan.....	74
dlesw_DisconnectAll.....	75
dlesw_WriteChanImage.....	76
dlesw_ReadChanImage.....	78
dlesw_ReadBusImage.....	81
dlesw_WriteBoardImage.....	82
dlesw_ReadBoardImage.....	84
dlesw_WriteBoxImage.....	86
dlesw_ReadChanRelayStates.....	88
dlesw_ReadBusRelayStates.....	90
dlesw_ReadBoardRelayStates.....	92
dlesw_ReadBoxImage.....	93
Library Call Code Examples.....	95
OVERVIEW OF TCP API.....	103

---



TCP Message Structure .....	104
TCP Command Summary Table .....	105
TCP Command Listing.....	106
0x01 Get Firmware Rev .....	106
0x02 Board Reset.....	107
0x05 Connect Channel.....	108
0x06 Disconnect Channel.....	109
0x07 Disconnect All.....	110
0x08 Number of Boards Present.....	111
0x09 Write Channel Image.....	112
0x0A Read Channel Image.....	113
0x0B Write Bus Image.....	114
0x0C Read Bus Image .....	115
0x0D Write Board Image.....	116
0x0E Read Board Image.....	118

0x0F Read Channel Relay State.....	120
0x10 Read Bus Relay State.....	121
0x11 Read Board Relay State.....	122
0x12 Relay Update.....	123
0x1B Get Instrument Model.....	124
0x1E Write Box Image.....	125
0x1F Read Box Image.....	129
0x20 Read Box Relay States.....	130
0x21 Set Relay Break Time.....	131
Appendix A - Replacement Parts.....	132
Fuses.....	132
Relays.....	132
DigESwitch I/O Connectors (J1 – J5).....	132
Appendix B - Warranty.....	134

Appendix C - Soft Front Panel .....	136
-------------------------------------	-----

***Installation***

# Hardware Installation

## Hardware Installation - 19" rack

1. Install the (2) rack mount brackets (0000-8337) to each side of the DigESwitch box using the (6) black 8-32 (2825-0167) screws provided.
2. Mount the DigESwitch box in the 19" rack using (4) clips (1400-1041) and (4) 10-32 screws (1400-3333) provided.
3. Connect power to the DigESwitch box using the power cord provided.
4. Connect the DigESwitch box to the network using the appropriate RJ-45 cable.
5. A shielded ethernet cable and shielded accessory cables should be used whenever possible to reduce noise.

# Software Installation

There are two different software packages that can be used to program the DigESwitch. One software package installs only the DLL files necessary to program the DigESwitch from any programming language that uses the C function calling syntax. The other software package installs the necessary files for the DigESwitch to be programmed from an IVI compatible programming language. Also available are National Instruments LabView™ VIs and National Instruments LabWindows/CVI™ front panels.

### **For IVI Installation Only:**

1. Install Windows Installer Version 2.0 or later. This software can be found at : <http://www.microsoft.com/downloads/release.asp?ReleaseId=32832>
2. Install the NI IVI Compliance Package Version 2.1 or later. This includes the latest IVI Shared Components. The latest version is available from: <http://www.ni.com>.
3. Install the NI-VISA Run-time Engine Version 3.0.1 or later. The latest version available from: <http://www.ni.com>.

## **Software Installation:**

1. If installing the IVI software, install the necessary components above to meet the Required Software Components.
2. Run the dlesw\_ivi\_c\_driver.msi program on the Installation CD to install the drivers. When prompted, select the Foundation option if the IVI drivers are not required or select both to install both the C syntax and IVI drivers.
3. If the IVI drivers were installed, run the “ConfigStoreEditor Setup.exe” program on the Installation CD to install the Configuration Store Editor program.
4. Follow the DigESwitch Hardware Configuration instructions. The software is now ready to use.

## **IVI Configuration**

If installing the IVI drivers, use the Configuration Store Editor Program found at “C:\Program Files\Digalog\Configuration Store Editor and create the following setup in the order given.

### To Create A Hardware Asset

1. Right click on the “Hardware Asset” entry and select “Create New ...”. This will create a new form where the Hardware Asset information can be entered. The “Hardware Asset” entry is in the tree view on the left hand side of the Configuration Store Editor.
2. Enter a name that specifies the hardware asset that is being added. For example: “DigESwitch”.
3. Enter a description for the asset. This field is optional. An example would be “230 x 8 Matrix” or “460 x 4 Matrix.”
4. Enter the I/O Resource. The I/O Resource is in the format of “TCPIP :: (Host Address) :: 9000 :: Socket” where (Host Address) is either the IP address such as 192.168.1.213 or is the given host name.
5. The entry will be automatically saved.

### To Create A Driver Session

1. Right click on the “Sessions” entry and select “Create New ...”. This will create a new form where the Driver Session information can be entered. The



---

“Sessions” entry is in the tree view on the left hand side of the Configuration Store Editor.

2. Enter a name that specifies the driver session that is being added. For example: “DigESwitch”.
3. Enter a description for the driver session. This field is optional. An example would be “230 x 8 Matrix” or “460 x 4 Matrix.”
4. Select which Session Attributes are to be enabled.
5. Select the Software Module called dleSw.
6. Select a Hardware Asset from the list of previously created Hardware Assets. If the hardware asset for this card does not exist it must be created before finishing this Driver Session.
7. The Virtual Names tab can be used to assign virtual names to the channels on the card. This is done in the Names section. First, enter the virtual name. Then select the channel to assign that virtual name to from the drop down list. Please note, a software module needs to be selected on the Driver tab in order for the drop down list to be filled with channels names for the card. Once both are entered select “Add”. The new assignment will appear in the

## Installation

---

list of assignments on the left. An existing assignment can be deleted by selecting it from the list and then selecting “Remove”.

8. The Initial Settings tab only applies to equipment that use initial settings. Review the documentation for the equipment to see what, if any, initial settings are available.

### To Create A Logical Name

1. Right click on the “Logical Names” entry and select “Create New ...”. This will create a new form where the Logical Name information can be entered. The “Logical Names” entry is in the tree view on the left hand side of the Configuration Store Editor.
2. Enter a name that specifies the logical name that is being added. For example: “DigESwitch”. If an existing logical name is being used, select that logical name and then select the Driver Session created for this card.
3. Enter a description for the logical name. This field is optional. An example would be “230 x 8 Matrix” or “460 x 4 Matrix.”
4. Select a Driver Session from the list of previously created driver sessions.

*< This page intentionally left blank*

# Configuration and Setup



## DigESwitch Home Page

The DigESwitch Home page contains information about the DigESwitch box. The following information is displayed:

Instrument Model	The DigESwitch model number.
Manufacturer	Manufacturer name - Digalog Systems, Inc.
Serial Number	Serial number of the DigESwitch box.

Description	Description of the DigESwitch box. This string can be configured by the user and may be changed on the IP Configuration page.
Host Name	Host name as resolved by the DNS server. This field will contain the assigned IP address if no host name is assigned to the DigESwitch box.
MAC Address	Media Access Control address of the DigESwitch box.
TCP/IP Address	Current IP address assigned to the DigESwitch box.
Firmware Revision	Firmware revision of the DigESwitch box.

## Installation

---



### IP Configuration Page

The IP Configuration page allows the user to change the IP Configuration of the DigESwitch box.

The IP Configuration entries are:

Hostname	Hostname to use for all dynamic naming services. (20 characters maximum)
----------	--

Domain	Domain name to use for all dynamic naming services. (20 characters maximum)
Description	User configurable string that can be used to describe the DigESwitch box. (20 characters maximum)
TCP/IP Mode	Specifies the IP addressing mode to use. Auto mode uses the DHCP server to obtain the IP address. Manual mode attempts to connect to the network using the IP Address provided below.
IP Address	User assigned IP address. Only used if the TCP/IP Mode is manual.
Subnet Mask	Subnet mask.
Default Gateway	IP address of the default gateway. This entry may also be left blank if a default gateway is not specified.
DNS Server Addressing	Specifies the DNS Server Addressing mode. Auto mode specifies that the network addresses for the DNS servers are automatically obtained from the network. Manual mode uses the user supplied Preferred and Alternate DNS Server addresses.

## Installation

Preferred DNS Server	User supplied IP address of the preferred or primary DNS server. Entry used only with DNS Server Addressing Mode set to manual. This entry may also be left blank if the DNS Server Addressing mode is auto.
Alternate DNS Server	User supplied IP address of the alternate or secondary DNS server. Entry used only with DNS Server Addressing Mode set to manual. This entry may also be left blank if the DNS Server Addressing mode is auto or an alternate DNS server is not present.
Dynamic DNS	When enabled, the hostname requests are sent by the DigESwitch module.
Submit	Pressing the Submit button will update the IP Configuration settings. The name of any invalid entry will be marked in red. All entries must be valid before any of the IP Configuration settings are updated. Power to the DigESwitch box must be cycled before the new settings will take effect.



Reset	Resets all entries to their original settings when the page was first loaded provided that the Submit button hasn't been pressed.
-------	---

NOTE: All IP Addresses and SubNet Masks must be entered in the form of four numbers between 0 and 255 separated by periods. For example, 192.168.1.210 or 255.255.255.0.

NOTE: Power to the DigESwitch box must be cycled before the new settings will take effect.

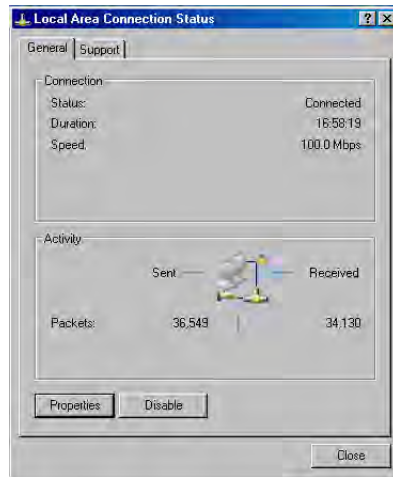
## Installation

---

### Changing the IP settings on the DigESwitch box

The DigESwitch box is configured in the factory to use a static IP Address of 197.2.2.2. The Subnet Mask is set to 255.255.255.0. The following outlines the steps required to reconfigure the IP Setting of the DigESwitch box:

1. Connect the DigESwitch box to a PC either directly using a crossover Ethernet cable or through an Ethernet switch using a standard Ethernet cable.
2. Configure the PC Network settings as follows:
  - a. From the *Start* menu, select *Settings* > *Network Connections*.
  - b. Double-click on the *Local Area Connection* icon under *LAN* or *High-Speed Internet*.
  - c. Click the *Properties* button under the *General* tab on the *Local Area Connection*



Status window.

d. Click once on *Internet Protocol (TCP/IP)*, then click the *Properties* button under the *General* tab on the *Local Area Connection Properties* window.

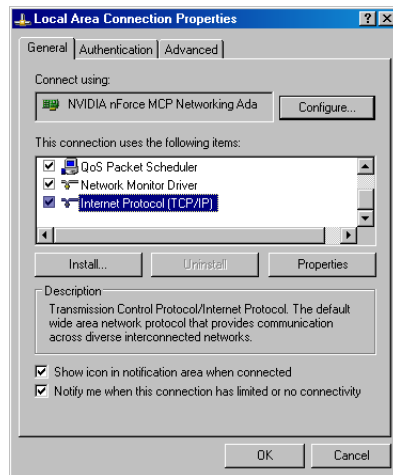
e. Under the *General* tab, click *Use the following IP address*. Enter 197.2.2.1 for the *IP address*, 255.255.255.0 for the *Subnet Mask* and leave the *Default Gateway* blank.

f. Click *Use the following DNS server addresses* and leave both entries blank.

g. Press the *OK* or *Close* buttons until you return to the *Network Connections* window.

3. Open an *Internet Explorer* window and type `http://197.2.2.2` in the address box. The *DigESwitch Home Page* should be displayed.

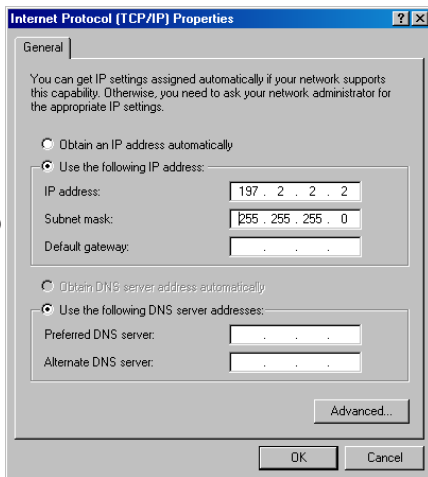
4. Click on the *LAN Configuration* link.



## Installation

5. Setup the IP Configuration as required.  
Press the Submit button when complete.
6. Return the PC to its original IP Configuration.
7. Cycle the power on the DigESwitch box to use the new IP Settings.

NOTE: Pressing the LAN Reset button on the back of the DigESwitch box will restore the factory default settings.



## Customer Support Page



**DigESwitch**  
Networked Test Solutions

[Home](#)  
[LAN Configuration](#)  
[Customer Support](#)

For help and support,  
please visit our website

### Customer Support

Corporate Website	<a href="http://www.digalogsystems.com">www.digalogsystems.com</a>
Customer Support Line	(262)785-8777
Customer Support Email	<a href="mailto:Support@DigalogSys.com">Support@DigalogSys.com</a>
Customer Support Fax	(262)797-8003

The Customer Support page lists the customer support contact information.

## Factory IP Configuration Default Settings

The factory default IP Configuration settings are:

## Installation

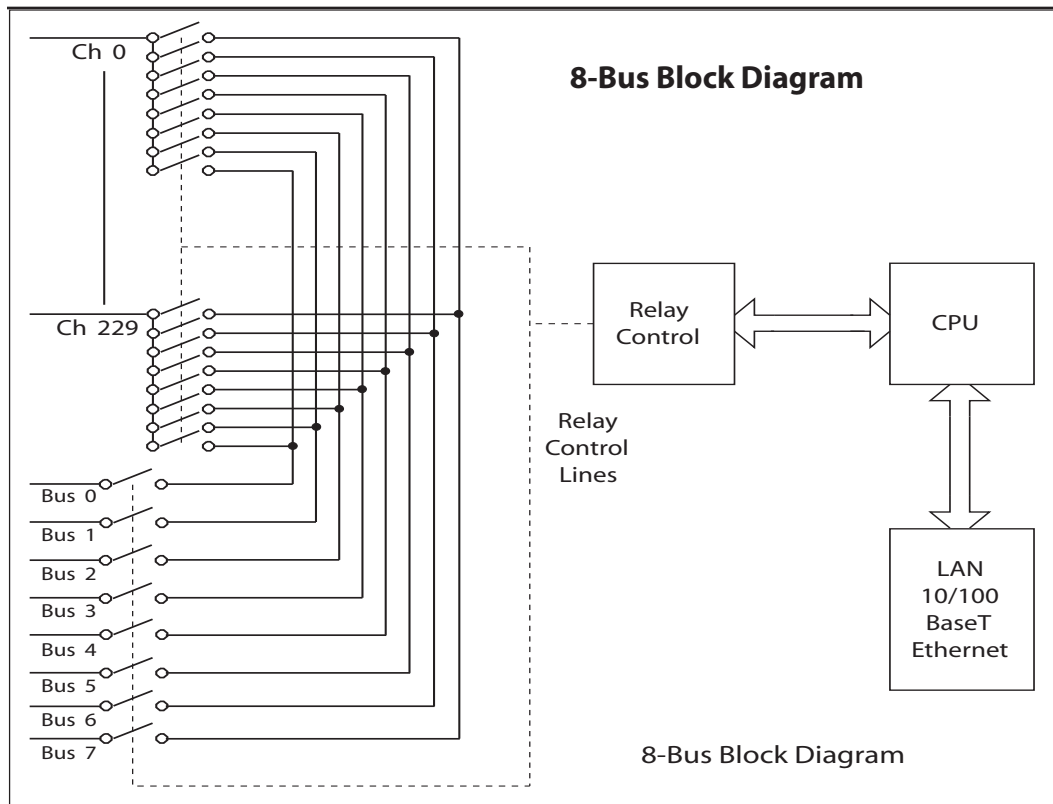
Hostname:	Blank
Domain:	Blank
Description:	230x8 Matrix Relay
TCP/IP Mode:	Manual
IP Address:	197.2.2.2
Subnet Mask:	255.255.255.0
Default Gateway:	Blank
DNS Server Addressing:	Auto
Preferred DNS Server:	Blank
Alternate DNS Server:	Blank
Dynamic DNS:	Disabled

The factory default settings can be restored by pressing the LAN Reset switch located on the back of the DigESwitch box.



***Overview***

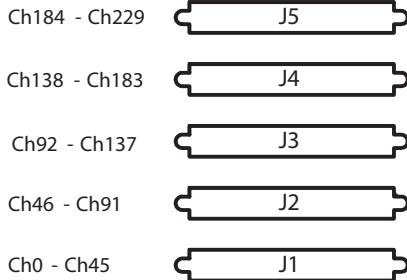




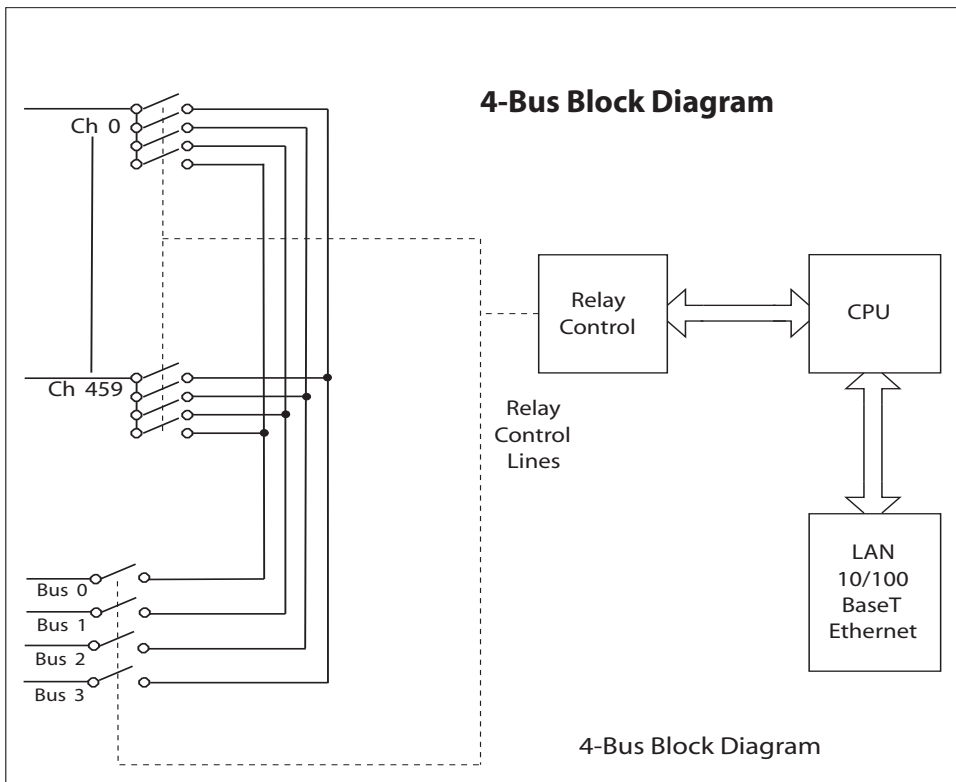
Ch23	Ch0
Ch24	Ch1
Ch25	Ch2
Ch26	Ch3
Ch27	Ch4
Ch28	Ch5
Ch29	Ch6
Ch30	Ch7
Ch31	Ch8
Ch32	Ch9
Ch33	Ch10
Ch34	Ch11
Ch35	Ch12
Ch36	Ch13
Ch37	Ch14
Ch38	Ch15
Ch39	Ch16
Ch40	Ch17
Ch41	Ch18
Ch42	Ch19
Ch43	Ch20
Ch44	Ch21
Ch45	Ch22
N/C	N/C
N/C	N/C
N/C	N/C
N/C	Bus0
N/C	Bus1
N/C	Bus2
N/C	Bus3
N/C	Bus4
N/C	Bus5
N/C	Bus6
Gnd	Bus7

Typical  
Connector  
Pinout  
(8 Bus)

## 8-Bus Block Connector Pinout & Layout



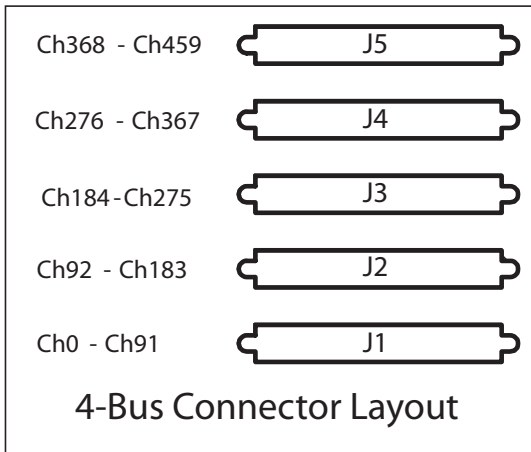
8-Bus Connector Layout



CH46	CH00
CH47	CH01
CH48	CH02
CH49	CH03
CH50	CH04
CH51	CH05
CH52	CH06
CH53	CH07
CH54	CH08
CH55	CH09
CH56	CH10
CH57	CH11
CH58	CH12
CH59	CH13
CH60	CH14
CH61	CH15
CH62	CH16
CH63	CH17
CH64	CH18
CH65	CH19
CH66	CH20
CH67	CH21
CH68	CH22
CH69	CH23
CH70	CH24
CH71	CH25
CH72	CH26
CH73	CH27
CH74	CH28
CH75	CH29
CH76	CH30
CH77	CH31
CH78	CH32
CH79	CH33
CH80	CH34
CH81	CH35
CH82	CH36
CH83	CH37
CH84	CH38
CH85	CH39
CH86	CH40
CH87	CH41
CH88	CH42
CH89	CH43
CH90	CH44
CH91	CH45
BUS2	BUS0
BUS3	BUS1

Typical  
Connector  
Pinout  
(4 Bus)

## 4-Bus Block Connector Pinout & Layout



---

# Overview

## Features:

- Up to 230 channels x 8 buses or 460 channels x 4 buses.
- Bus isolation relays
- 19" Rack Mount Chassis
- Ethernet 10/100BaseT communications
- Direct I/O Driver from C, C++
- Industry-Standard IVI Driver
- Support for National Instruments® Switch Executive™

The “DigESwitch” is a versatile and low-cost 10/100BaseT Ethernet-controlled relay switching solution. DigESwitch is housed in a compact 2U 19" rack-mount chassis, and in its full-capacity configuration features 230 channels x 8 buses or 460 channels by 4 buses. Bus lines can be individually connected and disconnected via bus isolation relays, which significantly simplifies external

## Overview

---

interface wiring. Each relay channel can switch up to 0.5A or carry up to 1.0A with 150VDC/100VAC off-state isolation.

The DigESwitch can be utilized as a standalone Ethernet appliance, or as an integrated component of existing or planned test systems. This system is designed to give users the capability of more complex routing and control of resources. Relay programming options include both a Direct I/O driver and the industry-standard IVI driver. It also supports the National Instruments® Switch Executive™.

*< This page intentionally left blank*

***Hardware***



## Front Panel

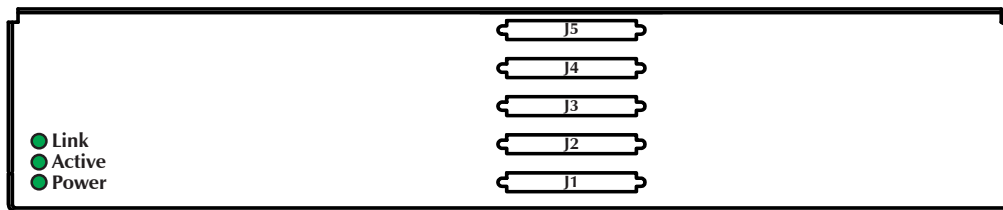
The front panel of the DigESwitch chassis houses the five (maximum configuration) board connectors and three status LEDs. The LEDs indicate the following status:

### Power LED

The green Power LED located on the front panel indicates the power status. This LED is lit when the Power Switch on the rear panel is closed and power is applied to the internal DC Power Supply.

### Active LED

The green Active LED located on the front panel indicates activity on the Ethernet bus. This LED will flash whenever transmit or receive activity occurs on the DigESwitch port.



### Link LED

The bi-color Link LED located on the front panel indicates the DigESwitch Ethernet link status. This LED is lit green if the link is okay or red if the link failed.

### Signal Connectors

The pin designations for each connector are shown below. The signal designations for each of the five different connectors are shown on the following pages.

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34

#### *8-Bus Matrix Model Connector Pinout*

35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68

01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48

#### *4-Bus Matrix Model Connector Pinout*

49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96

**Connector J1 (8-Bus Model)**

01 - CH0	18 - CH17	35 - CH23	52 - CH40
02 - CH1	19 - CH18	36 - CH24	53 - CH41
03 - CH2	20 - CH19	37 - CH25	54 - CH42
04 - CH3	21 - CH20	38 - CH26	55 - CH43
05 - CH4	22 - CH21	39 - CH27	56 - CH44
06 - CH5	23 - CH22	40 - CH28	57 - CH45
07 - CH6	24 - NC	41 - CH29	58 - NC
08 - CH7	25 - NC	42 - CH30	59 - NC
09 - CH8	26 - NC	43 - CH31	60 - NC
10 - CH9	27 - BUS 0	44 - CH32	61 - NC
11 - CH10	28 - BUS 1	45 - CH33	62 - NC
12 - CH11	29 - BUS 2	46 - CH34	63 - NC
13 - CH12	30 - BUS 3	47 - CH35	64 - NC
14 - CH13	31 - BUS 4	48 - CH36	65 - NC
15 - CH14	32 - BUS 5	49 - CH37	66 - NC
16 - CH15	33 - BUS 6	50 - CH38	67 - NC
17 - CH16	34 - BUS 7	51 - CH39	68 - GND

## Hardware

---

### Connector J2 (8-Bus Model)

01 - CH46	18 - CH63	35 - CH69	52 - CH86
02 - CH47	19 - CH64	36 - CH70	53 - CH87
03 - CH48	20 - CH65	37 - CH71	54 - CH88
04 - CH49	21 - CH66	38 - CH72	55 - CH89
05 - CH50	22 - CH67	39 - CH73	56 - CH90
06 - CH51	23 - CH68	40 - CH74	57 - CH91
07 - CH52	24 - NC	41 - CH75	58 - NC
08 - CH53	25 - NC	42 - CH76	59 - NC
09 - CH54	26 - NC	43 - CH77	60 - NC
10 - CH55	27 - BUS 0	44 - CH78	61 - NC
11 - CH56	28 - BUS 1	45 - CH79	62 - NC
12 - CH57	29 - BUS 2	46 - CH80	63 - NC
13 - CH58	30 - BUS 3	47 - CH81	64 - NC
14 - CH59	31 - BUS 4	48 - CH82	65 - NC
15 - CH60	32 - BUS 5	49 - CH83	66 - NC
16 - CH61	33 - BUS 6	50 - CH84	67 - NC
17 - CH62	34 - BUS 7	51 - CH85	68 - GND

**Connector J3 (8-Bus Model)**

01 - CH92	18 - CH109	35 - CH115	52 - CH132
02 - CH93	19 - CH110	36 - CH116	53 - CH133
03 - CH94	20 - CH111	37 - CH117	54 - CH134
04 - CH95	21 - CH112	38 - CH118	55 - CH135
05 - CH96	22 - CH113	39 - CH119	56 - CH136
06 - CH97	23 - CH114	40 - CH120	57 - CH137
07 - CH98	24 - NC	41 - CH121	58 - NC
08 - CH99	25 - NC	42 - CH122	59 - NC
09 - CH100	26 - NC	43 - CH123	60 - NC
10 - CH101	27 - BUS 0	44 - CH124	61 - NC
11 - CH102	28 - BUS 1	45 - CH125	62 - NC
12 - CH103	29 - BUS 2	46 - CH126	63 - NC
13 - CH104	30 - BUS 3	47 - CH127	64 - NC
14 - CH105	31 - BUS 4	48 - CH128	65 - NC
15 - CH106	32 - BUS 5	49 - CH129	66 - NC
16 - CH107	33 - BUS 6	50 - CH130	67 - NC
17 - CH108	34 - BUS 7	51 - CH131	68 - GND

## Hardware

---

### Connector J4 (8-Bus Model)

01 - CH138	18 - CH155	35 - CH161	52 - CH178
02 - CH139	19 - CH156	36 - CH162	53 - CH179
03 - CH140	20 - CH157	37 - CH163	54 - CH180
04 - CH141	21 - CH158	38 - CH164	55 - CH181
05 - CH142	22 - CH159	39 - CH165	56 - CH182
06 - CH143	23 - CH160	40 - CH166	57 - CH183
07 - CH144	24 - NC	41 - CH167	58 - NC
08 - CH145	25 - NC	42 - CH168	59 - NC
09 - CH146	26 - NC	43 - CH169	60 - NC
10 - CH147	27 - BUS 0	44 - CH170	61 - NC
11 - CH148	28 - BUS 1	45 - CH171	62 - NC
12 - CH149	29 - BUS 2	46 - CH172	63 - NC
13 - CH158	30 - BUS 3	47 - CH173	64 - NC
14 - CH151	31 - BUS 4	48 - CH174	65 - NC
15 - CH152	32 - BUS 5	49 - CH175	66 - NC
16 - CH153	33 - BUS 6	50 - CH176	67 - NC
17 - CH154	34 - BUS 7	51 - CH177	68 - GND

**Connector J5 (8-Bus Model)**

01 - CH184	18 - CH201	35 - CH207	52 - CH224
02 - CH185	19 - CH202	36 - CH208	53 - CH225
03 - CH186	20 - CH203	37 - CH209	54 - CH226
04 - CH187	21 - CH204	38 - CH210	55 - CH227
05 - CH188	22 - CH205	39 - CH211	56 - CH228
06 - CH189	23 - CH206	40 - CH212	57 - CH229
07 - CH190	24 - NC	41 - CH213	58 - NC
08 - CH191	25 - NC	42 - CH214	59 - NC
09 - CH192	26 - NC	43 - CH215	60 - NC
10 - CH193	27 - BUS 0	44 - CH216	61 - NC
11 - CH194	28 - BUS 1	45 - CH217	62 - NC
12 - CH195	29 - BUS 2	46 - CH218	63 - NC
13 - CH196	30 - BUS 3	47 - CH219	64 - NC
14 - CH197	31 - BUS 4	48 - CH220	65 - NC
15 - CH198	32 - BUS 5	49 - CH221	66 - NC
16 - CH199	33 - BUS 6	50 - CH222	67 - NC
17 - CH200	34 - BUS 7	51 - CH223	68 - GND

## Hardware

### Connector J1 (4-Bus Model)

001 - CH000	018 - CH017	035 - CH034	052 - CH049	069 - CH066	086 - CH083
002 - CH001	019 - CH018	036 - CH035	053 - CH050	070 - CH067	087 - CH084
003 - CH002	020 - CH019	037 - CH036	054 - CH051	071 - CH068	088 - CH085
004 - CH003	021 - CH020	038 - CH037	055 - CH052	072 - CH069	089 - CH086
005 - CH004	022 - CH021	039 - CH038	056 - CH053	073 - CH070	090 - CH087
006 - CH005	023 - CH022	040 - CH039	057 - CH054	074 - CH071	091 - CH088
007 - CH006	024 - CH023	041 - CH040	058 - CH055	075 - CH072	092 - CH089
008 - CH007	025 - CH024	042 - CH041	059 - CH056	076 - CH073	093 - CH090
009 - CH008	026 - CH025	043 - CH042	060 - CH057	077 - CH074	094 - CH091
010 - CH009	027 - CH026	044 - CH043	061 - CH058	078 - CH075	095 - BUS 2
011 - CH010	028 - CH027	045 - CH044	062 - CH059	079 - CH076	096 - BUS 3
012 - CH011	029 - CH028	046 - CH045	063 - CH060	080 - CH077	
013 - CH012	030 - CH029	047 - BUS 0	064 - CH061	081 - CH078	
014 - CH013	031 - CH030	048 - BUS 1	065 - CH062	082 - CH079	
015 - CH014	032 - CH031	049 - CH046	066 - CH063	083 - CH080	
016 - CH015	033 - CH032	050 - CH047	067 - CH064	084 - CH081	
017 - CH016	034 - CH033	051 - CH048	068 - CH065	085 - CH082	



**Connector J2 (4-Bus Model)**

001 - CH092	018 - CH109	035 - CH126	052 - CH141	069 - CH158	086 - CH175
002 - CH093	019 - CH110	036 - CH127	053 - CH142	070 - CH159	087 - CH176
003 - CH094	020 - CH111	037 - CH128	054 - CH143	071 - CH160	088 - CH177
004 - CH095	021 - CH112	038 - CH129	055 - CH144	072 - CH161	089 - CH178
005 - CH096	022 - CH113	039 - CH130	056 - CH145	073 - CH162	090 - CH179
006 - CH097	023 - CH114	040 - CH131	057 - CH146	074 - CH163	091 - CH180
007 - CH098	024 - CH115	041 - CH132	058 - CH147	075 - CH164	092 - CH181
008 - CH099	025 - CH116	042 - CH133	059 - CH148	076 - CH165	093 - CH182
009 - CH100	026 - CH117	043 - CH134	060 - CH149	077 - CH166	094 - CH183
010 - CH101	027 - CH118	044 - CH135	061 - CH150	078 - CH167	095 - BUS 2
011 - CH102	028 - CH119	045 - CH136	062 - CH151	079 - CH168	096 - BUS 3
012 - CH103	029 - CH120	046 - CH137	063 - CH152	080 - CH169	
013 - CH104	030 - CH121	047 - BUS 0	064 - CH153	081 - CH170	
014 - CH105	031 - CH122	048 - BUS 1	065 - CH154	082 - CH171	
015 - CH106	032 - CH123	049 - CH138	066 - CH155	083 - CH172	
016 - CH107	033 - CH124	050 - CH139	067 - CH156	084 - CH173	
017 - CH108	034 - CH125	051 - CH140	068 - CH157	085 - CH174	

## Hardware

---

### Connector J3 (4-Bus Model)

001 - CH184	018 - CH201	035 - CH218	052 - CH233	069 - CH250	086 - CH267
002 - CH185	019 - CH202	036 - CH219	053 - CH234	070 - CH251	087 - CH268
003 - CH186	020 - CH203	037 - CH220	054 - CH235	071 - CH252	088 - CH269
004 - CH187	021 - CH204	038 - CH221	055 - CH236	072 - CH253	089 - CH270
005 - CH188	022 - CH205	039 - CH222	056 - CH237	073 - CH254	090 - CH271
006 - CH189	023 - CH206	040 - CH223	057 - CH238	074 - CH255	091 - CH272
007 - CH190	024 - CH207	041 - CH224	058 - CH239	075 - CH256	092 - CH273
008 - CH191	025 - CH208	042 - CH225	059 - CH240	076 - CH257	093 - CH274
009 - CH192	026 - CH209	043 - CH226	060 - CH241	077 - CH258	094 - CH275
010 - CH193	027 - CH210	044 - CH227	061 - CH242	078 - CH259	095 - BUS 2
011 - CH194	028 - CH211	045 - CH228	062 - CH243	079 - CH260	096 - BUS 3
012 - CH195	029 - CH212	046 - CH229	063 - CH244	080 - CH261	
013 - CH196	030 - CH213	047 - BUS 0	064 - CH245	081 - CH262	
014 - CH197	031 - CH214	048 - BUS 1	065 - CH246	082 - CH263	
015 - CH198	032 - CH215	049 - CH230	066 - CH247	083 - CH264	
016 - CH199	033 - CH216	050 - CH231	067 - CH248	084 - CH265	
017 - CH200	034 - CH217	051 - CH232	068 - CH249	085 - CH266	

**Connector J4 (4-Bus Model)**

001 - CH276	018 - CH293	035 - CH310	052 - CH325	069 - CH342	086 - CH359
002 - CH277	019 - CH294	036 - CH311	053 - CH326	070 - CH343	087 - CH360
003 - CH278	020 - CH295	037 - CH312	054 - CH327	071 - CH344	088 - CH361
004 - CH279	021 - CH296	038 - CH313	055 - CH328	072 - CH345	089 - CH362
005 - CH280	022 - CH297	039 - CH314	056 - CH329	073 - CH346	090 - CH363
006 - CH281	023 - CH298	040 - CH315	057 - CH330	074 - CH347	091 - CH364
007 - CH282	024 - CH299	041 - CH316	058 - CH331	075 - CH348	092 - CH365
008 - CH283	025 - CH300	042 - CH317	059 - CH332	076 - CH349	093 - CH366
009 - CH284	026 - CH301	043 - CH318	060 - CH333	077 - CH350	094 - CH367
010 - CH285	027 - CH302	044 - CH319	061 - CH334	078 - CH351	095 - BUS 2
011 - CH286	028 - CH303	045 - CH320	062 - CH335	079 - CH352	096 - BUS 3
012 - CH287	029 - CH304	046 - CH321	063 - CH336	080 - CH353	
013 - CH288	030 - CH305	047 - BUS 0	064 - CH337	081 - CH354	
014 - CH289	031 - CH306	048 - BUS 1	065 - CH338	082 - CH355	
015 - CH290	032 - CH307	049 - CH322	066 - CH339	083 - CH356	
016 - CH291	033 - CH308	050 - CH323	067 - CH340	084 - CH357	
017 - CH292	034 - CH309	051 - CH324	068 - CH341	085 - CH358	

## Hardware

---

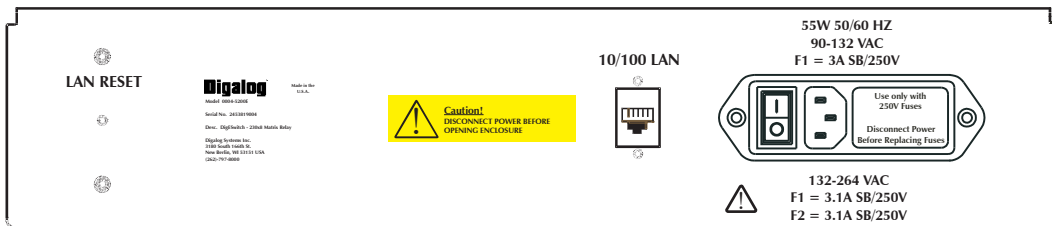
### Connector J5 (4-Bus Model)

001 - CH368	018 - CH385	035 - CH402	052 - CH417	069 - CH434	086 - CH451
002 - CH369	019 - CH386	036 - CH403	053 - CH418	070 - CH435	087 - CH452
003 - CH370	020 - CH387	037 - CH404	054 - CH419	071 - CH436	088 - CH453
004 - CH371	021 - CH388	038 - CH405	055 - CH420	072 - CH437	089 - CH454
005 - CH372	022 - CH389	039 - CH406	056 - CH421	073 - CH438	090 - CH455
006 - CH373	023 - CH390	040 - CH407	057 - CH422	074 - CH439	091 - CH456
007 - CH374	024 - CH391	041 - CH408	058 - CH423	075 - CH440	092 - CH457
008 - CH375	025 - CH392	042 - CH409	059 - CH424	076 - CH441	093 - CH458
009 - CH376	026 - CH393	043 - CH410	060 - CH425	077 - CH442	094 - CH459
010 - CH377	027 - CH394	044 - CH411	061 - CH426	078 - CH443	095 - BUS 2
011 - CH378	028 - CH395	045 - CH412	062 - CH427	079 - CH444	096 - BUS 3
012 - CH379	029 - CH396	046 - CH413	063 - CH428	080 - CH445	
013 - CH380	030 - CH397	047 - BUS 0	064 - CH429	081 - CH446	
014 - CH381	031 - CH398	048 - BUS 1	065 - CH430	082 - CH447	
015 - CH382	032 - CH399	049 - CH414	066 - CH431	083 - CH448	
016 - CH383	033 - CH400	050 - CH415	067 - CH432	084 - CH449	
017 - CH384	034 - CH401	051 - CH416	068 - CH433	085 - CH450	

## Rear Panel

The rear panel of the DigESwitch chassis houses the Reset Switch and the Power Receptacle On/Off Switch as shown below. The On/Off Switch assembly fuse(s) are rated at 3A250VAC, 3AG SloBlo (North America) or (2) 3.15A x 250VAC SloBlo (Europe). Complete specifications are listed in the next section.

The LAN Reset button resets the DigESwitch box and restores the IP configuration to the factory default settings.



Caution: Disconnect Power Before Opening Enclosure



Caution: See fuse specifications.

# Specifications

## DigESwitch IO Connectors (J1 – J5)

Mfg. P/N	AMP 2-174225-5 (8-Bus)	HONDA PSC-E96LMD (4-Bus)
Description	68-pin SCSI plug, 0.050" pitch	96-pin SCSI plug, 1.27mm pitch
Mating Connectors Source	<a href="http://www.tycoelectronics.com">www.tycoelectronics.com</a>	<a href="http://www.hondaconnectors.com">www.hondaconnectors.com</a>

## Relays

Type	SPST (Form A)
Static Contact Resistance	400m (Maximum Initial)
Max Switching Voltage	150VDC, 100VAC Peak
Max Switching Current	0.5ADC, 0.5AAC Peak
Max Switching Power	10W
Max Carry Current	1A, 1A-AC Peak
Mechanical Life Expectancy	1 x 10 <sup>9</sup>
Operating Time – Typical	2.5mS
Max number of relays on	500
Replacement Relays	Coto – 9001-05-01

---

## System Operating Environment

Operating Temperature	0 - 35°C, 32 - 95°F
Humidity	20% to 80% Relative Humidity
Altitude	less than 2000m

## DigESwitch Configurations

0004-5200A / 0004-5202A	46 x 8 / 92 x 4
0004-5200B / 0004-5202B	92 x 8 / 184 x 4
0004-5200C / 0004-5202C	138 x 8 / 276 x 4
0004-5200D / 0004-5202D	184 x 8 / 368 x 4
0004-5200E / 0004-5202E	230 x 8 / 460 x 4

## Power Requirements

Input Voltage	90 – 264VAC
Frequency	47 – 63Hz
Power	55W maximum

## Hardware

---

### Fuses

North America	3A250V, 3AG Slo-Blo, LittleFuse 313003P
Europe	(2) 3.15A250VAC, .5 x 20mm, Slo-Blo, LittleFuse 2183.15XP

### Switching Speeds

Dlesw_ConnectChan	3.2ms
Dlesw_DisconnectChan	4.0ms
Dlesw_WriteBoardImage (Normal)	8.4ms
Dlesw_WriteBoardImage(BBM)	10.0ms
Dlesw_WriteBoxImage (Normal)	30.1ms
Dlesw_WriteBoxImage(BBM)	32.1ms

### High Frequency Specifications

Gain at 2.5 MHz	-1dB
Gain at 3.5 MHz	-3dB
Interference	Less than -50dB at 2.5MHz



## Box Dimensions

19" Rackmount
2U (3.44") high x 18" deep
Weight - Less than 35Lbs. - Maximum Configuration



***IVI Switch Class***

## IVI Compliance Category

### IVI-C Custom Specific Instrument Driver

IviSwtchBase	Supported
IviSwtchScanner	Not Supported
IviSwtchSoftwareTrigger	Not Supported

### Optional Features:

Range Check	False
Query Instrument Status	False
State Caching	True
Simulate	False
Coercion Recording	False
Interchangeability Checking	False

### Driver Identification:

Driver Revision:	1.0
Driver Vendor:	Digalog Systems Inc.
Prefix:	DLESW
Description:	230x8 Matrix

**Hardware Information:**

Instrument Manufacturer:	Digalog Systems Inc.
Supported Instrument Models:	0004-5200A, B, C, D, E and 5202A ,B, C, D, E
Supported Bus Interfaces:	TCP/IP

**Software Information:**

Support Software Required:	National Instruments IVI Engine ver 2.1 or later
Source Code Availability:	Source code not available.

## IVI Switch Class Software

These calls adhere to IVI Switch Class Specification 3.0. Listed below are the calls that are supported by this driver. See the IVI Switch Class specification documents at the IVI Foundation’s web site (<http://www.ivifoundation.org>) for a full description of the calls.

**Supported Calls - Required Functions**

ViStatus IviSwitch\_init (ViRsrc logicalName, ViBoolean idQuery, ViBoolean resetDevice, ViSession \*vi);  
 ViStatus IviSwitch\_InitWithOptions (ViRsrc logicalName, ViBoolean IDQuery, ViBoolean resetDevice, ViConstString optionString, ViSession \*vi);

## IVI Switch Class

---

ViStatus IviSwTch\_close (ViSession vi);  
ViStatus IviSwTch\_reset (ViSession vi);  
ViStatus IviSwTch\_error\_message (ViSession vi, ViStatus statusCode, ViChar message[]);  
ViStatus IviSwTch\_revision\_query (ViSession vi, ViChar driverRev[], ViChar instrRev[]);

### Supported Calls - IviSwTchBase Capability Group Functions

ViStatus IviSwTch\_CanConnect (ViSession vi, ViConstString channel1, ViConstString channel2, Vilnt32 \*pathCapability);  
ViStatus IviSwTch\_Connect (ViSession vi, ViConstString channel1, ViConstString channel2);  
ViStatus IviSwTch\_Disconnect (ViSession vi, ViConstString channel1, ViConstString channel2);  
ViStatus IviSwTch\_DisconnectAll (ViSession vi);  
ViStatus IviSwTch\_GetChannelName (ViSession vi, Vilnt32 index, Vilnt32 bufferSize, ViChar name[]);  
ViStatus IviSwTch\_GetPath (ViSession vi, ViConstString channel1, ViConstString channel2, Vilnt32 bufferSize, ViChar pathList[]);  
ViStatus IviSwTch\_IsDebounced (ViSession vi, ViBoolean \*isDebounced);  
ViStatus IviSwTch\_SetPath (ViSession vi, ViConstString pathList);  
ViStatus IviSwTch\_WaitForDebounce (ViSession vi, Vilnt32 maxTime);

### Supported Calls - Utility Functions

ViStatus IviSwTch\_ResetWithDefaults (ViSession vi);  
ViStatus IviSwTch\_Disable (ViSession vi);

### Supported Calls - Set and Get Check Attribute Functions

ViStatus IviSwTch\_GetAttributeVilnt32 (ViSession vi, ViConstString channelName, ViAttr attributeld, Vilnt32 \*value);

```
ViStatus IviSwTch_SetAttributeViInt32 (ViSession vi, ViConstString channelName, ViAttr attributeld, ViInt32 value);  
ViStatus IviSwTch_GetAttributeViReal64 (ViSession vi, ViConstString channelName, ViAttr attributeld, ViReal64 *value);  
ViStatus IviSwTch_SetAttributeViReal64 (ViSession vi, ViConstString channelName, ViAttr attributeld, ViReal64 value);  
ViStatus IviSwTch_GetAttributeViString (ViSession vi, ViConstString channelName, ViAttr attributeld, ViInt32 bufferSize, ViChar value[]);  
ViStatus IviSwTch_SetAttributeViString (ViSession vi, ViConstString channelName, ViAttr attributeld, ViConstString value);  
ViStatus IviSwTch_GetAttributeViBoolean (ViSession vi, ViConstString channelName, ViAttr attributeld, ViBoolean *value);  
ViStatus IviSwTch_SetAttributeViBoolean (ViSession vi, ViConstString channelName, ViAttr attributeld, ViBoolean value);  
ViStatus IviSwTch_GetAttributeViSession (ViSession vi, ViConstString channelName, ViAttr attributeld, ViSession *value);  
ViStatus IviSwTch_SetAttributeViSession (ViSession vi, ViConstString channelName, ViAttr attributeld, ViSession value);
```

### Supported Calls - Lock and Unlock Functions

```
ViStatus IviSwTch_LockSession (ViSession vi, ViBoolean *callerHasLock);  
ViStatus IviSwTch_UnlockSession (ViSession vi, ViBoolean *callerHasLock);
```

### Supported Calls - Error Information Functions

```
ViStatus IviSwTch_GetError (ViSession vi, ViStatus *errorCode, ViInt32 bufferSize, ViChar description[]);  
ViStatus IviSwTch_ClearError (ViSession vi);
```

### Unsupported Calls - Interchangeability Checking Functions

ViStatus IviSwTch\_GetNextInterchangeWarning (ViSession vi, ViInt32 bufferSize, ViChar warning[]);  
ViStatus IviSwTch\_ClearInterchangeWarnings (ViSession vi);  
ViStatus IviSwTch\_ResetInterchangeCheck (ViSession vi);  
ViStatus IviSwTch\_GetNextCoercionRecord (ViSession vi, ViInt32 bufferSize, ViChar record[]);  
ViStatus IviSwTch\_GetSpecificDriverCHandle (ViSession vi, ViSession\* specificDriverCHandle);

### Unsupported Calls - IviSwTchScanner Extension Group Functions

ViStatus IviSwTch\_AbortScan (ViSession vi);  
ViStatus IviSwTch\_ConfigureScanList (ViSession vi, ViConstString scanList, ViInt32 scanMode);  
ViStatus IviSwTch\_ConfigureScanTrigger (ViSession vi, ViReal64 scanDelay, ViInt32 triggerInput, ViInt32 scanAdvancedOutput);  
ViStatus IviSwTch\_InitiateScan (ViSession vi);  
ViStatus IviSwTch\_IsScanning (ViSession vi, ViBoolean \*isScanning);  
ViStatus IviSwTch\_SetContinuousScan (ViSession vi, ViBoolean status);  
ViStatus IviSwTch\_WaitForScanComplete (ViSession vi, ViInt32 maxTime);

### Unsupported Calls - IviSwTchSoftwareTrigger Extension Group Functions

ViStatus IviSwTch\_SendSoftwareTrigger (ViSession vi);

### Unsupported Calls - Utility Functions

ViStatus IviSwTch\_InvalidDateAllAttributes (ViSession vi);  
ViStatus IviSwTch\_self\_test (ViSession vi, ViInt16 \*selfTestResult, ViChar selfTestMessage[]);  
ViStatus IviSwTch\_error\_query (ViSession vi, ViInt32 \*errorCode, ViChar errorMessage[]);  
ViStatus IviSwTch\_CheckAttributeViInt32 (ViSession vi, ViConstString channelName, ViAttr attributeld,



---

```
ViInt32 value);  
ViStatus IviSwTch_CheckAttributeViReal64 (ViSession vi, ViConstString channelName, ViAttr attributeld,  
ViReal64 value);  
ViStatus IviSwTch_CheckAttributeViString (ViSession vi, ViConstString channelName, ViAttr attributeld,  
ViConstString value);  
ViStatus IviSwTch_CheckAttributeViBoolean (ViSession vi, ViConstString channelName, ViAttr  
attributeld, ViBoolean value);  
ViStatus IviSwTch_CheckAttributeViSession (ViSession vi, ViConstString channelName, ViAttr attributeld,  
ViSession value);
```

## IVI-C Switch Class-Compliant Specific Software

If the IVI-C Switch class-compliant specific calls are to be used instead of the IVI-C Switch Class calls, the files for the IVI-C Switch class-compliant specific calls are placed in the directories given below:

C:\Program Files\IVI\Lib\msc	Contains the library file to include with the application
C:\Program Files\IVI\Include	Contains the header file to use with the application
C:\Program Files\IVI\Bin	Contains the DLL for the IVI Switch software

## IVI Switch Class

---

C:\Program Files\IVI\Drivers\dlesw	Contains the LabWindows/CVI function panels and help for the DigESwitch IVI class-compliant specific calls
------------------------------------	--

**Library Calls**

## Library Calls

---

### Library (.dll) Software

The Library calls support any programming language capable of calling a dynamic link library. Include files are included for the National Instruments LabWindows/CVI environment. This Library calls package is installed with the dlesw\_ivi\_c\_driver.msi installation program. The files for the Library calls are placed in the directories given below:

C:\digalog\cvi\docs\c	Contains the help documentation for the Library calls.
C:\digalog\cvi\lib\msc	Contains the library file to include with the application.
C:\digalog\cvi\include	Contains the header file to use with the application.
C:\digalog\cvi\functionpanels	Contains the LabWindows/CVI function panels for the Library calls.
C:\Windows\System32	Contains the DLL for the Library calls.

**Note:** The same functional calls work for both the 8-Bus and 4-Bus DigESwitch models. If a parameter value is passed that is not valid for a given model (i.e. Bus 7 on a 4-Bus model) the function may return an error indicating the invalid parameter. Please see the table of error codes at the end of this section.

## dlesw\_Open

This function establishes a connection to a DigESwitch box. The IP address and port number of the DigESwitch are passed into this function. This function returns a session handle to the DigESwitch box that the other calls will use to access the box.

Only one connection can be established at the same time.

The `dlesw_Close` function is used to close the connection.

**Header File:** `dlesw_foundation.h`

**CVI Declaration:** `int32 dlesw_Open (u_int32 *SessionHandle, char *LocalIP, u_int32 PortNumber);`

### Where:

#### **SessionHandle**

Returns the handle to the DigESwitch box. Note that a zero can be a valid handle.

#### **LocalIP**

This is the IP address or host name of the DigESwitch box. It can be either an alphabetical string such as `xyz.abc.com`, or a numerical string such as `192.168.7.100`.

#### **PortNumber**

The port number to the DigESwitch box, currently fixed at Port 9000.

### **dlesw\_Close**

This function terminates the connection to the DigESwitch box.

**Header File:** dlesw\_foundation.h

**CVI Declaration:** int32 dlesw\_Close (u\_int32 SessionHandle);

**Where:**

**SessionHandle**

Handle used to access the DigESwitch box. This handle is returned from dlesw\_Open().

## dlesw\_GetFirmwareRev

This function gets the firmware revision string from the DigESwitch box. The size of the character array to write the firmware revision string to must be at least 20 characters.

**Header File:** dlesw\_foundation.h

**CVI Declaration:** int32 dlesw\_GetFirmwareRev (u\_int32 SessionHandle, u\_int16 \*RevStrSize, char RevStr);

### Where:

#### SessionHandle

Handle used to access the DigESwitch box. This handle is returned from dlesw\_Open().

#### RevStrSize

Size of the RevStr character array in bytes. This parameter is updated with the actual size of the returned firmware revision string in bytes.

#### RevStr

Character array used to return the firmware revision string. This string must be at least 20 characters long.

### dlesw\_GetInstrumentModel

This function gets the model string from the DigESwitch box. The model depends on the number of boards in the box.

**Header File:** dlesw\_foundation.h

**CVI Declaration:** int32 dlesw\_GetInstrumentModel (u\_int32 SessionHandle, u\_int16 \*ModelStrSize, char \*ModelStr);

**Where:**

**SessionHandle**

Handle used to access the DigESwitch box. This handle is returned from dlesw\_Open().

**ModelStrSize**

Size of the ModelStr character array in bytes. This parameter is updated with the actual size of the returned model string in bytes.

**ModelStr**

Character array used to return the model string. This string must be at least 20 characters long.



## dlesw\_Reset

This function will open all matrix relay connections and clear all board images. This call resets all boards in the DigESwitch box.

**Header File:** dlesw\_foundation.h

**CVI Declaration:** int32 dlesw\_Reset (u\_int32 SessionHandle);

**Where:**

**SessionHandle**

Handle used to access the DigESwitch box. This handle is returned from dlesw\_Open().

### dlesw\_ConnectChan

This function will connect the given channel to the given on board bus. It will also connect the corresponding external bus pin to the given on board bus. This function uses the normal switching mode in which any relay can be connected or disconnected at will with the “break before make” feature disabled. Valid bus numbers are 0-7 for the 8-Bus DigESwitch and 0-3 for the 4-Bus model. Calling buses 4-7 with a 4-Bus model will make the function return an “Invalid Parameter” error.

**Header File:** dlesw\_foundation.h

**CVI Declaration:** int32 dlesw\_ConnectChan (u\_int32 SessionHandle, u\_int16 Chan, int16 Bus);

#### Where:

##### SessionHandle

Handle used to access the DigESwitch box. This handle is returned from dlesw\_Open().

##### Chan

The channel number to connect to one of the buses. The maximum channel number is dependent on the number of boards in the box. There are 46 or 92 channels per board, for the 8-Bus and 4-Bus models respectively. Therefore, for a one-board system the valid channel numbers are 0-45 or 0-91 respectively. For a two-board system, 0-91 and 0-183, etc.

##### Bus

The bus(es) to which the selected channel will be connected.

< 0 = All buses	0 = Bus 0	1 = Bus 1	2 = Bus 2
3 = Bus 3	4 = Bus 4	5 = Bus 5	6 = Bus 6
7 = Bus 7			

## dlesw\_DisconnectChan

This function will disconnect the given channel from the given bus. This function uses the normal switching mode in which any relay can be connected or disconnected at will with the “break before make” feature disabled. Since other channels may also be connected to a given bus pin, the dlesw\_DisconnectChan() call does not disconnect the external bus pin from the on-board bus. To disconnect the bus pins from the on-board bus, use the dlesw\_DisconnectAll() call.

**Header File:** dlesw\_foundation.h

**CVI Declaration:** int32 dlesw\_DisconnectChan (u\_int32 SessionHandle, u\_int16 Chan, int16 Bus);

### Where:

#### SessionHandle

Handle used to access the DigESwitch box. This handle is returned from dlesw\_Open().

#### Chan

The channel number to disconnect from one of the buses. The maximum channel number is dependent on the number of boards in the box. There are 46 or 92 channels per board, for the 8-Bus and 4-Bus models respectively. Therefore, for a one-board system the valid channel numbers are 0-45 or 0-91 respectively. For a two-board system, 0-91 and 0-183, etc.

#### Bus

The bus(es) from which the selected channel will be disconnected.

< 0 = All buses	0 = Bus 0	1 = Bus 1	2 = Bus 2
3 = Bus 3	4 = Bus 4	5 = Bus 5	6 = Bus 6
7 = Bus 7			

## Library Calls

---

### dlesw\_DisconnectAll

This function will disconnect all of the channels from all of the buses on the board specified.

The disconnections include every channel from all eight on-board buses and the eight external bus pins from the on-board buses.

**Header File:** dlesw\_foundation.h

**CVI Declaration:** int32 dlesw\_DisconnectAll (u\_int32 SessionHandle, int16 bdNum);

#### Where:

##### **SessionHandle**

Handle used to access the DigESwitch box. This handle is returned from dlesw\_Open().

##### **BdNum**

The board in the DigESwitch box to access.

< 0 – Update all boards in the DigESwitch box.

= 0 to 4.

The maximum board number is dependent on the number of boards in the box. Board #0 is the bottom board and corresponds to channels 0-45 on an 8-Bus model and 0-91 on a 4-Bus model.

---

## dlesw\_WriteChanImage

This function writes the image for a single channel to the on-board image. This function does not change any actual relay states. Use the `dlesw_RelayUpdate()` call to update the relay states.

This call will overwrite the existing image, so if the existing image is to be preserved, you must first read it, then OR it with the updated image and finally write the new “composite” image to the board.

Please note that this image only affects the channel to on-board bus connection. To connect an on-board bus to the external bus pin, use the `dlesw_WriteBusImage()` call.

**Header File:** `dlesw_foundation.h`

**CVI Declaration:** `int32 dlesw_WriteChanImage (u_int32 SessionHandle, u_int16 Chan, u_char ChanImage);`

### Where:

#### **SessionHandle**

Handle used to access the DigESwitch box. This handle is returned from `dlesw_Open()`.

#### **Chan**

The channel number whose image is to be changed.

= 0 to 229 (8-Bus models) or 0 to 459 (4-Bus Models).

The maximum channel number is dependent on the number of boards in the box. There are 46 channels per board on an 8-Bus model, 92 per board on a 4-Bus model. Therefore, for a one-board system the channel range is 0-45 or 0-91, for a two board system the range is 0-91 or 0-183, etc.

*(continued on next page)*

### ChanImage

Value to write to the on board image. This is an 8-bit value where each bit represents a different bus. If the bit is high the channel will be connected to the corresponding bus. The LSB is for bus #0 and the MSB is for bus #7. More than one bit can be set if the channel is to be connected to more than one bus.

For example:

A value of 0x01 will connect the channel to bus #0.

A value of 0x84 will connect the channel to buses #2 and #7.

The high-order nibble will be ignored on 4-Bus models.

---

## dlesw\_ReadChanImage

This function reads the image for a single channel from the on-board image. The upper nibble of the returned image will be empty for the 4-Bus models.

**Header File:** dlesw\_foundation.h

**CVI Declaration:** int32 dlesw\_ReadChanImage(u\_int32 SessionHandle, u\_int16 Chan, u\_char \*ChanImage);

### Where:

#### SessionHandle

Handle used to access the DigESwitch box. This handle is returned from dlesw\_Open().

#### Chan

The channel number whose image is to be read.

= 0 to 229 for 8-Bus models and 0-459 on 4-Bus models..

The maximum channel number is dependent on the number of boards in the box. There are 46 or 92 channels per board so for a one board system the channel range is 0-45 or 0-91, for a two board system the range is 0-91 or 0-183, etc.

#### ChanImage

Value read from the on board image. This is an 8-bit value where each bit represents a different bus. If a bit is high the channel will be connected to the corresponding bus. The LSB is for bus #0 and the MSB is for bus #7. More than one bit can be set if the channel is to be connected to more than one bus.

For example:

A value of 0x01 means the channel will be connected to bus #0.

A value of 0x84 means the channel will be connected to buses #2 and #7.

**Note:** This function reads the channel image, which may differ from the actual state of the channel relays.

## Library Calls

---

### dlesw\_WriteBusImage

This function writes the image for the on-board bus connection to the external bus pin for one board in the system. This function does not change any relay states. Use the `dlesw_RelayUpdate()` call to update the relay states.

This call will overwrite the existing image, so if the existing image is to be preserved, you must first read it, then OR it with the updated image and finally write the new “composite” image to the board..

Please note that this image only affects the connection between the on-board bus and the external bus pin. To connect a channel to the on-board bus, use the `dlesw_WriteChanImage()` call.

**Header File:** `dlesw_foundation.h`

**CVI Declaration:** `int32 dlesw_WriteBusImage (u_int32 SessionHandle, u_int16 BdNum, u_char *BusImage);`

#### Where:

##### **SessionHandle**

Handle used to access the DigESwitch box. This handle is returned from `dlesw_Open()`.

##### **BdNum**

The board in the system to access.  
= 0 to 4.

The maximum board number is dependent on the number of boards in the box. Board #0 is the bottom board and corresponds to channels 0-45 for the 8-Bus models and 0-91 for the 4-Bus versions.

*(continued on next page)*



### BusImage

Value to write to the on board image. This is an 8-bit value where each bit represents a different bus. If the bit is high the external bus pin will be connected to the corresponding on board bus. The LSB is for bus #0 and the MSB is for bus #7. More than one bit can be set if more than one external bus pin is to be connected.

For example:

A value of 0x01 will connect external bus #0 to on board bus #0.

A value of 0x84 will connect external buses #2 and #7 to on board buses #2 and #7.

A value of 0xFF will connect all eight external bus pins to their corresponding on board bus.

**Note:** The upper nibble will be ignored in the 4-Bus models.

### dlesw\_ReadBusImage

This function reads the connection image for the on-board buses to the external bus pins for one board in the DigESwitch box. The upper nibble of the returned image is ignored in the 4-Bus models.

**Header File:** dlesw\_foundation.h

**CVI Declaration:** int32 dlesw\_ReadBusImage (u\_int32 SessionHandle, u\_int16 BdNum, u\_char \*BusImage);

#### Where:

##### SessionHandle

Handle used to access the DigESwitch box. This handle is returned from dlesw\_Open().

##### BdNum

The board in the DigESwitch box to access.  
= 0 to 4.

The maximum board number is dependent on the number of boards in the box. Board #0 is the bottom board and corresponds to channels 0-45 or 0-91.

##### BusImage

Value read from the on board image. This is an 8-bit value where each bit represents a different bus. If the bit is high the external bus pin will be connected to the corresponding on-board bus. The LSB is for bus #0 and the MSB is for bus #7. More than one bit can be set if more than one external bus pin is connected.

For example:

A value of 0x01 means that external bus #0 will be connected to on board bus #0.

A value of 0x84 means that external buses #2 and #7 will be connected to on board buses #2 and #7.

A value of 0xFF means that all eight external bus pins will be connected to their corresponding on board bus.

---

## dlesw\_WriteBoardImage

This function writes the complete on board image for one board in the DigESwitch box. The complete image for an 8-Bus model consists of 46 bytes for the on-board bus images plus one byte for the external bus pin to on-board bus image. For the 4-Bus model the image is 93 bytes, one for each channel plus the external bus pins. This function does not change any relay states. Use the `dlesw_RelayUpdate()` call to update the relay states.

This call will overwrite the existing image, so if the existing image is to be preserved, you must first read it, then OR it with the updated image and finally write the new “composite” image to the board.

Since this call affects all 46 or 92 channels plus the external bus pin connections, the size of the image array must be 47 or 93 bytes respectively. If not, an error will be returned.

**Header File:** `dlesw_foundation.h`

**CVI Declaration:** `int32 dlesw_WriteBoardImage (u_int32 SessionHandle, u_int16 BdNum, u_int16 BdImageArrSize, u_char *BdImageArr);`

**Where:**

**SessionHandle**

Handle used to access the DigESwitch box. This handle is returned from `dlesw_Open()`.

*(continued on next page)*

### **MrlyBdNum**

The board in the DigESwitch box to access.

= 0 to 4.

The maximum board number is dependent on the number of boards in the box. Board #0 is the bottom board and corresponds to channels 0-45 or 0-91.

### **BdImageArrSize**

The size of the BdImageArr array. It must be at least 47 or 93 bytes.

### **BdImageArr**

Byte array containing the image to write to the given board. On the 8-Bus models the first byte is for channel #0, the 46th is for channel #45, and the 47th is for the external bus pin connections. On 4-Bus models the first byte is channel #0, the 92nd for channel 91 and the 93rd for the bus connections.

## dlesw\_ReadBoardImage

This function reads the complete image for one board in the DigESwitch box. The complete image for an 8-Bus DigESwitch consists of 47 bytes, one for each of the channel-to-bus images plus an additional byte for the external bus pin to on-board bus image. For a 4-Bus model, the image contains 93 bytes.

Since this call reads all 46 or 92 channels plus the external bus pin connections, the size of the image array must be 47 bytes or 93 bytes respectively. If not, an error will be returned.

**Header File:** dlesw\_foundation.h

**CVI Declaration:** int32 dlesw\_ReadBoardImage (u\_int32 SessionHandle, u\_int16 BdNum, u\_int16 BdImageArrSize, u\_char \*BdImageArr);

### Where:

#### SessionHandle

Handle used to access the DigESwitch box. This handle is returned from dlesw\_Open().

#### BdNum

The board in the DigESwitch box to access.  
= 0 to 4.

The maximum board number is dependent on the number of boards in the box. Board #0 is the bottom board and corresponds to channels 0-45 or 0-91.

#### BdImageArrSize

The size of the BdImageArr array. It must be at least 47 or 93 bytes.

*(continued on next page)*

### **BdImageArr**

Byte array containing the image read from the given board. For an 8-Bus model, the first byte is for channel #0, the 46th for channel #45, and the 47th is for the external bus pin connections. For a 4-Bus model the first byte is channel #0, the 92nd for channel 91, and the 93rd for the external bus pins.

**Note:** This function reads the board image which may differ from the actual state of the board's channel and bus relays.

---

## dlesw\_WriteBoxImage

This function writes the complete on board channel images for all boards in a DigESwitch box. The channel image consists of one byte for each of the channels in a box (up to 230 bytes for a fully configured 8-Bus DigESwitch or 460 bytes for a fully-configured 4-Bus model). This function automatically updates the images for the bus isolation relays based on the channel connections in the passed channel relay image. The actual relay states can be updated immediately using this call or at a later time using the `dlesw_RelayUpdate()` call.

**Header File:** `dlesw_foundation.h`

**CVI Declaration:** `int32 dlesw_WriteBoxImage (u_int32 SessionHandle, u_char RelayUpdate, u_int16 BoxImageArrSize, u_char BoxImageArr[]);`

### Where:

#### **SessionHandle**

Handle used to access the DigESwitch box. This handle is returned from `dlesw_Open()`.

#### **RelayUpdate**

Matrix relay state update option

=0, Do not update relay states, can be done later using `dlesw_RelayUpdate()` call.

=1, Update relay states using the normal switching mode with the “break before make” feature disabled.

=2, Update relay status using the “break before make” switching mode with a 2mS break time.

#### **BoxImageArrSize**

The size of the `BoxImageArr` array in bytes. Must be at least 46 x number of boards for an 8-Bus model or 92 x number of boards for a 4-Bus model.

*(continued on next page)*

### **BoxImageArr**

Byte array containing the image to write to the DigESwitch box. The first byte is for channel #0, the 2nd byte is for channel #1, etc. The array must contain 46 bytes per board for an 8-Bus DigESwitch and 92 bytes for a 4-Bus model, up to 230 or 460 bytes maximum respectively.



---

## dlesw\_ReadChanRelayStates

This function reads the current state for a single channel. This call is not reading the image but it is reading the actual state of the eight or four relays that connect the channel to the eight possible on board buses.

Use the `dlesw_ReadBusRelayStates()` call to read which on-board buses are actually connected to their corresponding external pin.

**Header File:** `dlesw_foundation.h`

**CVI Declaration:** `int32 dlesw_ReadChanRelayStates (u_int32 SessionHandle, u_int16 Chan, u_char *ChanRlyState);`

### Where:

#### **SessionHandle**

Handle used to access the DigESwitch box. This handle is returned from `dlesw_Open()`.

#### **Chan**

The channel number whose state is to be read.

= 0 to 229 (8-Bus models) or 0-459 (4-Bus models).

The maximum channel number is dependent on the number of boards in the box. There are 46 channels per board for the 8-Bus models and 92 on the 4-Bus models. Therefore, for a one-board system the channel range is 0-45 or 0-91, for a two board system the range is 0-91 or 0-183, etc.

*(continued on next page)*

### ChanRlyState

Value read from the board. This is an 8-bit value where each bit represents a different bus. If a bit is high the channel is connected to the corresponding bus. The LSbit is for bus #0 and the MSbit is for bus #7. More than one bit can be set if the channel is connected to more than one bus.

For example:

A value of 0x01 means the channel is currently connected to bus #0.

A value of 0x84 means the channel is currently connected to buses #2 and #7

---

## dlesw\_ReadBusRelayStates

This function reads the current state of the eight relays that connect the external bus pins to the eight on board buses. This call is not reading the image but it is reading the actual state of the eight relays that connect the external bus pin to the eight corresponding on board buses. In the 4-Bus model, the upper four bits will be 0 since there are only 4 buses.

Use the `dlesw_ReadChanRelayStates()` call to read the current state of the eight relays that connect one channel to the eight on board buses.

**Header File:** `dlesw_foundation.h`

**CVI Declaration:** `int32 dlesw_ReadBusRelayStates (u_int32 SessionHandle, u_int16 BdNum, u_char * BusRlyState);`

### Where:

#### **SessionHandle**

Handle used to access the DigESwitch box. This handle is returned from `dlesw_Open()`.

#### **BdNum**

The board in the DigESwitch box to access.  
= 0 to 4.

The maximum board number depends on the number of boards in the box. Board #0 is the bottom board and corresponds to channels 0-45 (8-Bus model) or 0-91 (4-Bus model).

*(continued on next page)*

### **BusRlyState**

Value read from the board. This is an 8-bit value where each bit represents a different bus. If the bit is high the external bus pin will be connected to the corresponding on board bus. The LSB is for bus #0 and the MSB is for bus #7. More than one bit can be set if more than one external bus pin is connected.

For example:

A value of 0x01 means external bus #0 is connected to on board bus #0.

A value of 0x84 means external buses #2 and #7 are connected to on board buses #2 and #7.

A value of 0xFF means all eight external bus pins are connected to their corresponding on board bus.

---

## dlesw\_ReadBoardRelayStates

This function reads the current state of all relays on one board. The complete state consists of 46 or 92 bytes for the 46 or 92 channel-bus states and one byte for the external bus pins to on board bus. This call is not reading the image but it is reading the actual state of all of the relays on one board.

Since this call reads all 46 or 92 channels plus the external bus pin connections, the state array must be 47 bytes for an 8-Bus DigESwitch or 93 bytes for a 4-bus model. If not, an error will be returned.

**Header File:** dlesw\_foundation.h

**CVI Declaration:** int32 dlesw\_ReadBoardRelayStates(u\_int32 SessionHandle, u\_int16 BdNum, u\_int16 BdRelayArrSize, u\_char \*BdRelayArr);

### Where:

#### SessionHandle

Handle used to access the DigESwitch box. This handle is returned from dlesw\_Open().

#### BdNum:

The board in the DigESwitch box to access.

= 0 to 4.

The maximum board number is dependent on the number of boards in the box. Board #0 is the bottom board and corresponds to channels 0-45 (8-Bus) or 0-91 (4-Bus).

#### BdRelayArrSize

Size of the BdRelayArrSize array (47 bytes for 8-Bus, 93 for 4-Bus).

#### BdRelayArr

Byte array containing the relay state read from the given board. The first byte is for channel #0, the 46th byte is for channel #46, and the 47th byte is for the external bus pin connections.

### dlesw\_ReadBoxImage

This function reads the complete on board channel image for all the boards in a DigESwitch box. The channel image consists of one byte for each of the channels in a box (up to 230 bytes for a fully configured 8-Bus DigESwitch or 460 bytes for a fully-configured 4-Bus model). Use the `dlesw_ReadBusImage` function to get the image settings for the bus isolation relays.

**Header File:** `dlesw_foundation.h`

**CVI Declaration:** `int32 dlesw_ReadBoxImage (u_int32 SessionHandle, u_int16 *BoxImageArrSize, u_char BoxImageArr[]);`

#### Where:

##### **SessionHandle**

Handle used to access the DigESwitch box. This handle is returned from `dlesw_Open()`.

##### **BoxImageArrSize**

The size of the `BoxImageArr` array in bytes. Must be at least 46 x number of installed boards for the 8-Bus models and 92 x the number of boards for the 4-Bus models. This parameter will be updated with the actual number of bytes written to the `BoxImageArr` array.

##### **BoxImageArr**

Byte array containing the channel relay image for the DigESwitch box. The first byte is for channel #0, the 2nd byte is for channel #1, etc. The image settings for the bus isolation relays can be read using the `dlesw_ReadBusImage` function.

**Note:** This function reads the board image which may differ from the actual state of the boards' channel and bus relays.

*(This page intentionally left blank)*

## Library Calls

---

### Library Call Code Examples

Example 1:

This example shows how to open a session to the DigESwitch box, manipulate the matrix connections, and then close the session.

```
unsigned int32 DleswHandle;
int DleswErr;

//Obtain a handle to the DigESwitch box.
if((DleswErr = dlesw_Open(&DleswHandle,"192.168.1.213",9000))!= 0)
{
    //An error occurred so handle it.
}

//Connect channel #3 to bus #5.
//NOTE: This will connect channel #3 to the on board bus #5 and
//the on board bus #5 to the external pin for bus #5.
if((DleswErr = dlesw_ConnectChan(DleswHandle, 3, 5)) != 0)
{
    //An error occurred so handle it.
}

//Perform the desired operation.
// ...
// ...

//Disconnect channel #3 from bus #5.
//NOTE: This will only disconnect channel #3 from the on board bus #5.
```



```
//The connection between the external pin for bus #5 and the
//on board bus #5 will remain.
if((DleswErr = dlesw_DisconnectChan(DleswHandle, 3, 5)) != 0)
{
    //An error occurred so handle it.
}

//Free the handle to the DigESwitch box.
if((DleswErr = dlesw_Close(DleswHandle)) != 0)
{
    //An error occurred so handle it.
}
```

### Example 2:

This example shows how to open a session to the DigESwitch box, manipulate the matrix connections using the channel image calls, and then close the session. NOTE: No error handling was performed in order to make this example more readable. See example 1 for error handling examples.

```
unsigned int32 DleswHandle;
int DleswErr;
unsigned char Curlmage;

//Obtain a handle to the DigESwitch box.
DleswErr = dlesw_Open(&DleswHandle, "192.168.1.213";9000);

//Configure the on board image to connect channel #32 to on board bus #0.
//First the current image for this channel needs to be read so current
//connections are not lost. Then write the combined image.
```

## Library Calls

---

```
DleswErr = dlesw_ReadChanImage(DleswHandle, 32, &CurlImage);
DleswErr = dlesw_WriteChanImage(DleswHandle, 32, CurlImage | 0x01);

//Configure the on board image to connect channel #40 to on board bus #7.
//First the current image for this channel needs to be read so current
//connections are not lost. Then write the combined image.
DleswErr = dlesw_ReadChanImage(DleswHandle, 40, &CurlImage);
DleswErr = dlesw_WriteChanImage(DleswHandle, 40, CurlImage | 0x80);

//Configure the on board image to connect the external bus pin for buses #0
//and #7 to their corresponding on board bus. The board number is #0 which
//contains channels #32 and #40 which were accessed above.
//First the current image needs to be read so current connections
//are not lost. Then write the combined image.
DleswErr = dlesw_ReadBusImage(DleswHandle, 0, &CurlImage);
DleswErr = dlesw_WriteBusImage(DleswHandle, 0, CurlImage | 0x81);

//Update the relays with the image stored above. This call will actually close
//the relays. All the relays configured above will be closed.
DleswErr = dlesw_RelayUpdate(DleswHandle, 0, 1);
//Update using the normal switching mode

//To connect an additional channel, set its bit to 1,
//all others to 0, and OR it with CurlImage.
//To disconnect a channel, set its bit to 0, all others to 1,
//and AND it with CurlImage.

//Perform the desired operation.

//Disconnect channel #32 from on board bus #0.
//First the current image for this channel needs to be read
```

```
// so current connections are not lost. Then write the combined image.
DleswErr = dlesw_ReadChanImage(DleswHandle, 32, &CurlImage);
DleswErr = dlesw_WriteChanImage(DleswHandle, 32, CurlImage & 0xFE);

//Disconnect channel #40 from on board bus #7.
//First the current image for this channel needs to be
//read so current connections are not lost. Then write the combined image.
DleswErr = dlesw_ReadChanImage(DleswHandle, 40, &CurlImage);
DleswErr = dlesw_WriteChanImage(DleswHandle, 40, CurlImage & 0x7F);

//Disconnect the external bus pin for buses #0 and #7 from their corresponding
//on board bus. The board number is #0 which contains channels #32 and #40
//which were accessed above.
//First the current image needs to be read so current
//connections are not lost. Then write the combined image.
DleswErr = dlesw_ReadBusImage(DleswHandle, 0, &CurlImage);
DleswErr = dlesw_WriteBusImage(DleswHandle, 0, CurlImage & 0x7E);

//Update the relays with the image stored above. This call will actually open
//the relays that were closed above.
DleswErr = dlesw_RelayUpdate(DleswHandle, 0, 1);
//Update using the normal switching mode

//Free the handle to the DigESwitch box.
DleswErr = dlesw_Close(DleswHandle);
```

## Library Calls

---

### Example 3:

This example shows how to open a session to the DigESwitch box, manipulate the matrix connections using the board image calls, and then close the session. NOTE: No error handling was performed in order to make this example more readable. See example 1 for error handling examples.

```
unsigned int32 DleswHandle;
int DleswErr;
unsigned char CurlImage[47];
unsigned int16 ImageArraysz;

//Obtain a handle to the DigESwitch box.
DleswErr = dlesw_Open(&DleswHandle, "192.168.1.213", 9000);
//Read the complete on board image for board #0 (channels 0-45).
ImageArraysz = 47;
dlesw_ReadBoardImage(DleswHandle, 0, &ImageArraysz, CurlImage);

//Modify the current image to disconnect channel #10 from bus #4 and to
//connect channel #11 to bus #4. In addition, guarantee that the external bus
//pin for bus #4 is connected to the on board bus.
CurlImage[10] &= 0xEF;
CurlImage[11] |= 0x10;
CurlImage[46] |= 0x10;

//Write the new complete image out to board #0.
DleswErr = dlesw_WriteBoardImage(DleswHandle, 0, 47, CurlImage);

//Update the relays with the image stored above. This call will actually close
//the relays. All the relays configured above will be closed.
DleswErr = dlesw_RelayUpdate(DleswHandle, 0, 1); //Update using the normal switching mode
```

```
//Perform the desired operation.
// ...
// ...

//Modify the current image to disconnect channel #11 from bus #4. In addition,
//disconnect the external bus pin for bus #4 from the on board bus. The image
//does not need to be read again because the 'CurlImage' array is still valid.
CurlImage[11] &= 0xEF;
CurlImage[46] &= 0xEF;

//Write the new complete image out to board #0.
DleswErr = dlesw_WriteBoardImage(DleswHandle, 0, 47, CurlImage);

//Update the relays with the new image.
DleswErr = dlesw_RelayUpdate(DleswHandle, 0, 1);
//Update using the normal switching mode

//Free the handle to the DigESwitch box.
DleswErr = dlesw_Close(DleswHandle);
```

#### Example 4:

This example shows how to open a session to the DigESwitch box, manipulate the matrix connections using the box image calls, and then close the session. NOTE: No error handling was performed in order to make this example more readable. See example 1 for error handling examples.

```
unsigned int32 DleswHandle;
int DleswErr;
unsigned char BoxImage[230];
unsigned int16 ImageArraysizes;
```

## Library Calls

---

```
//Obtain a handle to the DigESwitch box
DleswErr = dlesw_Open(&DleswHandle,"192.168.1.213",9000);

//Read the complete image from the DigESwitch box
ImageArraysize = 230;
DleswErr = dlesw_ReadBoxImage(DleswHandle, &ImageArraysize, BoxImage);

//Modify the current image to disconnect channel #10 from bus #4 and to
//connect channel #168 to bus #4.
BoxImage[10]&=0xEF;
BoxImage[168]|=0x10;

//Write the new box image to the DigESwitch box and
//update the relay driver states using normal switching mode.
DleswErr = dlesw_WriteBoxImage(DleswHandle, 1, 230, BoxImage);
//Perform the desired operation.
//...
//...

//Modify the current image to disconnect channel #168 from bus #4.
//The image does not need to be read again because the 'BoxImage'
// array is still valid.
BoxImage[168]&=0xEF;

//Write the new box image to the DigESwitch box and update
//the relay driver states using normal switching mode.
DleswErr = dlesw_WriteBoxImage(DleswHandle, 1, 230, BoxImage);

//Free the handle to the DigESwitch box
DleswErr = dlesw_Close(DleswHandle);
```

***TCP Application Programming Interface (API)***

# OVERVIEW OF TCP API

DigESwitch's Ethernet TCP communications protocol lets you control it using the TCP stack of a host computer running the operating system of your choice.

The command set detailed in the following pages gives a summary of each command's purpose along with a byte-by-byte definition of the data to be transmitted and received over TCP. The actual number of bytes involved for a particular command will in some cases depend on the exact DigESwitch model being used; i.e. the number of boards/channels installed and whether it is an 8-Bus or a 4-Bus model. These differences are noted with the commands affected.

Regardless of the host operating system or the unit model, the DigESwitch's internal microcontroller first initializes the board hardware and then enters a command loop, polling a "command received" flag to determine if the host has transmitted a command.

When the loop detects a change in the flag status, it breaks out of the "awaiting command" mode and executes the command code. Upon complete of the command, the microcontroller transmits a status byte and any return data to the host over Ethernet TCP.



## TCP Message Structure

The following table shows both the Host and Microcontroller TCP message structures. The microcontroller has a buffer to store incoming messages.

### Host Message Structure

### uC Message Structure

Byte	Description	Byte	Description
0	Command	0	Status
1	D0 - Data Byte 0	D0	Data Byte 0
2	D1 - Data Byte 1	Dn	Data Byte n
.			
.			
n	Dn - Data Byte n	Dn	Data Byte n

---

**TCP Command Summary Table**

<b>Command</b>	<b>Description</b>
0x01	Get Firmware Revision
0x02	Board Reset
0x03	Reserved
0x04	Reserved
0x05	Connect Channel
0x06	Disconnect Channel
0x07	Disconnect All
0x08	Number of Boards Present
0x09	Write Channel Image
0x0A	Read Channel Image
0x0B	Write Bus Image
0x0C	Read Bus Image
0x0D	Write Board Image
0x0E	Read Board Image
0x0F	Read Channel Relay State
0x10	Read Bus Relay State
0x11	Read Board Relay State

<b>Command</b>	<b>Description</b>
0x12	Relay Update
0x13	Reserved
0x14	Reserved
0x15	Reserved
0x16	Reserved
0x17	Reserved
0x18	Reserved
0x19	Reserved
0x1A	Reserved
0x1B	Get Instrument Model
0x1C	Reserved
0x1D	Reserved
0x1E	Write Box Image
0x1F	Read Box Image
0x20	Read Box Relay State
0x21	Set Relay Break Time

---

## TCP Command Listing

**Note:** In the following command tables, a Value/Byte type of “b” denotes a byte, and “w” denotes a word (2 bytes).

### 0x01 Get Firmware Rev

This function gets the firmware revision string from the DigESwitch box. On successful execution the function returns a byte array with a minimum length of 20.

#### Host TCP Command Packet

Byte#	Value/Type	Description
0	0x01	Command Byte

#### DigESwitch Response Packet

Byte#	Value/Type	Description
0	0x00	Status Byte / non-zero value = error code
1-n	b	Data Bytes (ASCII)

### 0x02 Board Reset

This function opens all the connections in a DigESwitch box and clears all board image locations.

#### Host TCP Command Packet

Byte#	Value/Type	Description
0	0x02	Command Byte

#### DigESwitch Response Packet

Byte#	Value/Type	Description
0	b	Status Byte / non-zero value = error code

## 0x05 Connect Channel

This function will connect the given channel to one or more on-board buses, and will also connect the corresponding external bus pins to those buses. The normal switching mode is employed, in which any relay can be connected or disconnected at will with the “break-before-make” feature disabled.

### Host TCP Command Packet

Byte#	Value/Type	Description
0	0x05	Command Byte
1-2	w	Channel (MSB-LSB)
3-4	w	Bus (MSB-LSB)

### DigESwitch Response Packet

Byte#	Value/Type	Description
0	0x00	Status Byte / non-zero value = error code

### 0x06 Disconnect Channel

This function will disconnect the given channel from one or more on-board buses. Since additional channels may be connected to the same buses, it leaves the connection in place between the on-board buses and the external bus pins.

#### Host TCP Command Packet

Byte#	Value/Type	Description
0	0x06	Command Byte
1-2	w	Channel (MSB-LSB)
3-4	w	Bus (MSB-LSB)

#### DigESwitch Response Packet

Byte#	Value/Type	Description
0	0x00	Status Byte / non-zero value = error code

## 0x07 Disconnect All

This function will disconnect all relays on the specified board.

### Host TCP Command Packet

Byte#	Value/Type	Description
0	0x07	Command Byte
1-2	w	Board (MSB-LSB)  Range: -1 (0xFF) = Disconnect all boards (0 -4) = Disconnect specified board only (See board specifications for valid channel range)

### DigESwitch Response Packet

Byte#	Value/Type	Description
0	0x00	Status Byte / non-zero value = error code

### 0x08 Number of Boards Present

This function retrieves the number of boards installed in the DigESwitch box.

#### Host TCP Command Packet

Byte#	Value/Type	Description
0	0x08	Command Byte

#### DigESwitch Response Packet

Byte#	Value/Type	Description
0	0x00	Status Byte / non-zero value = error code
1	b	Number of boards installed (1 - 5)



## 0x09 Write Channel Image

This function writes the image for a single channel into the on-board image. Only the image is written. To update the actual physical state of the relays with the newly-written image, use the Relay Update command.

### Host TCP Command Packet

Byte#	Value/Type	Description																																				
0	0x09	Command Byte																																				
1-2	w	Channel (MSB-LSB)																																				
3	b	<table border="0"> <tr> <td>Relay</td> <td>Image</td> <td>Bit</td> <td>Definitions</td> </tr> <tr> <td colspan="4">-----</td> </tr> <tr> <td>D7</td> <td>D6</td> <td>D5</td> <td>D4</td> <td>D3</td> <td>D2</td> <td>D1</td> <td>D0</td> </tr> <tr> <td colspan="4">-----</td> </tr> <tr> <td>CHxB7</td> <td>CHxB6</td> <td>CHxB5</td> <td>CHxB4</td> <td>CHxB3</td> <td>CHxB2</td> <td>CHxB1</td> <td>CHxB0</td> </tr> <tr> <td colspan="4">-----</td> </tr> <tr> <td colspan="4">CHxBn = Channel x to Bus n connection state image, which will update the relays at the next Relay Update command. Bit value 0 = channel will disconnect from bus. Bit value 1 = channel will connect to bus.</td> </tr> </table>	Relay	Image	Bit	Definitions	-----				D7	D6	D5	D4	D3	D2	D1	D0	-----				CHxB7	CHxB6	CHxB5	CHxB4	CHxB3	CHxB2	CHxB1	CHxB0	-----				CHxBn = Channel x to Bus n connection state image, which will update the relays at the next Relay Update command. Bit value 0 = channel will disconnect from bus. Bit value 1 = channel will connect to bus.			
Relay	Image	Bit	Definitions																																			
-----																																						
D7	D6	D5	D4	D3	D2	D1	D0																															
-----																																						
CHxB7	CHxB6	CHxB5	CHxB4	CHxB3	CHxB2	CHxB1	CHxB0																															
-----																																						
CHxBn = Channel x to Bus n connection state image, which will update the relays at the next Relay Update command. Bit value 0 = channel will disconnect from bus. Bit value 1 = channel will connect to bus.																																						

### DigESwitch Response Packet

Byte#	Value/Type	Description
0	0x00	Status Byte / non-zero value = error code

## 0x0A Read Channel Image

This function retrieves the image for a single channel from the on-board image. The retrieved image may not correspond to the actual physical state of the channel relays if a new image has previously been written but the Relay Update command has not yet been used to to change the relay states.

### Host TCP Command Packet

Byte#	Value/Type	Description																																				
0	0x0A	Command Byte																																				
1-2	w	Channel (MSB-LSB)																																				
3	b	<table border="0"> <tr> <td>Relay</td> <td>Image</td> <td>Bit</td> <td>Definitions</td> </tr> <tr> <td colspan="4">-----</td> </tr> <tr> <td>D7</td> <td>D6</td> <td>D5</td> <td>D4</td> <td>D3</td> <td>D2</td> <td>D1</td> <td>D0</td> </tr> <tr> <td colspan="4">-----</td> </tr> <tr> <td>CHxB7</td> <td>CHxB6</td> <td>CHxB5</td> <td>CHxB4</td> <td>CHxB3</td> <td>CHxB2</td> <td>CHxB1</td> <td>CHxB0</td> </tr> <tr> <td colspan="4">-----</td> </tr> <tr> <td colspan="4">CHxBn = Channel x to Bus n connection state image, which will update the relays at the next Relay Update command. Bit value 0 = channel is disconnected from bus. Bit value 1 = channel is connected to bus.</td> </tr> </table>	Relay	Image	Bit	Definitions	-----				D7	D6	D5	D4	D3	D2	D1	D0	-----				CHxB7	CHxB6	CHxB5	CHxB4	CHxB3	CHxB2	CHxB1	CHxB0	-----				CHxBn = Channel x to Bus n connection state image, which will update the relays at the next Relay Update command. Bit value 0 = channel is disconnected from bus. Bit value 1 = channel is connected to bus.			
Relay	Image	Bit	Definitions																																			
-----																																						
D7	D6	D5	D4	D3	D2	D1	D0																															
-----																																						
CHxB7	CHxB6	CHxB5	CHxB4	CHxB3	CHxB2	CHxB1	CHxB0																															
-----																																						
CHxBn = Channel x to Bus n connection state image, which will update the relays at the next Relay Update command. Bit value 0 = channel is disconnected from bus. Bit value 1 = channel is connected to bus.																																						

### DigESwitch Response Packet

Byte#	Value/Type	Description
0	0x00	Status Byte / non-zero value = error code

## 0x0B Write Bus Image

This function writes a new bus relay image into the on-board image. Only the image is written. To update the actual physical state of the relays with the newly-written image, use the Relay Update command.

### Host TCP Command Packet

Byte#	Value/Type	Description
0	0x0B	Command Byte
1-2	w	Board (MSB-LSB) (Valid values 0 - 4)
3	b	Bus Isolation Relay Image Bit Definitions ----- D7 D6 D5 D4 D3 D2 D1 D0 ----- Bus7 Bus6 Bus5 Bus4 Bus3 Bus2 Bus1 Bus0 ----- Bus $n$ = Bus isolation relay state, which will update the relays at the next Relay Update command. Bit value 0 = isolate bus from external connection. Bit value 1 = connect bus to external connection.

### DigESwitch Response Packet

Byte#	Value/Type	Description
0	0x00	Status Byte / non-zero value = error code

## 0x0C Read Bus Image

This function reads the bus relay image from the on-board image. The retrieved image may not correspond to the actual physical state of the bus isolation relays if a new image has previously been written but the Relay Update command has not yet been used to to change the relay states.

### Host TCP Command Packet

Byte#	Value/Type	Description
0	0x0C	Command Byte
1-2	w	Board (MSB-LSB) (Valid values 0 - 4)

### DigESwitch Response Packet

Byte#	Value/Type	Description																																										
0	0x00	Status Byte / non-zero value = error code																																										
1	b	<table border="0"> <tr> <td>Bus</td> <td>Isolation</td> <td>Relay</td> <td>Image</td> <td>Bit</td> <td>Definitions</td> </tr> <tr> <td colspan="6">-----</td> </tr> <tr> <td>D7</td> <td>D6</td> <td>D5</td> <td>D4</td> <td>D3</td> <td>D2 D1 D0</td> </tr> <tr> <td colspan="6">-----</td> </tr> <tr> <td>Bus7</td> <td>Bus6</td> <td>Bus5</td> <td>Bus4</td> <td>Bus3</td> <td>Bus2 Bus1 Bus0</td> </tr> <tr> <td colspan="6">-----</td> </tr> <tr> <td colspan="6">Busn = Bus isolation relay state, which will update the relays at the next Relay Update command. Bit value 0 = isolate bus from external connection. Bit value 1 = connect bus to external connection.</td> </tr> </table>	Bus	Isolation	Relay	Image	Bit	Definitions	-----						D7	D6	D5	D4	D3	D2 D1 D0	-----						Bus7	Bus6	Bus5	Bus4	Bus3	Bus2 Bus1 Bus0	-----						Busn = Bus isolation relay state, which will update the relays at the next Relay Update command. Bit value 0 = isolate bus from external connection. Bit value 1 = connect bus to external connection.					
Bus	Isolation	Relay	Image	Bit	Definitions																																							
-----																																												
D7	D6	D5	D4	D3	D2 D1 D0																																							
-----																																												
Bus7	Bus6	Bus5	Bus4	Bus3	Bus2 Bus1 Bus0																																							
-----																																												
Busn = Bus isolation relay state, which will update the relays at the next Relay Update command. Bit value 0 = isolate bus from external connection. Bit value 1 = connect bus to external connection.																																												

## 0x0D Write Board Image

This function writes the complete board image for one board in the DigESwitch box. The complete image consists of 47 or 93 bytes (8-Bus and 4-Bus models respectively), one byte for each of the channels on the board plus one more for the external bus connections.

### Host TCP Command Packet

Byte#	Value/Type	Description																																								
0	0x0D	Command Byte																																								
1-2	w	Board (MSB-LSB) (Valid values 0 - 4)																																								
3-4	w	Number of board connections (MSB, LSB)																																								
5 - 50 (8-Bus model) or 5 - 96 (4-Bus model)	b	<table border="0"> <tr> <td>Relay</td> <td>Image</td> <td>Bit</td> <td>Definitions</td> </tr> <tr> <td colspan="4">-----</td> </tr> <tr> <td>D7</td> <td>D6</td> <td>D5</td> <td>D4</td> <td>D3</td> <td>D2</td> <td>D1</td> <td>D0</td> </tr> <tr> <td colspan="4">-----</td> </tr> <tr> <td>CHxB7</td> <td>CHxB6</td> <td>CHxB5</td> <td>CHxB4</td> <td>CHxB3</td> <td>CHxB2</td> <td>CHxB1</td> <td>CHxB0</td> </tr> <tr> <td colspan="4">-----</td> </tr> <tr> <td colspan="8">           CHxBn = Channel x to Bus n connection state image, which will update the relays at the next Relay Update command. Bit value 0 = channel is disconnected from bus. Bit value 1 = channel is connected to bus. Buses 7 - 4 are ignored in the 4-Bus model.         </td> </tr> </table>	Relay	Image	Bit	Definitions	-----				D7	D6	D5	D4	D3	D2	D1	D0	-----				CHxB7	CHxB6	CHxB5	CHxB4	CHxB3	CHxB2	CHxB1	CHxB0	-----				CHxBn = Channel x to Bus n connection state image, which will update the relays at the next Relay Update command. Bit value 0 = channel is disconnected from bus. Bit value 1 = channel is connected to bus. Buses 7 - 4 are ignored in the 4-Bus model.							
Relay	Image	Bit	Definitions																																							
-----																																										
D7	D6	D5	D4	D3	D2	D1	D0																																			
-----																																										
CHxB7	CHxB6	CHxB5	CHxB4	CHxB3	CHxB2	CHxB1	CHxB0																																			
-----																																										
CHxBn = Channel x to Bus n connection state image, which will update the relays at the next Relay Update command. Bit value 0 = channel is disconnected from bus. Bit value 1 = channel is connected to bus. Buses 7 - 4 are ignored in the 4-Bus model.																																										

*(Table continued on next page)*

Byte#	Value/Type	Description
51 or 97	b	Bus Isolation Relay Image Bit Definitions <hr/> D7 D6 D5 D4 D3 D2 D1 D0 <hr/> Bus7 Bus6 Bus5 Bus4 Bus3 Bus2 Bus1 Bus0 <hr/> Bus $n$ = Bus isolation relay state, which will update the relays at the next Relay Update command. Bit value 0 = isolate bus from external connection. Bit value 1 = connect bus to external connection. Buses 7 - 4 are ignored in the 4-Bus model.

**DigESwitch Response Packet**

Byte#	Value/Type	Description
0	0x00	Status Byte / non-zero value = error code

## 0x0E Read Board Image

This function reads the complete board image for one board in the DigESwitch box. The complete image consists of 47 or 93 bytes (8-Bus and 4-Bus models respectively), one byte for each of the channels on the board plus one more for the external bus connections. The retrieved image may not correspond to the actual physical state of the board relays if a new image has previously been written but the Relay Update command has not yet been used to change the relay states.

### Host TCP Command Packet

Byte#	Value/Type	Description
0	0x0E	Command Byte
1-2	w	Board (MSB-LSB) (Valid values 0 - 4)

### DigESwitch Response Packet

Byte#	Value/Type	Description																																				
0	0x00	Status Byte / non-zero value = error code																																				
1 - 46 or 1 - 92	b	<table border="0"> <tr> <td>Relay</td> <td>Image</td> <td>Bit</td> <td>Definitions</td> </tr> <tr> <td colspan="4">-----</td> </tr> <tr> <td>D7</td> <td>D6</td> <td>D5</td> <td>D4</td> </tr> <tr> <td>D3</td> <td>D2</td> <td>D1</td> <td>D0</td> </tr> <tr> <td colspan="4">-----</td> </tr> <tr> <td>CHxB7</td> <td>CHxB6</td> <td>CHxB5</td> <td>CHxB4</td> </tr> <tr> <td>CHxB3</td> <td>CHxB2</td> <td>CHxB1</td> <td>CHxB0</td> </tr> <tr> <td colspan="4">-----</td> </tr> <tr> <td colspan="4">CHxBn = Channel x to Bus n connection state image, which will update the relays at the next Relay Update command. Bit value 0 = channel is disconnected from bus. Bit value 1 = channel is connected to bus.</td> </tr> </table>	Relay	Image	Bit	Definitions	-----				D7	D6	D5	D4	D3	D2	D1	D0	-----				CHxB7	CHxB6	CHxB5	CHxB4	CHxB3	CHxB2	CHxB1	CHxB0	-----				CHxBn = Channel x to Bus n connection state image, which will update the relays at the next Relay Update command. Bit value 0 = channel is disconnected from bus. Bit value 1 = channel is connected to bus.			
Relay	Image	Bit	Definitions																																			
-----																																						
D7	D6	D5	D4																																			
D3	D2	D1	D0																																			
-----																																						
CHxB7	CHxB6	CHxB5	CHxB4																																			
CHxB3	CHxB2	CHxB1	CHxB0																																			
-----																																						
CHxBn = Channel x to Bus n connection state image, which will update the relays at the next Relay Update command. Bit value 0 = channel is disconnected from bus. Bit value 1 = channel is connected to bus.																																						

*(Table continued on next page)*

Byte#	Value/Type	Description
47 or 93	b	<p>Bus Isolation Relay Image Bit Definitions</p> <hr/> <p>D7 D6 D5 D4 D3 D2 D1 D0</p> <hr/> <p>Bus7 Bus6 Bus5 Bus4 Bus3 Bus2 Bus1 Bus0</p> <hr/> <p>Busn = Bus isolation relay state, which will update the relays at the next Relay Update command. Bit value 0 = isolate bus from external connection. Bit value 1 = connect bus to external connection. Any data returned for buses 7 - 4 may be ignored in the 4-Bus model.</p>



## 0x0F Read Channel Relay State

This function retrieves the actual current channel-bus relay states (not just the image) for the specified channel.

### Host TCP Command Packet

Byte#	Value/Type	Description
0	0x0F	Command Byte
1-2	w	Channel (MSB-LSB) (Valid values 0 - 229 for 8-Bus DigESwitch, 0 - 459 for 4-Bus model)

### DigESwitch Response Packet

Byte#	Value/Type	Description
0	0x00	Status Byte / non-zero value = error code
1	b	Channel Status value read. An 8-bit value representing the channel-bus connections, with the MSB being Bus 7 and the LSB Bus 0. A "1" in a bit position indicates that the channel is connected to that bus, while a "0" indicates that the channel is disconnected from that bus. The upper nibble representing buses 7 - 4 can be ignored on the 4-Bus DigESwitch.

## 0x10 Read Bus Relay State

This function retrieves the actual current state (not just the image) of the eight relays that connect the external bus pins to the eight onboard buses.

### Host TCP Command Packet

Byte#	Value/Type	Description
0	0x10	Command Byte
1-2	w	Board (MSB-LSB) (Valid values 0 - 4)

### DigESwitch Response Packet

Byte#	Value/Type	Description
0	0x00	Status Byte / non-zero value = error code
1	b	Bus connection status value read. An 8-bit value representing the channel-bus connections, with the MSB being Bus 7 and the LSB Bus 0. A "1" in a bit position indicates that the on-board bus is connected to the external bus pin, while a "0" indicates that the on-board bus is disconnected from the external pin. The upper nibble representing buses 7 - 4 can be ignored on the 4-Bus DigESwitch.

## 0x11 Read Board Relay State

This function retrieves the actual current state (not just the image) of the all the relays on the designated board. The complete state consists of 47 bytes (8-Bus DigESwitch) or 93 bytes (4-Bus model), one byte for each channel on the board plus another byte for the external bus connection relays.

### Host TCP Command Packet

Byte#	Value/Type	Description
0	0x11	Command Byte
1-2	w	Board (MSB-LSB) (Valid values 0 - 4)

### DigESwitch Response Packet

Byte#	Value/Type	Description
0	0x00	Status Byte / non-zero value = error code
1 - 46 or 1-92	b	Channel connection status value read. An 8-bit value representing the channel-bus connections, with the MSB being Bus 7 and the LSB Bus 0. A "1" in a bit position indicates that the channel is connected to the on-board bus, while a "0" indicates that it is disconnected from that bus.
47 or 93	b	External bus connections status, with the MSB being Bus 7 and the LSB Bus 0. A "1" in a bit position indicates that the on-board bus is connected to the external bus pin, while a "0" indicates that it is disconnected.

## 0x12 Relay Update

This function updates the actual physical state of all relays on the selected board or boards by writing the on-board channel image information to the relay driver chips.

### Host TCP Command Packet

Byte#	Value/Type	Description
0	0x12	Command Byte
1-2	w	Board (MSB-LSB)  -1 (0xFF) = Disconnect all relays in box. (0 - 4) = Update the selected board only. (see board specifications for the channel range)
3	b	Mode:  1 = Update relay states using the "normal" switching mode. 2 = Update relay states using "break-before-make" mode (2ms delay)

### DigESwitch Response Packet

Byte#	Value/Type	Description
0	0x00	Status Byte / non-zero value = error code

## 0x1B Get Instrument Model

This function retrieves the model number of the DigESwitch box. The number corresponds to the number of boards installed in the box. The character array passed, which will contain the returned model number string, must be at least 20 characters in length.

### Host TCP Command Packet

Byte#	Value/Type	Description
0	0x1B	Command Byte

### DigESwitch Response Packet

Byte#	Value/Type	Description
0	0x00	Status Byte / non-zero value = error code
1 - 20	b	Model number string

## 0x1E Write Box Image

This function writes the complete on-board image for all boards in the DigESwitch box. The image consists of one byte for every channel in the box. The actual relay states may be simultaneously updated if desired, or this function can write only the image, with the intention that the actual relay states will be updated later using the Relay Update function.

### Host TCP Command Packet

Byte#	Value/Type	Description
0	0x1E	Command Byte
1	b	Relay Update Type  0 = Do not update actual relay states, just write image. 1 = Update relay states using normal switching mode. 2 = Update relay states using break-before-make switching mode (2ms delay).
2 -> (2n+1)	w	The number of connections (activated relays) per board, including channel-internal bus connections plus bus isolation relays. For example, if a board has one channel connected to three internal buses, a second channel connected to the other five internal buses, and all eight internal buses connected to the external bus pins, the total number of connections on the board is 16 (3 + 5 + 8).  <b><i>n = number of boards in the DigESwitch box.</i></b>

(Table continued on next page)

Byte#	Value/Type	Description
<u>Board #</u> <u>Byte Range</u> 0: (2n+2) -> (2n+47) 1: (2n+49) -> (2n+94) 2: (2n+96) -> (2n+141) 3: (2n+143) -> (2n+188) 4: (2n+190) -> (2n+235)  <b>(8-Bus DigESwitch)</b>	b	Channel Relay Image Bit Definitions ----- D7   D6   D5   D4   D3   D2   D1   D0 ----- CHxB7 CHxB6 CHxB5 CHxB4 CHxB3 CHxB2 CHxB1 CHxB0 ----- CHxBn = Channel x to Bus n connection state image, which will update the relays at the next Relay Update command. Bit value 0 = channel is disconnected from bus. Bit value 1 = channel is connected to bus.
<u>Board #</u> <u>Byte Range</u> 0: (2n+48) 1: (2n+95) 2: (2n+142) 3: (2n+189) 4: (2n+236)  <b>(8-Bus DigESwitch)</b>	b	Bus Isolation Relay Image Bit Definitions ----- D7   D6   D5   D4   D3   D2   D1   D0 ----- Bus7   Bus6   Bus5   Bus4   Bus3   Bus2   Bus1   Bus0 ----- Busn = Bus isolation relay state, which will update the relays at the next Relay Update command. Bit value 0 = isolate bus from external connection. Bit value 1 = connect bus to external connection.

(Table continued on next page)

Byte#	Value/Type	Description
<u>Board #</u> <u>Byte Range</u> 0: (2n+2) -> (2n+93) 1: (2n+95) -> (2n+186) 2: (2n+188) -> (2n+279) 3: (2n+281) -> (2n+372) 4: (2n+374) -> (2n+465)  <b>(4-Bus DigESwitch)</b>	b	Channel Relay Image Bit Definitions ----- D7    D6    D5    D4    D3    D2    D1    D0 ----- CHxB7 CHxB6 CHxB5 CHxB4 CHxB3 CHxB2 CHxB1 CHxB0 ----- CHxBn = Channel x to Bus n connection state image, which will update the relays at the next Relay Update command. Bit value 0 = channel is disconnected from bus. Bit value 1 = channel is connected to bus. Buses B7-B4 are ignored.
<u>Board #</u> <u>Byte Range</u> 0:            (2n+94) 1:            (2n+187) 2:            (2n+280) 3:            (2n+373) 4:            (2n+466)  <b>(4-Bus DigESwitch)</b>	b	Bus Isolation Relay Image Bit Definitions ----- D7    D6    D5    D4    D3    D2    D1    D0 ----- Bus7   Bus6   Bus5   Bus4   Bus3   Bus2   Bus1   Bus0 ----- Busn = Bus isolation relay state, which will update the relays at the next Relay Update command. Bit value 0 = isolate bus from external connection. Bit value 1 = connect bus to external connection. Bus7 - Bus4 are ignored.

(Table continued on next page)



**DigESwitch Response Packet**

<b>Byte#</b>	<b>Value/Type</b>	<b>Description</b>
0	0x00	Status Byte / non-zero value = error code

## 0x1F Read Box Image

This function reads the complete on-board image for all boards in the DigESwitch box. The image consists of one byte for every channel in the box, up to 230 bytes for the 8-Bus DigESwitch and up to 460 bytes for the 4-Bus model. The retrieved image may not correspond to the actual physical state of the box relays if a new image has previously been written but the Relay Update command has not yet been used to change the relay states. Use the Read Bus Image command to get the image of the bus isolation relays for a given board.

### Host TCP Command Packet

Byte#	Value/Type	Description
0	0x1F	Command Byte

### DigESwitch Response Packet

Byte#	Value/Type	Description
0	0x00	Status Byte / non-zero value = error code
1 -> (n * 46) or 1-> (n * 92)	b	Channel Relay Image Bit Definitions ----- D7   D6   D5   D4   D3   D2   D1   D0 ----- CHxB7 CHxB6 CHxB5 CHxB4 CHxB3 CHxB2 CHxB1 CHxB0 ----- CHxBn = Channel x to Bus n connection state at the next Relay Update command. Bit value 0 = channel is disconnected from bus. Bit value 1 = channel is connected to bus. Buses B7-B4 are ignored for the 4-Bus model. <b>n = the number of boards in the DigESwitch</b>

## 0x20 Read Box Relay States

This function reads the actual current state of all channel relays on all boards in the DigESwitch box. The relay state consists of one byte for each of the channels in the box, up to 230 bytes for a fully-configured 8-Bus model and 460 for a fully-configured 4-Bus model. To read the current state of the bus isolation relays, use the Read Bus Relay States command.

### Host TCP Command Packet

Byte#	Value/Type	Description
0	0x20	Command Byte

### DigESwitch Response Packet

Byte#	Value/Type	Description
0	0x00	Status Byte / non-zero value = error code
1 -> (n * 46) or 1-> (n * 92)	b	Channel Relay Image Bit Definitions ----- D7 D6 D5 D4 D3 D2 D1 D0 ----- CHxB7 CHxB6 CHxB5 CHxB4 CHxB3 CHxB2 CHxB1 CHxB0 ----- CHxBn = Channel x to Bus n connection state. Bit value 0 = channel is disconnected from bus. Bit value 1 = channel is connected to bus. Buses B7-B4 are ignored for the 4-Bus model. <b>n = the number of boards in the DigESwitch</b>

## 0x21 Set Relay Break Time

This command sets the break time for the “break-before-make” function which can be used when updating relay states using Mode 2 of the Relay Update and Write Box Image commands. These commands have a default break time of 2ms, so you only need to use Set Relay Break Time if you want a longer break.

### Host TCP Command Packet

Byte#	Value/Type	Description
0	0x21	Command Byte
1-2	w	Relay break time in milliseconds (valid values: 2 -> 500)

### DigESwitch Response Packet

Byte#	Value/Type	Description
0	0x00	Status Byte / non-zero value = error code

## Appendix A - Replacement Parts

### Fuses

North America	3A/250V, 3AG Slo-Blo, LittleFuse 313003P
Europe	(2) 3.15A/250VAC, .5 x 20mm, Slo-Blo, LittleFuse 2183.15XP

### Relays

Type	SPST (Form A)
Replacement Relays	Coto – 9001-05-01

### DigESwitch I/O Connectors (J1 – J5)

Mfg. P/N	AMP 2-174225-5 (8-Bus)	HONDA PSC-E96LMD (4-Bus)
Description	68-pin SCSI plug, 0.050" pitch	96-pin SCSI plug, 1.27mm pitch
Mating Connectors Source	<a href="http://www.tycoelectronics.com">www.tycoelectronics.com</a>	<a href="http://www.hondaconnectors.com">www.hondaconnectors.com</a>

**NOTE:** DigESwitch contains no user-serviceable parts other than the fuse. Disassembly and repair in the field is not supported.



## Appendix B - Warranty

Equipment manufactured by Digalog Systems is warranted free from defects in materials and workmanship for a period of one year from the date of delivery unless otherwise specified. Fuses, operating supplies, and other expendable parts are excluded from this warranty.

This warranty does not apply to defects caused by negligence, misuse, unauthorized modification, improper installation, accidents, or abnormal operating conditions. Digalog Systems disclaims any implied warranty of marketability or suitability of their products for a particular purpose.

Liability under this warranty is limited to the replacement of defective parts and/or the servicing of the equipment to return it to its normal operating condition. Equipment repaired under this warranty is to be returned to an authorized service center or to the factory. No equipment is to be returned to the factory or to a service center without prior authorization from Digalog Systems. Any equipment returned after authorization has been given will be shipped prepaid and insured at buyer's expense. All circuit boards to be returned to Digalog Systems must be shipped in static shielding bags, packed in anti-static packaging material. Digalog Systems will pay the return freight, within the continental United States, where a warranty adjustment is made.

## **Appendix B - Warranty**

---

Digalog Systems is not liable for consequential damages.

If Digalog Systems determines that no failure exists or that damage was created by negligence or misuse, the user will be invoiced for all repair and troubleshooting costs. Equipment not subject to warranty adjustment will be returned at buyer's expense.

Any defect or malfunction reported to Digalog Systems within one year after delivery and found to be within the scope of this warranty will be repaired by Digalog Systems. Charges for labor, parts, and material will be borne by Digalog Systems. Notification to Digalog Systems of problems with equipment must include model number, complete serial number, and problem(s) encountered.

Digalog Systems reserves the right to make design changes at any time without incurring any obligation to re-equip previously purchased systems with the same.

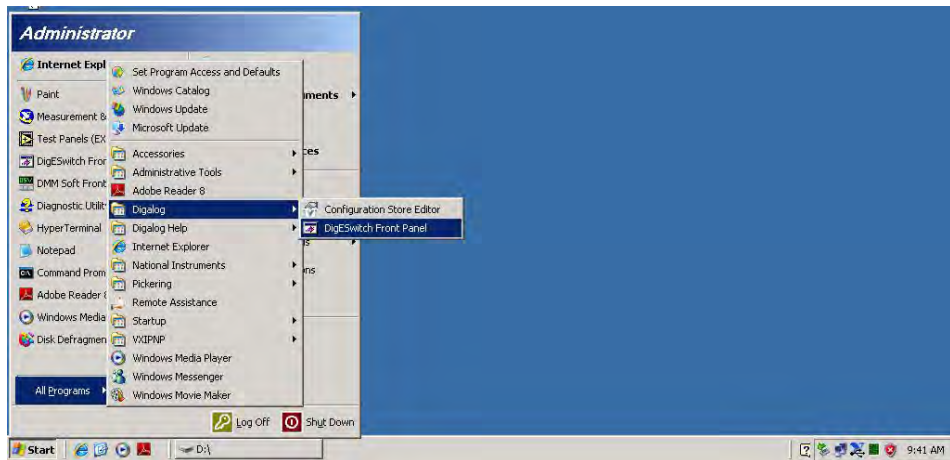
THIS WARRANTY IS IN LIEU OF ALL OTHER OBLIGATIONS OR LIABILITIES ON THE PART OF DIGALOG SYSTEMS. DIGALOG SYSTEMS DOES NOT AUTHORIZE ANY OTHER PARTY TO ASSUME ANY OTHER REPRESENTATION OR WARRANTY ON DIGALOG SYSTEM'S BEHALF, NOR DOES IT AUTHORIZE ANY OTHER PARTY TO ASSUME LIABILITY IN CONNECTION WITH THE SALE OF DIGALOG SYSTEM'S PRODUCTS.



## Appendix C - Soft Front Panel

**Purpose:** The DigESwitch SFP lets you directly connect and disconnect any of the unit's channel and bus relays in any desired combination with just a few mouse clicks. This makes it easy to test and confirm switching paths during automated test development and during system troubleshooting. The SFP does not require any programming knowledge and can therefore be used by test engineers and maintenance technicians alike.

**Starting the SFP:** Click Start -> Programs -> Digalog -> DigESwitch Soft Front Panel to run the SFP as shown below:



## Appendix C - Soft Front Panel

**Using the SFP:** The SFP opens into an introductory screen as shown on the left below, which explains how to use it and how to interpret the displayed channel and bus status indicators. The first step is to enter the DigESwitch's correct IP address in the indicated box. This address will have been determined when the DigESwitch was first configured in accordance with the "Configuration and Setup" section of this manual.

Next, click the Reset button to ensure that all relays are initially disconnected. This will also bring up the main screen as shown on the right. From here, simply use the Chan and Bus spinner controls to select channels and the buses to which you want to connect them. As explained in the last paragraph of the introductory screen, the connection state of each channel and bus will be displayed as a hex value. In the right-hand figure, Channel 0 is connected to Bus 0 and Channel 1 is connected to Bus 0 and Bus 1.

