United Electronic Industries

The High-Performance Alternative

DNA/DNR-SL-508 Serial Line Communication Module

—

User Manual

8-port RS-232 or RS-422/485 (Serial Port)
Board for the PowerDNA Cube or DNR RACKtangle I/O Chassis

December 2009 Edition
Version 1.4
PN Man-DNx-SL-508-1209

Information furnished in this manual is believed to be accurate and reliable. However, no responsibility is assumed for its use, or for any infringements of patents or other rights of third parties that may result from its use.

All product names listed are trademarks or trade names of their respective companies.

See UEI's website for complete terms and conditions of sale:

http://www.ueidaq.com/company/terms.aspx

Contacting United Electronic Industries

Mailing Address:

27 Renmar Avenue
Walpole, MA 02081
U.S.A.

For a list of our distributors and partners in the US and around the world, please see
http://www.ueidaq.com/partners/

Support:

Telephone: (508) 921-4600
Fax: (508) 668-2350

Also see the FAQs and online "Live Help" feature on our web site.

Internet Support:

Support	support@ueidaq.com
Web-Site	www.ueidaq.com
FTP Site	ftp://ftp.ueidaq.com

Product Disclaimer:

WARNING!

DO NOT USE PRODUCTS SOLD BY UNITED ELECTRONIC INDUSTRIES, INC. AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS.

Products sold by United Electronic Industries, Inc. are not authorized for use as critical components in life support devices or systems. A critical component is any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness. Any attempt to purchase any United Electronic Industries, Inc. product for that purpose is null and void and United Electronic Industries Inc. accepts no liability whatsoever in contract, tort, or otherwise whether or not resulting from our or our employees' negligence or failure to detect an improper purchase.

Note: Specifications shown in this document are subject to change without notice. Please check with UEI for current status.

Table of Contents

Chapter 1 Introduction 1
1.1 Organization of this manual 1
1.2 Frequently Asked Questions 2
1.3 DNx-SL-508 Module 2
1.4 What is Serial Communication? 3
1.5 Architecture 4
1.6 Module Capabilities 6
1.7 Wiring & Connectors 6
1.8 Preferred Configuration Using an Accessory Terminal Panel 8
1.8.1 Pinout of STP-SL-508 Panel 9
1.9 Jumper Settings for DNA Version 9
Chapter 2 Programming with High-Level API 11
2.1 Creating a Session 11
1.2 Configuring the Serial Ports 11
1.3 Configuring the Timing 12
1.4 Reading Data 12
1.5 Writing Data 12
1.6 Cleaning-up the session 12
Chapter 3 Programming with the Low-Level API 13
Appendix A – Accessories 14
Index 15

Table of Figures

Chapter 1 Introduction 1
1-1 Photos of DNR-SL-508 and DNA-SL-508 Modules 2
1-2 RS-485 Topologies 3
1-3 UART Data Frames for RS-232 and RS-485 4
1-4 Logic Block Diagram: DNA/DNR-SL-508 Overview 4
1-5 DNx-SL-508 Pinout Diagram 7
1-6 DNx-STP-SL-508 Screw Terminal Panel Connections 8
1-7 Diagram of DNA-SL-508 Layer Position Jumper Settings 10
1-8 Physical Layout of DNA-SL-508 Layer Base Board (60x) 10

Chapter 1 Introduction

This document outlines the feature-set and operation of the UEI DNA/DNR-SL-508 Serial Line Communication module. This I/O module offers eight independent serial interfaces software-configurable as RS-232 or RS-422/485.

The module is offered in two versions: one for the PowerDNA Cube called the DNA-SL-508 and another for use with the PowerDNR RACKtangle I/O Chassis called the DNR-SL-508. The two versions are functionally identical, but differ in the type of bus connector used. Both versions are typically used with the DNA / DNR-STP-SL-508 accessory screw terminal panel, which serves as a convenient connection interface between the 62-pin DB connector on the module and the individual cables for each of the eight serial lines.

Be sure that you have the PowerDNA Software Suite installed before you attempt to run any examples.

1.1 Organization of this manual

This DNx-SL-508 User Manual is organized as follows:

- **Introduction**
  This section provides an overview of the document content.
- **The SL-508 Module (2 boards)**
  This section provides an overview of the device architecture, connectivity, and logic of the board.
- **Wiring & Connectors**
  This section provides wiring schemes, notes, and specifications.
- **Programming with the High-Level API**
  This section explains how to create a session, how to configure the serial port bus communication of the board, and how to interpret results on the SL-508 series module.
- **Programming with the Low-level API**
  This section describes Low-Level API commands for configuring and using the SL-508 series module.
- **Appendix – Accessories**
  This appendix provides a list of accessories available for SL-508 module.
- **Index**
  This is an alphabetical listing of topics covered in this manual.

Manual Conventions

To help you get the most out of this manual and our products, please note that we use the following conventions:

Tips are designed to highlight quick ways to get the job done, or reveal good ideas you might not discover on your own.

NOTE: Notes alert you to important information.

CAUTION! Caution advises you of precautions to take to avoid injury, data loss, and damage to your boards or a system crash.

Text formatted in bold typeface generally represents text that should be entered verbatim. For instance, it can represent a command, as in the following example: “You can instruct users how to run setup using a command such as setup.exe.”

1.2 Frequently Asked Questions

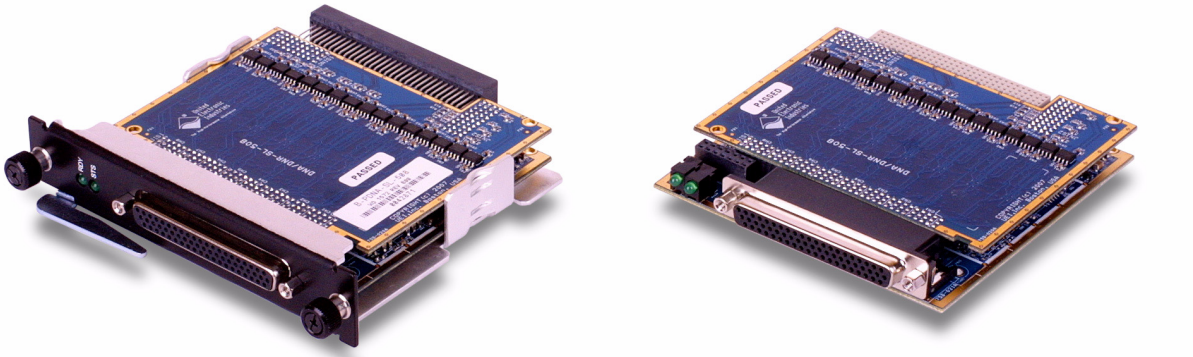
For frequently answered questions, application notes, and support, visit us online:

http://www.ueidaq.com/faq/

1.3 DNx-SL-508 Module

The DNx-SL-508 modules have the following features:

- Eight (8) independent ports
- Each port software-configurable as RS-232 or RS-422/485
- Max. speed of 256Kbit/s for RS-232 and 1Mbit/s for RS-422/485
- Completely independent bit rate settings for every port
- 350V isolation between ports, ports and circuitry; 15kV ESD
- Compatible with RS-422 networks when used in RS-485 mode
- Half- and full-duplex support for RS-485



DNR-SL-508 **DNA-SL-508**

Figure 1-1. Photos of DNR-SL-508 and DNA-SL-508 Modules

1.4 What is Serial Communication?

A serial port sends and receives data one bit at a time over one line (composed of a send, a receive, and one common ground wire).

RS-232 is a standard for serial binary data interconnection between a DTE (Data terminal equipment) and a DCE (Data communication equipment) and normally operates in a bipolar range of –10 to 10V.

RS-485 (a.k.a EIA-485) is a physical layer electrical specification of a two-wire, half-duplex, multipoint serial connection. A full duplex RS-485 system can be constructed by using two twisted-pair connections (transmit/receive pairs) together as shown in **Figure 1-2**. UART data frames for RS-232 and RS-485 are shown in **Figure 1-3**.

RS- is an abbreviation for "Recommended Standard".

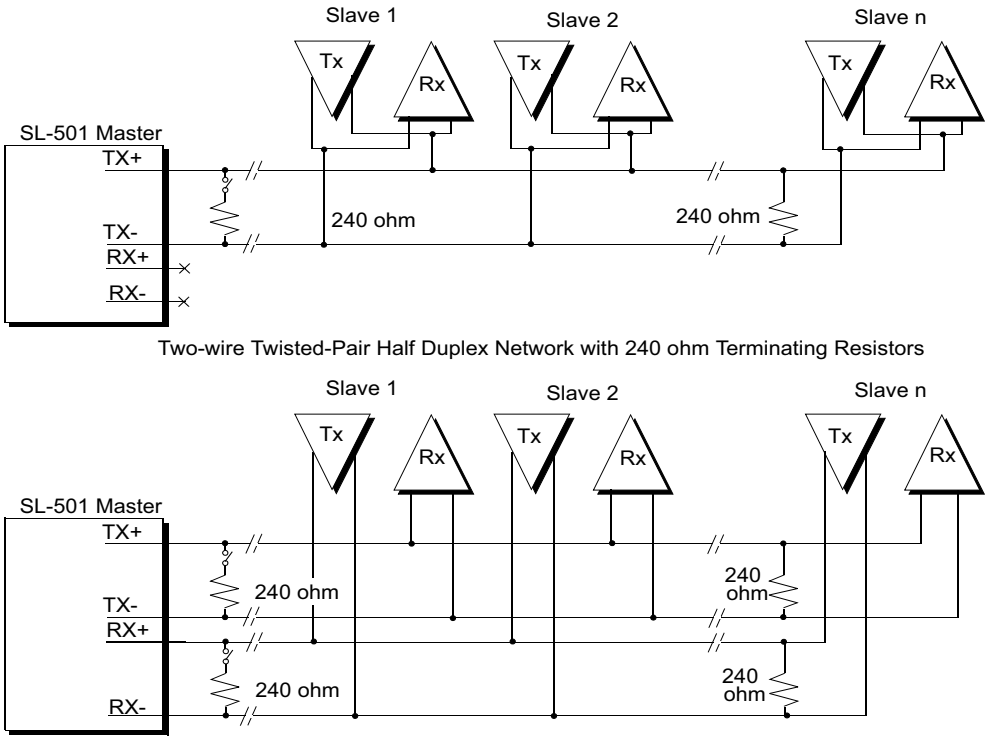


Figure 1-2. RS-485 Topologies

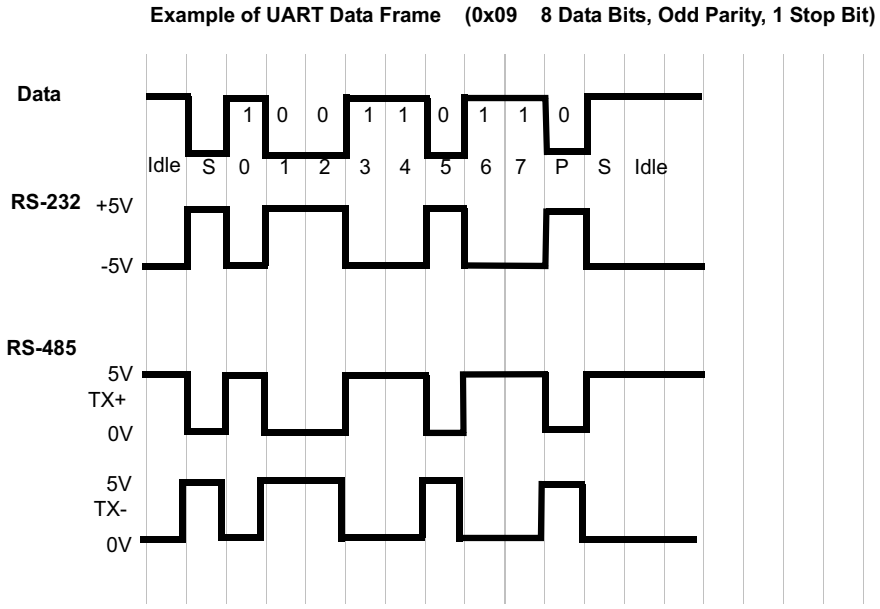


Figure 1-3. UART Data Frames for RS-232 and RS-485

1.5 Architecture

The architecture of the DNx-SL-508 is illustrated in the block diagram shown in **Figure 1-4**.

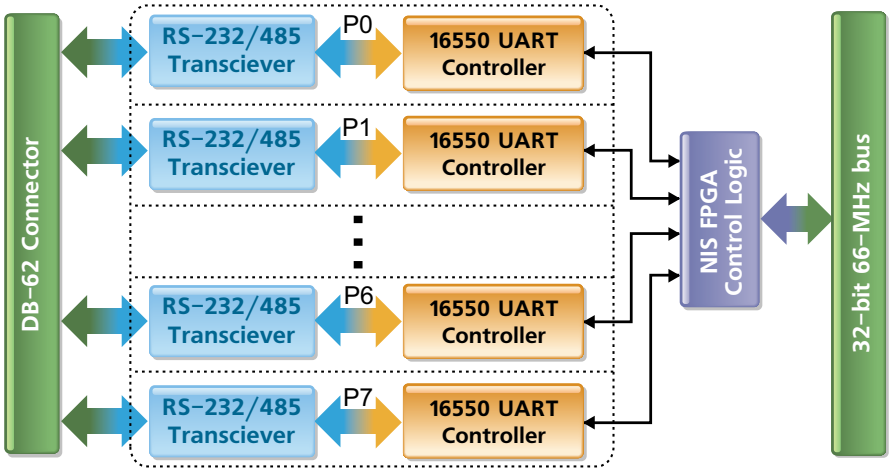


Figure 1-4. Logic Block Diagram: DNA/DNR-SL-508 Overview

An electrical signal on an RS-232/485/422 serial line flows through a DB-9 connector (or a terminal block on an DNx-STP-SL-508 accessory panel) to a DB-62 connector on the accessory panel, through a 62-conductor cable (DNA-CBL-62) to a mating connector on the SL-508 module, and then to a MAX3106E serial transceiver chip. The MAX3106E acts as an interface to the UART16550 serial interface controller — it assists the UART16550 by handling transmission/reception to/from the serial line.

The transceiver and controller are isolated from each other by a high-speed isolation IC capable of withstanding 350V channel-to-channel or 15kV ESD. There are eight MAX3106E » isolation » UART16550 structures, one per port; isolation is per-port.

The UART16550 is in turn controlled by a FPGA Control Chip, the board control chip. FPGA works in conjunction with the core module logic of the DNA or DNR chassis.

The SL-508 Module consists of two PC boards, one of which is a 60x Base Board and another, which is the eight port 508-specific I/O board.The 508-specific board plugs into a bus connector on the base board.

1.6 Module Capabilities

Using the RS-232 or RS-485 standard, the controller is capable of communicating at speeds of 256Kbit/s for RS-232 and 1Mbit/s for RS-485. When in RS-485 mode, the SL-508 is compatible with RS-422 networks.

The UART16550 runs at a base-block frequency of 66MHz, with a FIFO size of 1024 bytes.

Each port has independently programmable:

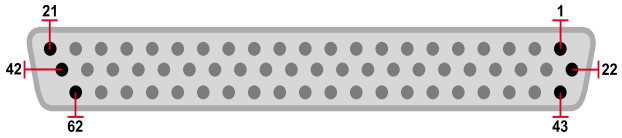
- Baud/bit rate
- UART interrupt
- Timeout interrupt
- TX/RX FIFO interrupt
- Error interrupts (4 per port)

1.7 Wiring & Connectors

The following signals are located at the DB-62 connector on the SL-508 module:

- GNDn - Isolated ground for the corresponding serial port
- TXDn/RXDn RS-232: Transmit/Receive
- RTSn/CTSn RS-232: Request to Send/Clear to Send
- TXn+/TXn– RS-422/485: Transmit pair
- RXn+/RXn– RS-422/485: Receive pair

The 62-pin female D-Sub connector on the SL-508 module is divided into eight 9-pin serial ports as shown in the pinout of **Figure 1-5**. A user can connect eight serial lines to this connector either through a custom made cable or by connecting to a DNA / DNR-STP-SL-508 accessory panel, as described in “Preferred Configuration Using an Accessory Terminal Panel” on page 8.



Pin	232 signal	422 signal	Pin	232 signal	422 signal	Pin	232 signal	422 signal
1	-	-	22	Gnd1	Gnd1	43	CTS1	RX1-
2	RTS1	TX1+	23	TX1	TX1-	44	RX1	RX1+
3	-	-	24	-	-	45	Gnd2	Gnd2
4	RX2	RX2+	25	CTS2	RX2-	46	-	-
5	RTS2	TX2+	26	TX2	TX2-	47	-	-
6	Gnd3	Gnd3	27	-	-	48	-	-
7	RX3	RX3+	28	CTS3	RX3-	49	-	-
8	RTS3	TX3+	29	TX3	TX3-	50	-	-
9	RX4	RX4+	30	CTS4	RX4-	51	Gnd4	Gnd4
10	RTS4	TX4+	31	TX4	TX4-	52	-	-
11	-	-	32	-	-	53	CTS5	RX5-
12	RTS5	TX5+	33	TX5	TX5-	54	RX5	RX5+
13	-	-	34	Gnd5	Gnd5	55	Gnd6	Gnd6
14	RX6	RX6+	35	CTS6	RX6-	56	-	-
15	RTS6	TX6+	36	TX6	TX6-	57	-	-
16	Gnd7	Gnd7	37	-	-	58	-	-
17	RX7	RX7+	38	CTS7	RX7-	59	-	-
18	RTS7	TX7+	39	TX7	TX7-	60	-	-
19	-	-	40	Gnd8	Gnd8	61	CTS8	RX8-
20	RTS8	TX8+	41	TX8	TX8-	62	RX8	RX8+
21	-	-	42	-	-			

Gndn	Isolated ground for the corresponding Serial Port “n”
TXn+/**RX**n	RS-232 Transmit/Receive, Port n
RTSn/**CTS**n	RS-232: Request To Send/Clear To Send
TXn+/**TX**n–	RS-485: Transmit Pair, Port n
RXn+/**RX**n–	RS-485:Receive Pair, Port n
—	No Internal Connection

Figure 1-5. DNx-SL-508 Pinout Diagram

1.8 Preferred Configuration Using an Accessory Terminal Panel

A DNA-CBL-62 62-conductor cable (see **Figure 1-6**) is available to connect the SL-508 module to a DNA / DNR-STP-SL-508 accessory terminal panel. The DNx-STP-SL-508 accessory panel provides a convenient interface for connecting the eight serial cables for the serial ports to the SL-508 module. The panel accepts eight 9-pin DB-9 connectors and also has eight screw terminal blocks that may be used for individual wire connections, if preferred. The eight DB-9 cable connectors are labeled JS1 to JS8 and the eight 5-terminal screw terminal blocks are labeled JT1 through JT8, as shown in **Figure 1-6**.

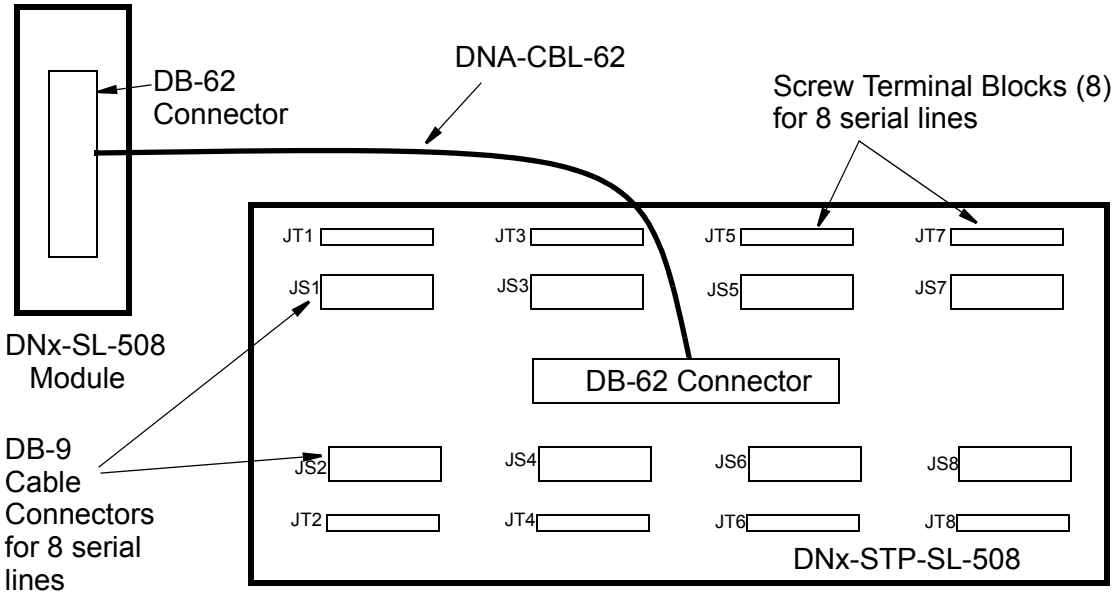


Figure 1-6. DNx-STP-SL-508 Screw Terminal Panel Connections

1.8.1 Pinout of STP-SL-508 Panel

The following diagram shows the pinout connections for the STP-SL-508 panel.

Table 1-1 . Pinout of STP-SL-508 Screw Terminal Panel

DB-62 Pin No.	232 Signal	485 Signal	DB-9 Pin No.	DB-62 Pin No.	232 Signal	485 Signal	DB-9 Pin No.	DB-62 Pin No.	232 Signal	485 Signal	DB-9 Pin No.	DB-9 Connector
1	–	–		22	Gnd1	Gnd1	5	43	CTS1	RX1–	8	JS1
2	RTS1	TX1+	7	23	TX1	TX1–	3	44	RX1	RX1+	2	
3	–	–		24	–	–		45	Gnd2	Gnd2	5	JS2
4	RX2	RX2+	2	25	CTS2	RX2–	8	46	–	–		
5	RTS2	TX2+	7	26	TX2	TX2–	3	47	–	–		
6	Gnd3	Gnd3	5	27	–	–		48	–	–		JS3
7	RX3	RX3+	2	28	CTS3	RX3–	8	49	–	–		
8	RTS3	TX3+	7	29	TX3	TX3–	3	50	–	–		
9	RX4	RX4+	2	30	CTS4	RX4–	8	51	Gnd4	Gnd4	5	JS4
10	RTS4	TX4+	7	31	TX4	TX4-	3	52	–	–		
11	–	–		32	–	–						
12	RTS5	TX5+	7	33	TX5	TX5–	3	53	CTS5	RX5–	8	JS5
13	–	–		34	Gnd5	Gnd5	5	54	RX5	RX5+	2	
14	RX6	RX6+	2	35	CTS6	RX6–	8	55	Gnd6	Gnd6	5	JS6
15	RTS6	TX6+	7	36	TX6	TX6–	3	56				
								57	–	–		JS7
16	Gnd7	Gnd7	5	37	–	–		58	–	–		
17	RX7	RX7+	2	38	CTS7	RX7–	8	59	–	–		
18	RTS7	TX7+	7	39	TX7	TX7–	3	60	–	–		
19	–	–		40	Gnd8	Gnd8	5	61	CTS8	RX8–	8	JS8
20	RTS8	TX8+	7	41	TX8	TX8–	3	62	RX8	RX8+	2	
21	–	–		42	–	–						

1.9 Jumper Settings for DNA Version

The base board of a DNA-SL-508 module (layer) has a jumper block that assigns the position of the module within a PowerDNA Cube. The jumpers must be set to match the physical position of an I/O board or layer in the Cube. This function is not required with DNR version boards.

NOTE: Since all layers are assembled in Cubes before shipment to a customer, you should never have to change a jumper setting unless you change a layer from one position to another in the field.

A diagram of the jumper block is shown in **Figure 1-7**. To set the layer address, place jumpers as shown in **Figure 1-7**.

		Layer's Position as marked on the Faceplate*					
		I/O **1**	I/O **2**	I/O **3**	I/O **4**	I/O **5**	I/O **6**
Jx Pins	9-10	Closed	Open	Closed	Open	Closed	Open
	11-12	Closed	Closed	Open	Open	Closed	Closed
	13-14	Closed	Closed	Closed	Closed	Open	Open
	15-16	Closed	Closed	Closed	Closed	Closed	Closed

* *All I/O Layers are sequentially enumerated from top to the bottom of the Cube*

Open – **Open** Closed – **Closed**

Figure 1-7. Diagram of DNA-SL-508 Layer Position Jumper Settings

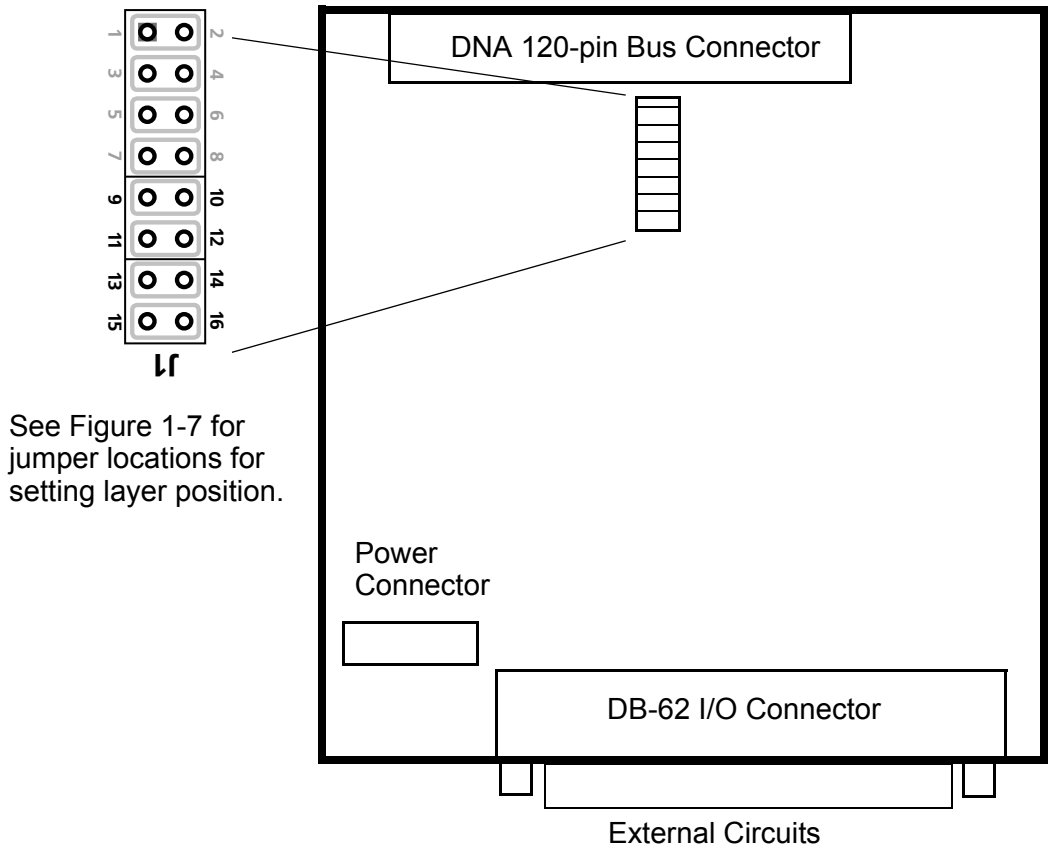


Figure 1-8. Physical Layout of DNA-SL-508 Layer Base Board (60x)

Chapter 2 Programming with High-Level API

This section describes how to program the DNx-SL-508 using the high-level UeiDaq Framework API.

UeiDaq Framework is object oriented and its objects can be manipulated in the same manner in various development environments, such as Visual C++, Visual Basic, LabVIEW, or DASYLab.

UeiDaq Framework comes bundled with examples for supported programming languages. These are located under the UEI programs group in:

```
Start » Programs » UEI » Framework » Examples
```

The following subsections focus on the C++ API, but the concept is the same regardless of programming language.

Please refer to the “UeiDaq Framework User Manual” to get more information on using other programming languages.

2.1 Creating a Session

The Session object controls all operations on your PowerDNA device. Therefore, the first task is to create a session object:

```
CUeiSession session;
```

1.2 Configuring the Serial Ports

Framework uses resource strings to select which device, subsystem, and channels to use within a session. The resource string syntax is similar to a web URL:

```
<device class>://<IP address>/<Device Id>/<Subsystem><Channel list>
```

For PowerDNA, the device class is **pdna**.

For example, the following resource string selects serial ports 0,2,3 on device 1 at IP address 192.168.100.2:

```
“pdna://192.168.100.2/Dev1/Com0,2,3”.
```

In addition to the resource, you will also configure:

- Port mode (RS-232, RS-485 half-duplex or RS-485 full duplex)
- Bit rate (bits per second)
- Number of data bits
- Parity
- Number of stop bits

```
// Configure Com ports 0, 2 and 3 on device 1
session.CreateSerialPort(pdna://192.168.100.2/Dev1/Com0,2,3,
                         UeiSerialModeRS232,
                         UeiSerialBitsPerSecond57600,
                         UeiSerialDataBits8,
```

```
                    UeiSerialParityNone,
                    UeiSerialStopBits1);
```

1.3 Configuring the Timing

You need to configure the SL-508 to use the “messaging” timing mode. A message is represented by an array of bytes:

The SL-508 can be programmed to wait for a certain number of bytes to be received before notifying the session.

It is also possible to program the maximum amount of time to wait for the specified number of bytes before notifying the session.

The following sample shows how to configure the messaging I/O mode to be notified when 10 bytes have been received or every second, whichever is less. (Note that if the serial port receives fewer than 10 bytes per second, it will return whatever number of bytes are available every second).

```
session.ConfigureTimingForMessagingIO(10, 1.0);
```

1.4 Reading Data

Reading data from the SL-508 is done using a reader object. As there is no multiplexing of data (contrary to what’s being done with AI, DI, or CI sessions), you need to create one reader object per serial port to be able to read from each port in the port list.

The following sample code shows how to create a reader object tied to port 1 and read up to 10 bytes from the serial port.

```
// Create a reader and link it to the session’s stream, port 1
reader = new CUeiSerialReader(session.GetDataStream(), 1);
// read up to 10 bytes, numBytesRead contains the
// number of bytes actually received.
Unsigned char bytes[10];
reader->Read(10, bytes, &numBytesRead);
```

1.5 Writing Data

Writing data to the SL-508 is done using a writer object. As there is no multiplexing of data (contrary to what’s being done with AO, DO, or CO sessions), you need to create one writer object per serial port to be able to write to each port in the port list.

The following sample code shows how to create a writer object tied to port 2 and send one byte to the serial port.

```
// Create a writer and link it to the session’s stream, port 2
writer = new CUeiSerialWriter(session.GetDataStream(), 2);

// Write 1 byte, numBytesWritten contains the
// number of bytes actually sent
unsigned char bytes[2] = {0x23, 0};
writer->Write(1, bytes, &numBytesWritten);
```

1.6 Cleaning-up the session

The session object cleans itself up when it goes out of scope or when it is destroyed. However, you can manually clean up the session (to reuse the object with a different set of channels or parameters).

```
session.CleanUp();
```

Chapter 3 Programming with the Low-Level API

The low-level API offers direct access to PowerDNA DaqBIOS protocol and allows you to access device registers directly.

We recommend that, where possible, you use the UeiDaq Framework tool (*see Chapter 2*), as it is easier to use. You should need to use the low-level API only if you are using an operating system other than Windows.

Please refer to the `API Reference Manual` document under:

`Start » Programs » UEI » PowerDNA » Documentation`

for pre-defined types, error codes, and functions for use with this board.

Appendix

A. Accessories

This appendix provides a list of accessories available for SL-508 board(s).

DNA-CBL-62 Cable

A DNA-CBL-62 cable has 62 conductors and may be connected directlly to user, OEM equipment, or to a UEI DNx-STP-SL-508 screw terminal panel.

DNA / DNR-STP-SL-508

A DNA/DNR-STP-SL-508 screw terminal panel is an accessory that serves as a convenient wiring interface between a 62-pin connector that plugs into a mating connector on the SL-508 module and either DB-9 serial cable connectors or eitght 5-pin screw terminal blocks on the DNx-STP-SL-508 accessory panel.

Index

A
Accessories 14
B
Block diagram 4
C
Capabilities 6
Cleaning-up 12
Configuring ports 11
Configuring timing 12
Creating a session 11
D
DNA-CBL-62 Cable 14
DNA-STP-62 14
DNx-STP-SL-508 accessory panel 8
F
Features 2
J
Jumper Settings 10
L
Layer position jumper settings 10
P
Physical layout 10
Pinout 7
Programming - High Level API 11
Programming with the Low-Level API 13
S
Serial communication 3
Software API
  reading data 12
  writing data 12
W
Wiring 6