DNx-AI-222

—

User Manual

12-Channel, fully isolated RTD / resistance input board
for the PowerDNA Cube and RACK series chassis

May 2017

PN Man-DNx-AI-222

See the UEI website for complete terms and conditions of sale:

http://www.ueidaq.com/cms/terms-and-conditions/

CE

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 contact a member of our support team:

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.

Specifications in this document are subject to change without notice. Check with UEI for current status.

Table of Contents

List of Figures

Chapter 1 Introduction

This document outlines the feature set and use of the DNx-AI-222 Resistance Temperature Detector (RTD) analog input boards.

The DNx-AI-222 is a 12-channel, high accuracy RTD/Resistance input board for the PowerDNA Cube and RACK chassis.

The following sections are provided in this chapter:

- Organization of Manual (Section 1.1)
- AI-222 Board Overview (Section 1.2)
- Features (Section 1.3)
- Specification (Section 1.4)
- Device Architecture (Section 1.5)
- Indicators (Section 1.6)
- Pinout (Section 1.7)

1.1 Organization of Manual

This AI-222 User Manual is organized as follows:

- **Introduction**
  Chapter 1 provides an overview of DNx-AI-222 analog input board features, device architecture, connectivity, and logic.
- **Programming with the High-Level API**
  Chapter 2 provides an overview of the how to create a session, configure the session, and interpret results with the Framework API.
- **Programming with the Low-Level API**
  Chapter 3 is an overview of low-level API commands for configuring and using the AI-222 series board.
- **Appendix A - Accessories**
  This appendix provides a list of accessories available for use with the DNx-AI-222 board.
- **Index**
  This is an alphabetical listing of the topics covered in this manual.

NOTE: A glossary of terms used with the PowerDNA Cube/RACK and I/O boards can be viewed or downloaded from www.ueidaq.com.

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 to 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**.”

Bold typeface will also represent field or button names, as in “Click **Scan Network**.”

Text formatted in `fixed` typeface generally represents source code or other text that should be entered verbatim into the source code, initialization, or other file.

Examples of Manual Conventions

Before plugging any I/O connector into the Cube or RACKtangle, be sure to remove power from all field wiring. Failure to do so may cause severe damage to the equipment.

Usage of Terms

Throughout this manual, the term “Cube” refers to either a PowerDNA Cube product or to a PowerDNR RACKtangle^{TM} rack mounted system, whichever is applicable. The term DNR is a specific reference to the RACKtangle, DNA to the PowerDNA I/O Cube, and DNx to refer to both.

1.2 AI-222 Board Overview

The DNx-AI-222 board is a 12-channel fully isolated, simultaneously sampling RTD/Resistance input board compatible with UEI’s Cube and RACK chassis.

DNA-AI-222, DNR-AI-222, and DNF-AI-222 boards are compatible with the UEI Cube, RACKtangle, and FLATRACK chassis respectively. These board versions are electronically identical except for the mounting hardware. The DNA version is designed to stack in a Cube chassis. The DNR/F versions are designed to plug into the backplane of a RACK chassis.

1.2.1 ADC Input Configuration

The DNx-AI-222 features 24-bit A/D converters with 50 µA excitation current for resistances up to 40 kΩ. The AI-222 provides an input resolution of better than 0.005 °C or 0.002 Ω when measuring 100 Ω RTDs using a range of 0-312 Ω.

An A/D per channel configuration allows simultaneous sampling at rates up to 150 S/s each (1800 S/s aggregate). Using an A/D per channel also virtually eliminates input cross talk and channel settling time issues.

1.2.2 RTD Measurement

The DNx-AI-222 provides high accuracy RTD measurements resulting in total errors of less than 0.2 °C on all standard 100 Ω RTDs.

UEI-provided software includes all required RTD linearization and compensation and returns data optionally in °C, °K, °F or °R. The software also allows the temperature conversion to be disabled and inputs to measure resistance in ohms up to 40 kΩ.

1.2.3 Software Support

Software included with the DNx-AI-222 provides a comprehensive yet easy to use API that supports all popular Windows programming languages as well as supporting programmers using Linux and most real-time operating systems including QNX, RTX, InTime, VXworks and more. The UEIDAQ Framework supplies complete support for those creating applications in data acquisition software packages such as LabVIEW, MATLAB/Simulink, or any application which supports ActiveX or OPC servers.

1.3 Features

The AI-222 board has the following features:

- 12 fully differential analog input channels
- 2-, 3-, and 4-wire RTD connections (per channel selectable)
- 100 Ω DIN and US measurements
- Channel-to-channel and channel-to-chassis isolation
- 0.005°C resolution, 0.2°C accuracy
- 0 to 40 kΩ resistance range
- Built-in 50, 60 Hz rejection at < 19.7 S/s
- Built-in 400 Hz rejection at < 150 S/s
- Weight of 120 g or 4.24 oz for DNA-AI-222; 630 g or 22.2 oz with PPC5

1.4 Specification

The technical specifications for the DNx-AI-222 board are listed in **Table 1-1**.

Table 1-1 DNx-AI-222 Technical Specifications (for 19.7 SPS unless otherwise noted)

Number of channels:	12 fully differential	
ADC resolution	24 bits, < 0.005 °C with 100 Ω PT RTDs < 0.002 Ω in resistance measurement mode	
Sampling rate	up to 150 samples/sec per channel 1800 S/S board aggregate	
Measurement error	(@ 25 ±5°C / over -40°C to +85°C)	
	3/4-wire	**2-wire**
RTDs (100 Ω PT)	±0.2 °C / ±0.4 °C	±0.8 °C / ±1.6 °C
0 to 156 or 312 Ω range	±0.05 Ω / ±0.1 Ω	±0.2 Ω / ±0.4 Ω
0 to 1250 Ω range	±0.1 Ω / ±0.2 Ω	±0.4 Ω / ±0.8 Ω
0 to 5000 Ω range	±0.2 Ω / ±0.4 Ω	±0.8 Ω / ±1.6 Ω
0 to 40,000 Ω range	±0.2 Ω ±0.02% / ±1% of reading	±0.8 Ω ±0.08% / ±1% of reading
Excitation current	50 µA typical (all ranges)	
Resistance range	0 - 40,000 Ω	
General A/D specifications		
Gain error	±0.005 % (typical)	
Input INL error	6 ppm typical, 15 ppm max	
Input impedance	>5000 MΩ	
Anti-Aliasing filtering	@47.6% of sample rate, ~100 dB/decade	
50/60/400 Hz notch filtering	>70 dB at sample rate of 19.7 Hz or less	
Chan to Chan crosstalk	< 0.03 Ω or 0.08 °C with a 100Ω PT RTD.	
Isolation	350 Vrms, chan-to-chan & chan-to-chassis	
Overvoltage protection	-15 V to +15 V (power on or off, current must be limited to ±20 mA)	
Power consumption	4 W max	
Operating temp. (tested)	-40°C to +85°C	
Operating humidity	95%, non-condensing	
Vibration *IEC 60068-2-6* *IEC 60068-2-64*	5 g, 10-500 Hz, sinusoidal 5 g (rms), 10-500Hz, broadband random	
Shock *IEC 60068-2-27*	50 g, 3 ms half sine, 18 shocks @ 6 orientations 30 g, 11 ms half sine, 18 shocks @ 6 orientations	
MTBF	230,000 hours	

For gain ranges below 5 kΩ, the AI-222 uses a ratiometric measurement, which eliminates most of the temperature-related drift. The ADC also applies gain and FIR filtering to the input data to eliminate noise introduced via the cable. Due to these effects, for longer length cables, the maximum input range is reduced by approximately 10%. When using longer cables (25 feet or more), select the AI-222 input range accordingly.

1.5 Device Architecture

Figure 1-1 shows a block diagram of the architecture of the AI-222 board.

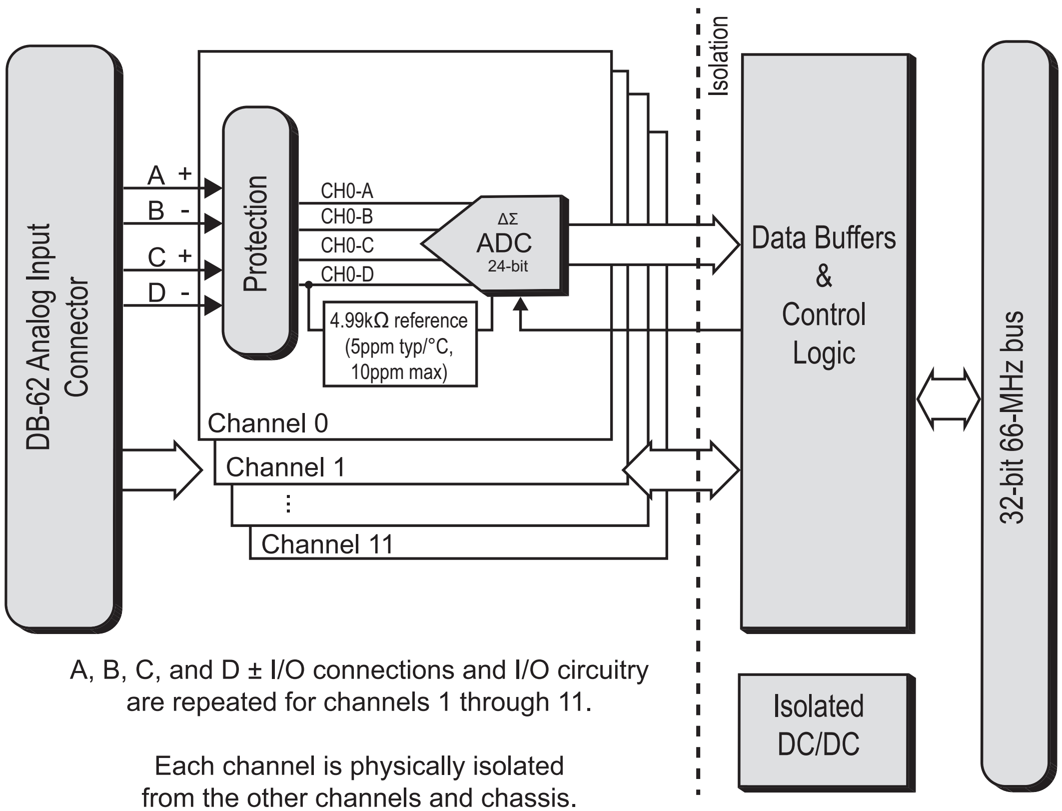


Figure 1-1 Block Diagram of the AI-222 Board

Each AI-222 has twelve differential input channels designed for high-resolution analog voltage signal measurement. Each channel has 24-bit sampling resolution and maximum sampling rate of 150 samples per second (one sample every $6.\overline{6}$ msec), up to 1800 max samples per second per board aggregate. Each channel is individually isolated and uses dedicated components to avoid crosstalk/noise between channels and also ensures that one channel's failure (e.g., due to accidental over-voltage) does not affect other channels or the board.

1.5.1 Channel Input & Overvoltage Protection

Each channel consists of four analog input and excitation lines. The analog input voltages are differentially compared against one another before digital sampling. The AI-222 design is intended for RTDs, but any resistance from 0 to 40 kΩ may be measured.

Analog input lines (A/B/C/D in **Figure 1-1**) enter through the DB-62 connector pins into a SP724 protection circuit designed to redirect ESD / transients to the channel's isolated power supply before entering the A/D converter.

The AI-222 supports 2-, 3-, and 4-wire RTD connections. Refer to **Figure 1-4** for wiring diagrams.

1.5.2 A/D Converter

Each input channel is configured with one 24-bit delta-sigma (ΔΣ) A/D converter. The ADC provides a low-noise programmable-gain-amplifier (PGA) with user-programmable gains that are selected by the firmware to best match the resistance range required for your system.

The ADC provides digital filtering that settles in one cycle. The use of a single dedicated ADC per input channel virtually eliminates all input cross talk and channel settling time issues, even when inputs are connected to high impedance signal sources. Additionally, the use of ADC sample rates lower than 19.7 Hz provides the greatest 50/60 Hz noise rejection and produces the most accurate readings.

1.5.3 Isolation

The converter control and data lines are wired through an isolation barrier to the board's main controller. The 350V isolation barrier ensures that disturbances or damage to one channel does not affect other channels, allowing the AI-222 to be an ideal solution for precise temperature measurement that must occur over long periods of time.

1.5.4 Control Logic

The main controller, in addition to containing the standard logic to provide DNx functionality, also contains AI-222 specific logic blocks, which sends commands to all 12 channels and provides clocking and synchronization. The logic controller receives and processes data from each channel and provides sampled data and configuration access to the Cube and RACK applications.

Data acquisition consists of the main controller sending commands to sample up to all 12 analog inputs simultaneously, and then receiving the 24-bit values from the conversion chip and storing the data in an output buffer. The output buffer is accessible to standard DNx function calls and is read using software calls in either the UEIDAQ Framework (see **Chapter 2**) or the low-level API (see **Chapter 3**).

1.6 Indicators

The DNx-AI-222 indicators are described in **Table 1-2** and illustrated in **Figure 1-2**.

Table 1-2 AI-222 Indicators

LED Name	Description
RDY	Indicates board is powered up and operational
STS	Indicates which mode the board is running in: • **OFF**: Configuration mode, (e.g., configuring channels, running in point-by-point mode) • **ON**: Operation mode, (e.g., running in DMap or ACB mode)

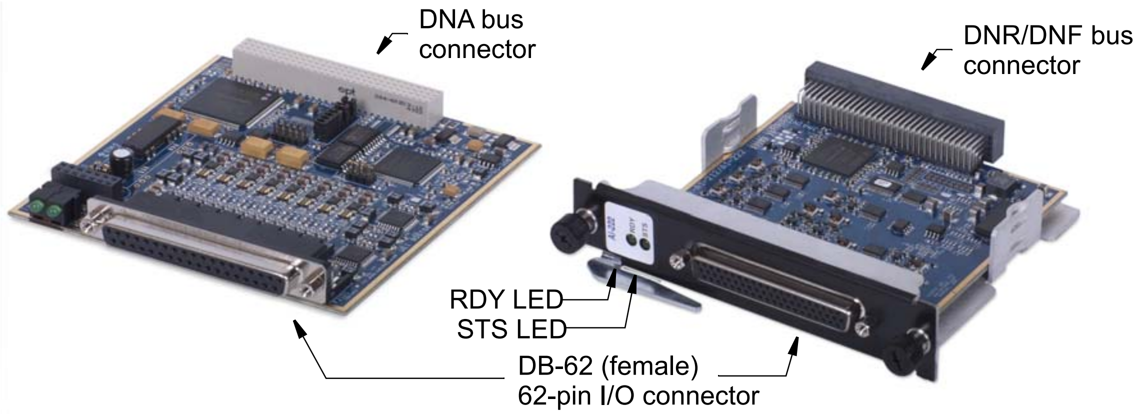
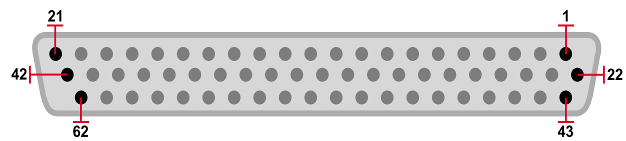


Figure 1-2 Photo of DNA-AI-222 and DNR-AI-222 Analog Input Boards

1.7 Pinout

Figure 1-3 below illustrates the pinout of the AI-222. The AI-222 board uses a 62-pin D-sub connector. The following signals are located at the connector:

- AI0 to AI11: analog input channel. Wiring diagrams provided on next page
- +5VDC (100mA max): provides isolated power for external accessories; Gnd for 5VDC provides a reference ground for +5VDC power supply
- Rsvd — do not connect, Reserved pins



Pin	Signal	Pin	Signal	Pin	Signal
1	Rsvd	22	Rsvd	43	Rsvd
2	Rsvd	23	Rsvd	44	Rsvd
3	5 VDC (100 mA Max)	24	Gnd for 5VDC	45	Gnd for 5VDC
4	Rsvd	25	Rsvd	46	CH 11.A
5	CH 11.C	26	CH 11.B	47	CH 11.D
6	CH 10.B	27	CH 10.A	48	CH 10.C
7	CH 9.A	28	CH 10.D	49	CH 9.B
8	CH 9.D	29	CH 9.C	50	CH 8.A
9	CH 8.C	30	CH 8.B	51	CH 8.D
10	CH 7.B	31	CH 7.A	52	CH 7.C
11	CH 6.A	32	CH 7.D	53	CH 6.B
12	CH 6.D	33	CH 6.C	54	CH 5.A
13	CH 5.C	34	CH 5.B	55	CH 5.D
14	CH 4.B	35	CH 4.A	56	CH 4.C
15	CH 3.A	36	CH 4.D	57	CH 3.B
16	CH 3.D	37	CH 3.C	58	CH 2.A
17	CH 2.C	38	CH 2.B	59	CH 2.D
18	CH 1.B	39	CH 1.A	60	CH 1.C
19	CH 0.A	40	CH 1.D	61	CH 0.B
20	CH 0.D	41	CH 0.C	62	DO 1
21	DI 0	42	DIO Gnd		

Figure 1-3 Pinout Diagram of the AI-222 Board

NOTE: If you are using an accessory panel with the AI-222, please refer to Appendix A for a description of the panel.

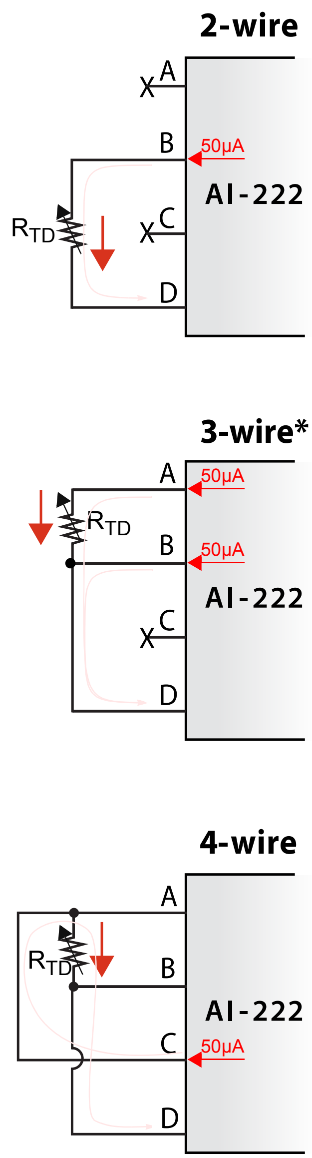


Figure 1-4 Wiring for 2/3/4-wire RTD for the AI-222 Board

NOTE: *In 3-wire mode, any imbalance between the resistance of wire A and wire B will be 1:1 reflected in the measurements.

Chapter 2 Programming with the High-Level API

This chapter provides the following information about using the UeiDaq high-level Framework API to program the DNx-AI-222:

- About the High-Level Framework (Section 2.1)
- Creating a Session (Section 2.2)
- Configuring the Resource String (Section 2.3)
- Configuring the Input (Section 2.4)
- Configuring the Timing (Section 2.5)
- Reading Data (Section 2.6)
- Cleaning-up the Session (Section 2.7)

2.1 About the High-Level Framework

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

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

- *Start » Programs » UEI » Framework » Examples*

The following sections focus on the C++ API, but the concept is the same no matter which programming language you use.

Please refer to the “UeiDaq Framework User Manual” for more information on use of other programming languages.

2.2 Creating a Session

The Session object controls all operations on your PowerDNx device. The first task is to create a session object:

```
// create a session object for input
                    CUeiSession aiSession;
```

2.3 Configuring the Resource String

UeiDaq 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 and RACKtangle, the device class is **pdna**.

For example, the following resource string selects analog input lines 0,1,2,3 on device 1 at IP address 192.168.100.2: “pdna://192.168.100.2/Dev1/Ai0:3” as a range, or as a list “pdna://192.168.100.2/Dev1/Ai0,1,2,3”.

2.4 Configuring the Input

The following section provides examples for configuring your input for resistance and RTD/Resistance measurements.

2.4.1 Resistance Measurement

Resistance measurements are configured using the Session object method "`CreateResistanceChannel`".

```
// The following example code adds three channels (0, 2 and 4) to the
// channel list and configures the channels to measure a resistance
// between 0 and 5000 Ohms.
// The maximum resistance that the AI-222 is capable of measuring is
// 40 KOhms. In this example, the max resistance expected is
// programmed to 5 KOhms; this value is used by system firmware to select
// the optimal gain.
// In this example, the resistance is connected to the DAQ device
// using four wires.

aiSession.CreateResistanceChannel(
      "pdna://192.168.100.2/dev0/Ai0,2,4", // Resource string (channels)
      0,                                 // min resistance expected
      5000.0,                            // max resistance expected
      UeiFourWires,                      // wiring scheme
      0.0,                               // lead resistance in 2-wire mode
      UeiAIChannelInputModeDifferential    // input mode to measure v
);
```

Note that reading a resistance beyond the programmed range, (e.g. a 6 kΩ resistance when configured for 0-5 kΩ expected) will appear as overrange or as the maximum value. The maximum resistance supported by the AI-222 is 40 kΩ.

2.4.2 RTD Measurement

RTD measurements are configured using the Session object method "`CreateRTDChannel`".

RTD sensors are resistive sensors whose resistance varies with temperature. Knowing the resistance of an RTD, we can calculate the temperature using the "Callendar Van-Dusen" equations. RTD sensors are specified using the "alpha" (α) constant. It is also known as the temperature coefficient of resistance, which defines the resistance change factor per degree of temperature change. The RTD type is used to select the proper coefficients A, B and C for the Callendar Van-Dusen equation, which is used to convert resistance measurements to temperature.

You must configure the RTD type and its nominal resistance at 0° Celsius, as shown in the following example:

```
// Add 4 channels (0 to 3) to the channel list and configure them to
// measure a temperature between 0.0 and 200.0 °C.
// The RTD sensor is connected to the DAQ device using two wires
// The RTD alpha coefficient is 0.00385, the nominal resistance at 0° C
// is 100 Ohms, and the measured temperature will be returned in °C.

aiSession.CreateRTDChannel("pdna://192.168.100.2/dev0/Ai0:3",
                           0, 1000.0, UeiTwoWires, 0.0,
                           UeiRTDType3850, 100.0,
                           UeiTemperatureScaleCelsius,
                           UeiAIChannelInputModeDifferential);
```

2.5 Configuring the Timing

You can configure the AI-222 to run in simple mode (point by point), high-throughput buffered mode (ACB mode), or high-responsiveness (DMAP) mode. In simple mode, the delay between samples is determined by software on the host computer. In buffered mode, the delay between samples is determined by the AI-222 on-board clock and data is transferred in blocks between PowerDNA and the host PC.

The following sample shows how to configure the simple mode. Please refer to the “UeiDaq Framework User’s Manual” to learn how to use other timing modes.

```
// configure timing of input for point-by-point (simple mode)
aiSession.ConfigureTimingForSimpleIO();
```

2.6 Reading Data

Reading data is done using *reader* object(s). There is a reader object to read raw data coming straight from the A/D converter. There is also a reader object to read data already scaled to resistance or temperature.

The following sample code shows how to create a scaled reader object and read samples.

```
// create a reader and link it to the analog-input session’s stream
CUeiAnalogScaledReader aiReader(aiSession.GetDataStream());
// the buffer must be big enough to contain one value per channel
double data[2];
// read one scan, where the buffer will contain one value per channel
aiReader.ReadSingleScan(data);
```

2.7 Cleaning-up the Session

The session object will clean itself up when it goes out of scope or when it is destroyed. To reuse the object with a different set of channels or parameters, you can manually clean up the session as follows:

```
// clean up the session
aiSession.CleanUp();
```

Chapter 3 Programming with the Low-Level API

This chapter provides the following information about programming the AI-222 using the low-level API:

- About the Low-level API (Section 3.1)
- Low-level Functions (Section 3.2)
- Low-level Programming Techniques (Section 3.3)

3.1 About the Low-level API

The low-level API provides direct access to the DAQBIOS protocol structure and registers in C. The low-level API is intended for speed-optimization, when programming unconventional functionality, or when programming under Linux or real-time operating systems.

When programming in Windows OS, however, we recommend that you use the UeiDaq high-level Framework API (see **Chapter 2**). The Framework extends the low-level API with additional functionality that makes programming easier, faster, and less error-prone.

For additional information regarding low-level programming, refer to the "PowerDNA API Reference Manual" located in the following directory:

- On Linux systems:
  <PowerDNA-x.y.z>/docs
- On Windows systems:
  Start » All Programs » UEI » PowerDNA » Documentation

3.2 Low-level Functions

Table 3-1 provides a summary of AI-222-specific functions. All low-level functions are described in detail in the PowerDNA API Reference Manual.

Table 3-1 Summary of Low-level API Functions for DNx-AI-222

Function	Description
`DqAdv222Config`	Configures AI-222 channels for resistance measurements, (e.g., sets wiring scheme, resistance range)
`DqAdv222Read`	Acquires input data: • On the first `DqAdv222Config` call, firmware programs the AI-222 according to channel list supplied, and the board starts acquiring data. • Subsequent calls return the latest acquired data. Note that `DqAdv222Config()` must be setup first. `DqAdv222Read()` should only be used when the AI-222 is acquiring data in point-by-point mode, not in any of the streaming or mapping data acquisition modes. Refer to the low-level code samples (Section 3.3) for example code for using each of the data acquisition modes.

3.3 Low-level Programming Techniques

Application developers are encouraged to explore the existing source code examples when first programming the AI-222. Sample code provided with the installation is self-documented and serves as a good starting point.

Code examples are located in the following directories:

- For Linux: <PowerDNA-x.y.z>/src/DAQLib_Samples
- For Windows: *Start » All Programs » UEI » PowerDNA » Examples*

3.3.1 Setting Modes

AI-222 supported modes are summarized in **Table 3-2**. Modes specify the RTD wiring scheme (2-wire, 3-wire, or 4-wire) and resistance range, which are used by AI-222 driver software to select the required excitation current, voltage reference, multiplexer settings, and gain for the mode selected.

To configure and set channel modes, use the low-level function, `DqAdv222SetConfig():`

```
DqAdv222Config(
      int hd,            // Handle to IOM received from DqOpenIOM()
      int devn,          // Board device # inside the IOM chassis
      uint32 chan_mask,  // Bit mask of channels programming
      uint32 mode,       // Channel modes (summarized in Table 3-2)
      double lead_res);  // for 2-wire configurations only:
                         //     user-specified lead resistance
```

Table 3-2 DNx-AI-222 Modes of Operation

Mode: #define variable	**Wiring Scheme**	**Resistance Range (maximum expected resistance)**
DQ_AI222_RTD_2_WIRE_156	2-wire	156 Ω
DQ_AI222_RTD_3_WIRE_156	3-wire	156 Ω
DQ_AI222_RTD_4_WIRE_156	4-wire	156 Ω
DQ_AI222_RTD_2_WIRE_312	2-wire	312 Ω
DQ_AI222_RTD_3_WIRE_312	3-wire	312 Ω
DQ_AI222_RTD_4_WIRE_312	4-wire	312 Ω
DQ_AI222_RTD_2_WIRE_1_25K	2-wire	1250 Ω
DQ_AI222_RTD_3_WIRE_1_25K	3-wire	1250 Ω
DQ_AI222_RTD_4_WIRE_1_25K	4-wire	1250 Ω
DQ_AI222_RTD_2_WIRE_5K	2-wire	5 kΩ
DQ_AI222_RTD_3_WIRE_5K	3-wire	5 kΩ
DQ_AI222_RTD_4_WIRE_5K	4-wire	5 kΩ
DQ_AI222_RTD_2_WIRE_40K	2-wire	40 kΩ
DQ_AI222_RTD_3_WIRE_40K	3-wire	40 kΩ
DQ_AI222_RTD_4_WIRE_40K	4-wire	40 kΩ

Appendix A

A.1 Accessories

The following cables and STP boards are available for the AI-222 board.

DNA-CBL-62

This is a 62-conductor round shielded cable with 62-pin male D-sub connectors on both ends. It is made with round, heavy-shielded cable; 2.5 ft (75 cm) long, weight of 9.49 ounces or 269 grams; up to 10ft (305cm) and 20ft (610cm).

DNA-STP-62

The STP-62 is a Screw Terminal Panel with three 20-position terminal blocks (JT1, JT2, and JT3) plus one 3-position terminal block (J2). The dimensions of the STP-62 board are 4w x 3.8d x1.2h inch or 10.2 x 9.7 x 3 cm (with standoffs). The weight of the STP-62 board is 3.89 ounces or 110 grams.

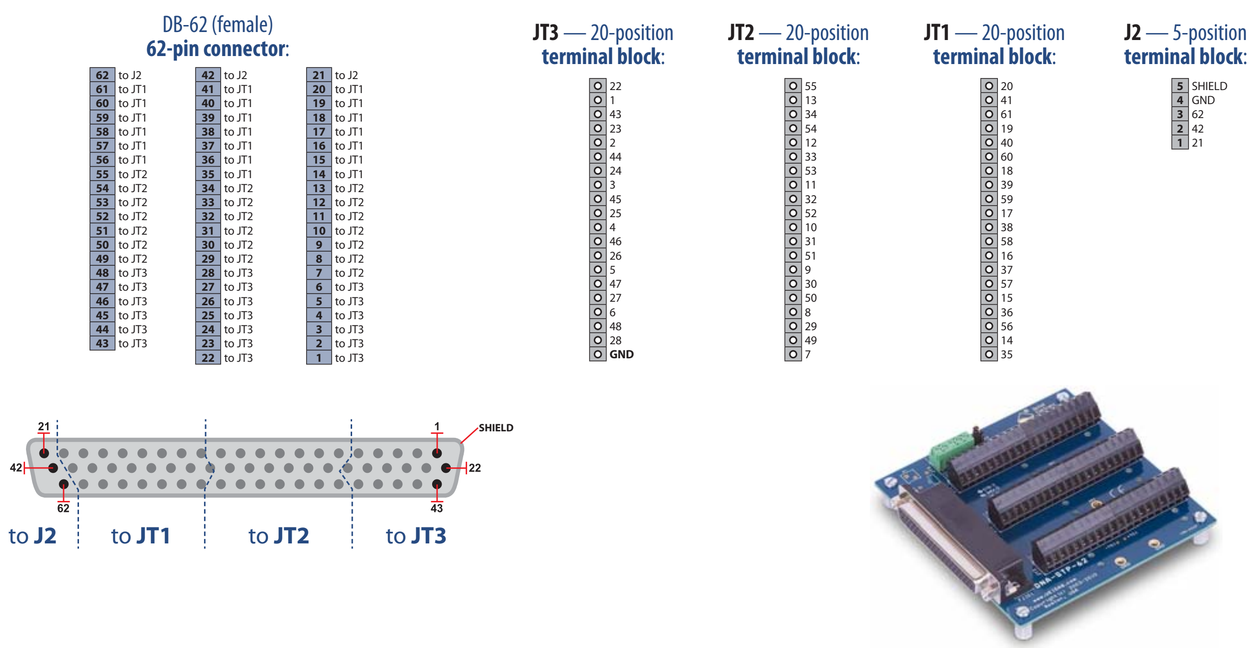


Figure A-1 Pinout and photo of DNA-STP-62 screw terminal panel

Index