United Electronic Industries

The High-Performance Alternative

DNx-AI-255

—

User Manual

2-Channel Synchro/Resolver I/O Interface Layer
for the PowerDNA Cube and PowerDNR RACKtangle

USPTO Patent: 7,957,942

Release 4.5

February 2012

PN Man-DNx-AI-255-212-DRAFT

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

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

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.

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 of the DNR- and DNA-AI-255 layer and how to use it for synchro/resolver applications.

1.1 Organization of Manual

This AI-255 User Manual is organized as follows:

- **Introduction**
  This section provides an overview of the AI-255 features.
- **Synchro/Resolver Input and Output/Simulator**
  This chapter provides an introduction to the synchro/resolver interface features, device architecture, and connectivity of the AI-256.
- **Programming with the High-Level API**
  This chapter provides an overview of the how to create a session, configure the session, and format relevant data with the Framework API.
- **Programming with the Low-Level API**
  Describes low-level API commands for configuring and using the AI-255 series layer for Synchro/Resolver/LVDT/RVDT operating modes.
- **Appendix A - Accessories**
  This appendix provides a list of accessories available for use with the DNx-AI-255 Synchro/Resolver interface board.
- **Appendix B - Connection Diagrams**
  This appendix contains connection diagrams for various operating and synchro/resolver excitation modes of the DNx-AI-255 interface board.
- **Index**
  This is an alphabetical listing of the 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 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**.”

Text formatted in `fixed` typeface generally represents source code or other text that should be entered verbadim 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 The AI-255 Interface Board

The DNx-AI-255 layer can act as a 2-channel synchro or resolver input or simulator output interface. It is suited for a wide variety of industrial, military, and simulator applications. The AI-255 functionality is similar to the AI-256, and uses the same subset of Synchro/Resolver software functions as the AI-256, but operates at lower frequency, lower currents and higher voltages than the AI-256.

The board provides two channels that can monitor either 3/4-wire (plus excitation) synchros or 4-wire (plus rotor excitation) resolvers or, as an alternative, provide simulated outputs for test and simulator applications. It is capable of angle measurement accuracies approaching ±2.6 arc-minutes. Each channel may be configured either as an input or an output, in any combination. Output accuracy is ±4 arc-minutes.

The inputs may be sampled at rates up to the excitation frequency of 4 kHz. Each channel provides its own Programmable Reference Voltage with outputs independently programmable from 2 to 28 V_{rms} at 1.2 VA and at frequencies from 50 to 4000Hz.

The DNx-AI-255 also provides two channels of Synchro or Resolver Output, ideal for driving devices such as attitude indicators or as test sources for a wide range of synchro or resolver input devices. The two outputs each accept an independent reference signal and offer 16-bit output resolution. Each channel can drive up to 28 V_{rms} at 1.2 VA without external buffering.

The AI-255 is available in two versions: the DNA-AI-255 for mounting in UEI Cube products, and the DNR-AI-255 for insertion into the UEI RACKtangle and HalfRACK chassis. The DNx-AI-255 is physically a two-board module composed of one of two types of base boards (one for the DNA version and another for the DNR version) plus an AI-255-specific daughter board. The DNA and DNR are functionally the same except for the bus connectors used.

The DNx-AI-255 is fully supported by the UEIDAQ Framework API, which provides a simple and complete software interface to all popular programming languages, operating systems, and data acquisition/control application packages such as LabVIEW, DASYLab, and MATLAB.

As with all UEI PowerDNA boards, the DNx-AI-255 can be operated in harsh environments and has been tested at 5*g* vibration, 50*g* shock, -40 to +85°C temperature, and altitudes up to 70,000 feet. Each board provides 350 V_{rms} isolation between channels and also between the board and its enclosure or any other installed boards as well as electro-shock-discharge (ESD) isolation.

Software for the DNx-AI-255 is provided as part of the UEI Framework. Framework provides a comprehensive, yet easy-to-use, API that is compatible with all popular Windows programming languages and that also supports programmers using Linux and most realtime operating systems such as QNX, RTX, or RT Linux. Also, UEI Framework can be used for creating applications in LabVIEW, MATLAB/SImulink, DASYLab, or any application that supports ActiveX or OPC servers.

1.3 Features

The features of the DNx-AI-255 that are important to note are:

- Two input or output channelsin any combination
- 16-bit Resolution
- 3/4-wire (plus excitation) Synchro and 4-wire (plus excitation) Resolver Inputs
- Reference (excitation) Output (28 V_{rms}) for each channel with 1.2 mV_{rms} resolution
- User-programmable Excitation Frequency Range of 50 Hz to 4 kHz (±0.5%) for Each Channel
- Each channel can drive up to 28 V_{rms} without External Buffering
- Isolation up to 350 V_{rms} between channel and between I/O's and GND
- Weight of 136 g or 4.79 oz for DNA-AI-255; 817 g or 28.8 oz with PPC5.
- Tested to withstand 5g Vibration, 50g Shock, -40 to +85°C Temperature, and Altitude up to 70,000 ft or 21'000 meters.
- UEI Framework Software API may be used with all popular Windows programming languages and most real time operating systems such as RT Linux, RTX, or QNX and graphical applications such as LabVIEW, MATLAB, DASYLab and any application supporting ActiveX or OPC.

1.4 Indicators

A photo of the DNx-AI-255 unit is illustrated below.

The front panel has two LED indicators:

- RDY: indicates that the layer is receiving power and operational.
- STS: can be set by the user using the low-level framework.

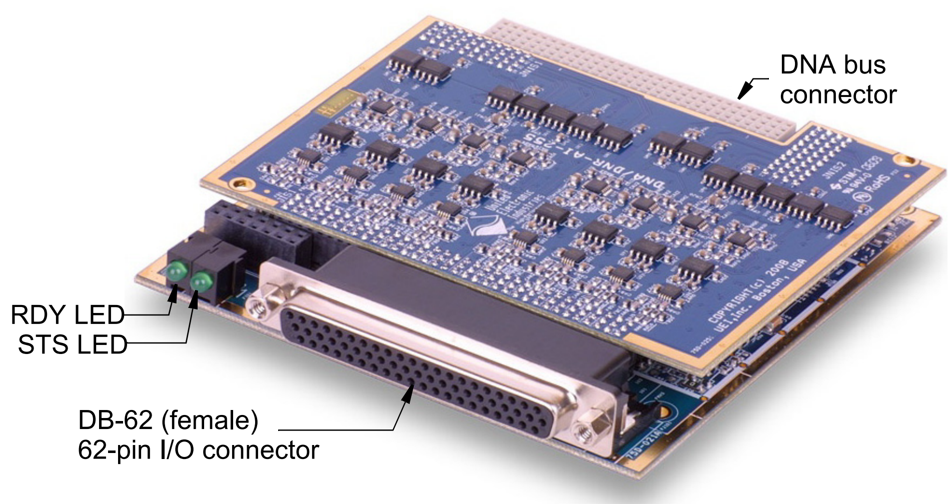


Figure 1-1. The DNR-AI-255 Analog-Input Layer

1.5 Specification

The technical specification for the DNx-AI-255 board are listed in **Table 1-1**.

Table 1-1. DNx-AI-255 Technical Specifications

Inputs	
Number of channels	2
Configuration	Synchro (3-wire) or Resolver (4-wire) may be selected via software
Resolution	16-bit
Accuracy	± 2.6 arc-minute
Frequency	50 Hz to 4.0 kHz
Signal Inputs	2-28 Vrms.
Acceleration	300 rps/s @ 60 Hz 450 rps/s @ 400 Hz 1000 rps/s @ 4000 Hz
Step response	800 mS - 179° @ 60 Hz 150 mS - 179° @ 2500 Hz
Update rate	Maximum update rate is equal to the excitation frequency.
Reference output	
Number of channels	2 (one per input channel)
Output voltage	28 Vrms up to 1.2 VA.
Voltage resolution	1.2 mVrms
Reference Frequency	50 Hz to 4 kHz (±0.5%)
Synchro / Resolver Outputs	
Number of channels	2 (total number of synchro/resolver inputs and simulated outputs is limited to 2.)
Configuration	Synchro (3-wire) or Resolver (4-wire)
Resolution	16-bit
Output Voltage	28 Vrms up to 1.2 VA.
Output Accuracy	±4 arc-minutes
General Specifications	
Operating temperature	Tested -40 °C to +85 °C
Vibration *IEC 60068-2-6* *IEC 60068-2-64*	5 g, 10-500 Hz, sinusoidal 5 g (rms), 10-500 Hz, broad-band 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
Humidity	5 to 95%, non-condensing
Altitude	0 to 70,000 feet
MTBF	275,000 hours

Chapter 2 Synchro/Resolver Mode

This chapter provides an overview of synchros/resolvers, how the DNx-AI-255 can be used to manipulate them, and supporting documentation to do so.

The DNx-AI-255 can act as a 2-channel *Synchro or Resolver interface* for UEI's PowerDNA and PowerDNR data acquisition systems.

The board provides two channels that can monitor either 3/4-wire (plus excitation) synchros or 4-wire (plus rotor excitation) resolvers or, as an alternative, provide simulated outputs for test and simulator applications. It is capable of angle measurement accuracies approaching ±2.6 arc-minutes. Each channel may be configured either as an input or an output, in any combination. Output accuracy is ±4 arc-minutes. The inputs may be sampled at rates up to the excitation frequency of 4 kHz. Each channel provides its own Programmable Reference Voltage with outputs independently programmable from 2 to 28 V_{rms} at 1.2 VA and at frequencies from 50 to 4000Hz.

The DNx-AI-255 also provides two channels of Synchro or Resolver Output, ideal for driving devices such as attitude indicators or as test sources for a wide range of synchro or resolver input devices. The two outputs each accept an independent reference signal and offer 16-bit output resolution. Each channel can drive up to 28 V_{rms} at 1.2 VA without external buffering.

2.1 Overview of Synchros & Resolvers

Synchros and resolvers are electromechanical transducers that are used either to detect and measure a rotary shaft position or to position a shaft at a desired angle. The devices can be further classified as transmitters, receivers, differentials, or control transformers.

2.1.1 The Synchro

Synchros were originally called "selsyns", for "self-synchronous". A generator and receiver are wired together so that the angular position of the generator (transmitter) shaft is automatically reproduced in the motor (receiver). Although they may appear to be similar in construction to synchronous motors and generators, the major difference between them is that the rotor of a synchro is excited with an AC voltage rather than a DC voltage. In other words, a synchro is a single phase device with AC rotor excitation and a synchronous motor or generator is typically a 3-phase (time-phase) device with DC rotor excitation.

The rotor of a synchro normally has a single-phase winding, usually referred to as a "dumbbell" rotor. The stator has 3 windings connected in a star configuration at 120°.

The AC voltage applied to the rotor winding induces AC voltages in three stator windings, which are spatially displaced 120° apart. The voltages induced in the stator windings are either in time-phase with the excitation voltage or 180° out of time-phase. The magnitudes of these voltages are:

$$V_{S1\text{-}3} = KV_{R2\text{-}1} \sin \theta$$

$$V_{S3\text{-}2} = KV_{R2\text{-}1} \sin (\theta + 120°)$$

$$V_{S2\text{-}1} = KV_{R2\text{-}1} \sin (\theta + 240°)$$

where

θ is the rotor position angle
V_{S1-3} is the voltage between S1 and S3 terminals
V_{R2-1} is the voltage between R2 and R1 terminals
and K is the maximum coupling transformation ratio V_{out}/V_{in}

Since the set of three voltages transmitted by the synchro generator is unique for each position of the rotor throughout a 360° rotation, a synchro receiver, whose rotor is excited in parallel with the generator, measures the magnitude and time-phase relation of the voltages and produces a torque that causes the receiver rotor to move to the same angular position as the transmitter. A synchro transmitter and receiver thus form a simple synchro system.

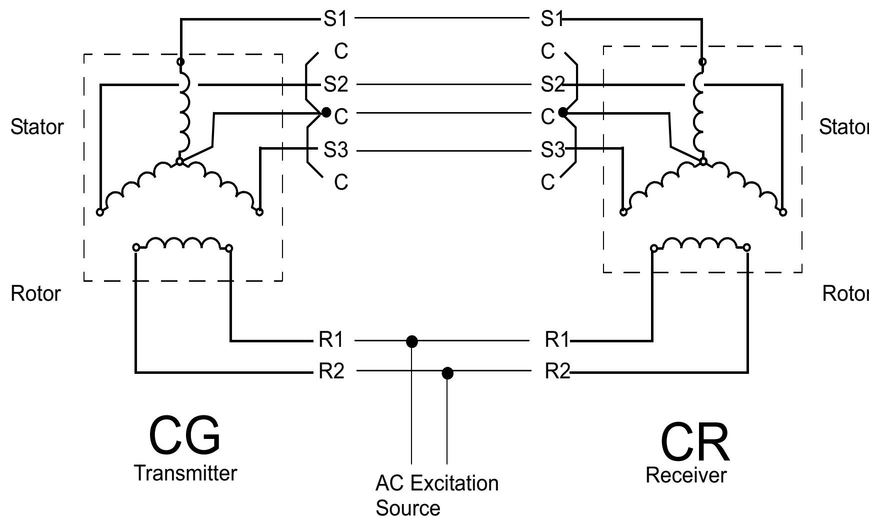


Figure 2-1. Typical Synchro Transmitter/Receiver

Receiver

When the transmitter and receiver rotors are in alignment, stator voltages are equal and no current flows. If the transmitter rotor is turned (relative to the receiver rotor), a force appears in the receiver, causing the rotor to track the transmitter rotor. The torque produced is proportional to the angle difference between the two rotors. Typical accuracy of such a system is 30 arc-minutes.

A single transmitter may be parallel-connected to multiple receivers, at the cost of reducing accuracy and increasing the power drain from the source.

Control Transformer

A control transformer has a Y-connected stator and a single-phase cylindrical drum rotor. When the stator is connected to the stator of a transmitter and the transmitter rotor is turned relative to the control transformer rotor, the magnitude of the control transformer stator field remains constant, but a voltage is induced in the rotor. The magnitude of this voltage varies with the sine of the angle between the axis of the rotor and that of the stator flux. The control transformer, therefore, provides information about the transmitter rotor angular position.

If the control transformer rotor angle differs from that of the transmitter, a voltage proportional to the sine of the angular difference appears on the control transformer rotor. This may be used as an input to a servo control system that causes the control transformer rotor to move to the same angle as the transmitter.

2.1.2 The Resolver

A resolver is a rotary transformer in which the magnitude of the energy through the resolver varies sinusoidally with rotation of the shaft. A resolver control transmitter has one primary winding (Reference Winding) and two secondary windings (the SIN and COS windings). The reference winding is located on the rotor and the SIN and COS windings are on the stator, displaced spatially by 90°. If the resolver is a brushless type, current is applied through a rotary transformer, which eliminates the problems of slip rings and brushes.

The reference winding is typically excited by an AC voltage. The induced voltages in the SIN and COS windings are equal to the reference voltage multiplied by the sine or cosine of the angle of the input shaft relative to a fixed zero position.

The connection arrangement of a brushless resolver control transformer is illustrated below in **Figure 2-2.**

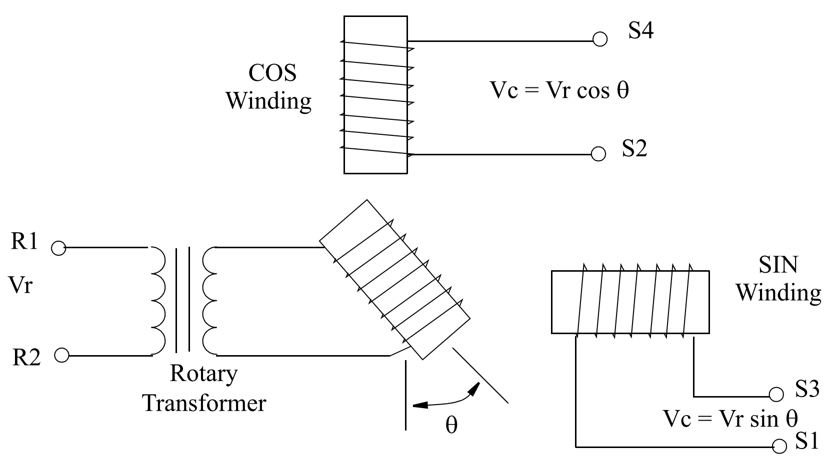


Figure 2-2. Brushless Resolver Control Transformer

2.1.2.1 Voltage Magnitude, Polarity, and Phase vs. Rotor Angle

A sine wave AC Excitation voltage $V_{exc}$is applied between R1 and R2. The voltage observed between S1 and S3 is $V_{sin} = V_{exc}$ sin A, where A is the rotor angle in radians. SImilarly, the voltage observed between S2 and S4 is $V_{cos} = V_{exc}$ cos A, where A is the rotor angle in radians. The two output voltages remain in phase with each other relative to the excitation voltage, but differ in magnitude and/or polarity (relative to excitation) as the rotor angle changes, as shown in **Figure 2-3.**

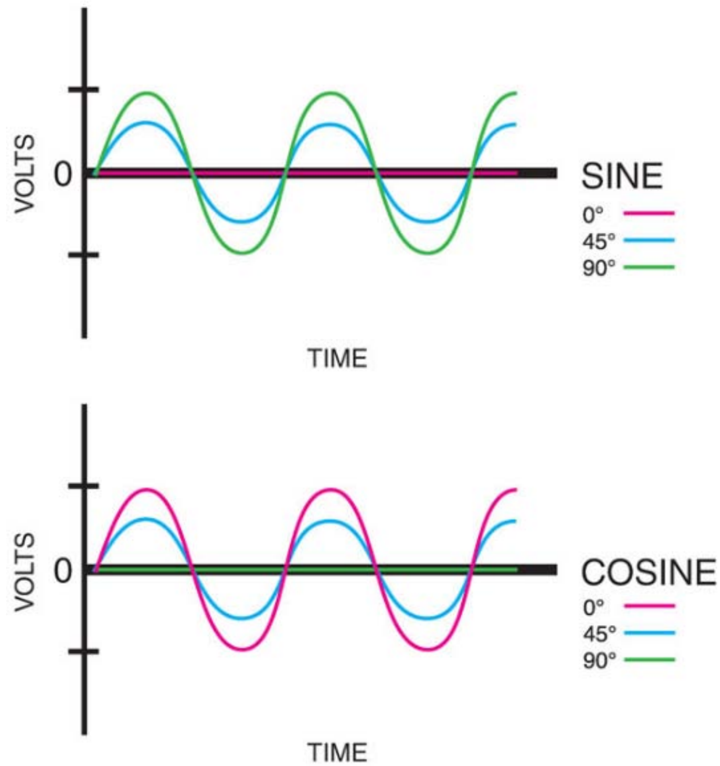


Figure 2-3. SIN and COS Output Voltages vs. Rotor Angle

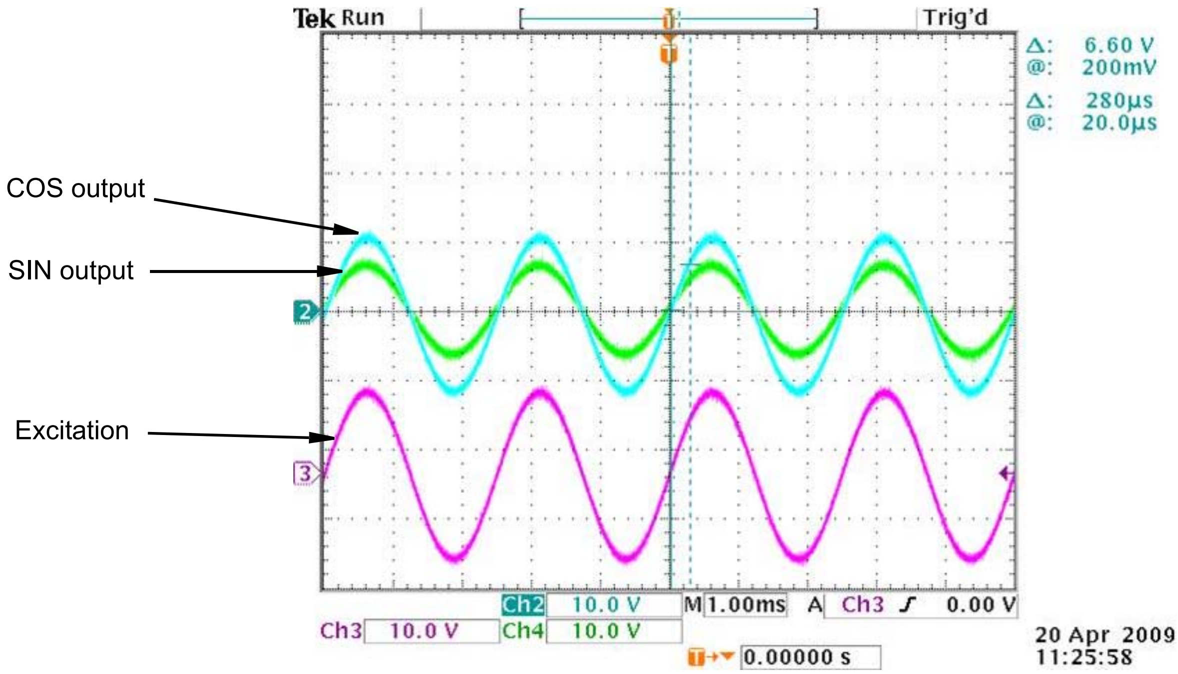


Figure 2-4. Resolver Waveforms at 30° Rotor Angle

Referring to **2-4**, The SIN and COS output voltages are in phase with excitation, have the same polarity relative to excitation, but have different magnitudes.

Referring to **2-5**, The SIN and COS output voltages are in phase with excitation, have the opposite polarity relative to excitation, but have equal magnitudes.

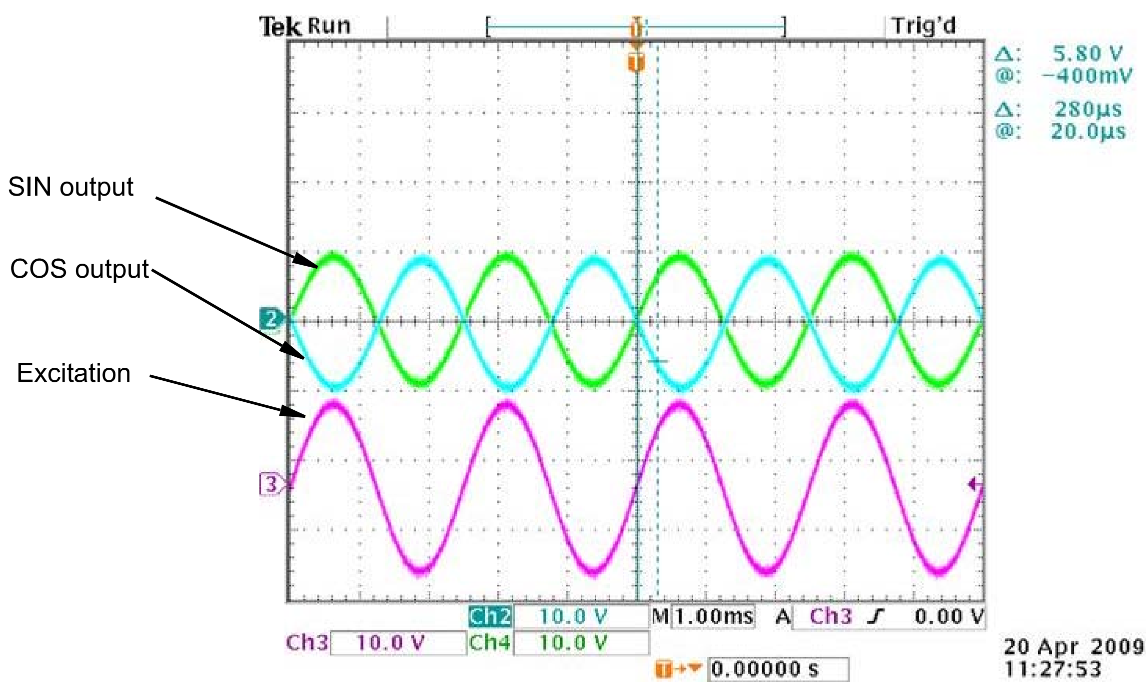


Figure 2-5. Resolver Waveforms at 135° Rotor Angle

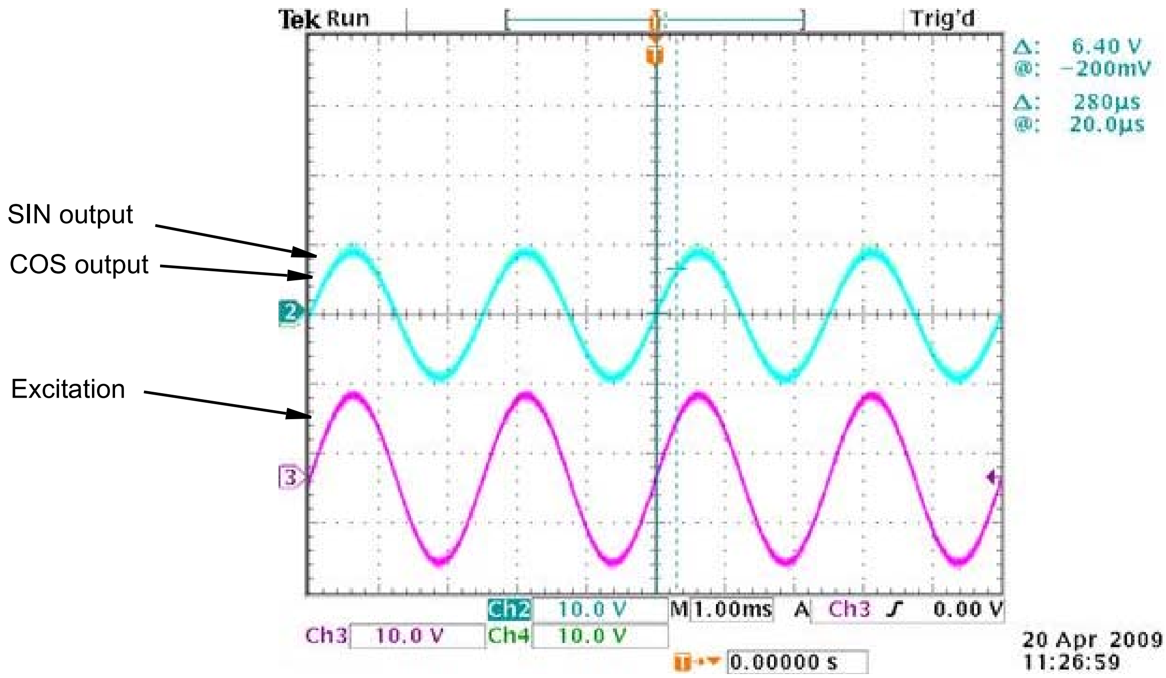


Figure 2-6. Resolver Waveforms at 45° Rotor Angle

Referring to **2-6**, The SIN and COS output voltages are in phase with excitation, have the same polarity relative to excitation, and also have equal magnitudes.

When a synchro is used, the excitation and output voltages appear as shown in **Figure 2-7.** Note that a synchro has three windings with angles between coils of 120°.

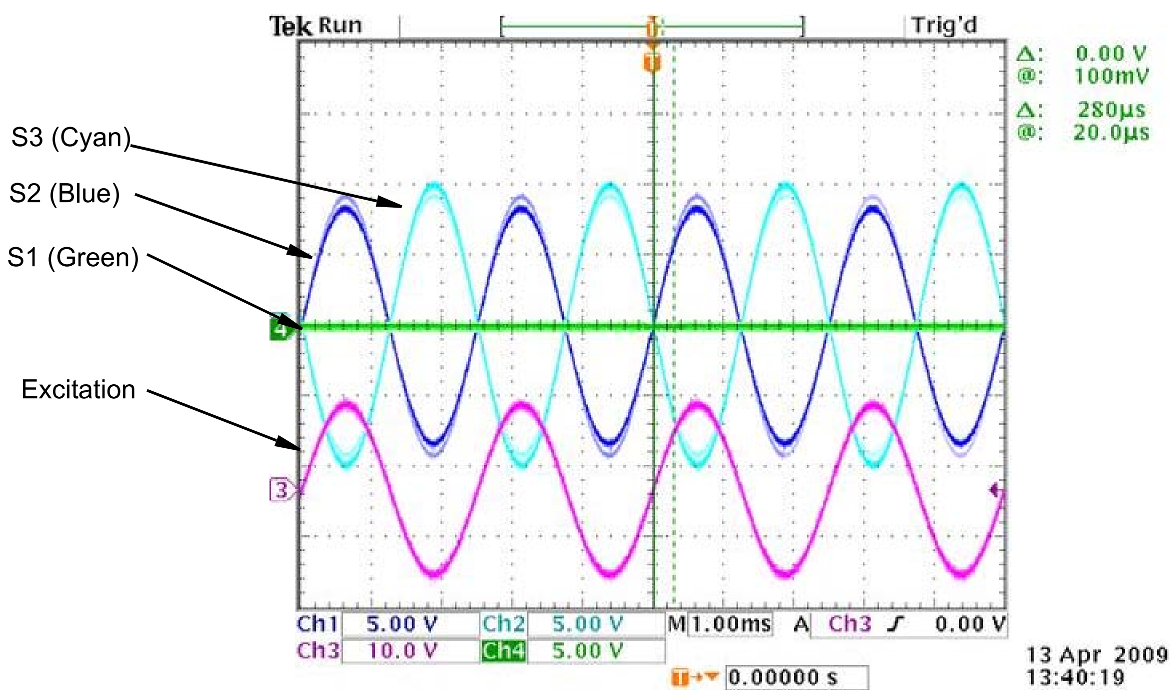


Figure 2-7. Synchro Waveforms at 0° Rotor Angle.

The SI coil output voltage (green line) is zero because the rotor is positioned plus or minus 90° relative to the S1 stator winding and therefore produces nothing. The S2 coil (blue line) shows a voltage in phase with excitation and with the same polarity as the excitation voltage. The S3 coil (cyan line) shows a voltage of polarity opposite to that of the excitation (or 180° out of phase).

NOTE: Coils on a synchro can be labeled in two different ways -- looking at the synchro from the shaft side as S1at the top, followed in a counterclockwise direction by S2 and S3, or looking at the collector side as S1 at the top, followed in a clockwise direction by S2 and S3. In datasheets from some companies, the labeling may be reversed.

NOTE: When using the AI-255 for simulation, you can attach a scope to the simulation outputs, ground the scope probes to AGND and read voltages between S1 and AGND, S2 and AGND, and S3 and AGND. Some synchros have the coil mid points between coils brought out, but most do not. To create a proper transform function, you need to consider the following formula:

$$\varphi_{actual} = 135\deg - \varphi_{calculated}$$

where the angle is calculated from the scope display.

Figure 2-8 below illustrates how the magnitudes of the SIN and COS output voltages vary with rotor angle. The table that follows lists the calculated data points for the graph.

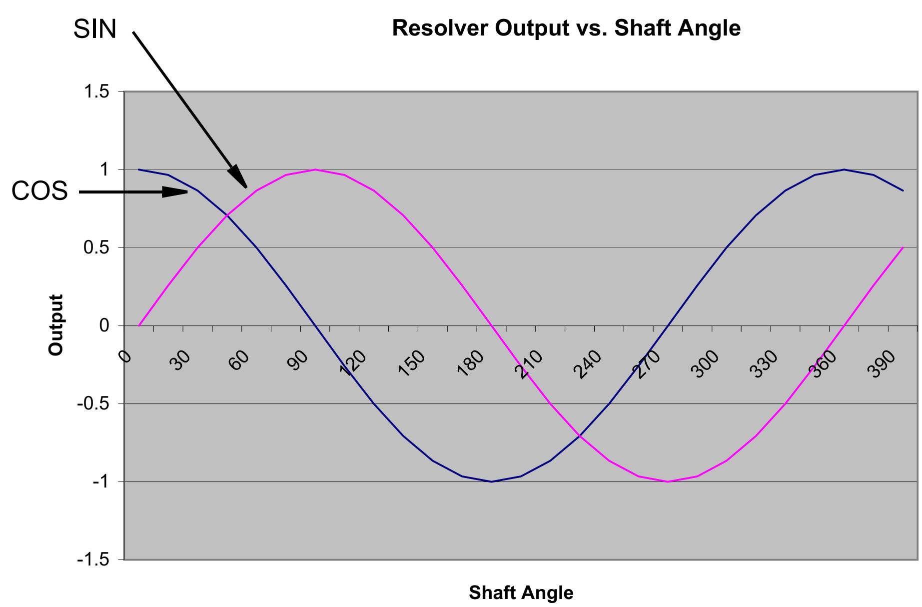


Figure 2-8. Magnitudes of SIN and COS Output RMS Voltages vs. Rotor Angle

2.2 Device Architecture

A block diagram of a DNx-AI-255 board is shown in **Figure 2-9.**

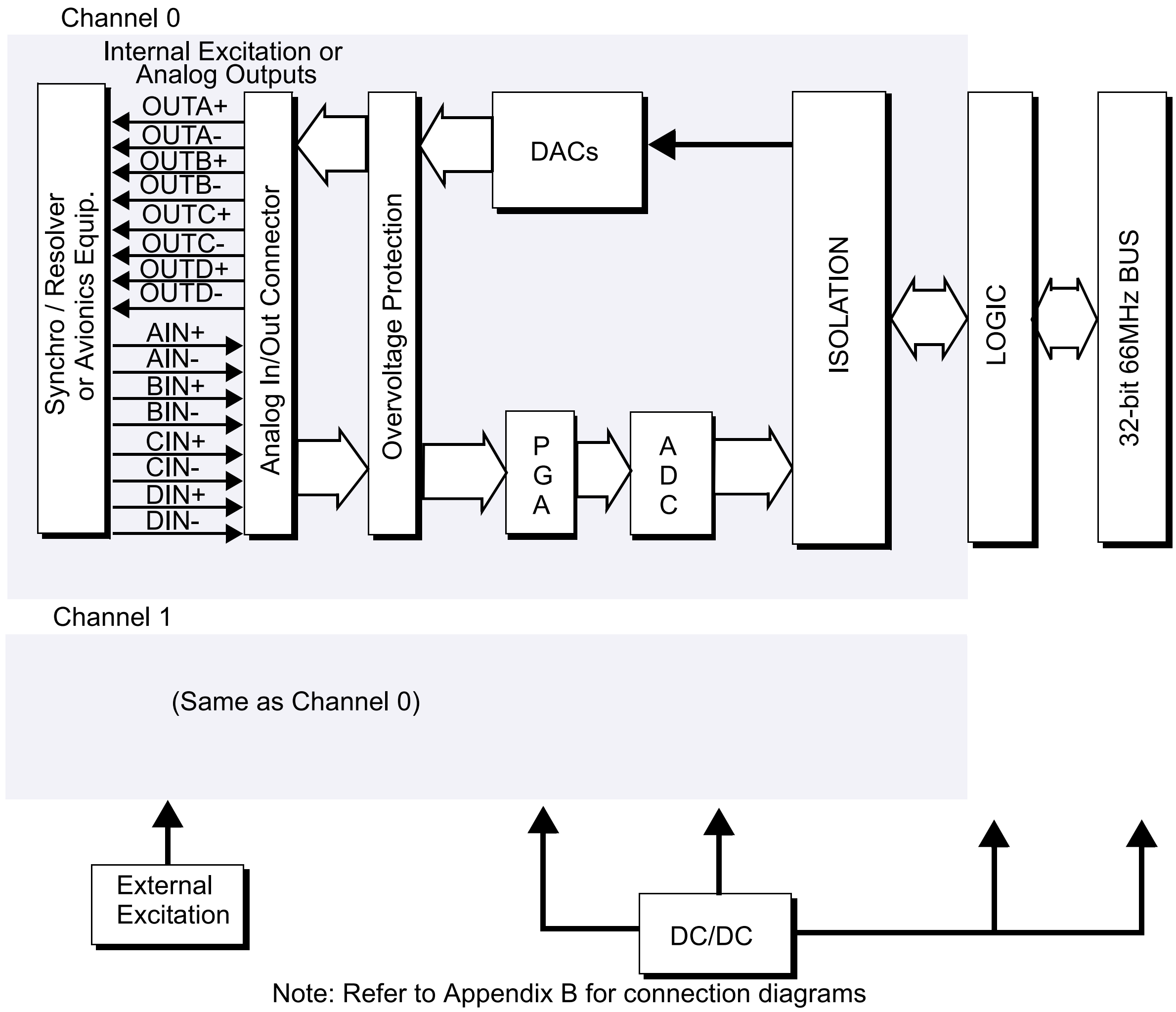


Figure 2-9. Block Diagram of DNx-AI-255 I/O Board

As shown in **Figure 2-9**, board logic is divided into isolated and non-isolated sections. The non-isolated logic complies with the full UEI Common Logic Interface standard. The isolated side handles all functions associated with the sensor input and output circuits.The non-isolated side handles all Cube or chassis-related operations.

Referring to **Fig. 2-9**, the two analog inputs from the synchro or resolver sensor are input to the programmable gain amplifiers (one per input pair) as differential inputs. Selected gain and offset are applied, producing single-ended outputs. These outputs are then converted to differential signals for input to 16-bit SAR analog to digital converters. These converters have serial interface outputs.

The signals from the ADC are passed to a 16-bit quad DAC, combined with $+V_{REF}$ voltage, amplified, and output to a sensor/resolver or avionic hardware. Refer to Appendix B for connection diagrams.

The DACs may be written from multiple sources, such as:

- Direct DAC write registers
- Direct DAC output word register
- Output FIFO. 1024 word FIFO may be used to hold DAC values or commands (in simulation mode)
- Waveform generator — which permits you to output a preloaded waveform on the output channels for simulation purposes
- Combination waveform and FIFO with FIFO in command mode — In this mode, output from FIFO may be directed to phase or gain registers or change may be delayed until the index of waveform is equal to zero.

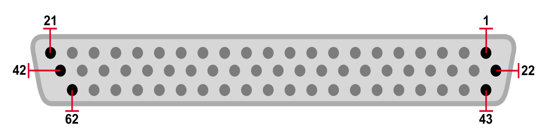
The logic computes moving averages of all analog inputs (updated with each sample) and also computes other values like (Sa-Sb)/(Sa+Sb) or (Sa-Sb)/Se, where Sa and Sb are computed values of moving averages of analog inputs and Se is the moving average of the excitation voltage value. Timing is controlled by counting pulses in half periods between zero crossings of the reference voltage signal. More information about the computations performed is available in Chapters 3 and 4 and the API Reference Manuals that describe the high-level and low-level functions used with this layer.

2.3 Setting Operating Parameters

For detailed instructions for configuring the board and setting operating modes and parameters, refer to the Functions **DqAdv255SetMode, DqAdv255SetExcitation**, and **DqAdv255GetWFMeasurements**, which are described in Chapter 3 and also in the PowerDNA API Reference Manual.

2.4 Pinout

The pinout of the DNx-AI-255 62-pin DB connector is shown in **Figure 2-10.**



	Pin	Signal	Pin	Signal	Pin	Signal
	1	Rsvd	22	Gnd	43	Out A-
	2	Out B+	23	Out B-	44	Out A+
	3	Rsvd	24	n/c	45	Gnd
	4	In A+	25	In A-	46	Rsvd
Chan 0	5	In B+	26	In B-	47	n/c
	6	Gnd	27	Rsvd	48	Rsvd
	7	Out C+	28	Out C-	49	Rsvd
	8	Out D+	29	Out D-	50	Rsvd
	9	In C+	30	In C-	51	Gnd
	10	In D+	31	In D-	52	Rsvd
	11	Rsvd	32	n/c	53	Out A-
	12	Out B+	33	Out B-	54	Out A+
	13	Rsvd	34	Gnd	55	Gnd
	14	In A+	35	In A-	56	Rsvd
	15	In B+	36	In B-	57	n/c
Chan 1	16	Gnd	37	Rsvd	58	Rsvd
	17	Out C+	38	Out C-	59	n/c
	18	Out D+	39	Out D-	60	Rsvd
	19	Rsvd	40	Gnd	61	In C-
	20	In D+	41	In D-	62	In C+
	21	Rsvd	42	n/c		

Dashed Line represents the isolation barrier between channels

Figure 2-10. Pinout Diagram for DNx-AI-255

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.

2.5 Synchro-Resolver Wiring Connections

The table shown below (**Table 2-1**) matches the terminal connections of the AI-255 with the corresponding terminals on a synchro or a resolver device or simulator in each of the various operating modes.

For a synchro, the terminals are S1, S2, S3 and C for the three stator windings and common and R1, R2 for the rotor. For a resolver, the terminals are S1/S3 and S2/S4 for stator windings and R1/R2 for the rotor. Exc+ and Exc– refer to excitation. For Connection Diagrams of the various modes, see Appendix B.

Table 2-1. Synchro-Resolver Wiring Connections for Various Modes

Signal Name	Pin No.		Input Mode, Internal Excitation		Input Mode, External Excitation		Simulator Mode, Internal Excitation		Simulator Mode, External Excitation	
	Ch 0	Ch 1	*Synchro*	*Resolver*	*Synchro*	*Resolver*	*Synchro*	*Resolver*	*Synchro*	*Resolver*
In A+	4	14	S1	S1	S1	S1	NC	NC	NC	NC
In A-	25	35	C	S3	C	S3	NC	NC	NC	NC
In B+	5	15	S3	S2	S3	S2	NC	NC	NC	NC
In B-	26	36	C	S4	C	S4	NC	NC	NC	NC
In C+	9	62	S2	NC	S2	NC	NC	NC	NC	NC
In C-	30	61	C	NC	C	NC	NC	NC	NC	NC
In D+	10	20	NC	NC	Exc+	Exc+(R1)	NC	NC	Exc+	Exc+
In D-	31	41	NC	NC	Exc-	Exc-(R2)	NC	NC	Exc-	Exc-
OutA+	44	54	NC	NC	NC	NC	S1	S1	S1	S1
OutA-	43	53	NC	NC	NC	NC	C	S3	C	S3
OutB+	2	12	NC	NC	NC	NC	S3	S2	S3	S2
OutB-	23	33	NC	NC	NC	NC	C	S4	C	S4
OutC+	7	17	NC	NC(R2)	NC	NC	S2	Opt+(R2)	S2	NC
OutC-	28	38	NC	NC(R4)	NC	NC	C	Opt-(R4)	C	NC
OutD+	8	18	R1	R1	NC	NC	Exc+	Exc+(R1)	NC	NC
OutD-	29	39	R2	R3	NC	NC	Exc-	Exc-(R3)	NC	NC
NC	24,47	32, 42, 57, 59	--	--	--	--	--	--	--	--
GND	6, 22, 45, 51	16, 34, 40, 55	--	--	--	--	--	--	--	--
Rsvd	1, 3, 27, 46, 48, 49, 50, 52	11, 13, 19, 21, 37, 56, 58, 60	--	--	--	--	--	--	--	--

2.5.0.1 Line-to-Line & Peak-to-Peak Measurement

The AI-255 performs measurement and stimulation to attached synchros and resolvers by comparing each of the inputs (S1, S2, S3) to a ground reference line at the AI-255's ADC using peak-to-peak voltage (V_{pp}) values.

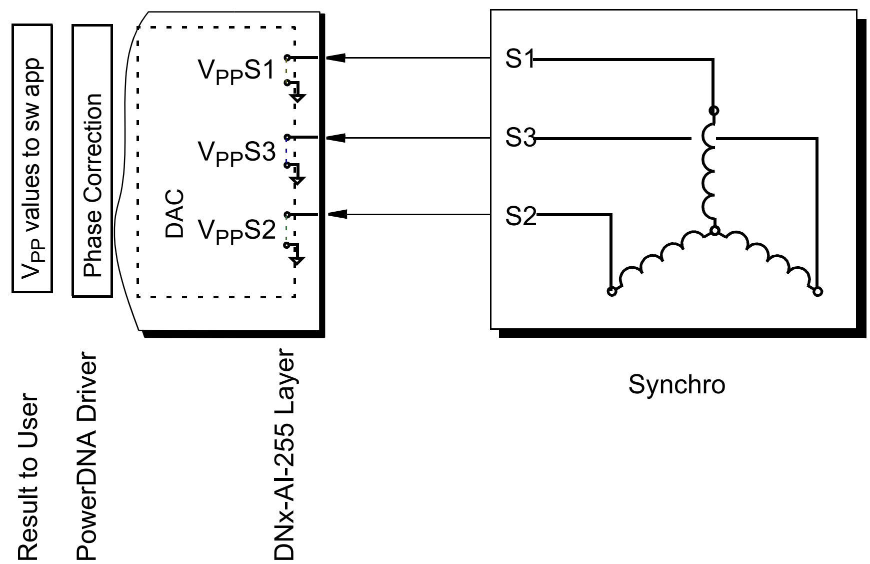


Figure 2-11. Peak-to-peak voltage measurement of Synchro

The synchro, however, is most commonly rated to use the root mean squared (rms) voltage as measured between two lines (not ground referenced). This is the rms line to line voltage (V_{LL}) measured across two of the three stator wires (V_{S1-3}, V_{S3-2}, or V_{S2-1}; as seen on page 6) or the excitation wires (V_{ext+}, V_{ext-}).

The user must convert between the line-to-line RMS voltage amplitude (V_{LL}) from the synchro specification and the peak-to-peak output-to-ground voltage amplitude (V_{PP}) that is the parameter needed by the AI-255 driver.

Convert between V_{LL} and V_{PP} with the relation:

$$V_{PP} = 2\sqrt{2}\frac{V_{LL}}{2\sin(120°)} \cong 1.63299V_{LL}$$

or using the low-level software macro from *powerdna.h*:

```
#define DQ_AI255_RMS_LN_LN_TO_PP(V) ((V)*1.633)
```

where input parameter V is V_{LL}, and the resulting output is V_{PP}.

Additionally, the low-level API also defines a macro to convert from V_{LL} to V_{RMS}:

```
#define DQ_AI255_RMS_LN_LN_TO_RMS(V) ((V)*0.5774)
```

where input parameter V is the line-to-line voltage V_{LL}, and the result is in volts RMS referenced to ground (V_{RMS}). The constant 0.5774 is 1/(2·sin(120º)).

As an example, a synchro with the rms excitation voltage of 23.5V between V_{ext+} and V_{ext-} will need a ground-referenced peak-to-peak voltage of 66.4V set for the AI-255, since 66.4V ≈ 2·√(2)·23.5V (no phase adjustment necessary). The same synchro's rms stator voltage of 11.8V that is the rms voltage between any two stator connections (responsible for positioning the rotor) should have a maximum peak-to-peak voltage span of 19.26V ≈ 11.8V/1.633·2·√(2).

Exercise caution when wiring and double-check that correct voltage is set on the AI-255 to avoid overloading and permanently damaging the synchro or resolver.

In Figure 2-11 on page 16, once the data has been sampled, the Cube logic corrects the phase by subtracting 30º, 150º, and 270º from S1, S2, and S3 resulting waveforms to yield an ideal voltage representation. This transformation is transparent, the final result appears as the final V_{PP} value to user application.

2.5.0.2 Z-grounded mode

It is possible to ground the z (S3) lines of some synchros to the vehicle's common ground to save on wiring - this is called a synchro in *z-grounded mode*. For z-grounded synchros, the S3 input/output on the AI-255 is left unconnected. The mode must be set for the AI-255 in software when using z-grounded mode. The modes for the low-level API are listed in the table below; refer to **Chapter 4** for information on low-level software programming.

Mode of Operation	Description
DQ_AI255_MODE_SI_INTZ	Synchro input, int. exc., Z grounded
DQ_AI255_MODE_SI_EXTZ	Synchro input, ext. exc., Z grounded
DQ_AI255_MODE_SS_INTZ	Synchro output, int. exc., Z grounded
DQ_AI255_MODE_SS_EXTZ	Synchro output, ext. exc., Z grounded

Table 2-2. Z-grounded modes of operation from powerdna.h

The connection diagram for wiring a synchro in z-grounded mode is as follows:

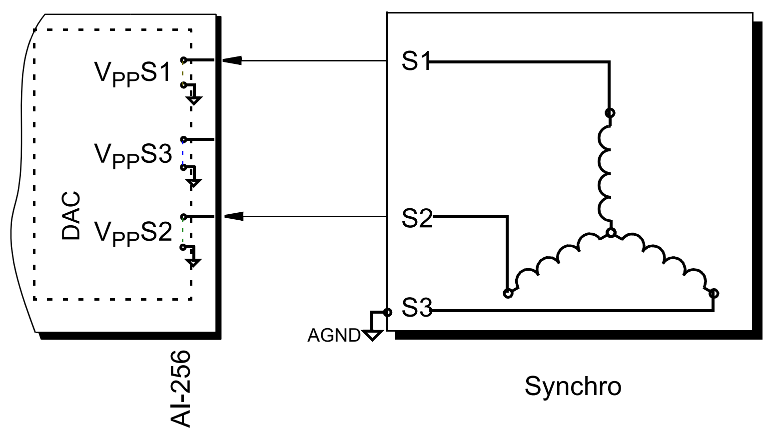


Figure 2-12. Connection in Z-grounded mode of Synchro

2.5.0.3 Trouble-shooting

This section describes some of the symptoms observed when the synchro is not wired correctly to the layer. Incorrect wiring can be as mild as inaccurate rotor position or as severe as permanently damaging the synchro.

In the mildest cases, the synchro or rotor lines may be in incorrect positions in the terminal panel. Reversing the rotor (V_{ext+}, V_{ext-}) or stator (S1, S2, S3) wires can cause the position of the rotor to be at a wrong angle, or rotate clockwise.

In the more severe cases, the rotor may move in a jerky or erratic manner, the synchro may hum and may be warm/hot to the touch, indicating a possible open connection. Warm or hot synchro's may also also indicate a short circuit.

Whereas the layer does have overvoltage protection up to $350V_{RMS}$ and thermal protection, the synchro may be permanently damaged by a bad voltage setting. It is recommended to check the configured voltage with an oscilloscope (best set to measure in true RMS mode) to ensure that the output voltages are correct. Unusual waveforms on an oscilloscope may indicate that thermal limits are being reached (normally due to an overloaded synchro), and waveforms that drop to zero may indicate that the overvoltage protection was breached and the layer has shut down. Overvoltage messages will appear on the serial console and also returns as STS_POST_OVERCURRENT in the POST word of the layer status of the low level framework.

Chapter 3 Programming with the High Level API

This section describes how to control the DNx-AI-255 using the UeiDaq Framework High Level API.

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.

The following section focuses on the C++ API, but the concept is the same no matter what programming language you use.

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

Also see the BufferedAISynchroResolver and AnalogInBufferedSynchroResolver examples provided by the UEI Framework for a starting point.

3.1 Creating a Session

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

```
// create a session object for input, and a session object for output

CUeiSession aiSession;
CUeiSession aoSession;
```

3.2 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 on device 1 at IP address 192.168.100.2: “pdna://192.168.100.2/Dev1/Ai0,1”

3.3 Configuring for Input

The AI-255 can be configured for synchro/resolver input.

Use the method **CreateSynchroResolverChannel()** to program the input channels and parameters associated with each channel.

The following call configures the analog input channels of an AI-255 set as device 1:

```
// Configure session synchro/resolver input

aiSession.CreateSynchroResolverChannel(
                                 "pdna://192.168.100.2/Dev1/Ai0,1",
                                  UeiSynchroMode,
                                  3.0,
                                  5000.0,
                                  false);
```

It configures the following parameters:

- **Sensor Mode**: the type of sensor (synchro or resolver) connected to the input channel.
- **Excitation Voltage**: the amplitude of the excitation sine wave in volts RMS.
- **Excitation Frequency**: the frequency of the excitation sine wave.
- **External Excitation**: specifies whether you wish to provide external excitation or use the excitation provided by the AI-255.

If you want to use different parameters for each channel, you can call **CreateSynchroResolverChannel()** multiple times with a different set of channels (0 or 1) in the resource string.

Note that the external excitation amplitude value that comes back from firmware is a peak-to-peak voltage that is converted to an RMS value by the framework on the assumption that it is a sinusoidal excitation signal. However, position transducers may use a square wave or a pulse for excitation. As a result, the amplitude for these signals will appear to be low, and only serve to verify the existence of a signal. When using the framework, the actual RMS or peak-to-peak amplitude of the excitation signal should be measured using an oscilloscope to ensure correctness.

3.4 Simulated Synchro/ Resolver Output

The AI-255 can be configured for synchro/resolver output. When using the AI-255 in Synchro/Resolver Mode, you can also use the AI-255 to simulate a Synchro or a Resolver output.

The following call configures an analog output channel of an AI-255 set as device 1:

```
// Configure session for synchro/resolver output

aoSession.CreateSimulatedSynchroResolverChannel(
                                   "pdna://192.168.100.2/Dev1/AO0",
                                    UeiResolverMode,
                                    3.0,
                                    5000.0,
                                    false);
```

It configures the following parameters:

- **Sensor Mode**: the type of sensor (synchro or resolver) to be simulated.
- **Excitation Voltage**: the amplitude of the excitation sine wave in volts RMS.
- **Excitation Frequency**: the frequency of the excitation sine wave.
- **External Excitation**: specifies whether you wish to provide external excitation or use the excitation provided by the AI-255.

3.5 Configuring the Timing

You can configure the AI-255 to run in simple mode (point by point), buffered mode (ACB mode), or DMAP mode.

NOTE: ACB (buffered) mode is not currently supported for the AI-255, but will be available in the near future. If you have a specific need for this feature, please contact UEI for current availability status

In simple mode, the delay between samples is determined by software on the host computer. In DMAP mode, the delay between samples is determined by the AI-255 on-board clock and data is transferred one scan at a time between PowerDNA and the host PC. In buffered mode, the delay between samples is determined by the AI-255 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) for AI
aiSession.ConfigureTimingForSimpleIO();
// configure timing of input for point-by-point (simple mode) for AO
aoSession.ConfigureTimingForSimpleIO();
```

3.6 Read Data

Reading data is done using *reader* object(s). 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);
```

3.7 Write Data

Writing data is done using a *writer* object.

The following sample shows how to create a scaled writer and write samples. The AI-255 simulates angle positions entered in radians.

```
// create a writer and link it to the session's analog-output stream
CUeiAnalogScaledWriter aiWriter(aoSession.GetDataStream());
// to write a value, the buffer must contain one value per channel
double data[2] = { 1.0, 2.0 };
// write one scan, where the buffer contains one value per channel
aoWriter.WriteSingleScan(data);
```

3.8 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 sessions
aiSession.CleanUp();
aoSession.CleanUp();
```

Chapter 4 Programming with the Low-level API

This chapter illustrates how to program the PowerDNA cube using the low-level API. The low-level API offers direct access to PowerDNA DAQBios protocol and also allows you to access device registers directly.

However, we recommend that, when possible, you use the UeiDaq Framework High-Level API (see **Chapter 3**), because 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.

NOTE: This chapter contains descriptive information about the various operating modes and wiring connections of the AI-255, and descriptions of the low-level functions that may be used in programming this module. These functions can also be used by the AI-256. Note, however, that the AI-255 only has lower frequencies and current than the AI-256.

For additional information about low-level programming of the AI-255, please refer to the PowerDNA API Reference Manual document under:

Start » Programs » UEI » PowerDNA » Documentation

For a good starting point, please consider reviewing the examples for the AI-255 that are Sample255 (input) and Sample255_Simulation (output) under:

Start » Programs » UEI » PowerDNA » Examples

4.1 DNx-AI-255 Modes of Operation

The basic modes of operation supported by an DNx-AI-255 channel are:

- Synchro Input
- Resolver Input
- Synchro Output/Simulator
- Resolver Output/Simulator

There are eight Synchro or Resolver modes of operation.

Functions performed by the eight modes of operation supported by this layer may be described as follows:

- **Synchro Input with internal excitation**: The AI-255 reads the voltages on the stator coils as analog inputs and also supplies the excitation voltage to the rotor coil.
- **Synchro Input with external excitation:** The AI-255 reads the voltages on the stator coils as analog inputs. An external source supplies the excitation voltage to the rotor coil, which is readback by the AI-255 as an analog input.
- **Resolver Input with internal excitation**: The AI-255 reads the voltages on the stator coils as analog inputs and also supplies the excitation voltage to the rotor coil(s).
- **Resolver Input with external excitation:** The AI-255 reads the voltages on the stator coils as analog inputs. An external source supplies the excitation voltage to the rotor coil(s), which is readback by the AI-255 as an analog input.
- **Synchro Simulation with internal excitation:** The AI-255 outputs voltages that simulate the analog signals from stator coils of a synchro. It also outputs an analog excitation voltage generated in the AI-255.
- **Synchro Simulation with external excitation:** The AI-255 outputs voltages that simulate the analog signals from stator coils of a synchro. Excitation voltage is supplied by an external source and read back by the AI-255 as an analog input.
- **Resolver Simulation with internal excitation:** The AI-255 outputs voltages that simulate the analog signals from stator coils of a resolver. It also outputs an analog excitation voltage generated in the AI-255.
- **Resolver Simulation with external excitation:** The AI-255 outputs voltages that simulate the analog signals from stator coils of a resolver. Excitation voltage is supplied by an external source and read back by the AI-255 as an analog input.

All the modes, including additional z-ground modes, are defined in powerdna.h.

The above eight modes of operation are set by the following definitions:

```
#define DQ_AI255_MODE_SI_INT0  // Synchro input, internal excitation
#define DQ_AI255_MODE_RI_INT1  // Resolver input, int.exc.
#define DQ_AI255_MODE_SI_EXT2  // Synchro input, ext. exc. - readback exc. on D
#define DQ_AI255_MODE_RI_EXT3  // Resolver input, ext. exc. - readback exc. on D
#define DQ_AI255_MODE_SS_INT4  // Synchro drive/simulation, int. exc. - fully sourced
#define DQ_AI255_MODE_RS_INT5  // Resolver drive/simulation, int.exc. - fully sourced
#define DQ_AI255_MODE_SS_EXT6  // Synchro drive/simulation, ext. exc. - readback on D
#define DQ_AI255_MODE_RS_EXT7  // Resolver drive/simulation, ext. exc. - readback on D
```

4.1.1 Wiring

A synchro has three stator coils S1, S2, S3 connected in a star or delta fashion to the Common. The rotor primary coil (exciter) has wires R1 and R2.

Resolver stator coils are S1-S3 and S2-S4. Rotor coil is R1-R3 (and R2-R4 when two rotor windings are used).

Table 4-1 describes the wiring connections for various modes of operation.

Table 4-1. Wiring Connections for Various Synchro/Resolver Operating Modes

Mode	Synchro (three 120º coils)		Resolver (two 90º coils)	
	AOut	**AIn**	**AOut**	**AIn**
Input, internal excitation, $28V_{rms}$ 400Hz-4kHz.	R1 and R2 connected to D+ and D- (optionally use A, B or C in parallel).	S1 to A+, S2 to B+, S3 to C+.	R1 and R3 connected to D+ and D-, optionally R2 and R4 to C+ and C- (two windings per rotor).	S1 to A+, S3 to A-, S2 to B+, S4 to B-.
Input, external excitation $28V_{rms}$ (from A/C bus).	N/C	S1 to A+, S2 to B+, S3 to C+, Excitation readback to D+/D-.	N/C	S1 to A+, S3 to A-, S2 to B+, S4 to B-, Excitation readback to D+/D-.
Output, internal excitation, $28V_{rms}$.	S1 to A+, S2 to B+, S3 to C+, R1 to D+, R2 to D- and Common to AGND.	N/C	S1 to A+, S3 to A-, S2 to B+, S4 to B- R1 to D+, R3 to D- and Common to AGND. Optionally, R2 to C+ and R4 to C-.	N/C
Output, external excitation $28V_{rms}$ (from A/C bus), internal drive (resolver only).	S1 to A+, S2 to B+, S3 to C+ and Common to AGND.	Excitation readback to D+/ D.-	S1 to A+, S3 to A-, S2 to B+ and S4 to B-.	Excitation readback to D+/ D-.

Mode: Input, Internal Excitation

```
DQ_AI255_MODE_SI_INT:
DQ_AI255_MODE_RI_INT:

     exc_rate = 400.0; <- excitation frequency is the same for both calls
     exc_level = 26.0; <- level for the rotor coil
     adc_rate = 0; -> returns actual sampling rate
     ret = DqAdv255SetExcitation(hd0, DEVN, CHANNEL,
                                 DQ_AI255_ENABLE_EXC_D, // D channel only
                                 exc_rate, exc_level, &adc_rate);

     exc_level = 26.0; <- level for the rotor coil (from the datasheet)
     ret = DqAdv255SetMode(hd0, DEVN, CHANNEL, mode, flags,
                          (float)exc_rate, (float)exc_level);
```

Mode: Input, External Excitation

```
DQ_AI255_MODE_SI_EXT:
DQ_AI255_MODE_RI_EXT:

// Measure frequency and level on input D
ret = DqAdv255MeasureWF(hd0, DEVN, CHANNEL_COARSE,
                        &exc_rate, &exc_level, &exc_offs);

// use excitation frequency measured on the rotor winding
// use excitation voltage measured on the rotor winding
// this information is required to estimate A/D settings
ret = DqAdv255SetMode(hd0, DEVN, CHANNEL, mode, flags, exc_rate, exc_level);
```

Mode: Output, Internal Excitation

```
DQ_AI255_MODE_SS_INT:
DQ_AI255_MODE_RS_INT:

     exc_rate = 400.0; <- excitation frequency is the same for both calls
     exc_level = 26.0; <- level for the rotor coil
     adc_rate = 0; -> returns actual sampling rate
     ret = DqAdv255SetExcitation(hd0, DEVN, CHANNEL_SIM,
                                 DQ_AI255_ENABLE_EXC_D, // D channel only
                                 exc_rate, exc_level, &adc_rate);

    exc_level = 11.8; <- level for the stator coils (from the datasheet)
    ret = DqAdv255SetMode(hd0, DEVN, CHANNEL_SIM, mode, flags,
                          (float)exc_rate, (float)exc_level);
```

Mode: Output, External Excitation

```
DQ_AI255_MODE_SS_EXT:
DQ_AI255_MODE_RS_EXT:

// Measure frequency and level on input D
ret = DqAdv255MeasureWF(hd0, DEVN, CHANNEL_COARSE,
                        &exc_rate, &exc_level, &exc_offs);


exc_level = 11.8; <- level for the stator coils (from the datasheet)
ret = DqAdv255SetMode(hd0, DEVN, CHANNEL, mode, flags, exc_rate, exc_level);
```

4.2 Low–Level Functions

The low-level synchro/resolver functions for an AI-255 layer used with synchro/resolvers are described in the PowerDNA API Reference Manual Release 4.5, Section 4.9 to 4.11.

The following functions are inherited by the AI-256 from the AI-255 for use with synchro/resolver applications:

Function	Description
`DqAdv255SetMode`	Sets up one of the S/R operating modes supported by the AI-255.
`DqAdv255SetExt`	Set up extra (additional) parameters.
`DqAdv255SetExcitation`	Sets excitation frequency and amplitude in internal exitation mode.
`DqAdv255GetWFMeasurements`	Returns the measured parameters of waveform on selected input(s).
`DqAdv255MeasureWF`	Simple form of DqAdv255GetWFMeasurements.
`DqAdv255Enable`	Refer to DqAdv256Enable.
`DqAdv255GetExcitation`	Gets layer excitation voltage parameters of excitation waveform.
`DqAdv255Read`	Read the calculated angle or special data for selected channels.
`DqAdv255Write`	Write a simulated position of a synchro or resolver or special data.
`DqAdv255ConvertSim`	Converts angle to raw data representation for gain and phase control.
`DqAdv255WriteBin`	Writes an angle or special data for selected channels.
`DqAdv255ReadDIn`	Reads digital inputs.
`DqAdv255WriteDOut`	Writes digital outputs and reads back digital inputs.

Appendix A

A. Accessories

The following cables and STP boards are available for the AI-255 layer.

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.

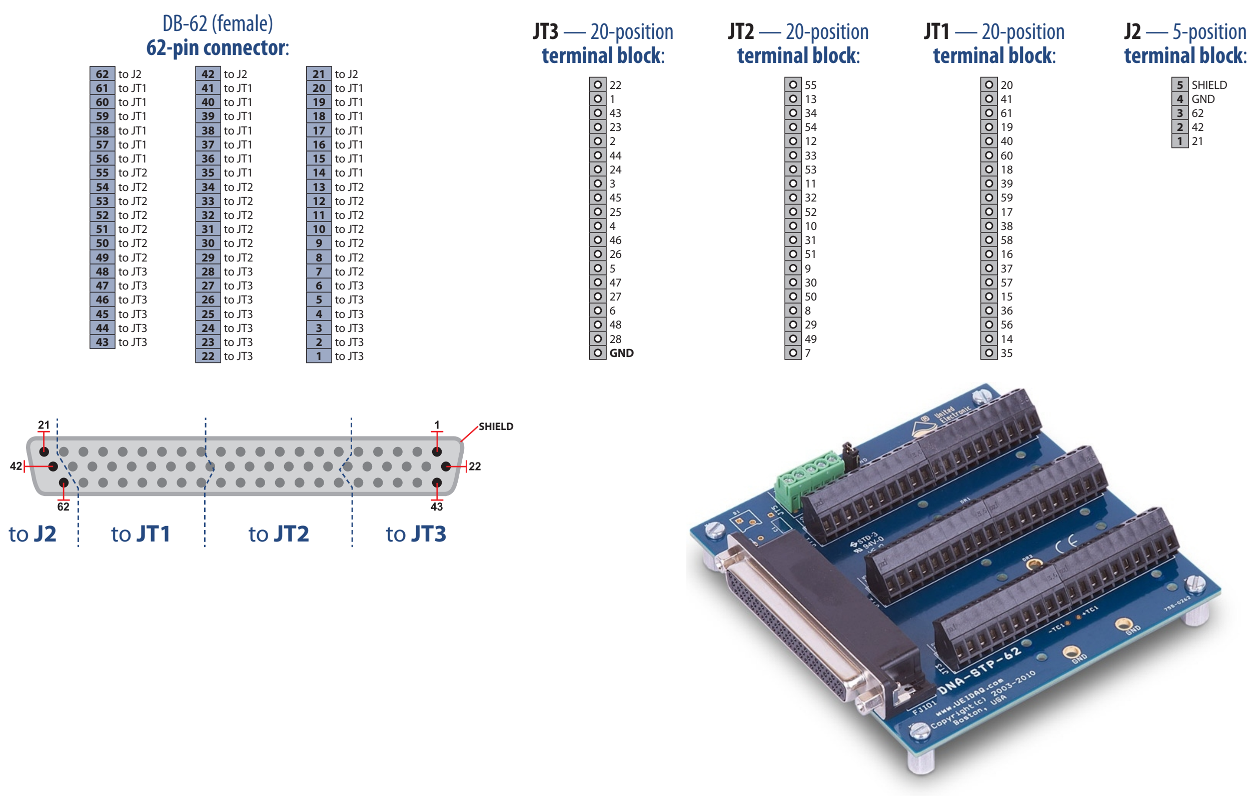


Figure A-1. Pinout and photo of DNA-STP-62 screw terminal panel

Appendix B

Connection Diagrams for S/R Operating and Excitation Modes

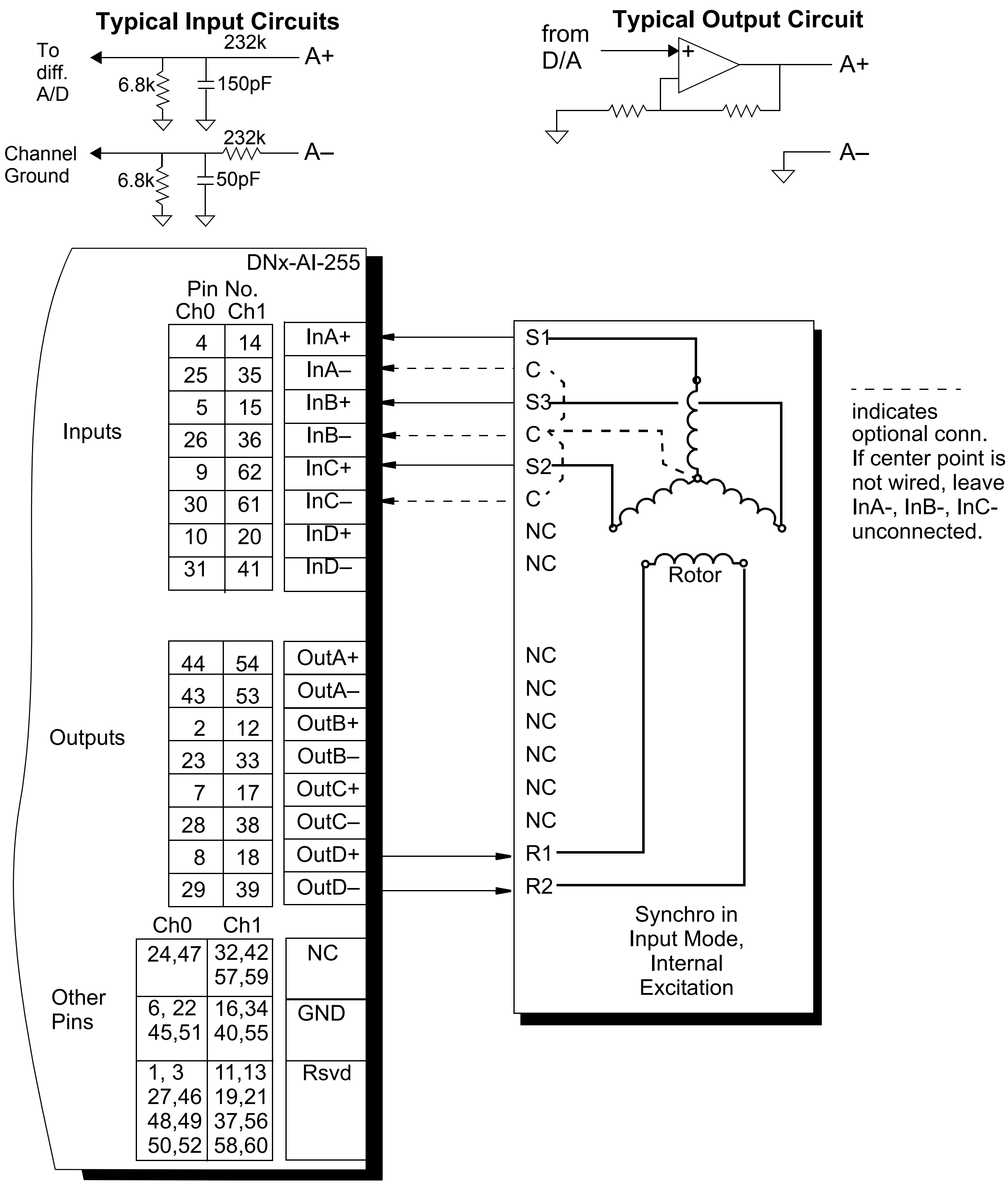


Figure B-1. AI-255 in Synchro Input Mode with Internal Excitation

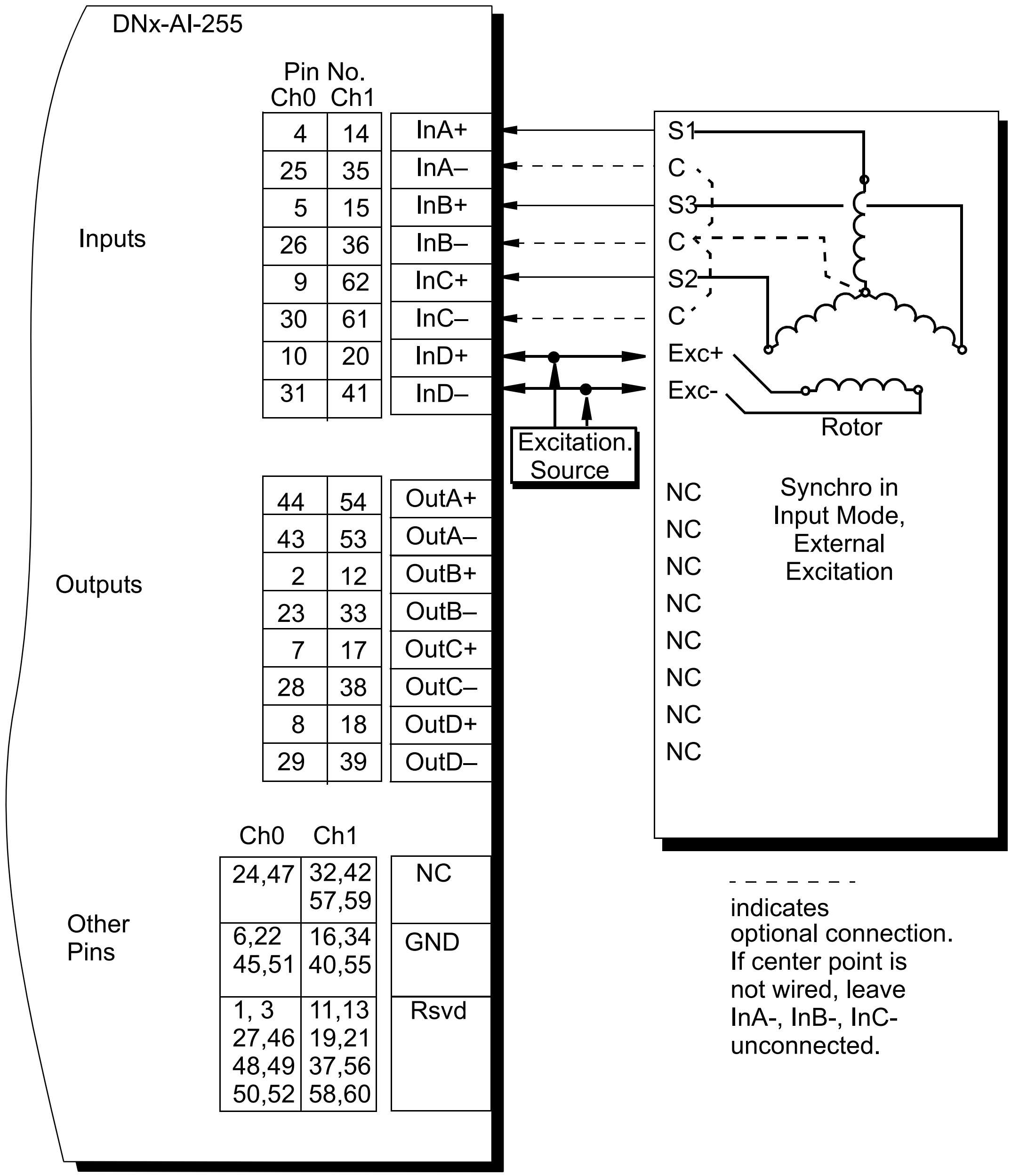
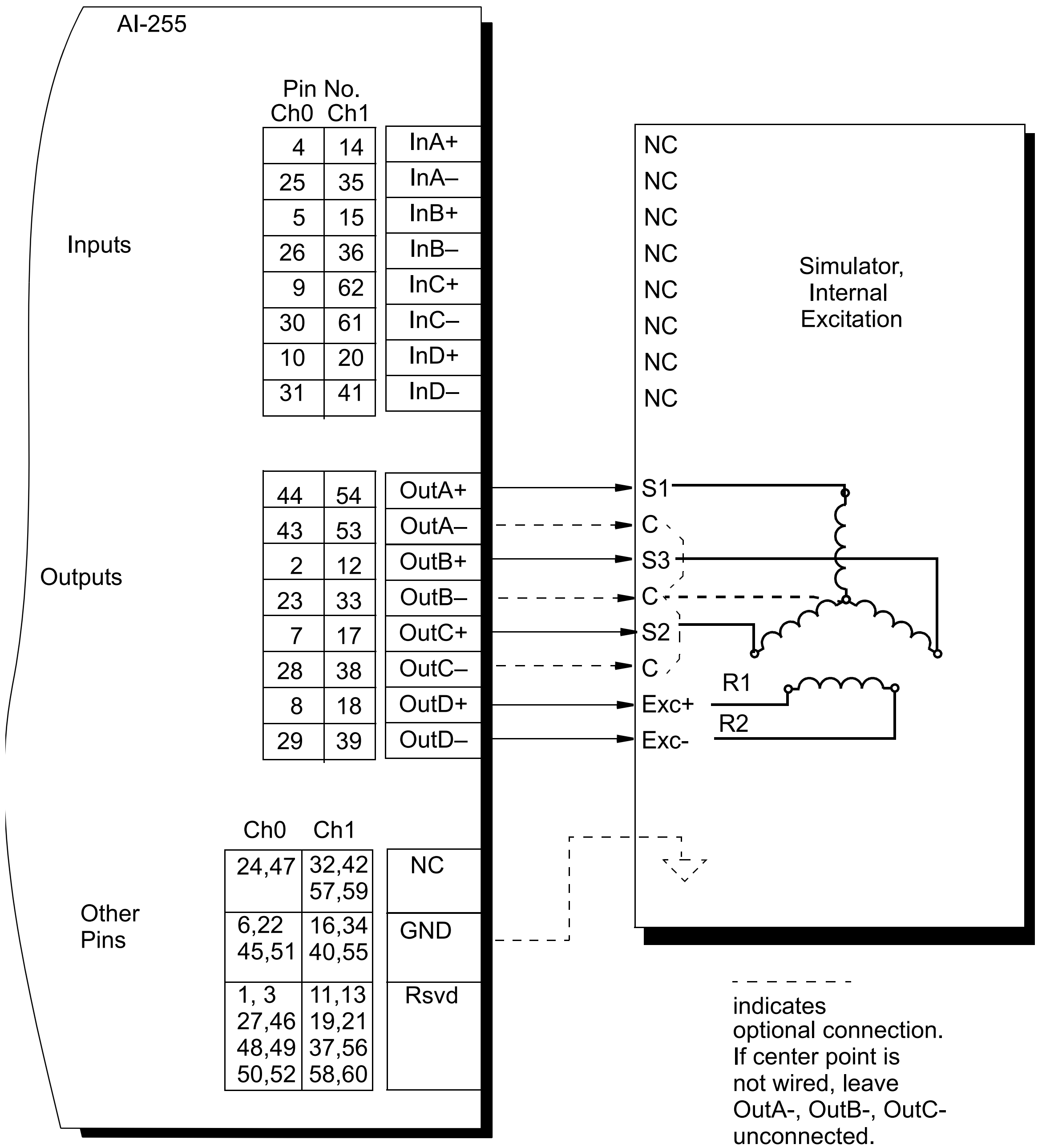


Figure B-2. AI-255 in Synchro Input Mode with External Excitation



NOTE: Most synchros do not require connections to OutA-, OutB-, and OutC-. Some devices, however, use electronic equivalents of synchros. In such cases, connect the channel ground to the device ground.

Figure B-3. AI-255 in Synchro Simulator Mode with Internal Excitation

As an example of the above configuration, here is a practical setup to drive a synchro with 3-wire stator (20.4V_{PP}) and 2-wire rotor (65V_{PP}) connected to Channel 0 of an AI-255 using a DNA-STP-62 board with pinout:

Signal	Synchro	Pin	Practical Notes
OutA+	S1	44	One of the three wires placed in a triangle (120º).
OutA-	(none)	43	Already wired internally to ground.
OutB+	S2	2	See above.
OutB-	(none)	23	See above.
OutC+	S3	7	See above.
OutC-	(none)	28	See above.
OutD+	Exc+	8	Striped wire on some synchro's, goes to the rotor coil. See synchro for excitation voltage (often RMS).
OutD-	Exc-	29	

Table 6-1. Setup for Simulator Mode w/ Int. Exc. example

PowerDNA Explorer configuration for the above setup on Channel 0:

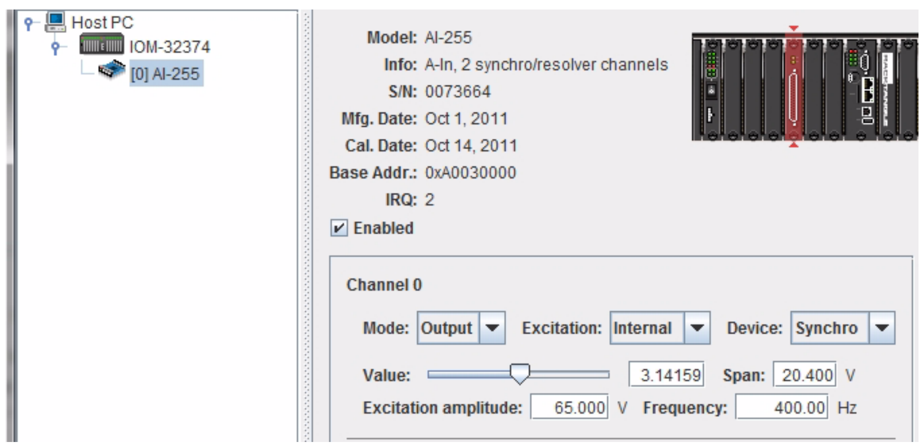


Figure B-4. PowerDNA Explorer in Simulator Mode, Int. Exc., 180º

Resulting waveform on an oscilloscope:

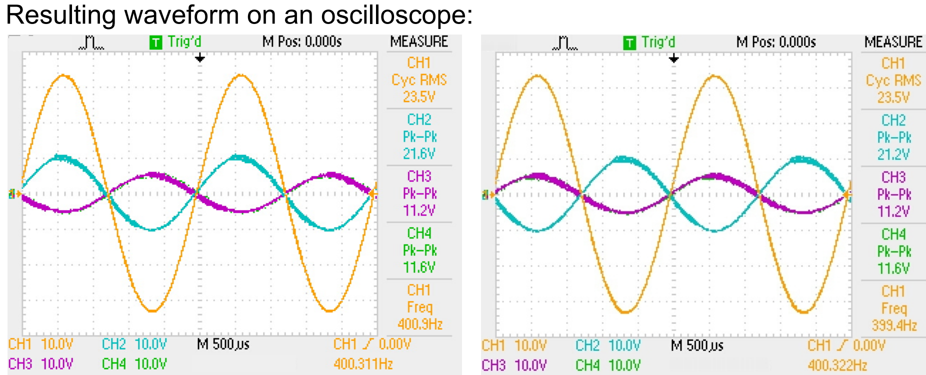


Figure B-5. Waveforms for Simulator Mode w/ Int. Exc. at 0º (left) and 180º (right)

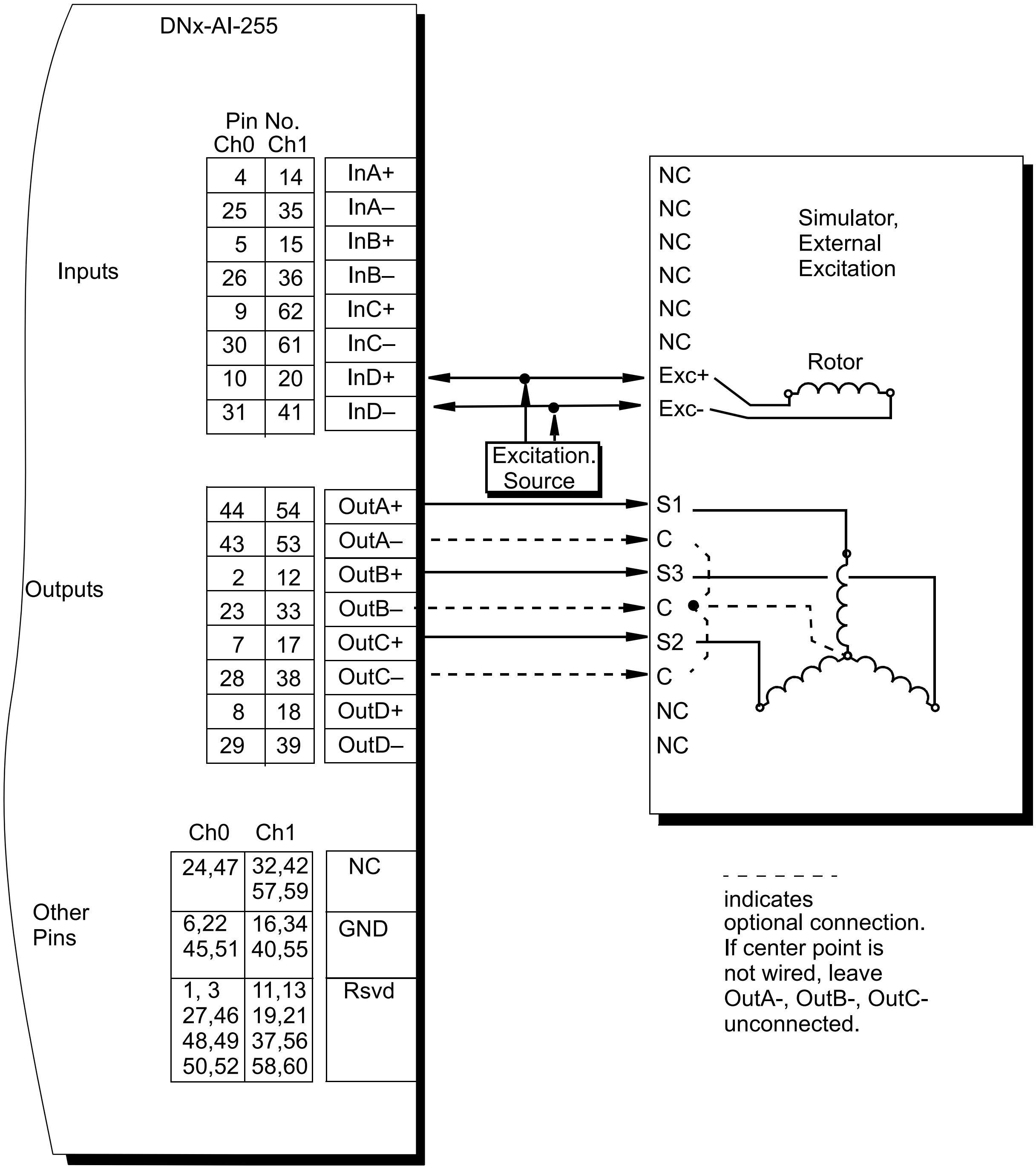
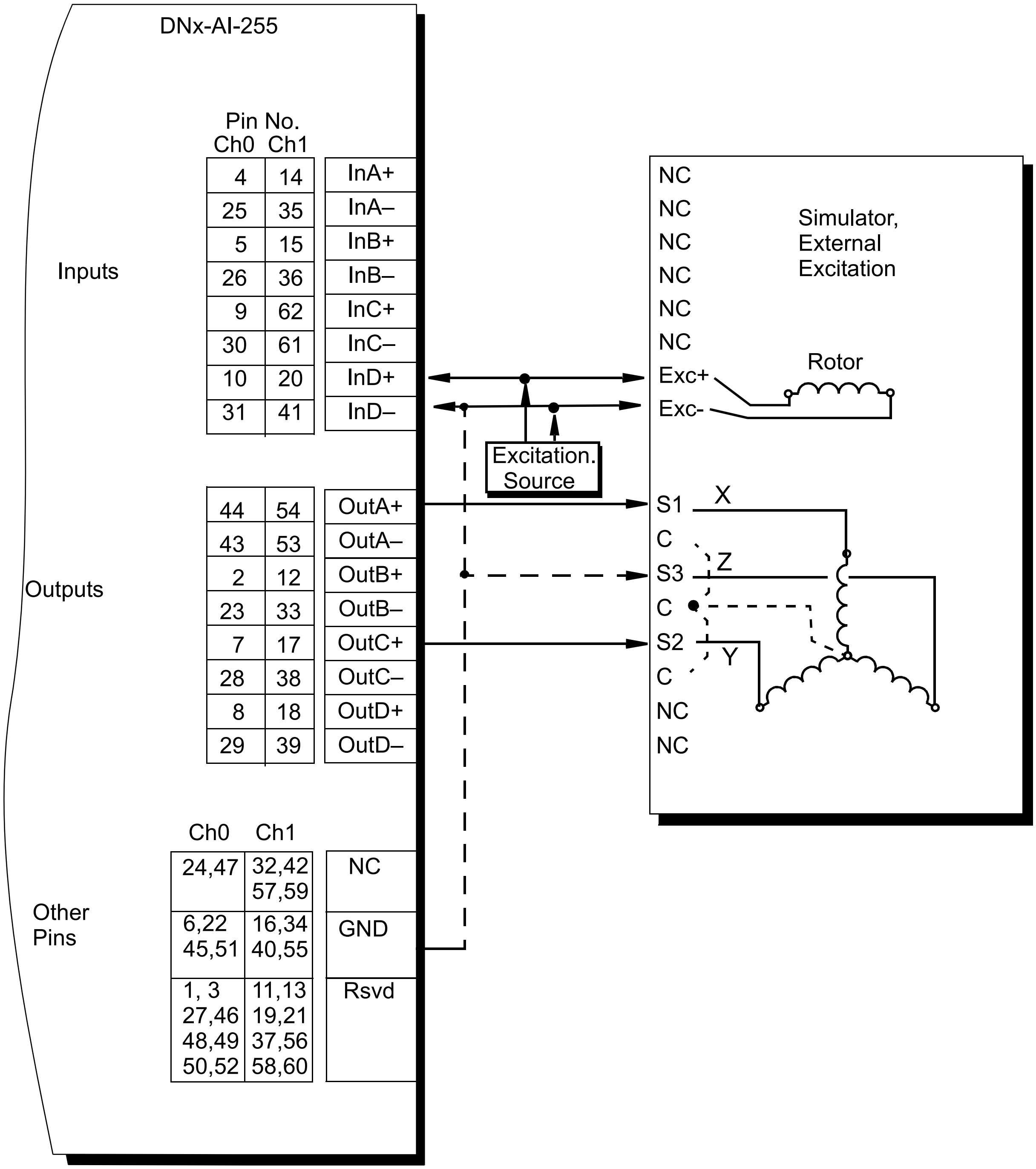


Figure B-6. AI-255 in Synchro Simulator Mode with External Excitation



AI-255 in Synchro Simulator Mode with External Excitation in Z-grounding Mode

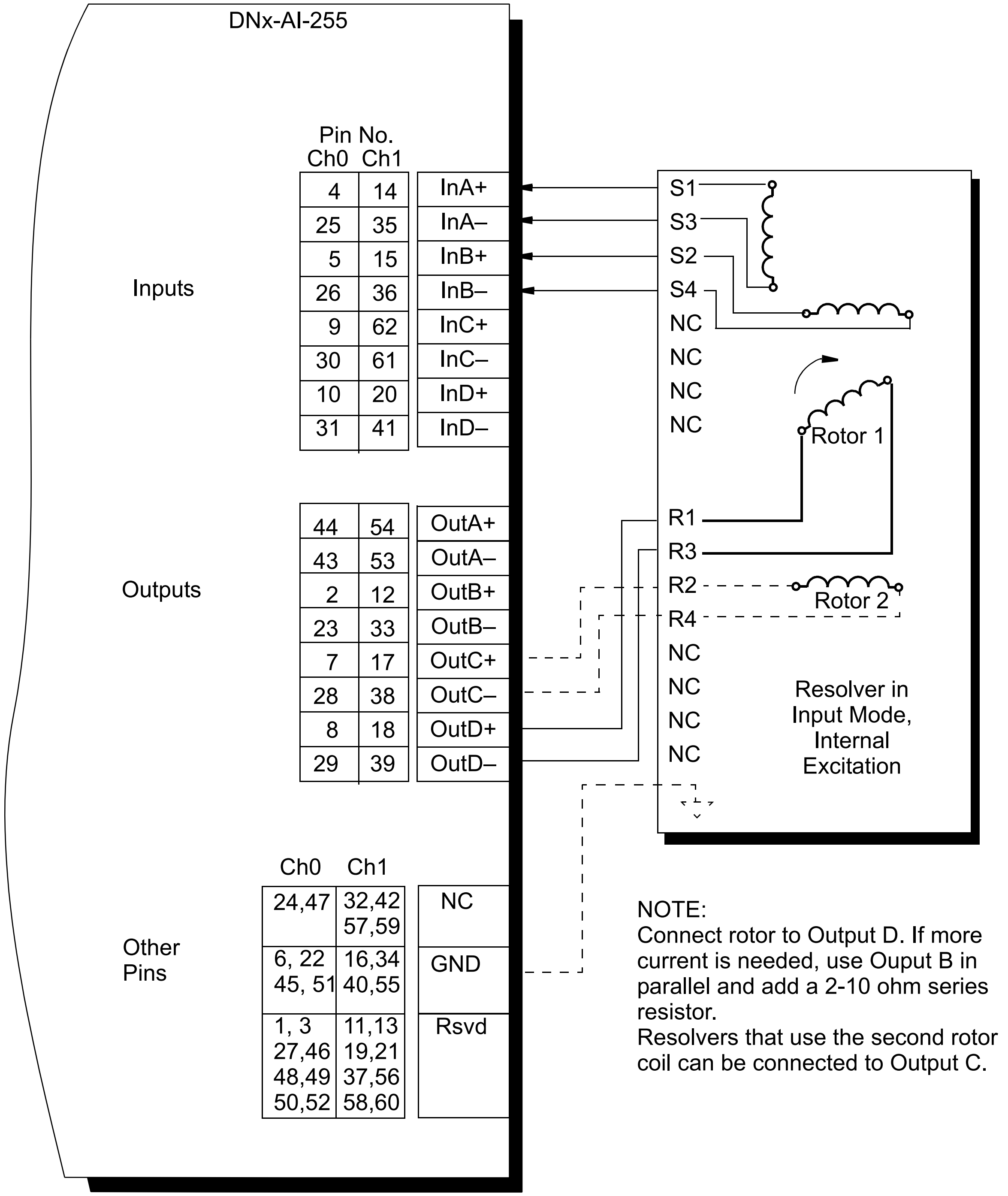


Figure B-7. AI-255 in Resolver Input Mode with Internal Excitation

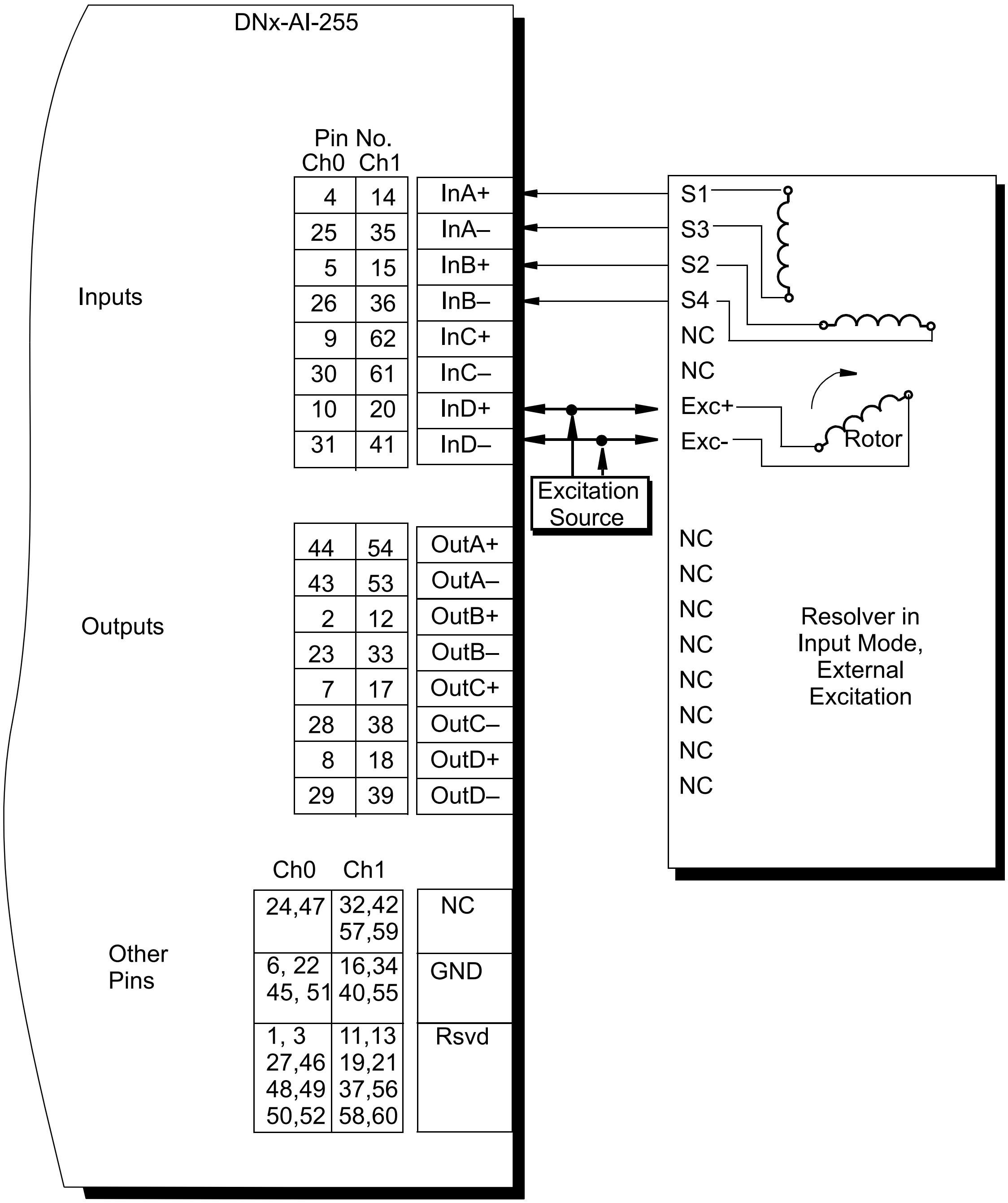


Figure B-8. AI-255 in Resolver Input Mode with External Excitation

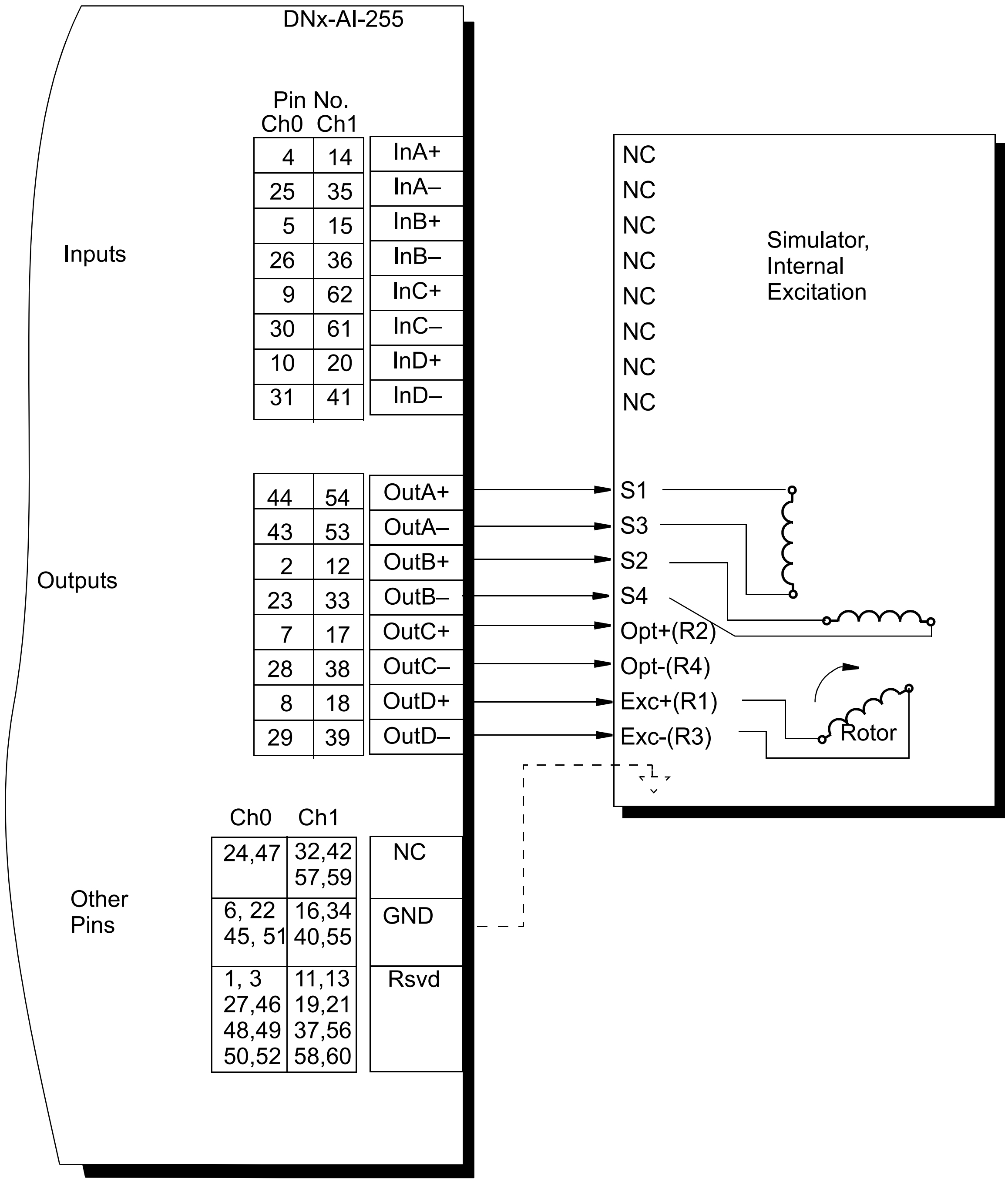


Figure B-9. AI-255 in Resolver Simulator Mode with Internal Excitation

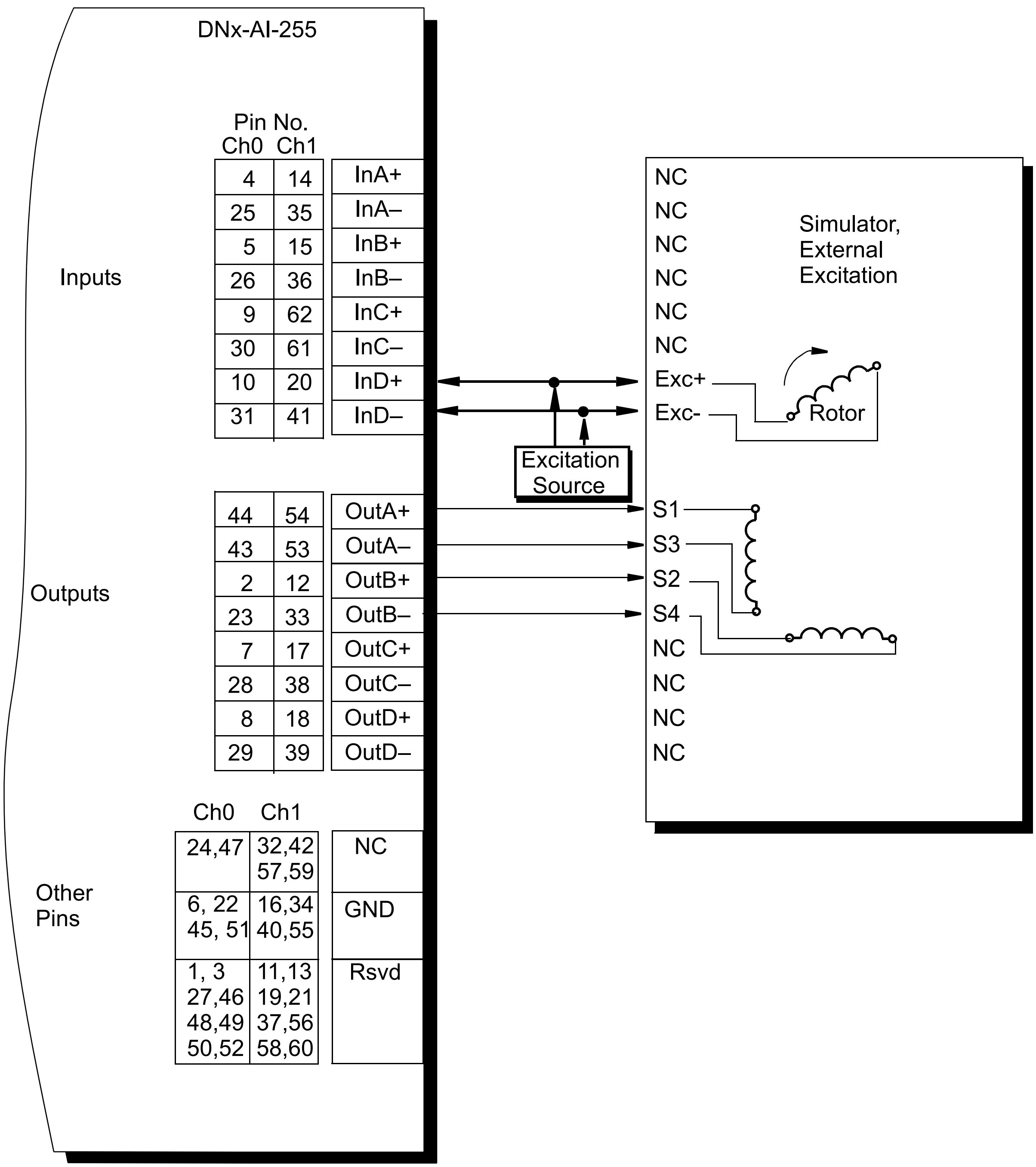


Figure B-10. AI-255 in Resolver Simulator Mode with External Excitation

Index